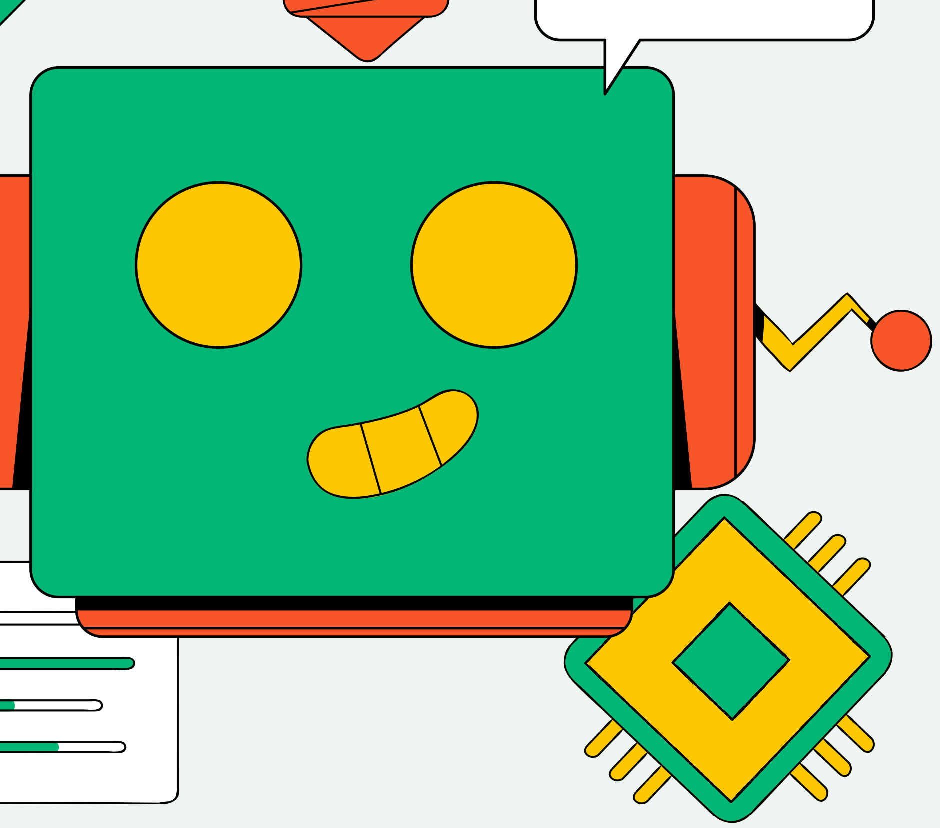
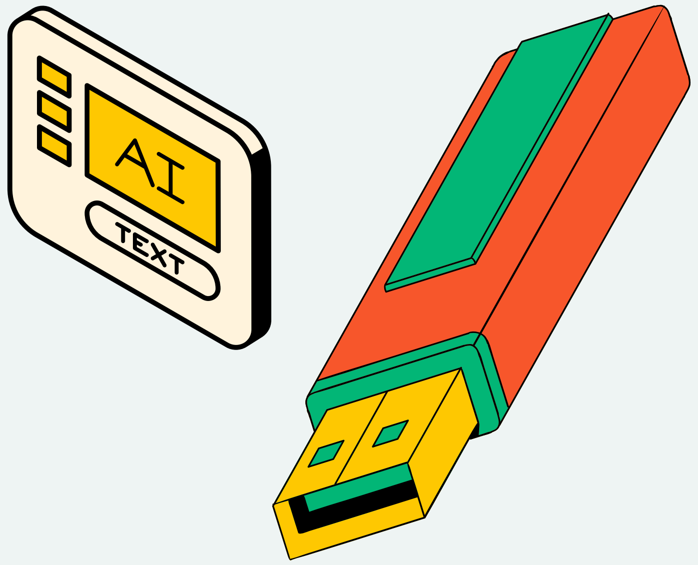




FRAUD DETECTION IN FINANCIAL TRANSACTIONS

PRESENTATION OUTLINE

1) INTRODUCTION

INTRODUCTION

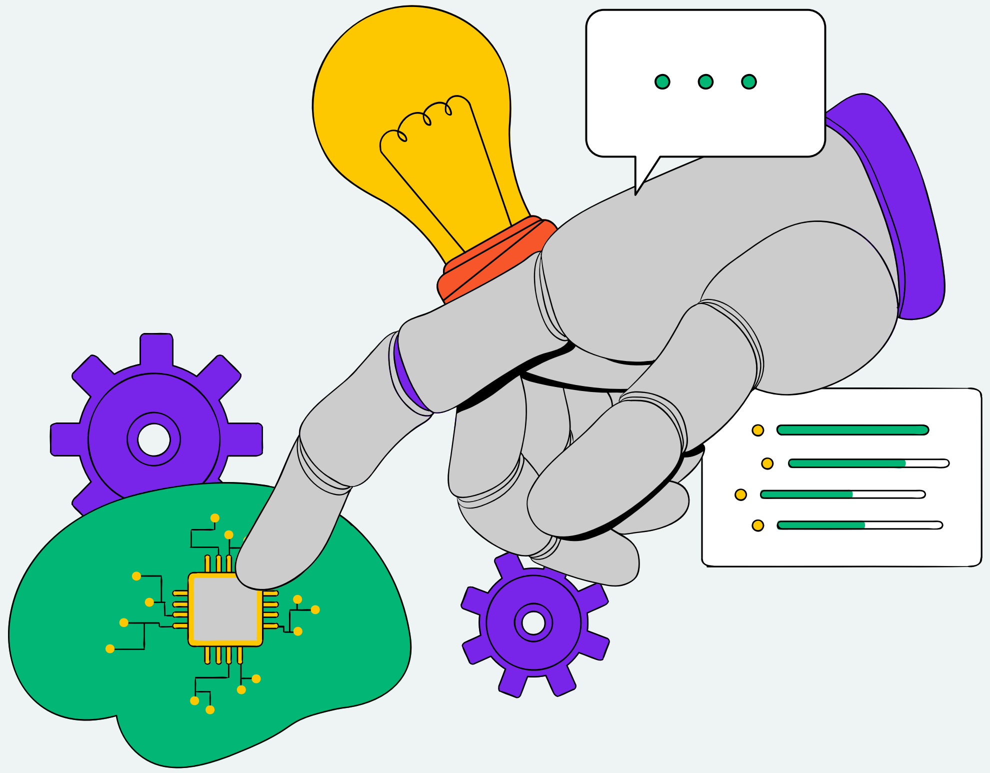
- Rapid growth of digital payments increases fraud risks
- Fraud detection is challenging due to extreme class imbalance
- Rule-based systems struggle to adapt to evolving fraud patterns
- Machine learning enables data-driven fraud detection
- This project compares supervised, unsupervised, and graph-based models for fraud detection

2) DATASET & PROBLEM SETTING

DATASET OVERVIEW

PaySim Mobile Money Fraud Dataset (Kaggle)

- Large-scale synthetic dataset derived from real mobile money transaction logs
- Simulates realistic financial activity over a 30-day period
- Preserves statistical properties of real-world transaction behavior
- Privacy-preserving and commonly used in fraud detection research



DATASET STRUCTURE

Each transaction includes:

- Temporal information: transaction time step
- Transaction details: type and amount
- Account balances: before and after the transaction (origin and destination)
- Fraud label: indicates whether the transaction is fraudulent

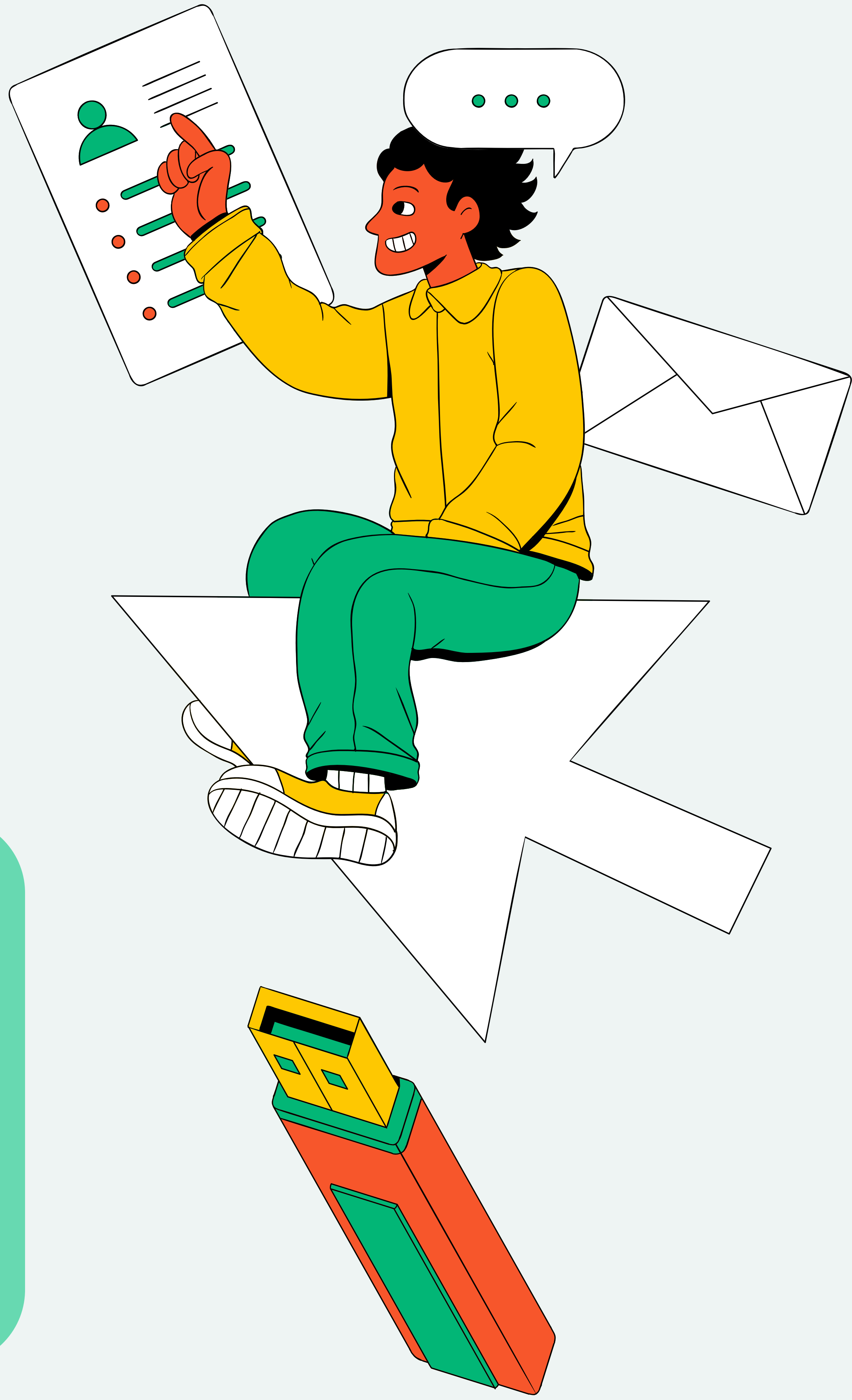
FRAUD DETECTION TASK

Objective

- Automatically identify fraudulent transactions in mobile money systems
- Binary classification: fraud vs. legitimate transactions

Key characteristics

- Fraud cases represent a very small portion of all transactions
- High cost associated with both false alarms and missed fraud



3) OVERALL METHODOLOGY (PIPELINE OVERVIEW)

END-TO-END PIPELINE

Overall workflow

1. Dataset acquisition and exploration
2. Data cleaning and preparation
3. Feature preprocessing
4. Model training (multiple paradigms)
5. Evaluation and comparison

All models share the same data preparation pipeline to ensure fair comparison.

DATA PREPARATION & PREPROCESSING

Data preparation

- Numeric conversion and consistency checks
- Duplicate removal
- Outlier mitigation using IQR-based clipping

Feature preprocessing

- Encoding transaction types
- Feature scaling
- Time-based train/validation/test split to prevent leakage

MODELING LAYER

Same preprocessed data used for:

- Supervised tabular models
- Unsupervised models (AE / VAE)
- Graph Neural Network (GraphSAGE)

Each approach captures different fraud characteristics.

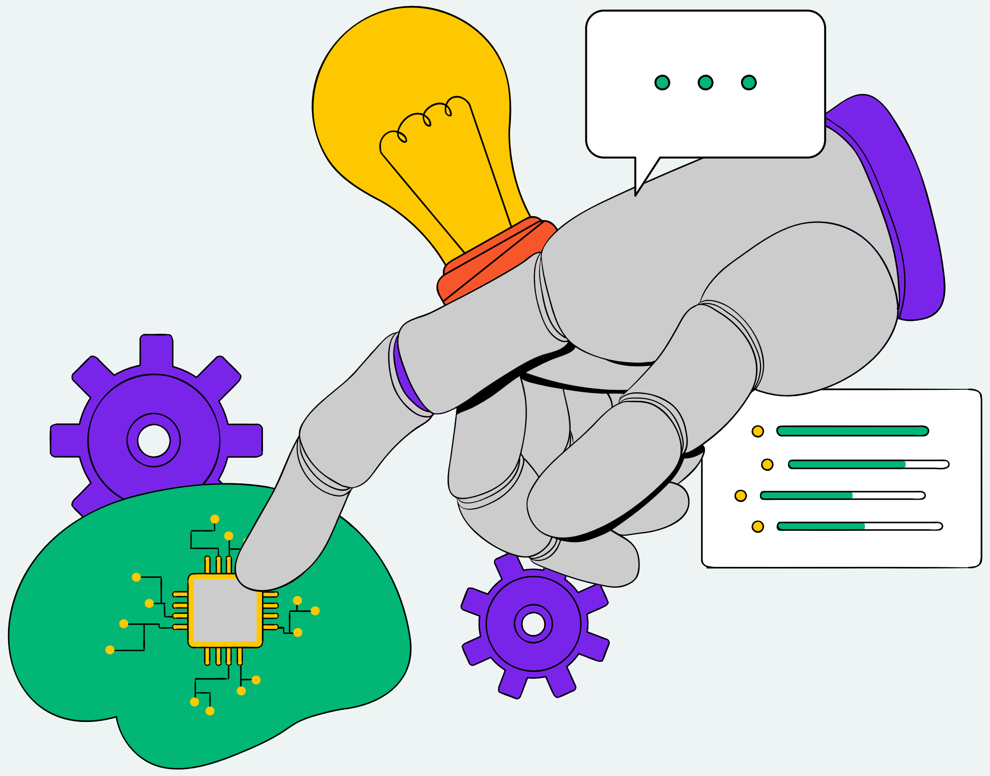
EVALUATION STRATEGY

Evaluation focus

- Fraud-class performance
- Precision, Recall, F1-score, AUC–PR

Goal

- Understand trade-offs between:
  - Detection accuracy
  - False positives
  - Model complexity



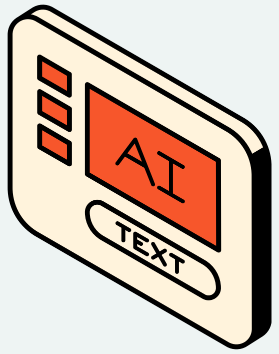
4) SUPERVISED TABULAR MODELS

4) MODELS - SUPERVISED

- Treat each transaction as independent
- Use fixed feature vectors
- Predict a probability of fraud per transaction

- Capture transaction networks
- They do not model: Transaction order, Transaction paths , Long-term behavior evolution
- They struggle to adapt to: New fraud strategies, New transaction patterns

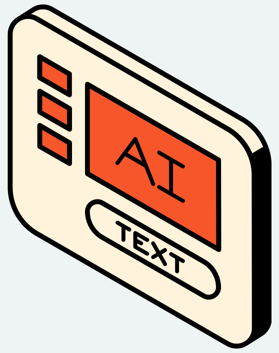
- Logistic Regression
- Random Forest
- XGBoost
- LightGBM



RESULTS

	Model	Accuracy	Precision (Fraud)	Recall (Fraud)	F1 (Fraud)
0	Logistic Regression	0.957112	0.476364	0.877054	0.617395
1	Random Forest	0.996422	0.981935	0.926354	0.953335
2	XGBoost	0.997599	0.947766	0.993914	0.970291
3	LightGBM	0.996350	0.919527	0.994522	0.955556

While Logistic Regression provides a useful baseline, ensemble tree-based models significantly outperform it in fraud detection. XGBoost and LightGBM achieve near-perfect recall, making them ideal for high-risk financial systems



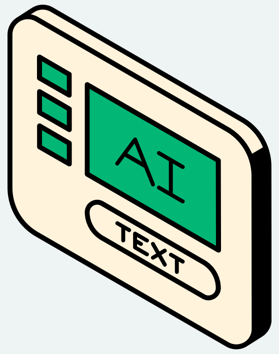
5) UNSUPERVISED ANOMALY DETECTION

5) MODELS – AE & VAE (UNSUPERVISED)

- **Autoencoder (AE):**
  - Neural network trained to reconstruct input
  - Trained using only normal transactions
  - Fraud detected by:
    - High reconstruction error

- **Variational Autoencoder (VAE):**
  - Probabilistic version of Autoencoder
  - Learns a distribution instead of fixed encoding

- **Training strategy:**
  - Only normal transactions used for training.



RESULTS – METRICS & VISUAL COMPARISON

WHY VAE IS BETTER FOR FRAUD DETECTION ?

- To decide what counts as " **fraud** " we set a threshold based on the normal transactions: only the top 0.5% of highest reconstruction errors are flagged as anomalies.
- Threshold (99.5 percentile) :

=== Autoencoder (Unsupervised) ===
Threshold: 0.003958569

	precision	recall	f1-score	support
0	0.9992	0.9955	0.9973	1270881
1	0.0938	0.3634	0.1491	1643
accuracy			0.9946	1272524
macro avg	0.5465	0.6794	0.5732	1272524
weighted avg	0.9980	0.9946	0.9962	1272524

=== Variational Autoencoder (VAE) ===
Threshold: 0.0058107376

	precision	recall	f1-score	support
0	0.9992	0.9953	0.9973	1270881
1	0.1033	0.4218	0.1659	1643
accuracy			0.9945	1272524
macro avg	0.5513	0.7085	0.5816	1272524
weighted avg	0.9981	0.9945	0.9962	1272524

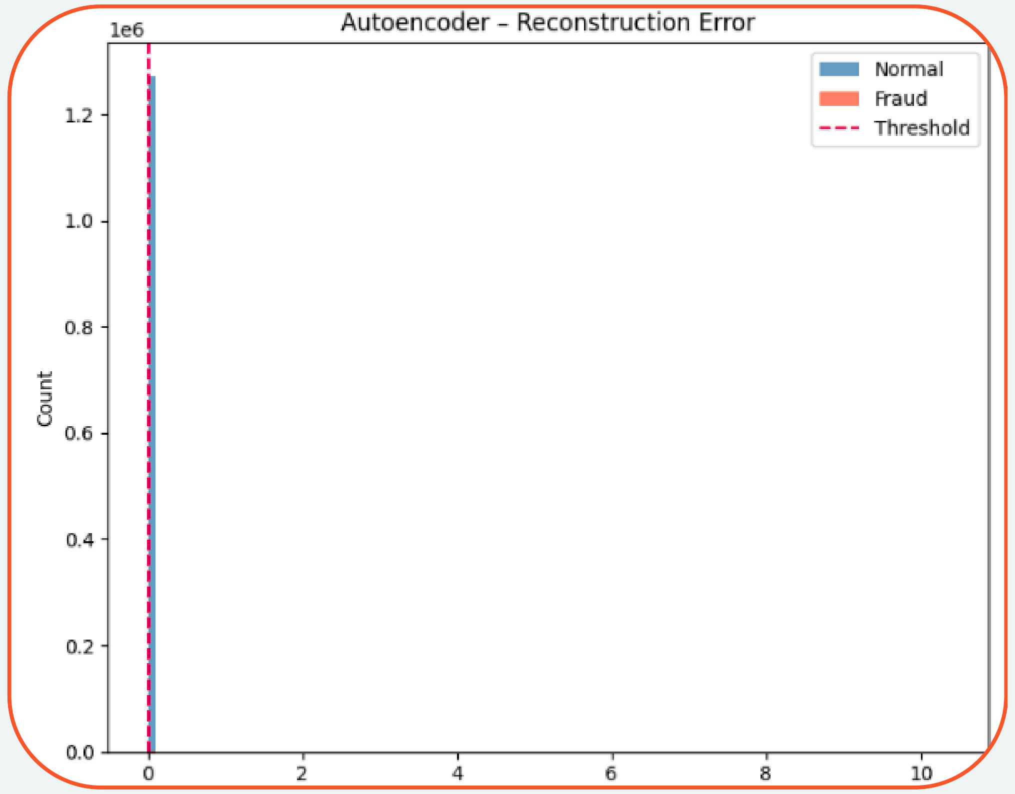
AE: 0.00396

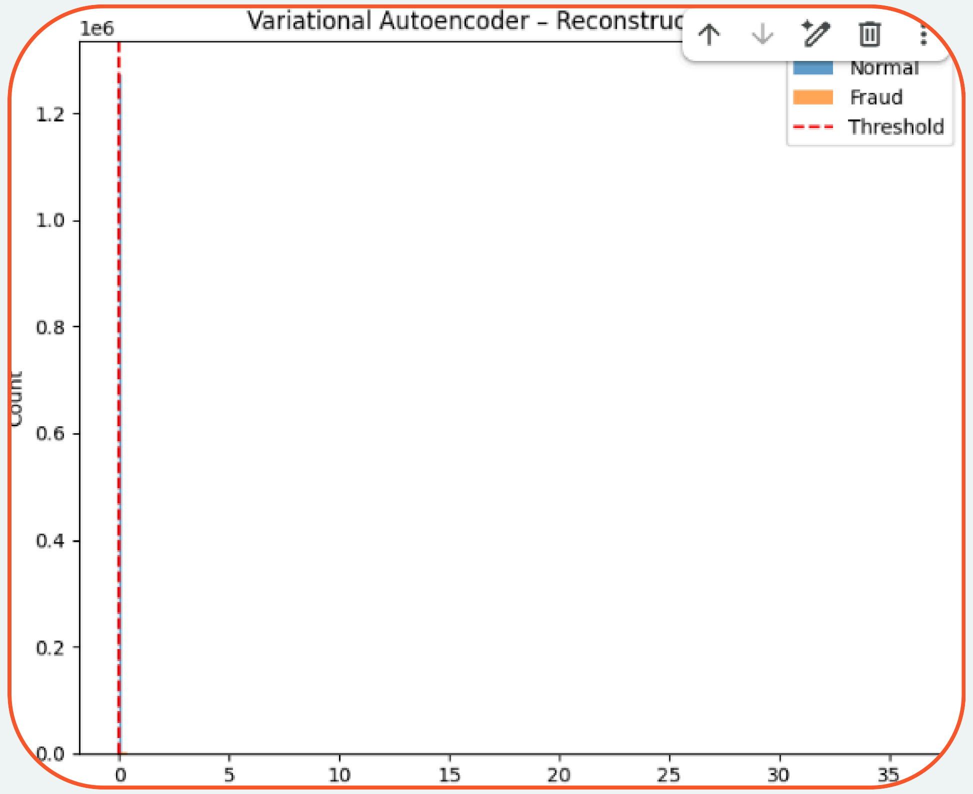
VAE: 0.00581

DISCUSSION & INSIGHTS

- **Key Observations**
  - Both AE and VAE learn normal transaction behavior successfully
  - VAE detects more fraud cases than AE
  - Reconstruction error is an effective anomaly score

- **Limitations**
  - Low fraud precision due to extreme class imbalance
  - Threshold selection strongly affects performance
  - Model does not explain why a transaction is fraudulent



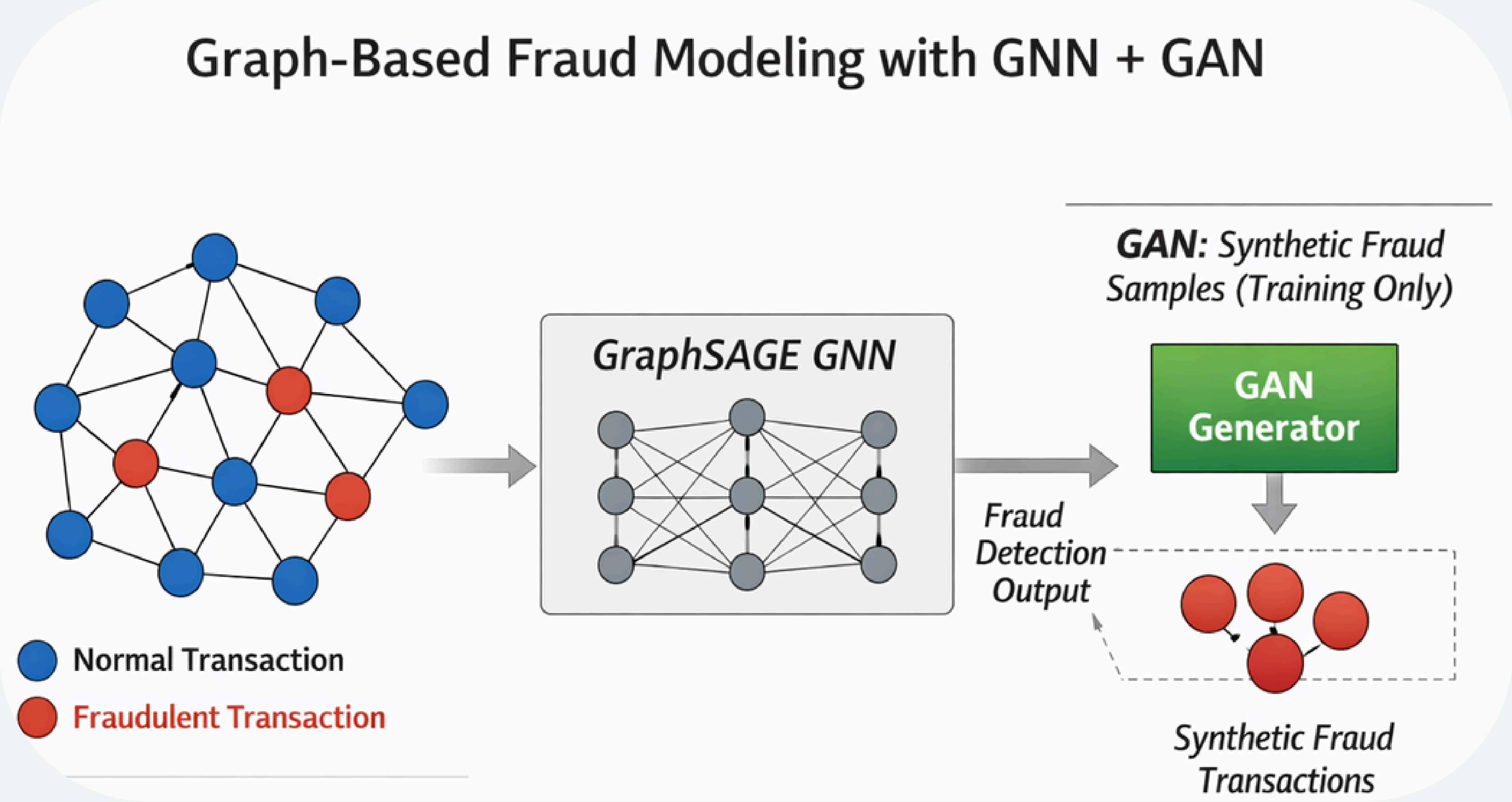


6) GRAPH NEURAL NETWORK WITH GAN AUGMENTATION

6) GRAPH NEURAL NETWORK

- Graph representation of transactions
- Nodes: transactions
- Edges: shared accounts / transaction relationships

```
Fraud distribution:
isFraud
0    6354407
1       8213
Name: count, dtype: int64
Fraud ratio: 0.0012908204481801 52
```



- Neighborhood aggregation: “summarize neighbors → get a richer transaction embedding”
- Mini-batch training (NeighborLoader): trains on small sampled neighborhoods instead of the full graph
- 2-layer setup: captures 1-hop and 2-hop relational patterns

GAN-BASED FRAUD AUGMENTATION

Problem: Fraud transactions are extremely rare

Our solution:

- Train a GAN using only fraud transactions
- GAN learns the distribution of fraudulent behavior
- Generates realistic synthetic fraud samples

```
Real training rows: 4072076
Fraud before synthetic: 5256

After adding synthetic fraud:
isFraud
0    4066820
1      45256
Name: count, dtype: int64
Total training rows: 4112076
New fraud ratio: 0.011005633164367585
```

How it is used:

- Synthetic fraud samples are added to the training data
- Helps the GNN see more diverse fraud patterns
- Improves learning in an imbalanced setting

RESULTS

Model behavior

- Performs well under extreme class imbalance
- Prioritizes fraud detection coverage over accuracy

Default decision threshold

- Favors high recall
- Detects most fraudulent transactions
- Produces a higher number of false positives

```
=== FINAL TEST RESULTS ===
AUPRC:      0.7137
Precision:  0.4401
Recall:     0.9704
F1-score:   0.6055
```

```
Best threshold: 1.0

=== OPTIMIZED TEST RESULTS ===
Precision: 0.7345
Recall:    0.9680
F1-score:  0.8352
```

Optimized decision threshold

- Improves precision while maintaining strong recall
- Provides a better operational balance
- More suitable for real-world deployment

7) COMPARISON OF MODELS

Unsupervised Models (AE / VAE)

- Detect unusual transactions based on reconstruction error
- Correctly identify some fraud cases
- Miss several fraud transactions that look similar to normal behavior

Graph Neural Network (GNN)

- Uses connections between transactions
- Detects fraud that appears across related transactions
- Identifies more fraud cases overall while keeping reasonable precision

Observed difference:

- Unsupervised models struggle with subtle fraud patterns
- GNN performs better when transaction relationships are included

8) CONCLUSION & FUTURE WORK

8) CONCLUSION

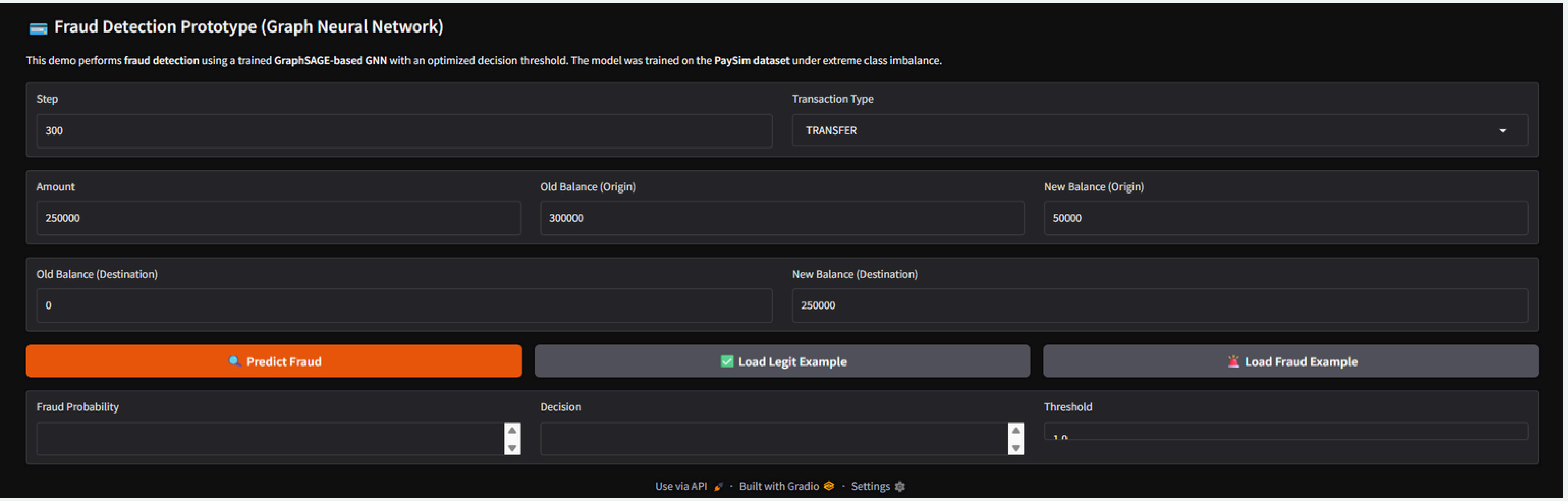
- Multiple fraud detection approaches were evaluated
- Supervised tabular models provide strong baselines
- Unsupervised models detect anomalies but may miss some fraud cases
- The GNN achieves a strong balance between precision and recall
- Graph-based modeling proves effective when transaction relationships are available

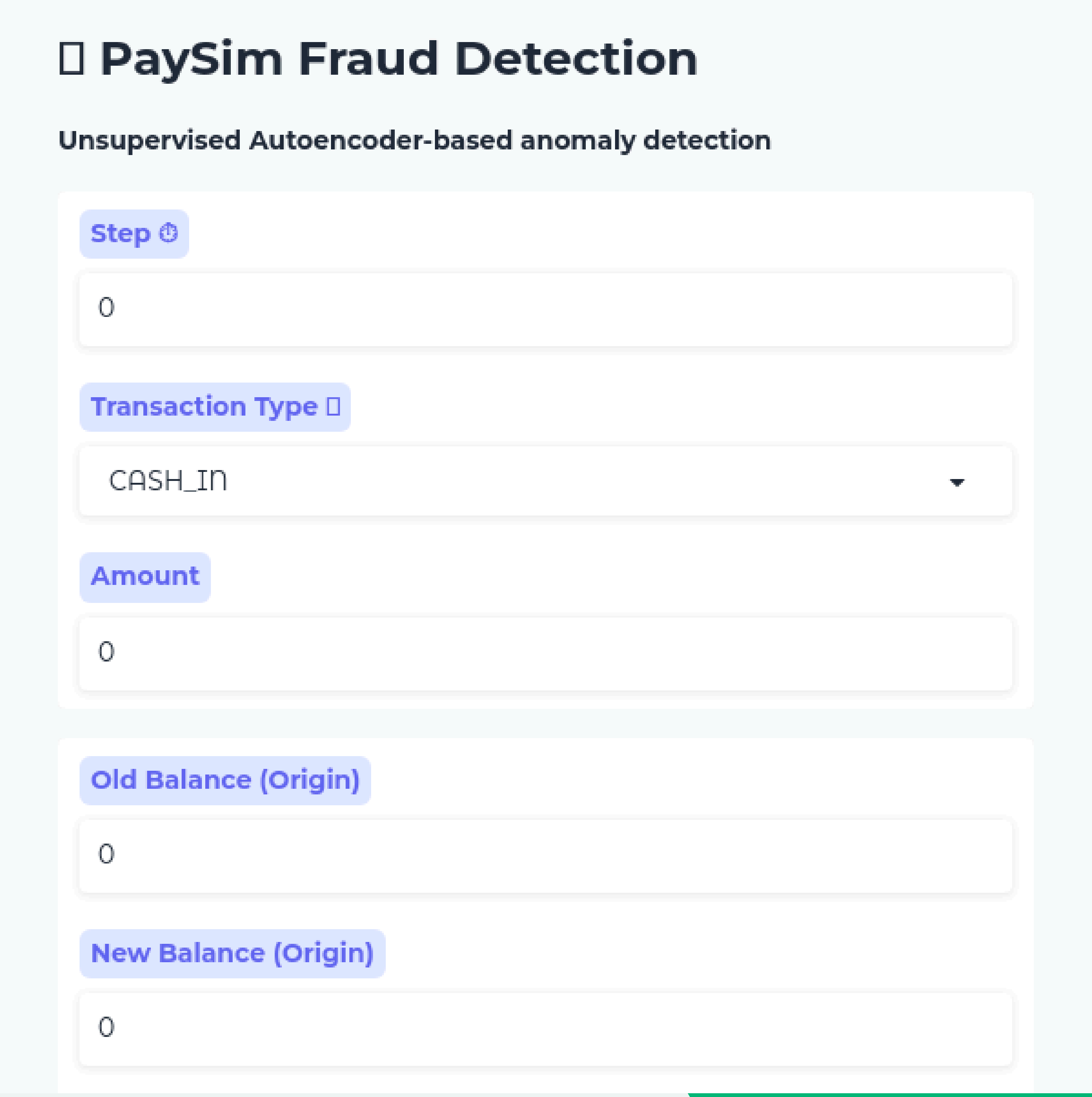
8) FUTURE WORK

- Extend to temporal or dynamic GNNs
- Evaluate on real-world datasets
- Add explainability for predictions
- Study real-time deployment feasibility

9) LIVE DEMO

LINKS:





https://huggingface.co/spaces/Houbrose/fraud-detection-demo

THANK YOU FOR YOUR ATTENTION