Project Synopsis

Generative Augmented Fraud Detection Framework

Candidate Name: [Your Name Here]

December 4, 2025

1 Introduction

Financial fraud detection is a critical challenge in cybersecurity, yet research is persistently hindered by the scarcity of high-quality, publicly available datasets due to strict privacy regulations (GDPR/PII). Traditional solutions often rely on static, anonymized datasets that lack the behavioral granularity required to detect complex, evolving fraud patterns.

To address this "data poverty," this project presents a **Generative Augmented Fraud Detection Framework**. Unlike standard statistical models, this system leverages **Generative AI (GenAI)**—specifically a **Conditional Tabular Generative Adversarial Network (CTGAN)**—to synthesize realistic, privacy-preserving financial transaction logs. This creates a self-contained "Fraud Laboratory" capable of training robust detection models without exposing sensitive real-world banking data.

2 Problem Statement

- **Data Scarcity:** Real financial fraud data is proprietary and legally restricted.
- **Class Imbalance:** Fraud occurs rarely ($< 1\%$), making standard classifiers biased toward the majority class.
- **Static Models:** Traditional rule-based systems fail to detect non-linear or novel "zero-day" attack vectors.

3 Objectives

1. To engineer a **Deep Generative Pipeline** using CTGANs that learns the high-dimensional joint probability distribution of transaction features to synthesize novel fraud samples.
2. To implement a **Hybrid Intelligence Strategy** that fuses deterministic domain heuristics (Rule Scores) with probabilistic Machine Learning (Random Forest) and Unsupervised Anomaly Detection (Isolation Forest).
3. To operationalize the framework via an interactive **Streamlit Dashboard**, enabling real-time sensitivity analysis and "Human-in-the-Loop" validation.

4 Methodology

The system operates on a linear "Generate-Augment-Detect" pipeline:

4.1 1. Deep Generative Augmentation

Instead of relying solely on statistical sampling, the system utilizes the **CTGAN (Conditional Tabular GAN)** architecture.

- **Generator (G):** A deep residual network that transforms latent noise into synthetic transaction rows.
- **Critic (D):** A discriminator that ensures the synthetic data is statistically indistinguishable from the baseline profiles.

This allows the system to "dream" new, complex fraud scenarios (e.g., high-value international transactions at off-peak hours) to enrich the training dataset.

4.2 2. Noise Injection & Engineering

To simulate production-grade data imperfections, a chaos engineering module injects missing values (2%) and statistical outliers (1%) prior to training. A heuristic engine then calculates a **Rule Score**, encoding domain knowledge (e.g., "Vampire Window" risk) directly into the feature space.

4.3 3. Hybrid Detection Model

The classification engine employs a dual-strategy:

- **Supervised:** A Random Forest classifier optimized via `RandomizedSearchCV` for maximum Recall.
- **Unsupervised:** An `IsolationForest` algorithm that detects geometric anomalies in the feature space, serving as a safety net for unknown attack patterns.

5 Technologies Used

- **Language:** Python 3.9+
- **Generative AI:** SDV (Synthetic Data Vault), PyTorch (CTGAN Backend)
- **Machine Learning:** Scikit-Learn (Random Forest, Isolation Forest), NumPy
- **Frontend:** Streamlit (Reactive Web Dashboard)

6 Expected Outcomes

The project delivers a fully functional, end-to-end fraud detection pipeline. Experimental results demonstrate that the **GenAI-Augmented Model** achieves a Recall of $> 96\%$, significantly outperforming baseline statistical models. The final deliverable is a deployable web application that allows stakeholders to simulate fraud scenarios and audit model decision-making in real-time.