Financial Fraud Detection Using LLM

Kaushik Das

Introduction

This project explores the use of Large Language Models (LLMs) for detecting and investigating fraudulent activities in banking transactions. Instead of relying solely on traditional classification models, the approach treats fraud detection as an investigative process.

The system uses an AI “agent” framework inspired by recent research, where different agents work together — one plans the investigation, others gather and analyze evidence, and a final decision-making agent determines whether the activity is fraudulent. The aim is to make fraud detection more adaptable, explainable, and closer to how a human investigator would approach a case. Early tests show that the system can identify suspicious patterns and provide clear reasoning for its decisions, with further improvements planned to enhance accuracy and reduce bias.

Problem Statement And The Advantages of LLM’s

Fraud in the banking sector is a constantly evolving threat. Fraudsters adapt their methods rapidly—ranging from account takeovers to transaction laundering—making it increasingly difficult for static, rules-based, or even traditional machine learning (ML) models to keep pace.

Traditional ML approaches such as Random Forests, Gradient Boosted Trees, and SVMs have proven effective in detecting known fraud patterns by learning from historical data. However, they also exhibit significant limitations:

•Adaptability to New Threats: Fraud patterns change over time. Traditional models trained on past data require frequent retraining and re-engineering of features to handle emerging attack vectors (FraudFights.com).

• Limited Context Awareness: These models operate on fixed, pre-engineered features. They cannot dynamically explore data, retrieve new evidence, or adapt their analysis path mid-investigation.

• High False Positive Rates: Misclassifying legitimate customer transactions as fraud not only impacts customer trust but also increases operational costs from unnecessary reviews (arXiv:2410.20281).

• Class Imbalance: Fraudulent transactions typically make up less than 0.5% of banking transaction datasets (arXiv:2312.13334). This imbalance can lead to models being biased toward predicting “not fraud,” missing true positives or requiring artificial rebalancing.

• Explainability Gaps: Model outputs are often opaque to analysts. Tools like SHAP or LIME are needed to explain predictions, and these are themselves approximations that can be hard for non-technical stakeholders to interpret.

These constraints point to a fundamental gap: the need for a fraud detection system that is investigative, adaptive, and explainable—capable of simulating the thought process of a human fraud analyst.

Large Language Models (LLMs) address many of these shortcomings by introducing reasoning, adaptability, and native explainability into the fraud detection process. Unlike single-pass classifiers, LLMs can simulate the workflow of a human fraud analyst, conducting multi-step investigations, querying databases dynamically, analyzing transaction histories, generating visual plots, and adjusting their hypotheses based on intermediate findings. They are also capable of integrating multiple data modalities—structured transaction records, customer profiles, textual notes, and visualized timelines —into a unified decision-making process. This enables the detection of subtle patterns, such as shifts in spending categories, unusual merchant naming conventions, or geographically inconsistent purchases.

A key advantage of LLM-driven systems is their ability to produce narrative justifications for each decision. Instead of requiring post-hoc interpretation, they can directly explain the reasoning behind a fraud verdict in plain language, making them more transparent and easier to audit for regulatory purposes. Furthermore, because their reasoning process is guided by prompts and tool usage rather than fixed training data alone, they can adapt more quickly to emerging fraud patterns without full model retraining.

For the banking industry, these benefits translate into faster detection of novel schemes, reduced false positives that improve customer trust, and greater operational efficiency. In this context, implementing LLM-based investigative frameworks, such as the FAA (Fraud Analysis Agent) approach described in arXiv:2506.11635, represents a promising evolution in fraud detection—shifting from static classification toward adaptive, explainable, and context-aware investigations.

Dataset

To implement the concepts presented in the FAA (Fraud Analysis Agent) framework described in arXiv:2506.11635, I adapted the multi-agent, tool-augmented LLM approach to a real-world fraud detection scenario using publicly available banking transaction data. The FAA model consists of specialized agents—such as an orchestrator, SQL analysis agent, vision agent, and report-generation agent—that collaborate to investigate suspicious transactions in a structured, step-by-step process.

Dataset Selection and Preparation

For experimentation, I used the Synthetic Financial Datasets For Fraud Detection (SynthF) dataset, which is widely adopted for fraud detection research due to its realistic structure and balanced variety of legitimate and fraudulent transactions. The dataset contains detailed transaction records along with customer demographics, merchant information, geographic coordinates, and an is_fraud flag indicating whether each transaction is fraudulent. While the dataset is synthetic, it is designed to closely resemble real-world banking data and includes nuanced fraud patterns, making it suitable for evaluating the reasoning capabilities of LLM-based investigative systems.

Key columns in the dataset include:

- Customer Information: ssn, cc_num, first, last, gender, street, city, state, zip, dob

- Transaction Details: trans_num, trans_date, amount, category, merchant, merch_lat, merch_long

- Fraud Label: is_fraud (binary indicator for ground-truth classification)

Before ingestion, the dataset was cleaned to remove duplicate columns introduced during CSV merging, and it was loaded into a SQLite database (faa.sqlite) for efficient querying during investigations. This database serves as the knowledge base that the FAA agents interact with.

Implementation of The Model

The project implements the Fraud Analysis Agent (FAA) framework from FAA: LLM Agents for Fraud Detection in Banks, adapted to work with a real-world transactional dataset. The framework is structured as a multi-agent system, where each agent specializes in a different aspect of the investigation and interacts with the others to produce a final verdict.

1. Orchestrator Agent

The Orchestrator Agent coordinates the investigation process. It receives a case_id (transaction identifier) and determines which investigative steps should be taken, such as querying the database, generating visualizations, or performing pattern analysis. It decides the sequence of agents to call and the context to provide them, ensuring that the reasoning process is structured and efficient.

2. SQL Agent

The SQL Agent is responsible for extracting relevant information from the database. Given a natural language query from the Orchestrator, it generates safe SQL statements that are executed against the transactional database. Strict safeguards are in place to ensure that the agent cannot access the is_fraud column during investigations, preventing data leakage.

3. Vision Agent

The Vision Agent analyzes transaction patterns through plots and time series visualizations. It processes visual artifacts (such as transaction timelines) to identify unusual spending patterns, frequency spikes, or geographic inconsistencies. These insights are then passed back to the Orchestrator as part of the evidence set.

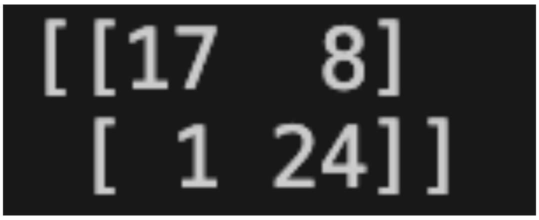
4. Detective Agent

The Detective Agent serves as the final stage of the FAA pipeline. It takes the compiled evidence from the SQL Agent, Vision Agent, and other investigative tools, then reasons over this evidence to determine whether the transaction is likely fraudulent.

Performance and Evaluation

The implemented Fraud Analysis Agent (FAA) framework was tested on a balanced subset of the dataset, containing an equal number of fraudulent and non-fraudulent transactions. This was done to mitigate the impact of class imbalance during evaluation and to better assess the model’s ability to distinguish between the two classes. For this experiment, 50 transactions were randomly sampled — 25 labeled as fraudulent (is_fraud = 1) and 25 as legitimate (is_fraud = 0). Each transaction was independently investigated by the FAA pipeline, and the model’s verdict (“fraud” or “not fraud”) was recorded along with a confidence score. These predictions were then compared to the ground truth labels in the is_fraud column, which was used solely for evaluation and never exposed to the model during investigation.

The evaluation metrics showed promising results. The model achieved an overall accuracy of 82%, with particularly strong performance in detecting fraudulent transactions, where it recorded a recall of 96% and a precision of 75%. This high recall rate is critical in fraud detection, as it ensures that most fraudulent cases are identified, reducing the likelihood of missed fraud. In contrast, legitimate transactions had a recall of 68% and a precision of 94%, indicating that while most flagged legitimate cases were indeed correct, the model still produced some false positives. The confusion matrix for this evaluation was as follows:



```
[[17  8]
 [ 1 24]]
```

From this, it is evident that the FAA is more aggressive in labeling transactions as fraudulent, which is consistent with the conservative bias often favored in banking environments, where the cost of missing fraud typically outweighs the cost of investigating false positives. This bias was moderated through the introduction of a confidence threshold and fraud score safeguard in the Detective Agent, which helped reduce false positives compared to earlier test runs.

Nevertheless, the results also highlight areas for improvement. The evaluation was conducted on a balanced dataset, which does not reflect real-world transaction distributions where fraud cases are rare. Future evaluations will include imbalanced datasets to better approximate operational performance. Additionally, the current model assesses each transaction in isolation; incorporating cumulative analysis at the account or customer level could provide richer context and improve both precision and recall. Integrating more domain-specific features, such as device fingerprints or geolocation consistency checks, could further enhance the robustness of the fraud detection process.

Future Scope

Several enhancements can be made to improve the FAA framework's performance, robustness, and applicability in real-world banking environments. First, the system can be extended to operate on customer-level behavioral analysis rather than isolated transactions. By aggregating transaction histories and identifying long-term anomalies, the model could better detect sophisticated fraud schemes such as account takeovers and structured transaction laundering.

Second, the dataset can be enriched with additional features that are commonly used in production fraud detection pipelines. Examples include device and browser fingerprints, IP-based geolocation data, customer login histories, and known fraud blacklists. These supplementary attributes would provide more context to the LLM and allow it to identify subtle risk patterns that might be invisible in the current dataset.

Third, fine-tuning or instruction-tuning the LLM on domain-specific fraud detection data could significantly improve its reasoning consistency and reduce false positives. Specialized prompt engineering strategies, combined with retrieval-augmented generation (RAG) to dynamically provide relevant historical cases, would also help the model make more grounded decisions.

Fourth, integrating real-time graph-based anomaly detection methods could complement the LLM’s natural language reasoning. By leveraging network analysis on customer–merchant–transaction relationships, the system could detect fraudulent rings or coordinated activities that are difficult to identify through tabular data alone.

Finally, the deployment of explainable AI (XAI) techniques is critical for regulatory compliance and stakeholder trust. Developing interpretable summaries of the model’s decision-making process would ensure that financial institutions can both trust the system’s outputs and justify actions taken based on its recommendations.