FrAug: Enhanced Fraud Detection in Interbank Transfers via Augmented Account Features

Seonkyu Lim[*1], **Jeongwhan Choi**[*2], **Jaehoon Lee**[*3], **Noseong Park**[4]

[1]Korea Financial Telecommunications and Clearings Institute
[2]Yonsei University
[3]LG AI Research
[4]Korea Advanced Institute of Science and Technology
sklim@kftc.or.kr, jeongwhan.choi@yonsei.ac.kr, jaehoon.lee@lgresearch.ai, noseong@yonsei.ac.kr

Abstract

In bank transfers, detecting fraud in interbank transfer is crucial for maintaining the integrity of financial systems. While graph analysis has been widely used in fraud detection systems (FDS), there have been limited attempts to apply it to interbank transfer transaction data such as that from the Housing Finance Information Network (HOFINET). This paper addresses this gap by exploring the application of the structural information inherent in transcation networks to enhance fraud detection in interbank transfers. We propose **Fr**aud detection via feature **Aug**mentation (FrAug), a novel approach that uses the structural information inherent in transaction networks. FrAug extracts graph centrality features and money laundering patterns from these networks, integrates them to capture inter-dependencies, and augments them to the original feature set. This method enables traditional machine learning models to capture complex patterns in graph structures, particularly forcusing on the unique characteristics of interbank transfers. We shows the effectiveness of FrAug using real-world dataset from HOFINET. Our experiments show that models enhanced with FrAug outperform their non-augmented counterparts and surpass graph neural networks (GNNs) in detecting fraudulent interbank transfers. Furthermore, our analysis provides evidence that the graph information created by FrAug are crucial for effective fraud detection.

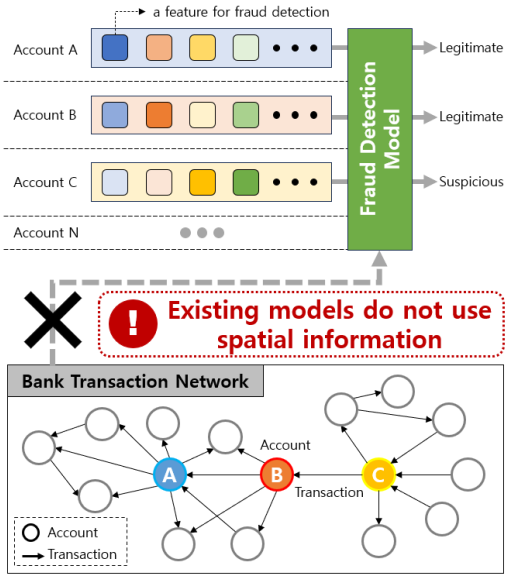


Figure 1: Existing fraud detection models do not consider interbank transfer graphs which is one of the important information for fraud detection.

Introduction

Bank fraud, a federal offense, involves deceptively seeking financial gain at the expense of banking institutions. Because these frauds cause billions of dollars in losses each year, detecting suspicious activity has become a fundamental problem, and most financial institutions have tried to develop and enhance fraud detection systems (FDS) (Sarma et al. 2020; Hilal, Gadsden, and Yawney 2022). For example, pioneering financial messaging infrastructures, such as the Society for Worldwide Interbank Financial Telecommunication (Swift), have long collaborated with a community of more than 11,500 financial institutions to develop novel methods to detect fraudulent transactions[1].

Recently, financial institutions have introduced machine learning and deep learning-based methods to FDS, using their capabilities in classification tasks such as fraud detection (Jung et al. 2019; Leo, Sharma, and Maddulety 2019; Ali et al. 2023; Priscilla and Prabha 2021; Noviandy et al. 2023). While these methods have shown promise, they often overlook the structural information inherent in transaction networks, where bank accounts represent nodes and transactions form edges (see Fig. 1), especially in the case of machine learning models (Chen and Guestrin 2016; Ke et al. 2017; Prokhorenkova et al. 2018).

The information from transaction networks is essential for fraud detection, because some fraud patterns, such as money laundering, are better captured with this graph information. Therefore, existing models that fail to capture the network structure of transactions lead to a decrease in the accuracy of fraud detection.

We propose that this network structure, particularly when

[*]These authors contributed equally.
Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
[1]https://customers.microsoft.com/en-au/story/1637929534319366070-swift-banking-capital-markets-azure-machine-learning

analyzed using graph centrality measures and money laundering patterns, can provide insights for detecting fraud in interbank transfers. Graph centrality metrics, such as degree, closeness, and betweenness centrality, can reveal the importance and connectivity of accounts within the financial transaction network. These measures can help identify unusual patterns or influential nodes that may be indicative of fraudulent activity. Moreover, certain graph patterns are associated with known money laundering techniques (Teichmann 2020; Gerbrands et al. 2022). For instance, 'fan-in' patterns (multiple accounts sending funds to a single account) and 'fan-out' patterns (one account distributing funds to many accounts) are often indicative of attempts to obscure the origin or destination of illicit funds. cyclic transaction patterns, where money moves through a series of accounts and returns to the origin, can also be a red flag for money laundering activities.

To use this structural information and known money laundering patterns, we introduce a novel framework called feature **Aug**mentation for **Fr**aud detection (**FrAug**). FrAug extracts critical graph centrality features and money laundering pattern indicators from transaction networks and augments them to the original feature set. To effectively combine these diverse features, FrAug utilizes the Mahalanobis distance to integrate all centrality measures and money laundering pattern indicators into one. This approach enables traditional machine learning methods to capture complex patterns in graph structures, particularly focusing on the structure characteristics of interbank transfers and potential money laundering activities.

We test our method using a real-world dataset from the Housing Finance Information Network (HOFINET). HOFINET provides detailed data on interbank transfers made by customers through mobile or internet banking. Our experiments demonstrate that machine learning models enhanced with FrAug outperform their non-augmented counterparts in detecting fraudelent interbank transactions. We also show the efficiency of our method by comparing the machine learning methods with FrAug to those of GNNs. Moreover, our analyses show that the centrality-based features and money laundering pattern indicators generated by FrAug are crucial for effective fraud detection.

To sum up, our main contributions are as follows:

- We propose FrAug, a novel method that use structural information of transaction networks for fraud detection in interbank transfers.
- FrAug extracts and integrates graph centrality features and money laundering pattern indicators using Mahalanobis distance.
- We demonstrate the machine learning models enhanced with FrAug outperforms both their non-augmented counterparts and GNNs on HOFINET dataset.

Related Work

Fraud Detection in Bank Transfer

Fraud detection in bank transfers is crucial for financial institutions. (Ranjan et al. 2022) emphasize the importance of machine learning models and data preprocessing to detect fraudulent transactions, while (Bukhori and Munir 2023) focus on leveraging transaction data and addressing the complexity of fraud detection. (Lv et al. 2019) use convolutional neural networks (CNNs) to identify fraudulent bank accounts through transaction relationships. To handle data imbalance in fraud detection, (Hsin et al. 2022) suggest preprocessing and resampling strategies. Furthermore, (Höppner et al. 2022) propose cost-sensitive algorithms to minimize financial losses due to fraud. These studies improve the detection and prevention of fraudulent bank transfers.

Utilizing Graph Information in Fraud Detection

Graph information, such as centrality, is frequently used in fraud detection to extract important features from financial data. (Prusti, Das, and Rath 2021) demonstrates how graph algorithms can be integrated with credit card datasets to improve fraud detection accuracy. In addition, graph-based features have been further refined using the Gaussian mixture model to capture more complex patterns (Prusti, Behera, and Rath 2022). In online auctions, (Bangcharoensap et al. 2015) utilizes weighted degree centrality in a semi-supervised learning framework to detect fraudsters. Similarly, centrality measures, such as degree and betweenness, have also been used to detect money laundering in factoring company data (Colladon and Remondi 2017). xFraud utilizes centrality measures to detect fraud in eBay's network (Rao et al. 2020).

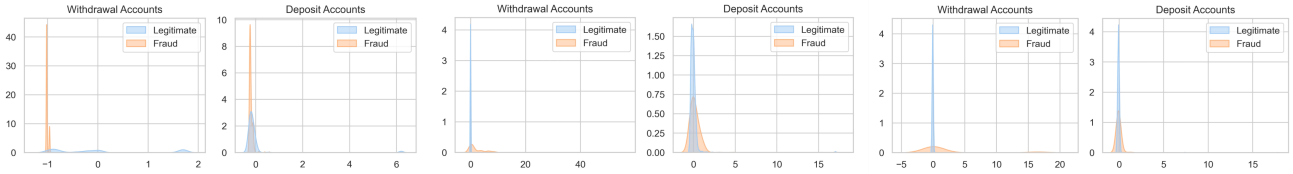
While graph-based methods are widely used in fraud detection, their application to *interbank transfers remains underexplored*. Therefore, we aim to address this challenge by proposing methods to use graph information to detect fraudulent activities in interbank transfers.

Graph Neural Networks for Fraud Detection

Recently, there has been an emergence of GNN-based models (Kipf and Welling 2016; Hamilton, Ying, and Leskovec 2017; Dou et al. 2020; Cheng et al. 2020; Liu et al. 2021; Zhang et al. 2021; Xiang et al. 2023) that utilize structural information of financial transactions to detect fraud. These models have demonstrated excellent performance in learning complex relationships and patterns within financial transaction networks. However, GNN-based models often require significant computational resources, limiting their practical use in real-world fraud detection systems. In addition, their lack of interpretability makes it difficult to understand how specific features contribute to the detection process (Ying et al. 2019). In contrast, machine learning models such as XGBoost offer advantages in computational efficiency and provide interpretability (Shwartz-Ziv and Armon 2022).

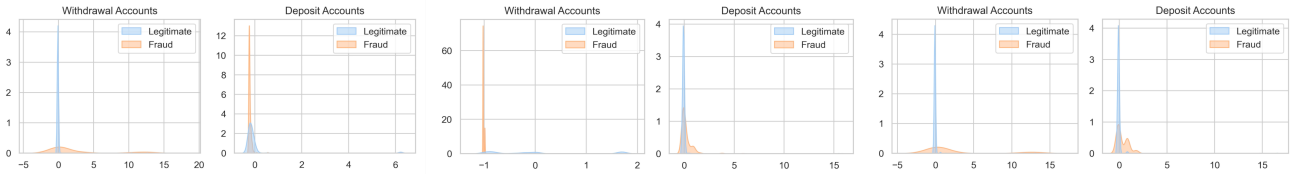
Methodology

In this section, we describe FrAug. Specifically, Section and Section explain what kinds of features are extracted from transaction networks. Subsequently, in Section , we provide how these extracted features are augmented for machine learning models.



(a) Degree Centrality (b) Closeness Centrality (c) Betweenness Centrality

Figure 2: KDE plots of normalized centrality measures for deposit and withdrawal account graphs. 'Withdrawal Accounts' denotes the centrality of withdrawal accounts when given transactions are legitimate or fraudulent. Likewise, 'Deposit Accounts' denotes the centrality of deposit accounts.



(a) Fan-in (b) Fan-out (c) Simple Cycle

Figure 3: KDE plots of money laundering pattern indicators in deposit and withdrawal accounts.

Graph Centrality Analysis

Graph centrality measures how important nodes are within a graph. Centrality analysis plays a crucial role in identifying the influence, importance, and position of nodes across various graphs (Kuzubaş, Ömercikoğlu, and Saltoğlu 2014; Temizsoy, Iori, and Montes-Rojas 2017; Xu and Corbett 2019; Martínez-Ventura, Mariño-Martínez, and Miguélez-Márquez 2023). As such, we examine the centrality distributions of an account in a financial transaction network. We explore various centrality measures, including degree, closeness, and betweenness centrality (Freeman et al. 2002). For this analysis, we use a representation of the transaction network as a directed graph structure, where withdrawal and deposit accounts are source and target nodes, respectively.

Formulation of Centrality Let $G = (V, E)$ be a transaction graph where $v \in V$ is an account (node) and $(u, v) \in E$ is a transaction (edge) in which u and v are withdrawal (source) and deposit (target) accounts, respectively. Then, each centrality is formulated as follows:

- **Degree centrality** measures the level of an account's activity within the network through its direct transactions by calculating the number of edges connected to nodes: Degree_Cen(v)=$\sum_{u \in V}(\mathbf{1}_{(u,v)\in E} + \mathbf{1}_{(v,u)\in E})$, where $\mathbf{1}_{\text{condition}}$ equals 1 if the condition is satisfied and 0 otherwise.
- **Closeness centrality** measures how quickly an account can interact with all other accounts in the network by calculating the average length of the shortest path from the node to all other nodes: Closeness_Cen(v)=$\sum_{u \in V}(1/d(u,v)+1/d(v,u))$, where $d(u, v)$ denotes the length of shortest path from u to v.
- **Betweenness centrality** indicates how much a node bridges others in a network. It is based on a node's participation in the shortest path between a pair of nodes: Betweenness_Cen(v)=$\sum_{u_1,u_2 \in V} 1_{v \in p(u_1,u_2)}$, where $p(u_1, u_2)$ denotes the shortest path from u_1 to u_2.

Analysis of Centrality in Transaction Networks In Fig. 2, we provide the distribution of withdrawal and deposit accounts when transactions are legitimate or fraudulent. As shown in the figure, fraudulent and legitimate transactions exhibit different distributions. For degree centrality, legitimate transactions typically have larger centrality values than fraudulent ones, particularly in withdrawal accounts. Previous studies have noted that nodes with low degree centrality in money laundering networks are associated with roles such as organizers and recruiters (Dreżewski, Sepielak, and Filipkowski 2015). Our observations may align with this statement. In terms of closeness and betweenness centrality, fraudulent transactions tend to have a wider range of centrality values than normal ones. Fraudulent transactions generally have closeness and betweenness centrality values that deviate from the normal range.

Based on this analysis, we utilize centrality indicators such as degree, closeness, and betweenness as additional features to enhance the performance of machine learning models. This is because we anticipate that incorporating these centrality measures will enable models to better understand the importance and connectivity structure of accounts within the financial transaction graph, aiding in accurately identifying fraudulent transactions.

Money Laundering Patterns

Money laundering is a representative form of criminal activity in which funds obtained through illegal activities are disguised to appear as though they originate from legitimate sources. This illicit process often manifests in specific patterns within financial transactions. Research has shown that

money laundering frequently follows identifiable graph patterns, as detailed in (Altman et al. 2024). By analyzing and extracting features from the transaction graphs, we improve our ability to identify suspicious transactions and uncover fraudulent schemes.

Formulation of Money Laundering Patterns To extract features related to money laundering, we formulate each pattern as follows:

- **Fan-out pattern** is related to the number of outgoing edges. This pattern happens when one account gives money to many other accounts. It is used to hide where the money came from by putting it in many accounts. Fan_Out(v)=$\sum_{u \in V} \mathbf{1}_{(v,u) \in E}$.
- **Fan-in pattern** indicates the number of incoming edges in contrast to the fan-out case. The fan-in pattern involves multiple accounts sending funds to a single account. This is used to consolidate funds from various sources into a single account, making it difficult to trace the origins. Fan_In(v)=$\sum_{u \in V} \mathbf{1}_{(u,v) \in E}$.
- **Cycle pattern** measures the number of cycles including a given node. Money moves through a series of accounts and eventually returns to the original account. This can be a sign of money laundering because it shows money moving in a circle to hide the trail. Cycle(v)=$\sum_{\mathbf{c} \in \mathbf{C}} \mathbf{1}_{v \in \mathbf{c}}$, where $\mathbf{C}$ denotes a set of existing cycle paths $\mathbf{c}$ in transaction graphs.

Analysis of Money Laundering Patterns in Transaction Networks In addition to examining the centrality metrics, we also present the KDE plots of various money laundering patterns in Fig. 3. As expected, the distributions of fraudulent and legitimate cases exhibit noticeable differences. This observation suggests that these features are valuable for distinguishing between fraudulent and legitimate transactions. Recognizing their potential, we integrate these features into our fraud detection system to enhance its accuracy and effectiveness. By augmenting our model with these additional insights, we aim to improve its ability to identify and prevent fraudulent activities.

Node Feature Augmentation

Up to this point, we have focused on extracting valuable features from transaction graphs. In this section, we will discuss how to incorporate these features into our fraud detection system. The most straightforward method would be to simply add all the generated features to the original feature set. However, this approach has limitations in capturing complex correlations between features, which can result in ineffective FDS.

To address this issue, we propose an approach based on the Mahalanobis distance (De Maesschalck, Jouan-Rimbaud, and Massart 2000). Let $\mathbf{a} \in R^d$ represent the concatenated vector of all generated node features, where d is the number of generated node features. This approach condenses the information contained in $\mathbf{a}$ into a single scalar value $\texttt{MAH}(\mathbf{a}) \in R$: MAH(a) = $\sqrt{(\mathbf{a}-\mu)^T \Sigma^{-1} (\mathbf{a}-\mu)} \in R, \quad \mathbf{a} \in R^d$, where μ and Σ are empirical mean and variance of $\mathbf{a}$, respectively. After calculating $\texttt{MAH}(\mathbf{a})$, we augment $\texttt{MAH}(\mathbf{a})$ to an original feature set.

Table 1: Statistics of HOFINET transfer transaction dataset.

Time Range	# Accounts	# Transfers	# Suspicious
Mar. 2023	26,227	141,494	260 (0.1838%)

We use Mahalanobis distance, which can captures the correlations between features rather than the simpler Euclidean distance, which is important for detecting complex fraud patterns (Astivia 2024). It provides a more precise measure of variability in multivariate data (Gath and Hayes 2011). By condensing multiple features into a single distance value, this approach retains the richness of multivariate relationships and helps reduce overfitting, particularly when the dataset has limited observations relative to the number of features. Therefore, the Mahalanobis distance enhances the ability of the model to generalize well.

Although simpler methods such as feature aggregation can reduce computational steps, they often fail to capture the complex inter-dependencies between features, leading to suboptimal fraud detection results. In contrast, the Mahalanobis distance offers a balanced approach that integrates feature complexity with improved detection capability.

We can generate the feature vector $\mathbf{a}$ using centrality measures and money laundering patterns. In our experiments, we systematically test each of these cases to evaluate their effectiveness.

Experiments

We proceed to assess the performance of FrAug in fraud detection by applying it to a real-world transfer transaction dataset from Housing Finance Information Network (HOFINET). The primary goal of our experiments is to demonstrate the superior accuracy of our method when compared to existing baseline models.

Experimental Setting

All experiments are conducted in the following software and hardware environments: UBUNTU 18.04 LTS, NVIDIA TESLA T4, CUDA 11.4, PYTHON 3.8, PYTORCH 1.12, NETWORKX 2.8.4.

Datasets For our analysis, we utilize the HOFINET dataset, which consists of real-world interbank fransfers. Detailed statistics about this dataset can be found in Table 1. The HOFINET dataset comprises 9 distinct features, each of which is described in detail in Table 2. The HOFINET tabular dataset was converted into a graph structure by treating each unique combination of account number and bank code as an individual node and transfer transactions between these accounts as edges connecting the respective nodes. We split the dataset into training, validation, and test datasets in the following proportions: 75%, 15%, and 15%.

Models & Hyperparameters For our experiment, we use various models in our experiments. In this section, we ex-

Table 2: The details of the HOFINET dataset.

Field	Description
Transaction Date	The date of the transaction.
Transaction Time	The time of the transaction.
Amount	The amount of money.
Media Type	Transaction medium (e.g., Mobile).
Fund Type	Type of funds (e.g., salary).
Withdrawal Bank Code	Bank identification code.
Withdrawal Account Number	Account number for withdrawal.
Deposit Bank Code	Bank identification code.
Deposit Account Number	Account number for deposit.

Table 3: Result of each model in fraud detection. The best scores in each column are in boldface, and the best scores in each model are underlined.

	Model	Pre.	Rec.	F1	Geo-m.	IBA	AUC
Machine Learning	XGBoost	0.7333	0.5789	0.6470	0.7607	0.5544	0.7893
	w/ CEN MAH	0.7586	0.5789	0.6567	0.7608	0.5544	0.7893
	w/ MOL MAH	0.6279	**0.7105**	0.6667	**0.8426**	**0.6895**	**0.8549**
	w/ ALL MAH	0.7667	0.6053	**0.6765**	0.7779	0.5812	0.8025
	LightGBM	0.8095	0.4474	0.5763	0.6688	0.4226	0.7236
	w/ CEN MAH	0.8571	0.4737	0.6102	0.6882	0.4487	0.7368
	w/ MOL MAH	0.9524	0.5263	0.6557	0.7254	0.5013	0.7631
	w/ ALL MAH	0.9524	0.5263	0.6780	0.7255	0.5014	0.7631
	HGBoost	0.8571	0.4737	0.6102	0.6882	0.4487	0.7368
	w/ CEN MAH	0.7407	0.5263	0.6154	0.7254	0.5012	0.7630
	w/ MOL MAH	0.8696	0.5263	0.6557	0.7254	0.5013	0.7631
	w/ ALL MAH	0.8261	0.5000	0.6230	0.7070	0.4749	0.7499
	CatBoost	0.8571	0.3157	0.4615	0.5619	0.2942	0.6578
	w/ CEN MAH	**1.0000**	0.2895	0.4490	0.5380	0.2689	0.6447
	w/ MOL MAH	**1.0000**	0.2895	0.4490	0.5380	0.2689	0.6447
	w/ ALL MAH	0.9231	0.3158	0.4706	0.5619	0.2942	0.6579
GNN	PC-GNN	0.1667	0.0513	0.0784	0.2264	0.0561	0.5254
	STAGN	0.1087	0.3030	0.1600	0.5492	0.2809	0.6496
	GTAN	0.0102	0.1026	0.0185	0.3173	0.0918	0.5574

plain what kinds of models we use for experiments and provide search space for the hyperparameters of each model. Selected hyperparameters are in italics and boldface. To begin with, we offer a detailed description of the machine learning models we have selected, which are outlined as follows:

- **XGBoost** (Chen and Guestrin 2016): XGBoost is a widely used gradient-boosted decision tree algorithm. The search space for the number of estimators L is $\{100, 200, \textbf{\textit{500}}, 1000, 2000\}$, maximum tree depth is $\{2, 3, 4, 6, \textbf{\textit{8}}, 10\}$, and regularization weights are $\{0, 0.1, 1, 5, \textbf{\textit{10}}\}$.
- **LightGBM** (Ke et al. 2017): It is an efficient gradient boosting algorithm aimed at reducing computational costs. The hyperparameter search spaces are the same as XGBoost, with L set to ***2000***, maximum tree depth to ***3***, and regularization weight to ***0.1***.
- **HGBoost** (Pedregosa et al. 2011): It is a histogram-based variant of gradient boosting. The hyperparameter search spaces are the same as other boosting algorithms, with L, maximum tree depth, and regularization weight set to ***1000***, ***10***, and ***5***, respectively.
- **CatBoost** (Prokhorenkova et al. 2018): It is a gradient-boosting algorithm designed for categorical features. We use the same search spaces for hyperparameters as other boosting algorithms. L, the maximum depth of tree, and the weight of regularization terms are set to ***1500***, ***8***, and ***5***, respectively.

Next, we utilize recent GNN-based models as follows:

- **PC-GNN** (Liu et al. 2021): It is a GNN designed to address class imbalance using a picking and choosing scheme. The neighbor sampling ratio is in $\{0.1, 0.3, 0.5, \textbf{\textit{0.7}}, 0.9\}$ and the embedding size in $\{32, \textbf{\textit{64}}, 128, 256\}$.
- **STAGN** (Cheng et al. 2020): It is a GNN that captures spatial and temporal information with attention mechanisms. We set the search space for attention hidden dimension to $\{32, \textbf{\textit{64}}, 128, 150\}$.
- **GTAN** (Xiang et al. 2023): It is a semi-supervised GNN that employs gated temporal attention for message passing. We set the hidden dimension to $\{\textbf{\textit{64}}, 128, 256, 512\}$ and the number of GTAN layers to $\{1, \textbf{\textit{2}}, 3\}$.

Table 4: Performance comparison of LightGBM with individual features and Mahalanobis distances (MAH). DC, CC, and BC denote degree centrality, closeness centrality, and betweenness centrality, respectively. Also, FO, FI, and SC denote fan-out, fan-in, and simple cycle in money laundering patterns, respectively.

	Datasets	Pre.	Rec.	F1	Geo-m.	IBA	AUC
	Original model	0.8095	0.4474	0.5763	0.6688	0.4226	0.7236
CEN	w/ DC	0.8947	0.4474	0.5965	0.6688	0.4226	0.7236
CEN	w/ CC	0.8636	0.5000	0.6333	0.7071	0.4749	0.7499
CEN	w/ BC	0.7917	0.5000	0.6129	0.7070	0.4749	0.7499
CEN	w/ MAH	0.8571	0.4737	0.6102	0.6882	0.4487	0.7368
MOL	w/ FO	0.9048	0.5000	0.6441	0.7071	0.4750	0.7500
MOL	w/ FI	0.8333	**0.5263**	0.6452	0.7254	0.5013	**0.7631**
MOL	w/ SC	0.8636	0.5000	0.6333	0.7071	0.4749	0.7499
MOL	w/ MAH	**0.9524**	**0.5263**	0.6557	0.7254	0.5013	**0.7631**
	w/ ALL MAH	**0.9524**	**0.5263**	**0.6780**	**0.7255**	**0.5014**	**0.7631**

Evaluation Metrics Our task is fraud detection, which is a kind of classification. Therefore, we report various classification metrics, including precision (Pre.), recall (Rec.), F1-score, geometric mean (Geo-m.), index balanced accuracy (IBA), and AUC — note that we do not report accuracy scores because of the very small proportion of suspicious labels (See Table 1).

Fraud Detection Performance

Table 3 shows the experimental results of the fraud detection model after including additional features such as the Mahalanobis distance for the centrality metric (CEN MAH), the Mahalanobis distance for the money laundering pattern

Table 5: Comparison of feature importance with MAH in LightGBM. Withdrawal/Deposit CEN MAH and MOL MAH refer to the Mahalanobis distance of centrality measures and money laundering pattern indicators for withdrawal/deposit banks and accounts, respectively. The best importance is in boldface, and the second best is underlined.

Features	w/ CEN MAH	w/ MOL MAH	w/ ALL MAH
Transaction Date	2460	568	1254
Transaction Time	526	35	387
Transaction Amount	**17,044**	<u>4498</u>	**3864**
Media Type	186	194	69
Fund Type	0	0	0
Withd. Bank Code	590	320	142
Withd. Acc. Num.	903	337	270
Deposit Bank Code	621	107	233
Deposit Acc. Num.	1215	306	472
Withd. CEN MAH	<u>13,895</u>	-	1180
Deposit CEN MAH	12,843	-	1500
Withd. MOL MAH	-	**4556**	1699
Deposit MOL MAH	-	4079	<u>1758</u>

metric (MOL MAH), and both (ALL MAH). We use these enhanced features with various machine learning models to evaluate their impact on performance.

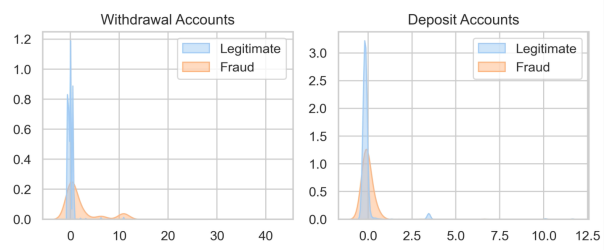
Significant performance improvements are observed. The XGBoost model with ALL MAH shows a significant improvement over the original model, with a precision of 0.7667, a recall of 0.6053, and an F1 score of 0.6765. Similarly, LightGBM with ALL MAH also performed remarkably well, achieving a precision of 0.9524, recall of 0.5263, and F1 score of 0.6780. These results show that the Mahalanobis distances of centrality measures and money laundering pattern indicators improve model performance by providing a more comprehensive feature set that captures the underlying graph patterns in the data. As for GNN models, most of them fail in the fraud detection task, showing worse performance than machine learning models with FrAug. This result shows that our FrAug is more helpful for capturing graph information in fraud detection than GNNs.

The experimental results demonstrate that the proposed feature enhancements, which integrate various centrality measures and money laundering pattern indicators into the Mahalanobis distance, improve machine learning models by enabling them to capture graph information. These findings emphasize the potential of integrating centrality and money laundering pattern features to improve the effectiveness of fraud detection systems in financial networks.

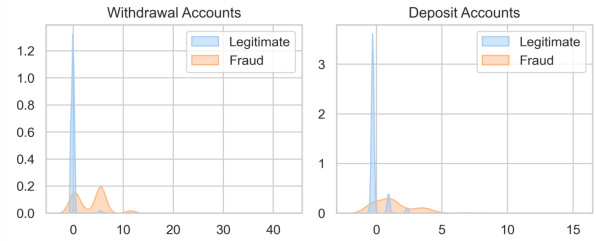
Analyses

In this section, we explain the effectiveness of our method by explaining why we use Mahalanobis distance to augment features in FrAug, analyzing the distribution of this distance, and providing an interpretation of the importance of the features.

Effectiveness of augmentation based on Mahalanobis distance As stated in Section , we augment the generated



(a) Centrality Mahalanobis Distance (CEN MAH)



(b) Money Laundering Mahalanobis Distance (MOL MAH)

Figure 4: KDE plots for the distributions of Mahalanobis distance based on centrality and money laundering patterns for withdrawal and deposit accounts.

features using Mahalnobis distance. In this section, we provide the performance of these approaches compared to the case when each feature is simply augmented without applying Mahalnobis distance methods. In the case without Mahalnobis distance methods, we augment each feature individually to make the number of augmented features similar to the case with Mahalnobis distance methods. Table 4 shows the performance of LightGBM in this experiment with individual features and Mahalanobis distances (MAH). A comparison of the performance with individual centrality measures with that of the Mahalanobis distance reveals that the latter, which integrates all centrality measures into one Mahalanobis distance (CEN MAH) and all money laundering pattern indicators into another (MOL MAH), significantly enhances the performance. Furthermore, using both information (ALL MAH) achieves the highest scores.

These results show that combining centrality measures and money laundering pattern indicators into a unified feature using Mahalanobis distance improves the performance of fraud detection models. We conjecture that this can better capture the complex inter-dependencies between features, thereby improving performance over using individual features separately.

Importance of features augmented by FrAug Table 5 shows the feature importance in the LightGBM with the inclusion of Mahalanobis distance metrics for centrality measures (CEN MAH) and money laundering pattern indicators (MOL MAH). Using CEN MAH, the most important feature is the transaction amount, followed by the Mahalanobis distance of centrality measures for withdrawal banks and accounts. When MOL MAH is included, the Mahalanobis distance of money laundering pattern indicators for with-

Table 6: Result of GNN-based models with or without our method in fraud detection. As in the case of machine learning models, we report scores without our node feature augmentation and with Mahalanobis distance based on centrality (CEN MAH), money laundering pattern (MOL MAH), and both (ALL MAH).

Model	Pre.	Rec.	F1	Geo-m.	IBA	AUC
PC-GNN	0.1667	0.0513	0.0784	0.2264	0.0561	0.5254
w/ CEN MAH	0.1035	0.0769	0.0882	0.2772	0.0698	0.5378
w/ MOL MAH	0.1000	0.0769	0.0870	0.2772	0.0697	0.5378
w/ ALL MAH	0.1000	0.0769	0.0870	0.2772	0.0697	0.5378
STAGN	0.1087	0.3030	0.1600	0.5492	0.2809	0.6496
w/ CEN MAH	**0.1875**	**0.4167**	**0.2586**	**0.6445**	**0.3913**	**0.7068**
w/ MOL MAH	**0.1875**	**0.4167**	**0.2586**	**0.6445**	**0.3913**	**0.7068**
w/ ALL MAH	**0.1875**	**0.4167**	**0.2586**	**0.6445**	**0.3913**	**0.7068**
GTAN	0.0102	0.1026	0.0185	0.3173	0.0918	0.5574
w/ CEN MAH	0.0132	0.0769	0.0225	0.2759	0.0692	0.6600
w/ MOL MAH	0.0084	0.1795	0.0161	0.4153	0.1590	0.6425
w/ ALL MAH	0.0090	0.1026	0.0165	0.3169	0.0916	0.6884

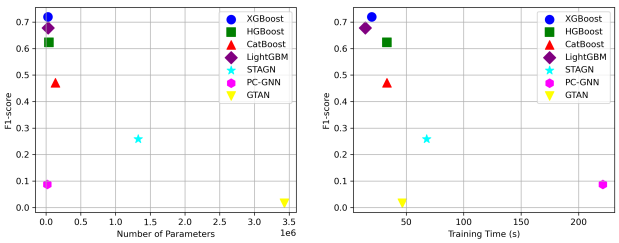
drawal banks and accounts becomes the most important feature. Using both CEN MAH and MOL MAH (ALL MAH), the transaction amount remains the most important feature, with the Mahalanobis distance of money laundering pattern indicators for deposit banks and accounts also showing high importance. These findings indicate that Mahalanobis distances for centrality measures and money laundering pattern indicators play a crucial role in fraud detection.

Distributions of augmented features with Mahalnobis distance Fig. 4 shows the distribution of the Mahalanobis distance for centralities and money laundering pattern indicators. The plots show that combining various features in the Mahalanobis distance leads to more distinct separation between legitimate (blue) and fraudulent (orange) transactions. These results demonstrate the effectiveness of the Mahalanobis distance in capturing complex inter-dependencies among features and enhancing fraud detection capabilities.

GNN with FrAug In this section, we evaluate the performance of GNN-based models when using feature sets augmented by FrAug. The results presented in Table 6 demonstrate that FrAug significantly benefits GNN-based approaches. While GNNs are effective at capturing graph-based information, they may not fully exploit all the available graph data for fraud detection tasks. Our findings suggest that FrAug effectively uncovers crucial information that enhances fraud detection in interbank transfers, thereby improving the overall performance of GNN models.

Model Efficiency

We assess the efficiency of various models by comparing their F1-scores in relation to the number of parameters and training times. The results, illustrated in Fig. 5, highlight the performance and efficiency of different models. Our analysis shows that boosting models generally have shorter training times compared to GNN-based models. Notably, XGBoost and LightGBM achieve high F1-scores while also maintaining relatively short training times and requiring fewer parameters. In contrast, GNN-based models required more parameters and had longer training times. This is because machine learning models are generally less complex than GNN-based models. In addition, the inference time for XGBoost (0.03s) is shorter compared to GTAN (0.28s).



(a) F1-score vs Number of Parameters (b) F1-score vs Training Time

Figure 5: The results of model efficiency with F1 in fraud detection. Note that all reported models are equipped with FrAug.

Conclusion and Future Work

In this paper, we proposed FrAug where graph centralities and representative money laundering patterns are extracted and augmented to an original feature set. Based on the findings and discussions presented in this paper, our FrAug significantly advances fraud detection within financial transaction networks. By using graph information through node feature augmentation, FrAug allows traditional machine learning models to effectively capture patterns indicative of fraudulent behavior, which are often missed by conventional approaches. Our extensive experiments confirm that models enhanced with FrAug outperform their non-augmented counterparts and achieve superior performance compared to various GNNs.

Future research will broaden the way to extract useful features and augment them. These expansions will pave the way for more robust and reliable fraud prevention strategies in the evolving landscape of financial transactions.

References

Ali, A. A.; Khedr, A. M.; El-Bannany, M.; and Kanakkayil, S. 2023. A Powerful Predicting Model for Financial Statement Fraud Based on Optimized XGBoost Ensemble Learning Technique. *Applied Sciences*, 13(4): 2272.

Altman, E.; Blanuša, J.; Von Niederhäusern, L.; Egressy, B.; Anghel, A.; and Atasu, K. 2024. Realistic synthetic financial transactions for anti-money laundering models. *Advances in Neural Information Processing Systems*, 36.

Astivia, O. L. O. 2024. A method to simulate multivariate outliers with known mahalanobis distances for normal and non-normal data. *Methods in Psychology*, 11: 100157.

Bangcharoensap, P.; Kobayashi, H.; Shimizu, N.; Yamauchi, S.; and Murata, T. 2015. Two step graph-based semi-supervised learning for online auction fraud detection. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part III 15*, 165–179. Springer.

Bukhori, H. A.; and Munir, R. 2023. Inductive link prediction banking fraud detection system using homogeneous graph-based machine learning model. In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, 0246–0251. IEEE.

Chen, T.; and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Cheng, D.; Wang, X.; Zhang, Y.; and Zhang, L. 2020. Graph neural network for fraud detection via spatial-temporal attention. *IEEE Transactions on Knowledge and Data Engineering*, 34(8): 3800–3813.

Colladon, A. F.; and Remondi, E. 2017. Using social network analysis to prevent money laundering. *Expert Systems with Applications*, 67: 49–58.

De Maesschalck, R.; Jouan-Rimbaud, D.; and Massart, D. L. 2000. The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1): 1–18.

Dou, Y.; Liu, Z.; Sun, L.; Deng, Y.; Peng, H.; and Yu, P. S. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM international conference on information & knowledge management*, 315–324.

Dreżewski, R.; Sepielak, J.; and Filipkowski, W. 2015. The application of social network analysis algorithms in a system supporting money laundering detection. *Information Sciences*, 295: 18–32.

Freeman, L. C.; et al. 2002. Centrality in social networks: Conceptual clarification. *Social network: critical concepts in sociology. Londres: Routledge*, 1: 238–263.

Gath, E.; and Hayes, K. 2011. Bounds for a multivariate extension of range over standard deviation based on the Mahalanobis distance. *Linear algebra and its applications*, 435(6): 1267–1276.

Gerbrands, P.; Unger, B.; Getzner, M.; and Ferwerda, J. 2022. The effect of anti-money laundering policies: an empirical network analysis. *EPJ Data Science*, 11(1): 15.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Hilal, W.; Gadsden, S. A.; and Yawney, J. 2022. Financial fraud: a review of anomaly detection techniques and recent advances. *Expert systems With applications*, 193: 116429.

Höppner, S.; Baesens, B.; Verbeke, W.; and Verdonck, T. 2022. Instance-dependent cost-sensitive learning for detecting transfer fraud. *European Journal of Operational Research*, 297(1): 291–300.

Hsin, Y.-Y.; Dai, T.-S.; Ti, Y.-W.; Huang, M.-C.; Chiang, T.-H.; and Liu, L.-C. 2022. Feature engineering and resampling strategies for fund transfer fraud with limited transaction data and a time-inhomogeneous modi operandi. *IEEE Access*, 10: 86101–86116.

Jung, C.; Mueller, H.; Pedemonte, S.; Plances, S.; and Thew, O. 2019. Machine learning in UK financial services. *Bank of England and Financial Conduct Authority*.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Kuzubaş, T. U.; Ömercikoğlu, I.; and Saltoğlu, B. 2014. Network centrality measures and systemic risk: An application to the Turkish financial crisis. *Physica A: Statistical Mechanics and its Applications*, 405: 203–215.

Leo, M.; Sharma, S.; and Maddulety, K. 2019. Machine learning in banking risk management: A literature review. *Risks*, 7(1): 29.

Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2021. Pick and choose: a GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the web conference 2021*, 3168–3177.

Lv, F.; Wang, W.; Wei, Y.; Sun, Y.; Huang, J.; and Wang, B. 2019. Detecting fraudulent bank account based on convolutional neural network with heterogeneous data. *Mathematical Problems in Engineering*, 2019(1): 3759607.

Martínez-Ventura, C.; Mariño-Martínez, R.; and Miguélez-Márquez, J. 2023. Redundancy of centrality measures in financial market infrastructures. *Latin American Journal of Central Banking*, 4(4): 100098.

Noviandy, T. R.; Idroes, G. M.; Maulana, A.; Hardi, I.; Ringga, E. S.; and Idroes, R. 2023. Credit Card Fraud Detection for Contemporary Financial Management Using XGBoost-Driven Machine Learning and Data Augmentation Techniques. *Indatu Journal of Management and Accounting*, 1(1): 29–35.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12: 2825–2830.

Priscilla, C. V.; and Prabha, D. P. 2021. A two-phase feature selection technique using mutual information and XGB-RFE for credit card fraud detection. *Int. J. Adv. Technol. Eng. Explor*, 8: 1656–1668.

Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A. V.; and Gulin, A. 2018. CatBoost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31.

Prusti, D.; Behera, R. K.; and Rath, S. K. 2022. Hybridizing graph-based Gaussian mixture model with machine learning for classification of fraudulent transactions. *Computational Intelligence*, 38(6): 2134–2160.

Prusti, D.; Das, D.; and Rath, S. K. 2021. Credit card fraud detection technique by applying graph database model. *Arabian Journal for Science and Engineering*, 46(9): 1–20.

Ranjan, P.; Santhosh, K.; Kumar, A.; and Kumar, S. 2022. Fraud detection on bank payments using machine learning. In *2022 International Conference for Advancement in Technology (ICONAT)*, 1–4. IEEE.

Rao, S. X.; Zhang, S.; Han, Z.; Zhang, Z.; Min, W.; Chen, Z.; Shan, Y.; Zhao, Y.; and Zhang, C. 2020. xFraud: explainable fraud transaction detection. *arXiv preprint arXiv:2011.12193*.

Sarma, D.; Alam, W.; Saha, I.; Alam, M. N.; Alam, M. J.; and Hossain, S. 2020. Bank fraud detection using community detection algorithm. In *2020 second international conference on inventive research in computing applications (ICIRCA)*, 642–646. IEEE.

Shwartz-Ziv, R.; and Armon, A. 2022. Tabular data: Deep learning is not all you need. *Information Fusion*, 81: 84–90.

Teichmann, F. 2020. Recent trends in money laundering. *Crime, Law and Social Change*, 73: 237–247.

Temizsoy, A.; Iori, G.; and Montes-Rojas, G. 2017. Network centrality and funding rates in the e-MID interbank market. *Journal of Financial Stability*, 33: 346–365.

Xiang, S.; Zhu, M.; Cheng, D.; Li, E.; Zhao, R.; Ouyang, Y.; Chen, L.; and Zheng, Y. 2023. Semi-supervised credit card fraud detection via attribute-driven graph representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 14557–14565.

Xu, Y.; and Corbett, J. 2019. Using network method to measure financial interconnection. Technical report, National Bureau of Economic Research.

Ying, Z.; Bourgeois, D.; You, J.; Zitnik, M.; and Leskovec, J. 2019. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32.

Zhang, G.; Wu, J.; Yang, J.; Beheshti, A.; Xue, S.; Zhou, C.; and Sheng, Q. Z. 2021. Fraudre: Fraud detection dual-resistant to graph inconsistency and imbalance. In *2021 IEEE International Conference on Data Mining (ICDM)*, 867–876. IEEE.