FraudShield AI Case Study

Syed Armghan Ahmad

syedarmghanahmad.work@gmail.com | LinkedIn | GitHub

May 28, 2025

Part of the AI Case Studies Portfolio: github.com/SyedArmghanAhmad/Case-Studies

Contents

1 Vision

As a self-taught developer with a relentless drive to innovate, I created FraudShield AI, a full-stack fraud detection platform that redefines how financial institutions combat credit card fraud. By blending machine learning, generative AI, and workflow automation, FraudShield AI delivers real-time, explainable fraud predictions with a fintech-inspired Streamlit interface. Powered by XGBoost, LangChain, LlamaIndex, LangGraph, and Groq's Llama-3.1-8b, this project showcases my ability to tackle high-stakes problems with a fusion of technical precision and creative flair. My vision was to build a system that not only detects fraud with high accuracy but also empowers users with transparent, human-readable insights, setting a new standard for trust and usability in fintech.

2 Problem Statement

Credit card fraud costs billions annually, and traditional detection systems often struggle with imbalanced datasets, opaque predictions, and slow processing. Financial institutions need a solution that:

- Detects fraud in real-time with high precision and recall.
- Provides clear, actionable explanations for predictions, especially for borderline cases.
- Scales to handle high-volume transactions and complex workflows.
- Offers an intuitive, engaging interface for analysts and stakeholders.
- Ensures compliance with banking regulations through transparency and auditability.

With FraudShield AI, I aimed to address these challenges by creating a hybrid system that combines robust machine learning, generative AI for explainability, and a modular workflow for scalability.

3 My Problem-Solving Style

My approach is a dynamic blend of bold experimentation, user-centric design, and iterative mastery. I tackle problems like a fintech alchemist, diving into cutting-edge tools, testing hypotheses with fervor, and refining solutions to balance innovation with reliability. For this project, I:

1. Experimented Fearlessly: Integrated XGBoost with LangGraph and LlamaIndex, iterating through failures to perfect the pipeline.
2. Designed with Empathy: Crafted a fintech-inspired UI to make complex fraud analysis accessible and engaging.
3. Refined Obsessively: Tuned ML models, prompts, and workflows to achieve an AUPRC of 0.87 and recall of 0.86.
4. Anticipated Scale: Built a Dockerized, modular system ready for enterprise deployment.
5. Learned Through Immersion: Mastered LangGraph, SMOTE, and FastAPI through documentation and hands-on prototyping.

4 Key Features

- **Real-Time Fraud Detection**: Processes single or batch transactions with XGBoost, achieving 86% recall.
- **Explainable AI**: Uses LangChain and Groq's Llama-3.1-8b to generate human-readable explanations, with special handling for borderline cases.
- **Borderline Case Analysis**: Delegates uncertain predictions (probability 0.3–0.7) to LLMs for nuanced judgment.
- **Fraud Pattern Recognition**: Leverages LlamaIndex for semantic retrieval of fraud patterns from structured data.

- **Interactive Dashboard**: Streamlit UI with visualizations, risk badges, and confetti animations for fraud alerts. A demonstration video showcasing FraudShield AI's capabilities can be viewed here: GitHub Demo Video.
- **Scalable Workflow**: LangGraph orchestrates detection, pattern retrieval, and explanation generation.
- **Dockerized Deployment**: Fully containerized, hosted on Docker Hub for easy scalability.
- **Performance Metrics**: Achieves an AUPRC of 0.87 and recall of 0.86 on the Kaggle Credit Card Fraud Dataset test set.

5 Technical Implementation

- **Machine Learning Pipeline**:
  - Trained an XGBoost model on the Kaggle Credit Card Fraud Dataset, using SMOTE to address class imbalance.
  - Applied `RobustScaler` for transaction amount normalization, ensuring robust feature scaling.
  - Saved model and scaler artifacts with Joblib, with metadata (e.g., optimal threshold, top features) in JSON.
  - Achieved AUPRC of 0.87 and recall of 0.86, validated on test set.
- **Generative AI**:
  - Built a LangChain prompt chain with Groq's Llama-3.1-8b for fraud explanations, using chain-of-thought reasoning and Basel III guidelines.
  - Designed a separate prompt for borderline cases, requiring explicit fraud/legitimate verdicts.
  - Integrated LlamaIndex with `all-MiniLM-L6-v2` embeddings for vector-based fraud pattern retrieval.
- **Workflow Automation**:
  - Used LangGraph to create a stateful pipeline with nodes for detection, pattern retrieval, explanation generation, and error handling.
  - Defined a `FraudCheckState` schema for type-safe state management.
  - Implemented conditional edges for robust error handling.
- **Web Interface**:
  - Developed a Streamlit app with custom CSS for fintech styling (glass-morphism cards, gradient text, pulse animations).
  - Supported batch CSV uploads and single-transaction input with real-time feedback.
  - Visualized metrics (precision, recall, AUPRC) and transaction details with interactive widgets.
- **Backend and API**:
  - Built a FastAPI server to expose endpoints for transaction analysis.
  - Integrated FastAPI with Streamlit for seamless UI access.
- **Deployment**:
  - Dockerized the app with Python 3.10-slim, exposing ports 8000 (FastAPI) and 8501 (Streamlit).
  - Hosted on Docker Hub for public access, ensuring portability.
- **Core Technologies**:
  - XGBoost & Scikit-learn: Fraud prediction and preprocessing.
  - LangChain & Groq: Explainable AI with Llama-3.1-8b.
  - LlamaIndex: Fraud pattern indexing.
  - LangGraph: Workflow automation.
  - Streamlit & FastAPI: Interactive UI and API.

- Docker: Containerized deployment.

6 Challenges and Solutions

1. **Challenge**: Handling the highly imbalanced Kaggle dataset (0.2% fraud cases).
   Solution: Applied SMOTE for oversampling, tuned XGBoost hyperparameters, and optimized for AUPRC (0.87).
2. **Challenge**: Generating clear, compliant explanations for fraud predictions.
   Solution: Designed chain-of-thought prompts with Basel III references, validated through iterative testing with sample transactions.
3. **Challenge**: Scaling pattern retrieval for real-time analysis.
   Solution: Built a LlamaIndex vector store with compact embeddings, enabling fast semantic search.
4. **Challenge**: Ensuring robust workflow orchestration.
   Solution: Used LangGraph with error-handling nodes and conditional routing, ensuring pipeline reliability.
5. **Challenge**: Creating a professional, engaging UI.
   Solution: Implemented fintech-inspired Streamlit UI with custom CSS, animations, and risk visualizations, optimized for performance.

7 Impact

- **High Accuracy**: Achieved an AUPRC of 0.87 and recall of 0.86, detecting 86% of fraud cases with minimal false negatives.
- **Enhanced Transparency**: Explainable AI empowers analysts with clear, regulation-compliant insights, reducing manual review time.
- **User Engagement**: Fintech-inspired UI with interactive dashboards increases adoption among stakeholders.
- **Scalability**: Dockerized, LangGraph-based architecture supports enterprise-scale transaction volumes.
- **Skill Mastery**: Self-learned XGBoost, LangGraph, LlamaIndex, and FastAPI, preparing me for advanced fintech roles.
- **Portfolio Strength**: A flagship project showcasing my expertise in ML, generative AI, and full-stack development.

8 Evaluation Metrics

- **AUPRC (Area Under Precision-Recall Curve)**: 0.87, indicating excellent performance on the imbalanced dataset.
- **Recall**: 0.86, meaning 86% of fraudulent transactions were correctly identified, critical for minimizing financial losses.
- **Precision**: Balanced with high recall to reduce false positives, ensuring operational efficiency.

9 Lessons Learned

- **Hybrid AI**: Combining ML (XGBoost) with generative AI (Llama) maximizes accuracy and explainability.
- **Workflow Automation**: LangGraph streamlines complex pipelines, enabling scalability and modularity.
- **Imbalanced Data**: Techniques like SMOTE and AUPRC optimization are vital for fraud detection.
- **Prompt Engineering**: Precise prompts with domain context (e.g., Basel III) enhance LLM reliability.

- **UI Design**: Fintech applications demand polished, responsive interfaces to build trust.
- **Self-Learning**: Curiosity-driven prototyping and documentation unlocked mastery of advanced tools.

10 My Unique Problem-Solving Style

My problem-solving is a vibrant mix of curiosity, precision, and resilience. I approach challenges like a fintech trailblazer, dreaming big, iterating relentlessly, and grounding visions in practical outcomes. For FraudShield AI:

- **Chased Curiosity**: Explored LangGraph and LlamaIndex, mastering their nuances through experimentation.
- **Applied Precision**: Tuned models and prompts to achieve an AUPRC of 0.87, ensuring compliance and clarity.
- **Embraced Resilience**: Overcame pipeline failures by refining LangGraph workflows and error handling.
- **Balanced Innovation and Pragmatism**: Delivered a futuristic UI with a production-ready backend.
- **Learned with Passion**: Turned complex frameworks into allies, fueled by joy in discovery.

11 Future Enhancements

To productionize FraudShield AI:

- Migrate pattern index to Pinecone for distributed, scalable vector search.
- Implement Celery with Kafka for async batch processing of high-volume transactions.
- Cache predictions and LLM responses in Redis to reduce latency.
- Add OAuth2 authentication and role-based access for enterprise security.
- Integrate Prometheus and Grafana for real-time monitoring of AUPRC, recall, and system health.
- Fine-tune Llama-3.1 for fraud-specific explanations, reducing API costs.
- Optimize UI for mobile devices and low-bandwidth environments.

12 Conclusion

FraudShield AI is a pinnacle of my journey as a self-taught innovator, blending machine learning, generative AI, and fintech design to combat credit card fraud. With an AUPRC of 0.87 and recall of 0.86, it delivers accurate, transparent, and scalable fraud detection, setting a new benchmark for trust in financial systems. This project showcases my expertise in AI orchestration, explainable AI, and full-stack development, positioning me to drive transformative solutions in the fintech industry.