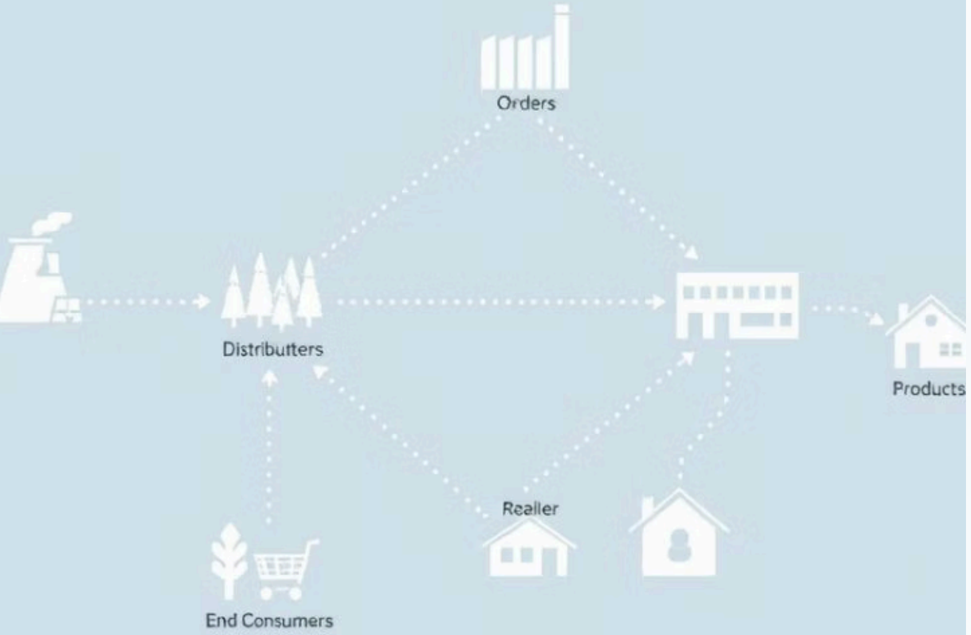


Graph Neural Networks for Supply Chain Fraud Detection with Explainability via Saliency Maps and RL-based Feature Masking

Aieh Eissa g00107537
Tsion regasa g00107807

6-05-2025

Table Of Content

Problem Identification

Intoduction

Supply chain fraud is subtle and relational.

Traditional Limitations

Rule-based systems miss subtle cross-entity fraud patterns.

Heterogeneous Graphs

Preserve customer, order, and product distinctions for context but usually lack explainability.

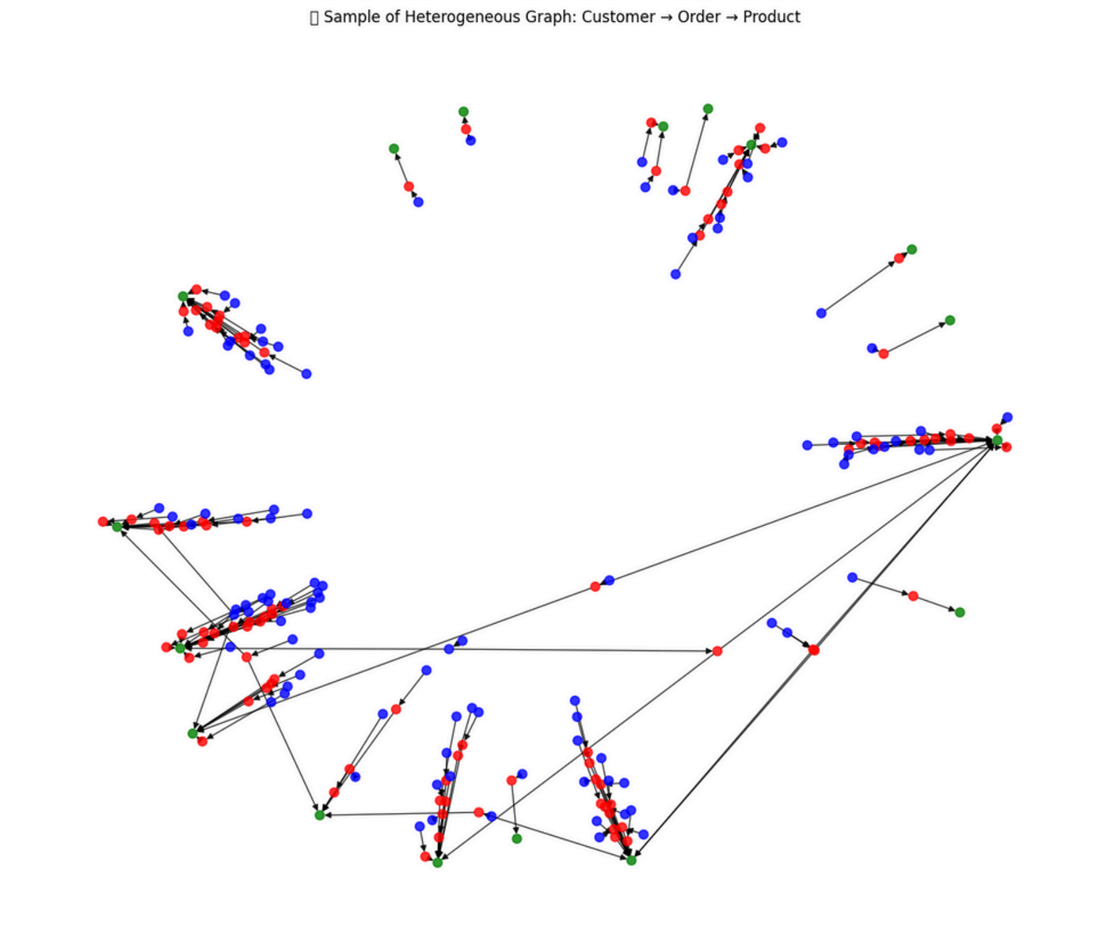
Objective

Detect fraudulent orders using a heterogeneous GNN and combining gradient saliency and RL-based feature masking for interpretability.

Data Preprocessing

- Dataset: Kaggle DataCo Supply Chain (synthetic, 18k+ rows
- Removed irrelevant/sensitive fields (e.g., email, password)
- Handled missing values (median for numerics, 'Unknown' for categoricals)
- Engineered new features: shipping delay, order hour, profit margin, discount flags
- Rule-based fraud labeling: 7.24% of data marked as fraudulent

Graph Construction



- Tripartite graph: Customers, Orders, Products
- Nodes: 2,072 customers, 9,351 orders, 1,859 products
- Directed edges customer → orders, orders → products
- Edges enriched with features: sales, profit, shopping delay, discount
- Preserved semantic structure
- Enabled detection of cross-entity fraud patterns

Model Architecture

Pytorch Geometric

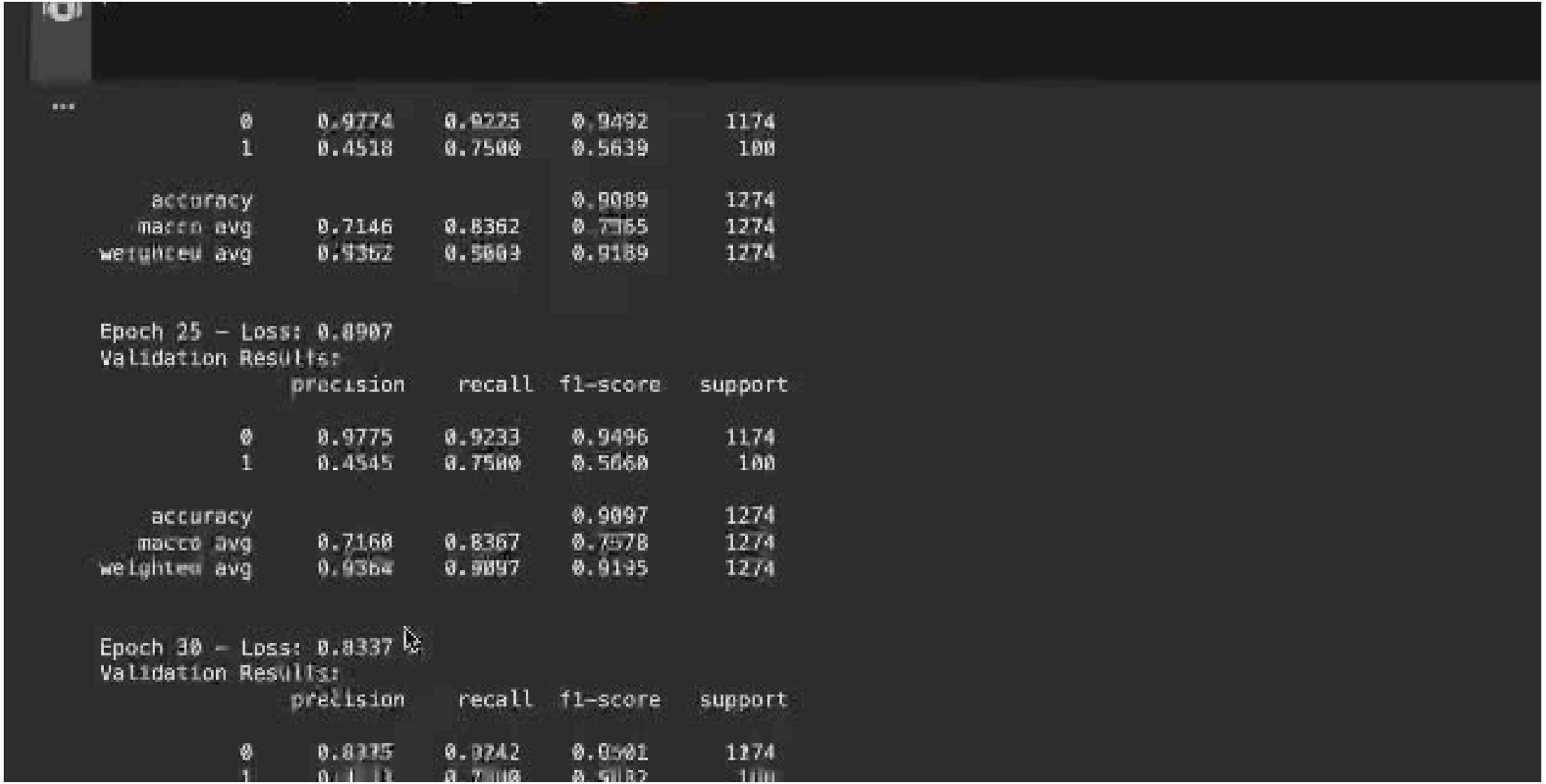
- Pytorch Geometric: Hetrogeneous GraphSAGE
- 2-layer-HetroConv: Distinct convolutions for edge types
- Layer 1: Customer → Order, Order → Product
- Layer 2: Product → Order (message passing)

Metrics

- Metrics: Accuracy, Precision, Recall, F1 (macro & weighted)
- Class-balanced binary cross-entropy loss

Training Set up

- 70/15/15 train/val/test split with stratified sampling
- Adam optimizer, LR = 0.005, weight decay = 5e-4
- Dropout = 0.3, evaluated every 5 epoch to monitor generalization and convergence


...
0 0.9774 0.9225 0.9492 1174
1 0.4518 0.7500 0.5639 100
accuracy 0.9089 1274
macro avg 0.7146 0.8362 0.7565 1274
weighted avg 0.9362 0.9089 0.9189 1274
Epoch 25 - Loss: 0.0907
Validation Results:
precision recall f1-score support
0 0.9775 0.9233 0.9496 1174
1 0.4545 0.7500 0.5660 100
accuracy 0.9097 1274
macro avg 0.7160 0.8367 0.7578 1274
weighted avg 0.9364 0.9097 0.9195 1274
Epoch 30 - Loss: 0.0337
Validation Results:
precision recall f1-score support
0 0.9775 0.9242 0.9501 1174

```
    Epoch 10, Reward: -0.6524, Mask sum: order: 7.00, customer: 5.00, product: 24.00
    Epoch 20, Reward: -0.5178, Mask sum: order: 8.00, customer: 5.00, product: 28.00
...  Epoch 30, Reward: -0.5975, Mask sum: order: 6.00, customer: 9.00, product: 25.00
    Epoch 40, Reward: -0.4421, Mask sum: order: 9.00, customer: 9.00, product: 28.00
    Epoch 50, Reward: -0.4381, Mask sum: order: 8.00, customer: 8.00, product: 26.00
    Epoch 60, Reward: -0.4473, Mask sum: order: 8.00, customer: 7.00, product: 32.00
    Epoch 70, Reward: -0.4437, Mask sum: order: 8.00, customer: 8.00, product: 32.00
    Epoch 80, Reward: -0.5264, Mask sum: order: 8.00, customer: 9.00, product: 30.00
    Epoch 90, Reward: -0.4645, Mask sum: order: 8.00, customer: 10.00, product: 29.00
    Epoch 0, Reward: -0.6152, Mask sum: order: 7.00, customer: 8.00, product: 23.00
    Epoch 10, Reward: -0.8264, Mask sum: order: 9.00, customer: 4.00, product: 24.00
    Epoch 20, Reward: -0.6770, Mask sum: order: 8.00, customer: 5.00, product: 28.00
    Epoch 30, Reward: -0.8022, Mask sum: order: 7.00, customer: 6.00, product: 28.00
    Epoch 40, Reward: -0.7161, Mask sum: order: 8.00, customer: 8.00, product: 32.00
    Epoch 50, Reward: -0.7279, Mask sum: order: 8.00, customer: 6.00, product: 29.00
    Epoch 60, Reward: -0.7051, Mask sum: order: 7.00, customer: 5.00, product: 31.00
    Epoch 70, Reward: -0.7692, Mask sum: order: 8.00, customer: 4.00, product: 30.00
```

Result Analysis

- Despite fraud comprising only 7.24% of the data, the model captured most of them.
- A healthy balance between catching fraud and avoiding false alarms.
- The model generalizes well across both fraud and non-fraud.
- The GNN successfully balances sensitivity and precision even in a highly imbalanced setting.

CLASS-WISE PERFORMANCE ON TEST SET

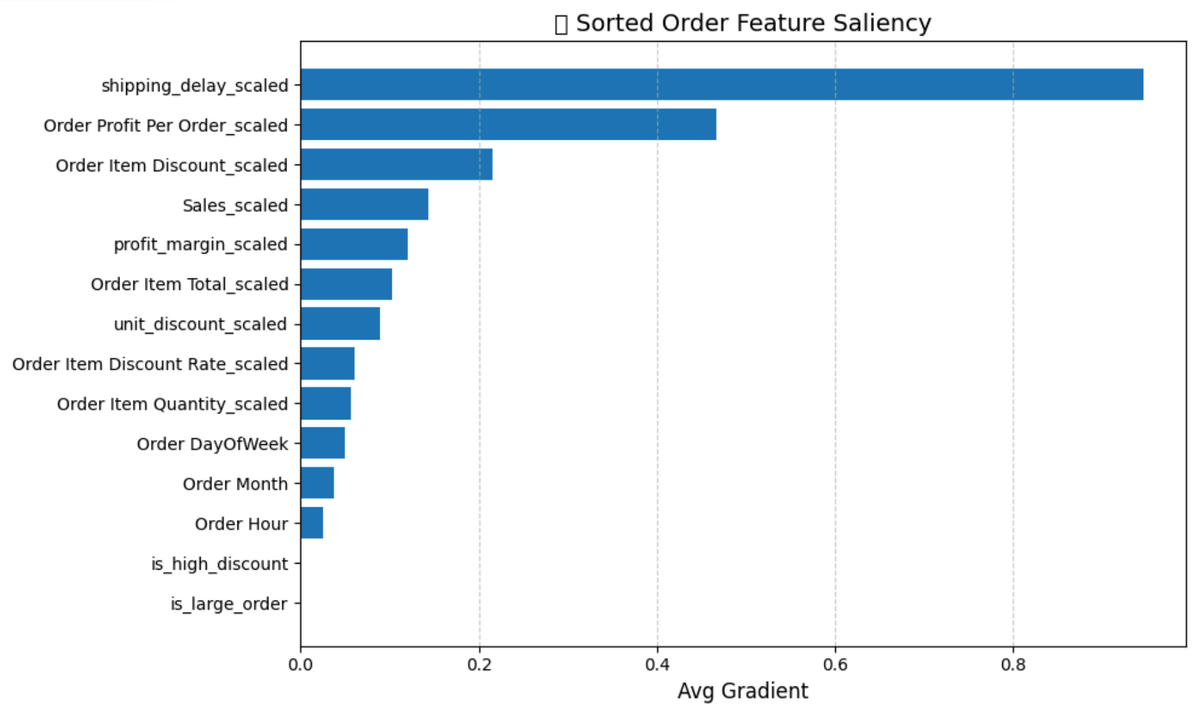
Class	Precision	Recall	F1-score	Support
Non-Fraud (0)	97.75%	99.15%	98.45%	1,182
Fraud (1)	86.84%	70.97%	78.11%	93

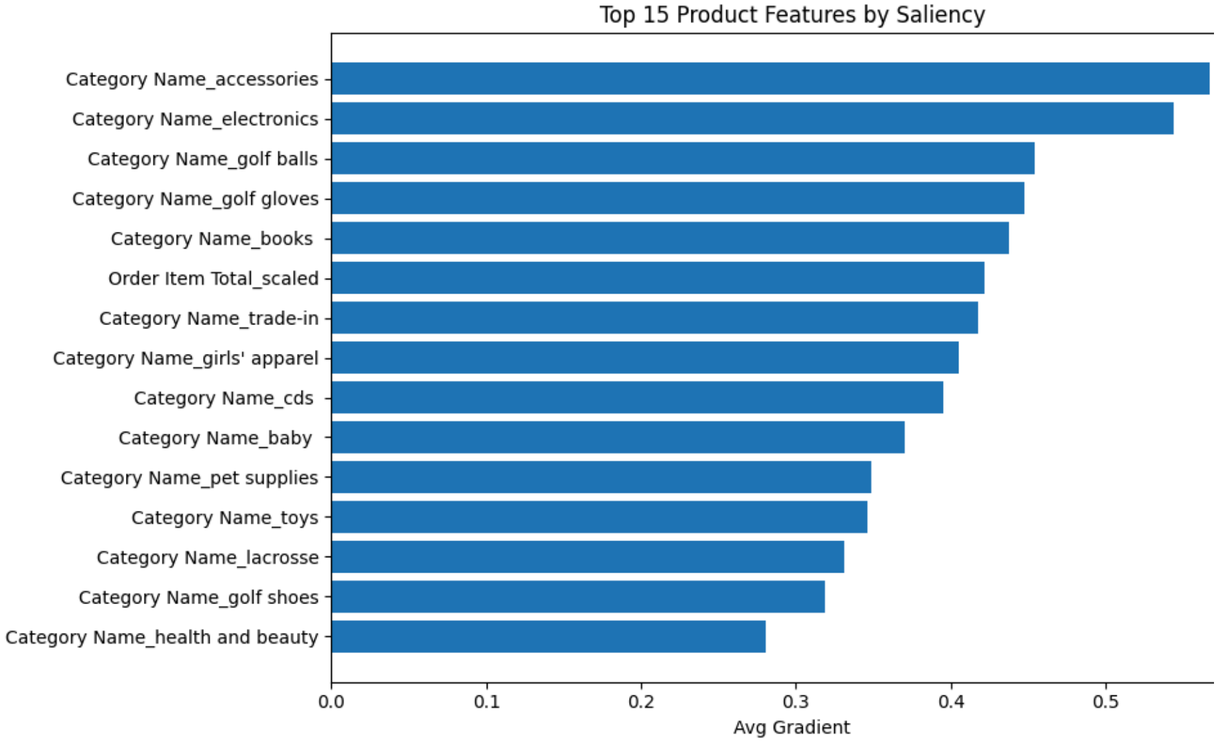
AGGREGATE METRICS ON TEST SET

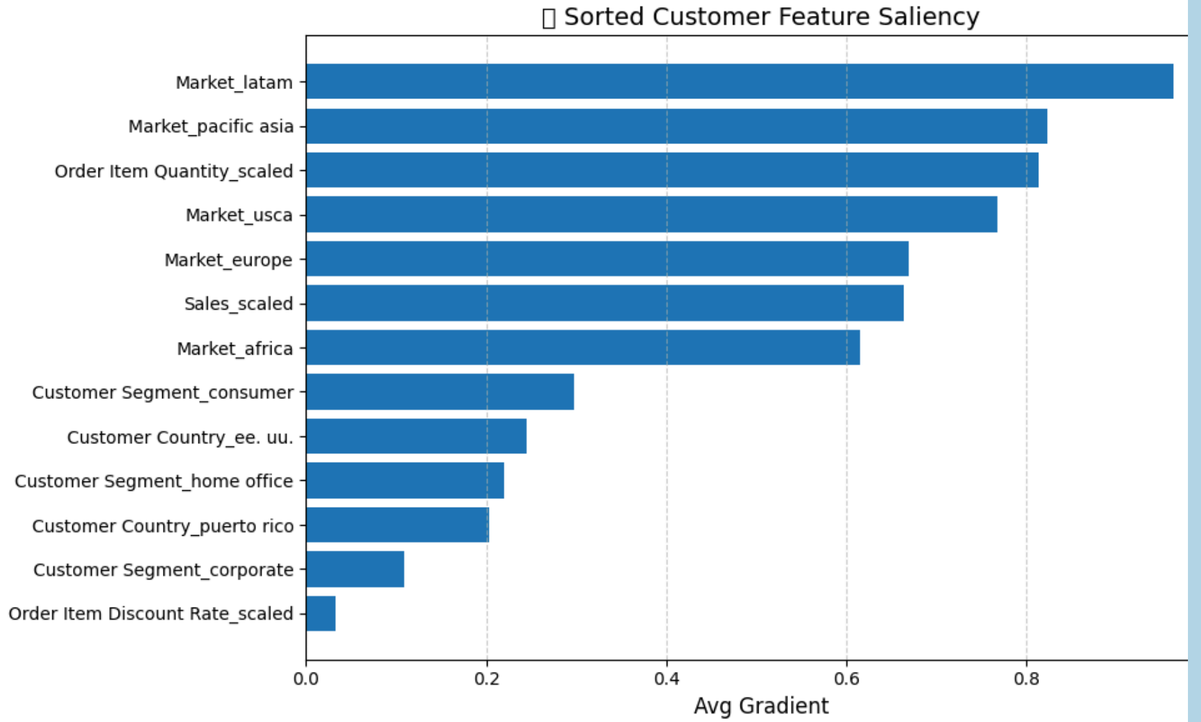
Metric	Score
Accuracy	97.10%
Macro Avg F1	88.28%
Weighted Avg F1	96.96%

Explainability: Gradient-based Saliency Analysis

- The saliency analysis reveals what the model pays attention to when flagging fraud.
- Order nodes: Features like shipping delays, high profit margins, and discount rates stood out — suggesting fraud is often tied to fulfillment or pricing anomalies.



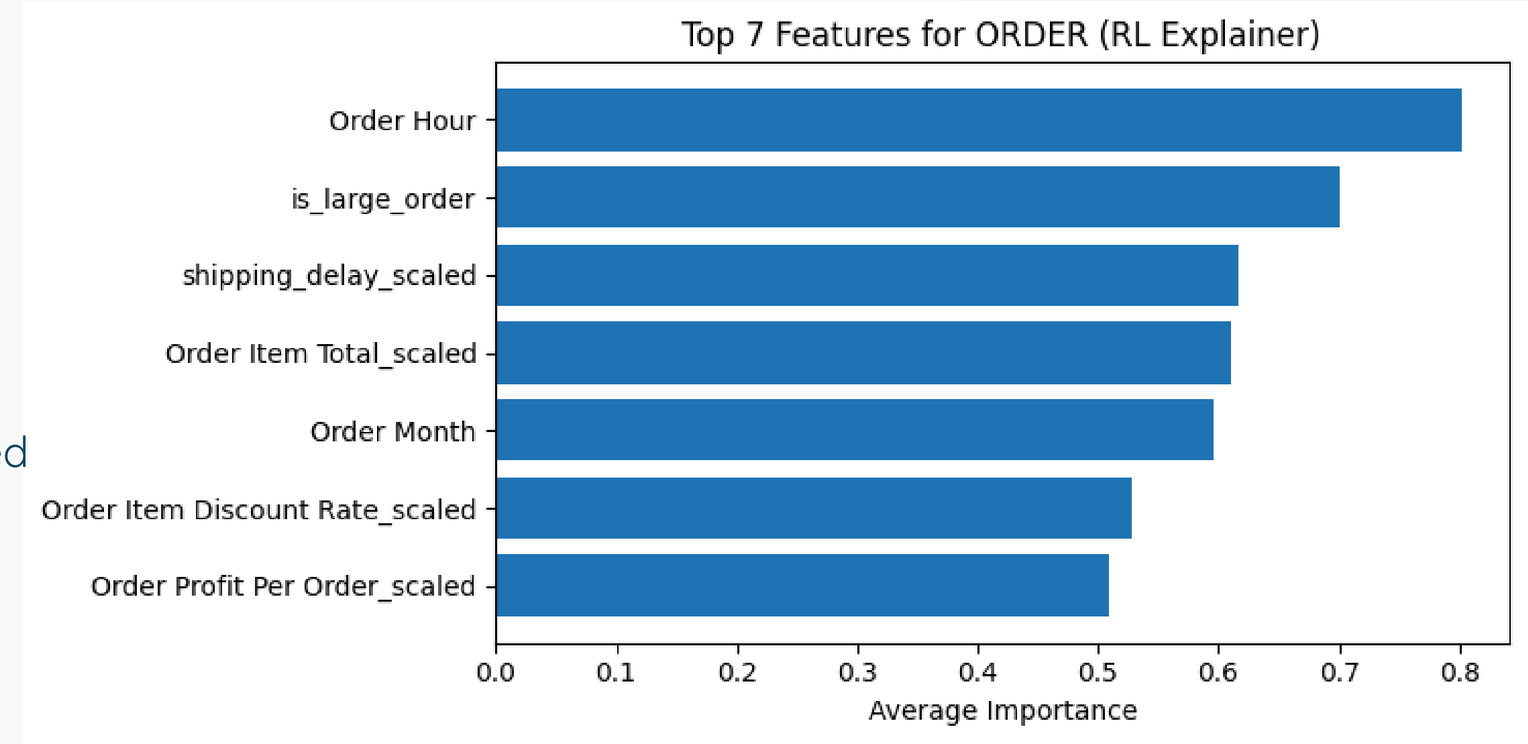




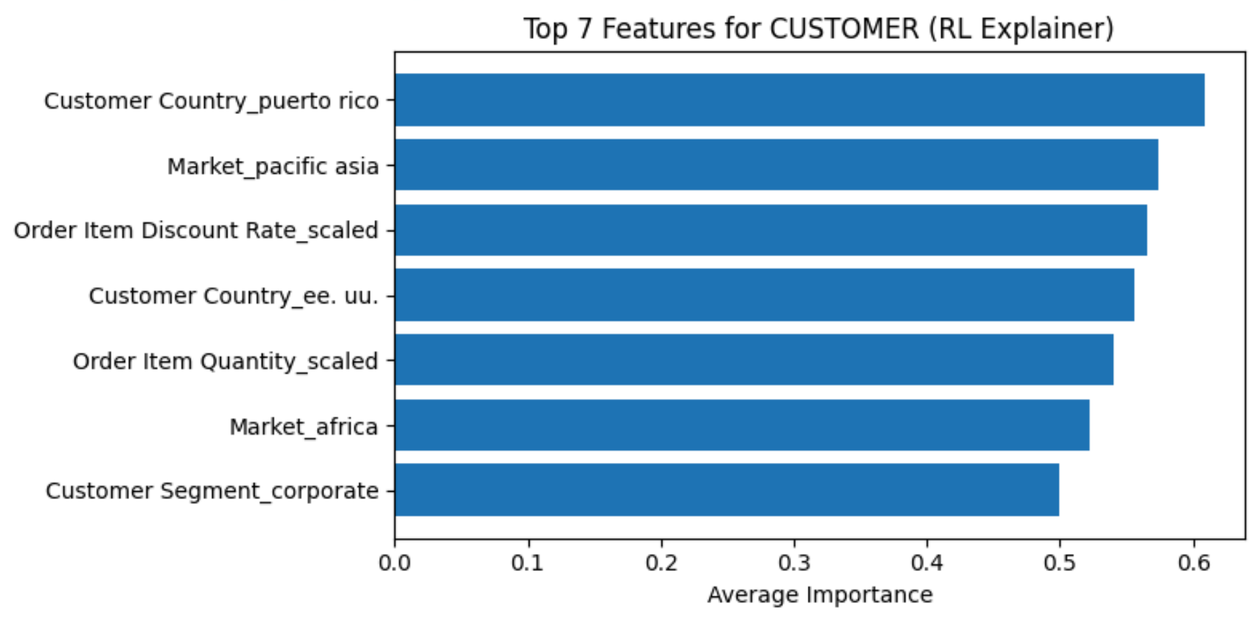
- Customer nodes: Regions such as Pacific Asia and Latin America showed higher saliency, pointing to potential region-specific patterns.
- Product nodes: Categories like electronics, golf gear, and accessories were among the most influential.
- These insights show the model is focusing on plausible, interpretable signals rather than noise.

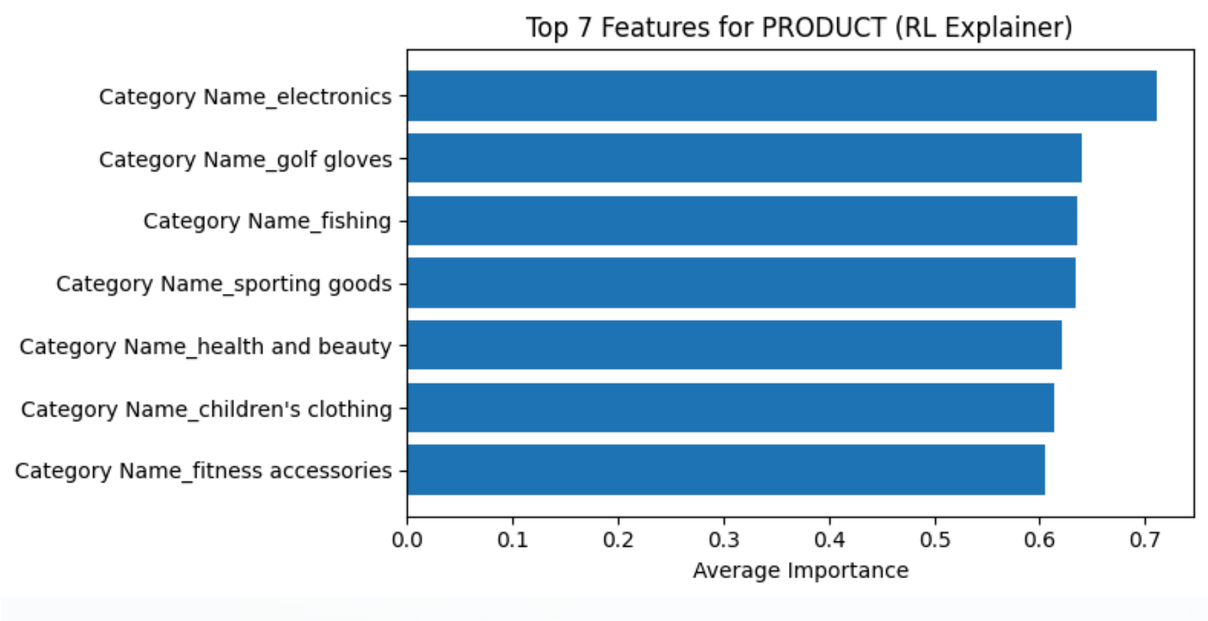
Explainability: Reinforcement Learning-Based Feature Masking

- The RL-based explainer generates sparse masks identifying the minimal features needed to preserve the model's fraud prediction.
- Order: features like order hour, is_large_order, and shipping_delay_scaled were most critical.



- Customers: geographic tags like Customer Country_puerto rico and regional indicators had high influence.
- For products, categories such as electronics, golf gloves, and fishing items dominated.
- This sparse, human-readable feature selection complements saliency and adds auditability to the pipeline





Comparative Insights

1

Saliency

Saliency offers global explanations by measuring gradient sensitivity. It highlights general patterns the model has learned over many predictions.

2

RL Explainer

RL-based explainer delivers sparse, instance-specific explanations by identifying the minimal feature set needed to preserve a prediction.

- The two methods sometimes agree (e.g., discount rate, profit margin) but also differ—RL highlights local anomalies that may be diluted in global averages.
- Together, they offer layered insights: saliency for model auditing, RL for transaction-level justification.

Discussion

Limitation

- Synthetic data and heuristic labels limit real-world generalizability.
- Class imbalance poses challenges for stability.
- Most explainability tools (e.g., GNNExplainer, PGExplainer) are incompatible with heterogeneous graphs.
- RL and saliency methods used here offer a workaround but are still post-hoc and sensitive to design.

Conclusions

- Addresses the need for accurate and interpretable fraud detection in structured supply chain environments.
- Employed heterogeneous GraphSAGE model and demonstrated its ability to capture subtle fraud cues across customer, order, and product interactions.

- The explainability strategy using both saliency and reinforcement learning adds interpretability to every prediction.
- Unlike previous models that focused either on black-box GNN performance or interpretable shallow learners, our model combines high detection accuracy (97.1%) with transparency.

The promise of integrating interpretable GNNs for supply chain risk monitoring in a scalable and insightful way.

References

- Wu et al., MultiFraud: A Heterogeneous Graph Neural Network for Financial Fraud Detection, 2024.
- Cui et al., FraudGNN-RL: Reinforcement Learning Enhanced Graph Neural Networks for Supply Chain Fraud Detection, 2025.
- Gandhi et al., Graph Convolutional Networks for Financial Fraud, 2025.
- Xie et al., DAST-GNN: Dual-Attention Spatiotemporal GNN for Financial Irregularities, 2024.
- Ying et al., GNNExplainer: Generating Explanations for Graph Neural Networks, NeurIPS 2019.
- Liu et al., PGExplainer: Probabilistic Graph Explainer, NeurIPS 2020.
- Vu & Thai, RELEx: Reinforcement Learning-Based Explanations for GNNs, ICLR 2022

Thank you