LIVE

Fraud Analytics: BT4012

Group Member: Dai Yang, Luo Haonan, Luoqi Xu, Yang Leyuan, Zhu Xiaoxiao

Problem Statement

In the wake of the digital transformation of the banking industry, the shift to digital currencies has led to a rise in security challenges, notably an increase in digital fraud and privacy breaches.

This research addresses the vulnerabilities within the Ethereum blockchain, particularly the prevalence of phishing scams, by leveraging advanced machine learning techniques, such as XGBoost and Random Forest.

Project Objective

To detect and classify phishing accounts within the Ethereum network, leveraging the Blockchain nodes and transactions.

Primary Challenge:

1. Classify an Ethereum account (node) as phishing or non-phishing.
2. Evaluate the model's ability to generalize across the vast number of unlabeled nodes in the dataset.
3. Predict behaviors exhibited by phishing nodes to preemptively detect newer phishing schemes.

Expected Outcome:

- Detect existing phishing nodes with high accuracy
- insights drawn from this analysis could potentially guide and inform anti-phishing strategies across other blockchain platforms.

Dataset

The dataset has been meticulously curated from the Ethereum blockchain, with phishing nodes sourced from the Etherscan labeled cloud.

Key Features

Nodes: Representing individual Ethereum accounts, the dataset encompasses 2,973,489 nodes, of which 1,165 are labeled as phishing nodes.

Key Attributes

1. Node Identifier: Unique Ethereum address identifying each node.
2. ISP (Indicator of Suspicious Activity): Binary attribute indicating whether a node is associated with phishing (1) or not (0), forming the basis for our classification model.

Dataset

The dataset has been meticulously curated from the Ethereum blockchain, with phishing nodes sourced from the Etherscan labeled cloud.

Key Features

Nodes: Representing individual Ethereum accounts, the dataset encompasses 2,973,489 nodes, of which 1,165 are labeled as phishing nodes.

Key Attributes

1. Node Identifier: Unique Ethereum address identifying each node.
2. ISP (Indicator of Suspicious Activity): Binary attribute indicating whether a node is associated with phishing (1) or not (0), forming the basis for our classification model.

Edges: A total of 13,551,303 edges have been documented.

Key Attributes

1. Transaction Amount: Represents the monetary value of each transaction between nodes.
2. Timestamp: The exact moment each transaction was executed, which can be critical for analyzing transaction patterns over time.

Sections

01 Data Preprocess

- Derive fundamental features
- EDA
- Data Balancing

02 Feature Engineering

- Generate aggregated features
- Feature scaling and normalisation
- Feature selection

03 Machine Learning Models

- KNN, RF, XGB, SVM
- GCN, GNN

04 Conclusion

- Business Usage
- Limitation
- Conclusion

02

Data Preprocessing

Fundamental Features Generation

1. Node Characteristics
2. Node Network & Component Characteristics
3. Edges Features

Node Characteristics

From the networks relationships between nodes, following features have been generated:

Features	Definition & Importance
In_degree	Incoming degrees of the nodes. Identify nodes' transactional relationships.
Out_degree	Outgoing degrees of the nodes. Identify nodes' transactional relationships.
PageRank	PageRank Score of the nodes. Evaluates importance based on transactional connections.
Weights_out	Sum of outgoing transactions amount of the nodes. Sum of transaction amounts for analysis.
Weights_in	Sum of incoming transactions amount of the nodes. Sum of transaction amounts for analysis.
Num_out	Total number of outgoing transactions of the nodes.
Num_in	Total number of incoming transactions of the nodes.
clustering_coefficient	Measures connections among a node's neighbors. Detects clusters or unusual node connections.
closeness_centrality	Proximity of a node to all others. Identify potential anomalies.
betweenness_centrality	Node's influence on information flow. Detects nodes crucial for transaction flow.
eigenvector_centrality	Importance based on connected high-scoring nodes. Highlights influential nodes in the network.

Network & Component Characteristics

Network Characteristics

Features	Definition & Importance
num_connected_components	Identifies isolated clusters, potentially highlighting segregated fraudulent activities.
network_density	Indicates overall interconnectivity, revealing potential areas of dense or sparse interactions that might signal fraud.
avg_path_length	Average distance between nodes reveals network efficiency and connections.

Component Characteristics

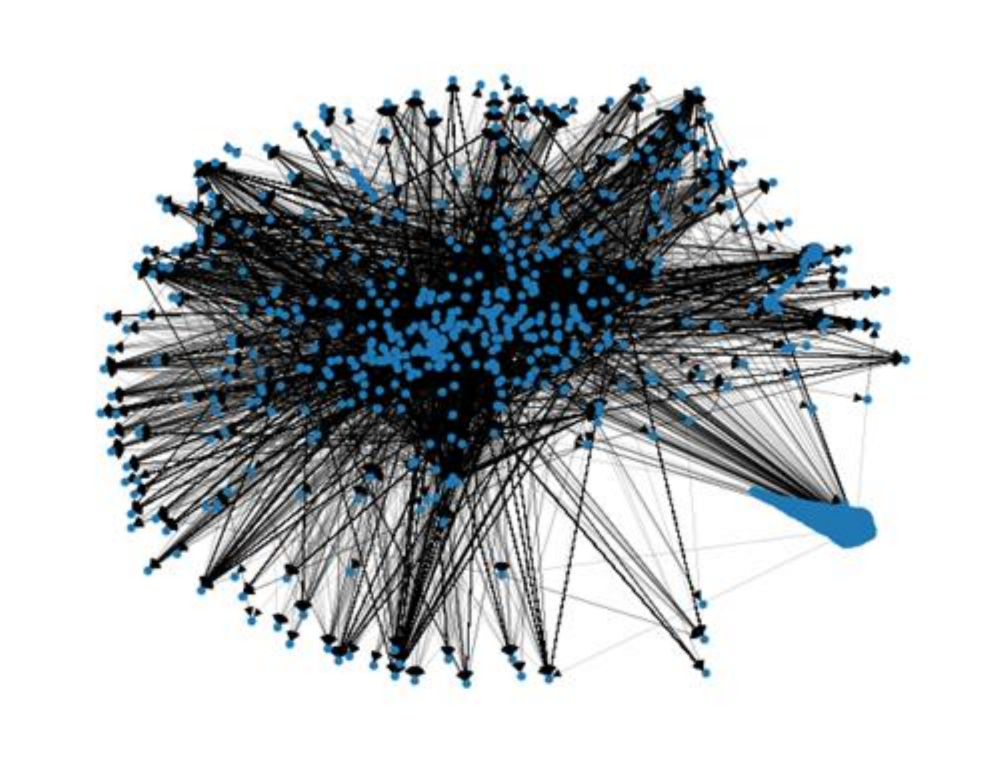
Features	Definition & Importance
component_size component_diameter component_eccentricity component_average_degree component_clustering_coefficient	Provide a nuanced understanding of component structures. Anomalously large diameters, high clustering coefficients in smaller components, or irregularly high average degrees could suggest potential fraudulent activities within those specific segments.

Edges Features

Features	Definition & Importance
fromnode	The identifier of the account initiating the transaction
tonode	The identifier of the account receiving the transaction
timestamp	The date and time when the transaction was executed

Nevertheless, the current fundamental features extracted from graph are insufficient for effectively identifying fraudulent nodes. The next step involves exploring additional graph features to enhance fraud detection capabilities.

Exploratory Data Analysis



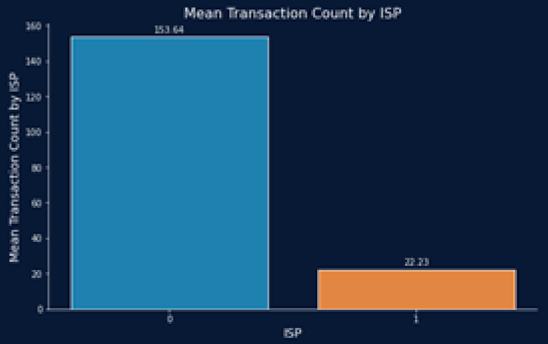
Strongly Connected Components:

- Found 15,704 components.
- Each node is reachable from any other via a directed path.

Weakly Connected Components:

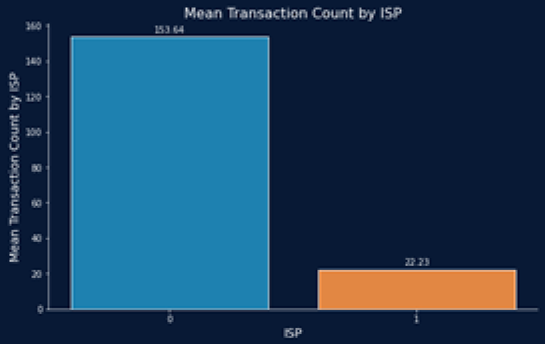
- Identified 16 components.
- One major component containing 29,411 nodes.
- Remaining components have a sparser distribution.

Edges Features

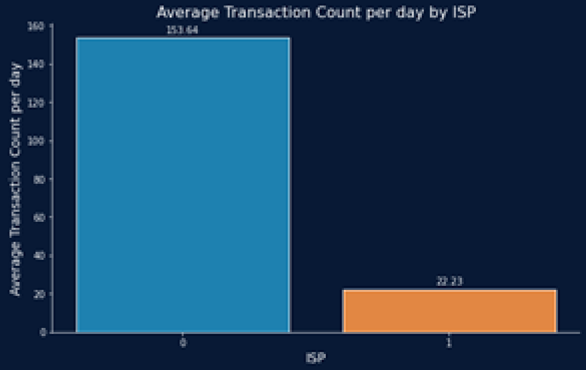


The average transactions count between non-fraudulent accounts are higher than fraudulent accounts

Edges Features

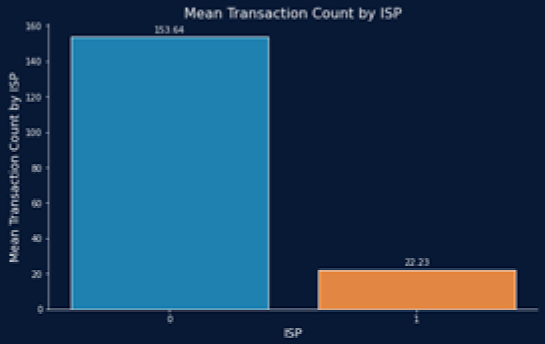


The average transactions count between non-fraudulent accounts are higher than fraudulent accounts

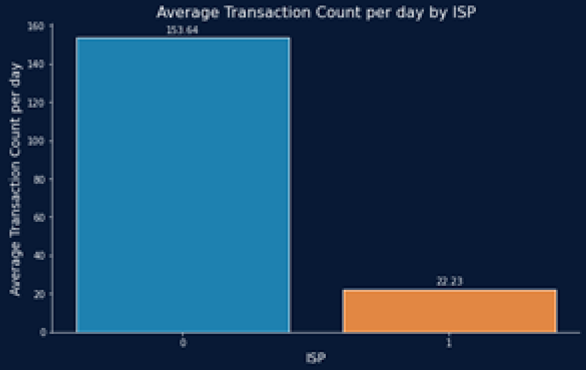


A consistent pattern can be found in daily average transactions count

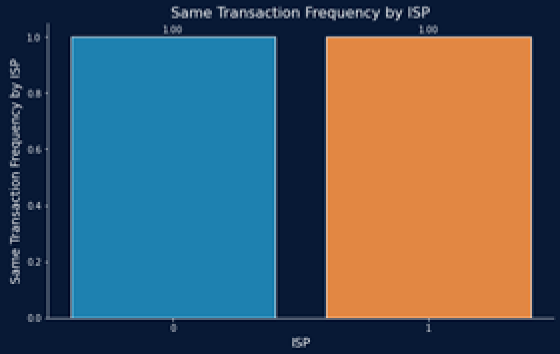
Edges Features



The average transactions count between non-fraudulent accounts are higher than fraudulent accounts

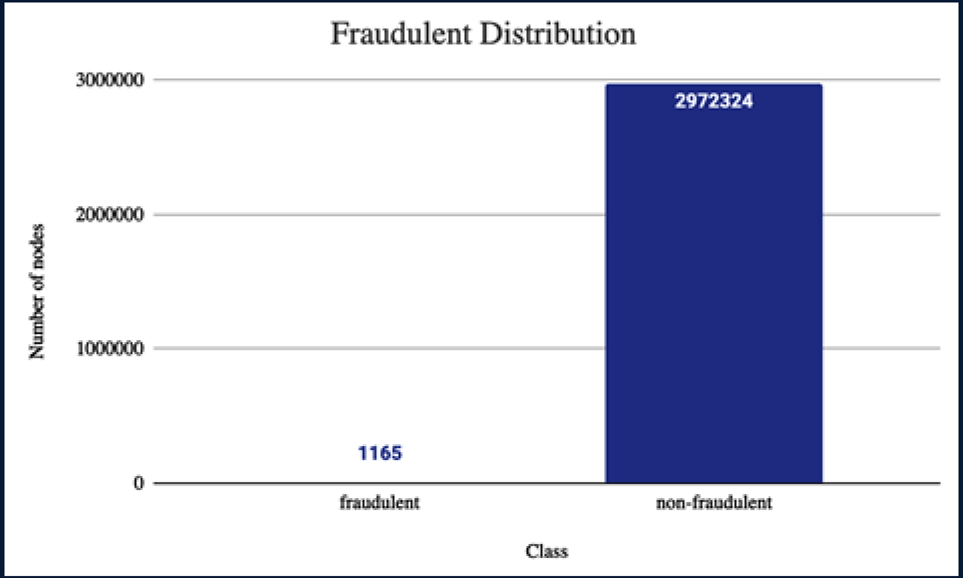


A consistent pattern can be found in daily average transactions count



There is no higher frequency of identical transactions for fraudulent accounts

Data Balancing



Imbalanced data: 25 non-fraud instances : 1 fraud instance

Oversampling Technique:

- SMOTE algorithm: Equalize the fraud class with non-fraud class

Undersampling Methods Tested:

- Near Miss-1: Minimum distance to three closest minority examples.
- Near Miss-2: Minimum distance to three furthest minority examples.
- Near Miss-3: Minimum distance to each minority example.

Best Performance: Near Miss-3 identified as the most effective.

02

Feature Engineering

Additional Temporal Features

In order to better capture the fraudulent activities, more features related to time range have been generated as following:

Features	Definition & Importance
Timestamp Related	year, hour-of-day, date, and day_of_week aiding in a more nuanced analysis of the timestamp impact on fraudulent transactions.
transaction_count	The frequency of transactions for each account.
transaction_per_day	Records the number of identical transactions, exploring the influence of the frequency of identical transactions occurring on the same day.
same_transaction_frequency	The numerical value of the transaction.
active_days	The number of unique days each node interacted on.
business_hours_interactions_count	Business hour, defined as between 9 am and 5 pm. The count of interactions during business hours.
hour-by-hour interaction	The count of interactions that node experienced during each respective hour.

Generate Aggregated Features

How to spot fake news

Node Pair

- Identifying transaction count differences
- exploring discrepancies

1

Time-Dependent

- Rolling metrics over time

2

Ratio Features

- Computing ratios between transactions or weights

3

Node Importance

- Aggregating centrality metrics

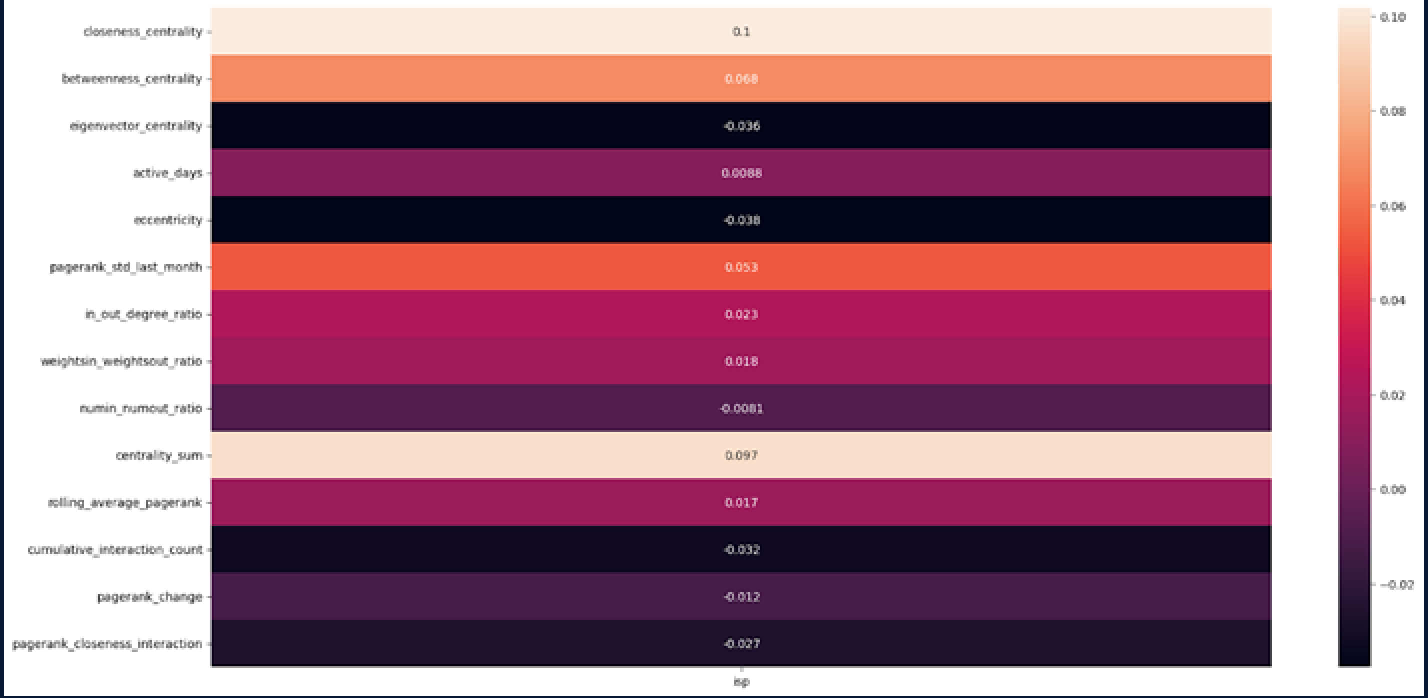
4

Generate Aggregated Features

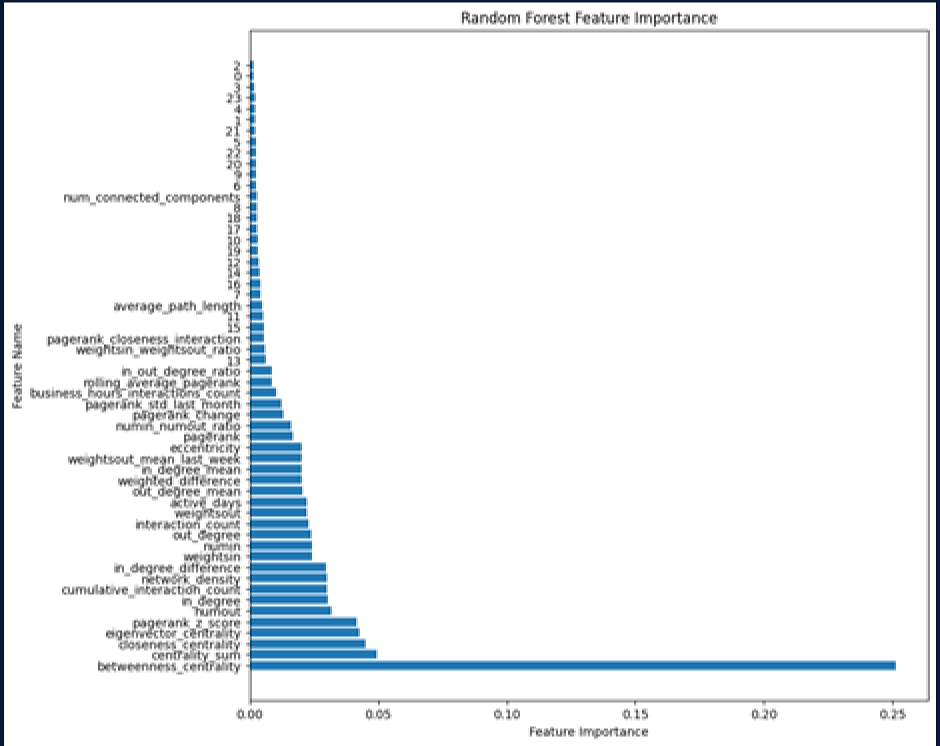
Historical	Statistical Anomalies	Change-Based	Cross Interaction
Utilizing rolling averages and cumulative counts	To identify nodes deviating significantly from average metrics	Tracking changes in pagerank values	Analyzing interactions between pagerank and centrality measure

Feature vs Target Correlation

Utilizing Pearson correlation, 14 features surpassing a threshold of 0.006 are chosen due to their strong association with the target.



Feature Importance (XGB)



This figure illustrates the 14 selected features based on their importance scores derived from the model training process.

03

Machine Learning Models

- KNN, RF, XGB, SVM
- GCN, GNN
- Evaluation

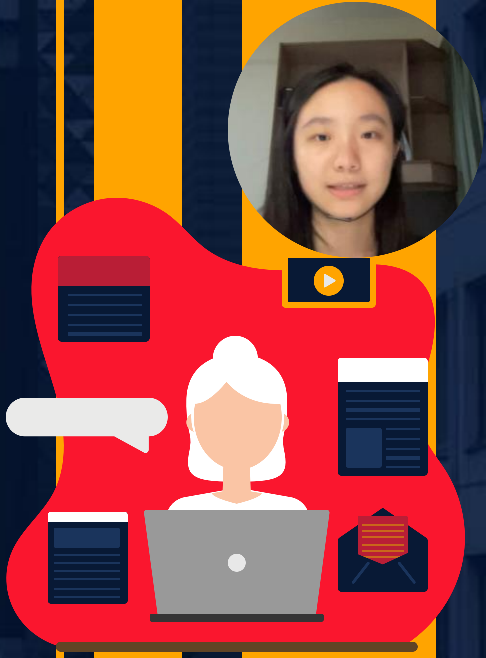
Non-Graph Model

K Nearest Neighbours
Logistic Regression
Random Forest Classifier
Extreme Gradient Boosting
Support Vector Machine
Ensemble Learning

Graph Model

Graph Convolutional Networks

Graph Neural Networks



LIVE

WHICH MODEL IS BETTER?!

A MILLION DOLLAR QUESTIONS! A MILLION DOLLAR QUESTIONS! A MILLION DOLLAR QUESTIONS! A MILLION DOLLAR QUESTIONS!

LIVE

Experiment Results

Non-Graph Model

Models	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
Logistic Regression	0.96	0.00	0.00	0.50
K Nearest Neighbours	0.96	0.04	0.07	0.51
Random Forest without scaling	0.99	0.72	0.81	0.86
Random Forest with scaling	0.99	0.70	0.80	0.85
XGBoost	0.99	0.27	0.41	0.95
Support Vector Machine	0.99	0.10	0.17	0.86
Ensemble Learning	0.99	0.45	0.54	0.72

Table 1: Without Data-Balancing

Non-Graph Model

Given the skewed distribution of data, with fraud instances forming a significantly small percentage, we have experiments with oversampling & under sampling in order to improve the recall score of class 1.

Models with Oversampling	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
Logistic Regression	0.93	0.08	0.08	0.52
K Nearest Neighbours	0.89	0.22	0.13	0.57
Random Forest without scaling	0.99	0.76	0.82	0.88
Random Forest with scaling	0.98	0.62	0.54	0.97

Models with Undersampling	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
Logistic Regression	0.93	0.08	0.08	0.52
K Nearest Neighbours	0.66	0.60	0.13	0.63
Random Forest without scaling	0.94	0.91	0.55	0.93
Random Forest with scaling	0.96	0.56	0.28	0.93

Non-Graph Model

Given the skewed distribution of data, with fraud instances forming a significantly small percentage, we have experiments with oversampling & undersampling in order to improve the recall score of class 1.

Models with Oversampling	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
XGBoost	0.93	0.72	0.23	0.92
Support Vector Machine	0.99	0.17	0.26	0.58
Ensemble Learning	0.93	0.08	0.08	0.52

Models with Undersampling	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
XGBoost	0.81	0.62	0.09	0.81
Support Vector Machine	0.86	0.82	0.15	0.84
Ensemble Learning	0.93	0.08	0.08	0.52

Graph Model

Models	Accuracy	Class 1 Recall	Class 1 F1 Score	ROC_AUC
GNN	0.95	0.01	0.01	**0.50**
GNN with class weights	0.95	0.01	0.01	**0.50**
GNN with oversampling	0.96	0.00	0.00	**0.50**
GNN with undersampling	0.96	0.00	0.00	**0.50**
GCN	0.96	0.00	0.00	**0.50**

1. Inadequate representation of features.
2. Imbalanced nature of data.

04

Conclusion

- Business Usage
- Limitation
- Conclusion

Business Usage


1. Enhanced Security for Cryptocurrency Exchanges and Wallets
2. Risk Management for Financial Institutions
3. Anti-Money Laundering for Regulators
4. Preventive Measures for Businesses

Limitation

1

Feature Effectiveness and Data Volume

-Importance of a substantial number of data samples for validation.
-Example: Centrality features require significant neighbors for validity.

2

Information Lag Issues

-Lag in capturing fraudulent transactions with novel data points.
-Some features need more data to be predictive.

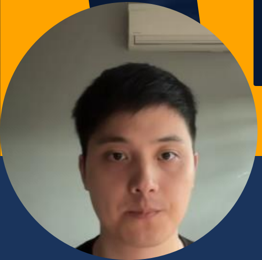
3

Selective Exploration Limitations

-Current focus on a subset of the dataset.
-Potential loss of valuable insights from unexplored patterns or trends.

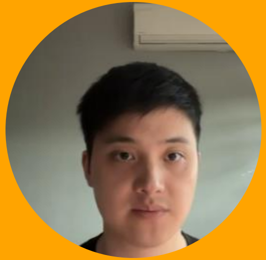
LIVE

Conclusion



Conclusion

- The objective of this project is to detect fraudulent transactions within the Ethereum network.
- We investigated techniques for feature extraction from graph datasets, employing the extracted features to enhance fraudulent identification ability with 89% recall.
- We believe this project offers tangible benefits for practical applications in real-world business scenarios, particularly in the domain of anti-fraud detection within user network data.



THANKS!