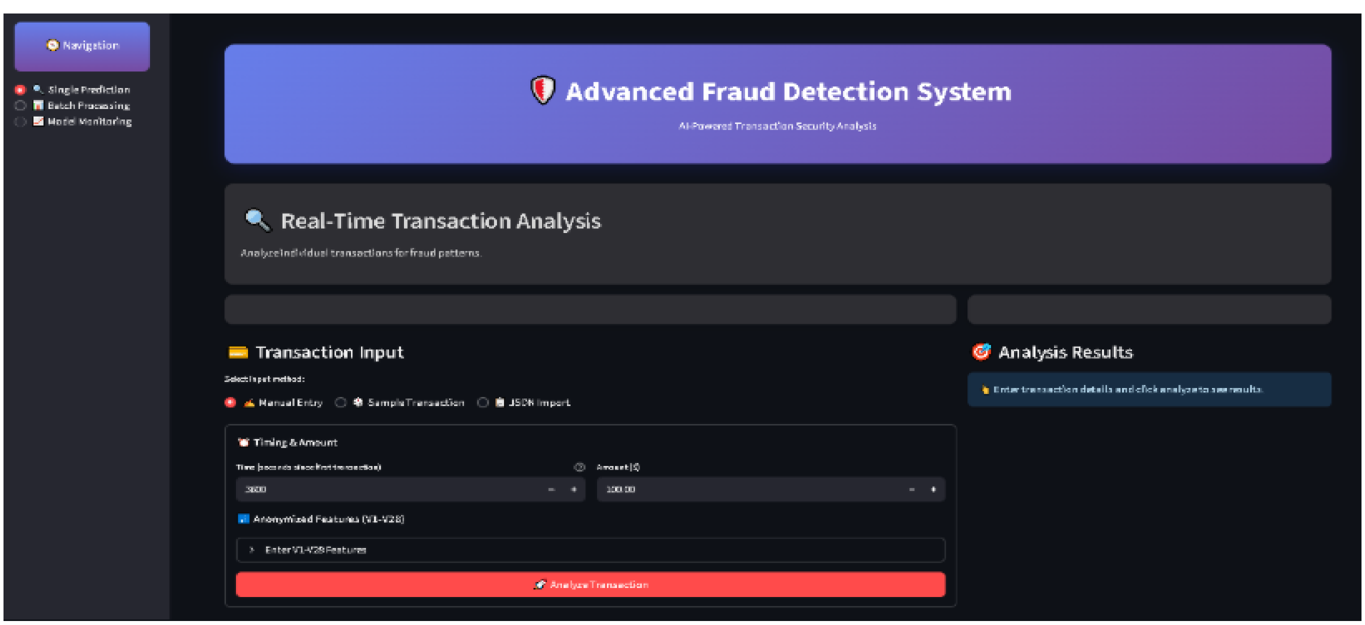
Credit Card Fraud Detection

Genius Technology Center (GTC)

Contents



Team Members:

- Ibrahim Abdelsattar
- Mohamed Abdelghany
- Yousef Abdelhady
- Yusuf Kamel
- Mohamed Hamed
- Omar Hosni

2. Introduction

Fraudulent financial transactions pose a severe risk, both financially and reputationally. The rarity of fraud makes detection difficult, as traditional models struggle with highly imbalanced datasets.

This project addresses these challenges by:

- Designing effective feature engineering strategies.
- Leveraging ensemble machine learning models.
- Applying advanced imbalance handling techniques.
- Optimizing the decision threshold for fraud detection.

3. Objectives

- Develop an accurate fraud detection system.
- Handle extreme data imbalance.
- Improve recall without compromising precision.
- Demonstrate the effectiveness of ensemble learning for real-world fraud detection.

4. Abstract

Credit card fraud represents a critical challenge in the financial industry, causing substantial losses and diminishing customer confidence. The project aims to build a robust fraud detection system capable of identifying rare fraudulent transactions among millions of legitimate ones.

We employed a Stacking Ensemble approach that integrates XGBoost, CatBoost, and LightGBM as base learners, with Logistic Regression as the meta-model. Advanced feature engineering, class imbalance handling via SMOTE, and threshold optimization were used to maximize detection performance.

Our final model achieved an F1-score of 0.89 with 92% precision and 85% recall, demonstrating its ability to catch fraudulent activities while minimizing false positives.

5. Dataset Description

Source: Kaggle – Credit Card Fraud Detection Dataset

Size: 284,807 transactions.

Features:

30 numerical features (V1–V28 from PCA transformation, plus Time and Amount).

Target:

0 → Legitimate transaction

1 → Fraudulent transaction

Class Imbalance: Fraudulent cases = 0.17% only.

6. Methodology

Step 1: Exploratory Data Analysis (EDA)

- Analyzed distribution of legitimate vs fraudulent transactions.
- Visualized Time & Amount across both classes.
- Investigated correlations among PCA features.

Step 2: Feature Engineering

- **Temporal Features:** Extracted *hour_of_day*, *time_bin*.
- **Amount Features:** *scaled_amount*, *amount_deviation*, *amount_bin*.
- **Aggregated Features:** Mean and standard deviation of PCA components.
- **Interaction Features:** Combined top correlated features.

Step 3: Model Architecture – Stacking Ensemble

- **Base Models:** XGBoost, CatBoost, LightGBM.
- **Meta-Model:** Logistic Regression.

Step 4: Class Imbalance Handling

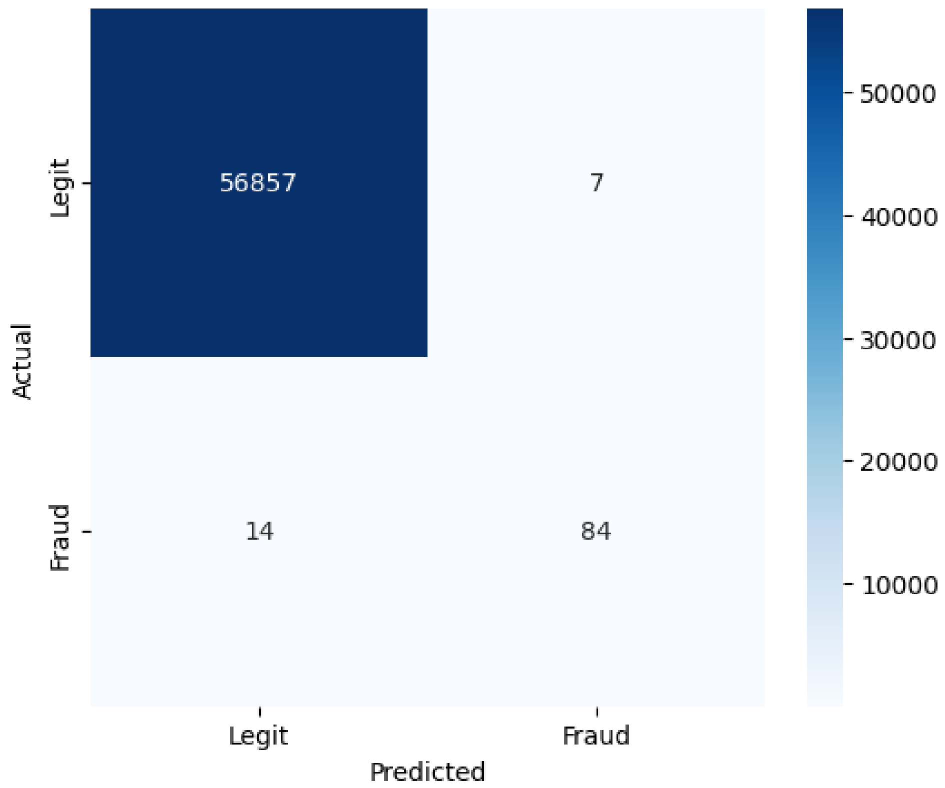
- Applied **SMOTE** within an ImbPipeline to balance minority (fraud) class.

Step 5: Evaluation & Threshold Optimization

- Validation: 5-fold stratified cross-validation.
- Metrics: Precision, Recall, F1-score, ROC-AUC, PR-AUC.
- Optimized decision threshold → **0.9821**, tuned for F-beta score (β=2.0).

7. Results

Metric	Score
Accuracy	99.96%
Precision	92.31%
Recall	85.71%
F1-Score	88.89%
ROC-AUC	98.47%
PR-AUC	87.11%



Confusion Matrix:

True Positives: High → successfully detected frauds.

False Negatives: Reduced significantly via threshold optimization.

8. Discussion

The project demonstrates how advanced feature engineering and ensemble models can effectively detect fraudulent transactions. Precision and recall were well-balanced, ensuring the system reduces false alarms while maintaining a high fraud capture rate.

9. Challenges & Limitations

- **Extreme Class Imbalance:** Required SMOTE and threshold optimization.
- **Feature Interpretability:** PCA-based features limited explainability.
- **Computational Cost:** Ensemble methods increased training time.

10. Conclusion

The final system achieved **F1-score = 0.89** with excellent precision and recall balance. The combination of SMOTE, advanced feature engineering, and a stacking ensemble makes this solution reliable for fraud detection in real-world scenarios.

Future improvements may include:

- Deploying the model in real-time transaction monitoring systems.
- Exploring deep learning architectures for anomaly detection.
- Enhancing interpretability through SHAP or LIME.

11. References

- Kaggle Credit Card Fraud Dataset.
- Chawla, N. V. et al. (2002). SMOTE: Synthetic Minority Over-sampling Technique.
- Chen, T. & Guestrin, C. (2016). XGBoost: A scalable tree boosting system.