

Deep Learning for Fraud Detection in Financial Transactions: A Novel Approach to Detect Hidden Anomalies



MCS

Author:

Nabeel Ahmad

(Registration No: **00000431936**)

Supervisor

Lt Col Dr Yasir Awais Butt

A thesis submitted to the faculty of Computer Software Engineering Department, Military College of Signals, National University of Sciences and Technology, Islamabad, Pakistan in partial fulfilment of the requirements for the degree of Master of Science in Computer Science

(Sep 2024)

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by Maj Nabeel Ahmad, Registration No.00000431936, of Military College of Signals has been vetted by undersigned, found complete in all respect as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial, fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the student have also been incorporated in the said thesis.

Signature: _____

Supervisor: Lt Col. Dr. Yasir Awais Butt

Date: _____

Signature (HoD): Brig Dr. Adnan Ahmad Khan

Date: _____

Signature (Dean/Principal): _____

Date: 23/9/24


Brig
Dean, MCS (NUST)
(Asif Masood, PhD)

NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY
MASTER THESIS WORK

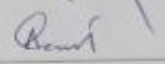
We hereby recommend that the dissertation prepared under our supervision by Maj Nabeel Ahmed, Regn No 00000431936 Titled: "Deep Learning for Fraud Detection in Financial Transactions: A Novel Approach to Detect Hidden Anomalies" be accepted in partial fulfillment of the requirements for the award of MS Computer Sciences degree.

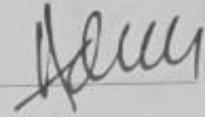
Examination Committee Members

1. Name: Maj Wajahat Sultan


Signature: 

2. Name: Asst Prof Dr. Asad Ullah

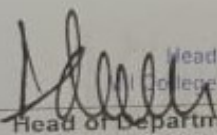
Signature: 

Co-Supervisor's Name Brig Adnan Ahmed Khan, PhD Signature: 

Supervisor's Name: Lt Col Yasir Awais Butt, PhD

Signature: 

Date: _____


Brig
Head of Dept of CSE
College of Sigs (NUST)
Head of Department

19/9/24
Date

COUNTERSIGNED

Date: 23/9/24


Brig
Dean, MCS (NUST)
(Asif Masood, Phd)
Dean

CERTIFICATE OF APPROVAL

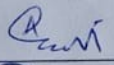
This is to certify that the research work presented in this thesis, entitled "Deep Learning for Fraud Detection in Financial Transactions: A Novel Approach to Detect Hidden Anomalies" was conducted by Nabeel Ahmad under the supervision of Lt Col Dr. Yasir Awais Butt. No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the Computer Software Engineering Department of Military College of Signals in partial fulfillment of the requirements for the degree of Master of Science in Field of Computer Software Engineering, Department of Software Engineering National University of Sciences and Technology, Islamabad.

Student Name: Nabeel Ahmad

Signature: 

Examination Committee:

a) External Examiner 1: Asst Prof Dr. Asad Ullah

Signature: 

b) External Examiner 2: Maj Wajahat Sultan

Signature: 

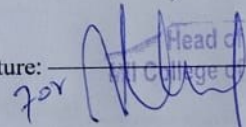
Name of Co-Supervisor: Brig Dr. Adnan Ahmed Khan

Signature: 

Name of Supervisor: Lt Col Dr. Yasir Awais Butt

Signature: 

Name of Dean/HOD: Brig Adnan Ahmed Khan

Signature:  Head of Dept of CSE
Military College of Signals (NUST)

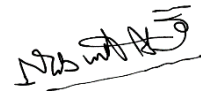
PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the thesis titled “**Deep Learning for Fraud Detection in Financial Transactions: A Novel Approach to Detect Hidden Anomalies**” is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero-tolerance policy of the HEC and National University of Sciences and Technology (NUST), Islamabad towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the University reserves the rights to withdraw/revoke my MS degree and that HEC and NUST, Islamabad has the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized thesis.

Student Signature:




Name: Maj Nabeel Ahmed

Date: Sep 2024

DECLARATION

I, Maj Nabeel Ahmed, hereby state that my MS thesis titled “**Deep Learning for Fraud Detection in Financial Transactions: A Novel Approach to Detect Hidden Anomalies**” is my own work and has not been submitted previously by me for taking any degree from National University of Sciences and Technology, Islamabad or anywhere else in the country/ world. At any time if my statement is found to be incorrect even after I graduate, the university has the right to withdraw my MS degree.

Student Signature:



Name: Maj Nabeel Ahmed

Date: Sep 2024

DEDICATION

“In the name of Allah, the most Beneficent, the most Merciful”

This research work is dedicated.

to

MY PARENTS, TEACHERS, AND FRIENDS

for their prayers, love, and endless support

whose encouragement remained the strongest pillar in successful completion of my thesis.

ACKNOWLEDGEMENTS

All praises to Allah for the strengths and His blessing in completing this thesis.

I would like to convey my gratitude to my supervisor, Lt Col Dr. Yasir Awais Butt, PhD, for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the experimental and thesis works are major contributions of this research's success.

I would also convey my special thanks to my Co-Supervisor, Brig Dr. Adnan Ahmed Khan, PhD, for his way of teaching and clearing concepts guidance towards completion of my thesis.

Also, I would thank my committee members; Maj Wajahat Sultan, and Asst Prof Dr. Asad Ullah for their support and knowledge regarding this topic. I would like to extend my feelings of gratitude towards my mother and my Father-in-law Muhammad Munir Khan (Retd Head Master) for their deemed necessary prayers all the time.

The most important part of my life my wife Dr. Fozia Munir and my children Essa Khan and Iltejah Nabeel who remained throughout with me in tough times of my thesis and achievements of results, their care, love, and endless support encouraged me throughout my times of stress and excitement.

Moreover, I am highly thankful to my all teachers who are my mentor. They have always stood by my dreams and aspirations and have been a great source of inspiration for me.

ABSTRACT

Money laundering, a critical issue in financial systems worldwide, involves the process of making illicitly-gained proceeds appear legitimate. As financial transactions grow increasingly complex, it has become harder for traditional methods to detect and prevent laundering activities effectively. The rise of sophisticated techniques such as cross-currency transactions and rapidly evolving fraudulent practices necessitates the development of more advanced, automated approaches for identifying suspicious activity. This research introduces a novel graph-based approach for detecting money laundering using advanced machine learning models—particularly Graph Convolutional Networks (GCN), GraphSAGE, and our proposed models, Adaptive Sampling Aggregated Graph Convolutional Network (ASA-GCN) and ASA-GNN. These models are designed to process graph-based data, such as financial transactions, and identify suspicious activity based on the relationships and interactions between entities. The primary objective of this thesis is to propose, develop, and evaluate a robust model for detecting money laundering. The models were trained and tested on the IBM Anti-Money Laundering (AML) dataset, which includes simulated financial transactions representing both legitimate and fraudulent activities. This dataset, rich in attributes such as transaction timestamps, amounts, currencies, and identifiers for the originating and receiving accounts, provides an ideal testing ground for assessing the performance of graph-based models. The results of this study demonstrate that the proposed ASA-GCN model consistently outperforms traditional graph-based models and baseline machine learning methods across several key metrics. ASA-GCN achieves an Area Under the Curve (AUC) score of 0.99, far exceeding the performance of GCN, GraphSAGE, and GAT, which typically range between 0.75 and 0.80. In addition, ASA-GCN demonstrates higher precision and recall, with an average precision (AP) score of 0.98, indicating its superior ability to identify both money

laundering transactions and non-money laundering transactions with minimal false positives and false negatives. Beyond the technical performance, this research highlights the interpretability of the ASA-GCN model. By examining the attention weights and node representations, we are able to understand how the model identifies suspicious transactions. This interpretability is essential for financial institutions that require transparency in their decision-making processes for compliance with anti-money laundering regulations. The findings of this thesis have broad implications for the future of anti-money laundering systems. As financial crimes become more intricate and the datasets grow larger, the use of graph-based machine learning models like ASA-GCN could revolutionize how banks, financial institutions, and regulatory agencies detect and prevent money laundering. The ability to process vast amounts of transactional data in real-time and accurately detect fraudulent activities makes these models indispensable tools in the fight against financial crime. In conclusion, this thesis presents ASA-GCN as a state-of-the-art model for money laundering detection. With its high accuracy, scalability, and interpretability, it holds great promise for practical deployment in financial institutions. However, future research could focus on optimizing the training time and exploring transfer learning methods to extend the applicability of the model to different domains. The potential for real-time implementation also opens avenues for further exploration, ensuring that financial institutions can stay ahead in the rapidly evolving landscape of financial crime.

Keywords: Fraud Detection, Financial Transactions, Graph Neural Networks, Anomaly Detection, Machine Learning

Table of Contents

ACKNOWLEDGEMENTS.....	viii
ABSTRACT.....	ix
LIST OF TABLES	xiv
LIST OF FIGURES	xvi
CHAPTER 1: INTRODUCTION.....	1
1.1. Fraud Detection in Financial Transactions.....	1
1.2. Conventional Fraud Detection Methods	2
1.3. Rule-Based Systems.....	2
1.3.1. Statistical Methods.....	2
1.3.2. Machine Learning Approaches	2
1.4. Challenges in Fraud Detection.....	3
1.4.1. Data Imbalance	3
1.4.2. Evolving Fraud Tactics	3
1.4.3. Scalability Issues.....	3
1.5. Background Study	4
1.6. Research Motivations.....	5
1.7. Problem Statement	6
1.8. Problem Formulations.....	7
1.8.1. Graph Anomaly Detection Optimization	7
1.8.2. Class Imbalance Handling in Fraud Detection	8
1.9. Research Aim and Objectives	9
1.10. Research Questions.....	10

1.11. Research Scope and Limitations	10
1.11.1. Research Scope	10
1.11.2. Research Limitations	10
1.12. Thesis Structure	11
CHAPTER 2: LITERATURE REVIEW	13
2.1. Introduction	13
2.2. Related Work	13
2.2.1. Importance of Fraud Detection in Financial Transactions	13
2.2.2. Traditional Fraud Detection Methods	17
2.2.3. Anomalies in Financial Transactions	20
2.2.4. GNN for Anomaly Detection	23
2.3. Literature Summary	30
2.4. Research Gap	30
CHAPTER 3: METHODOLOGY	31
3.1. Introduction	31
3.2. Dataset Description	32
3.2.1. Attributes of the Dataset	33
3.2.2. Dataset Size and Structure	39
3.3. Data Preprocessing	39
3.3.1. Encoding Categorical Features	39
3.3.2. Normalization of Numerical Features	40
3.3.3. Handling Class Imbalance	40
3.4. Proposed Model: ASA-GCN	40
3.4.1. Model Architecture	40
3.4.2. Training Procedure	44

3.5. Evaluation Metrics	46
3.5.1. Accuracy	46
3.5.2. Precision, Recall, and F1-Score.....	46
3.5.3. ROC-AUC	47
3.5.5. Additional Metrics	48
3.6. Conclusion.....	48
CHAPTER 4: RESULTS AND DISCUSSIONS	49
4.1 Introduction	49
4.2 Performance Metrics	49
4.2.1 AUC and ROC Curves Analysis.....	49
4.3 Loss Function Comparison.....	51
4.4 Neighbor Sample Size Impact.....	52
4.5 Node Representation Analysis	54
4.6 Impact of Bins and Thresholds on Model Performance.....	55
4.7 Model Complexity and Training Time Analysis	56
4.8 Model Scalability and Generalization.....	58
4.9 Discussion on the Interpretability of ASA-GCN.....	58
4.10 Summary of Findings	59
CHAPTER 5: CONCLUSIONS	61
5.1 Key Findings	61
5.1.1 Performance of ASA-GCN.....	61
5.2 Limitations	61
5.3 Recommendations	62
5.4 Future Work.....	63
5.5 Conclusion.....	64

REFERENCE..... 66

LIST OF TABLES

Table 2.1 Comparative table of previous study 27

Table 3.1 Dataset Feature Description 34

Table 4.1 ROC Curves of different models..... 50

Table 4.2 Loss over batches for different loss functions..... 52

Table 4.3 Performance comparison (Recall, F1 Score, and AUC) with varying neighbor sample sizes across six datasets 53

Table 4.4 Visualization of node representations (blue: legitimate nodes; red: fraudulent nodes) for different models: (a) GCN, (b) GraphSAGE, (c) GAT, (d) SSA, (e) RGCN, (f) ASA-GNN, and (g) ASA-GCN. 54

Table 4.5 Impact of Bins and Thresholds on Model Performance..... 55

Table 4.6 Model Complexity and Training Time Analysis 57

Table 4.7 Model Scalability and Generalization.....	58
Table 4.8 Discussion on the Interpretability of ASA-GCN	59
Table 4.9 Summary of Findings.....	59

LIST OF FIGURES

Figure 3.1 Proposed Work Flow	32
Figure 3.2 Cross Currency Transactions vs Non-Cross Currency Transactions.....	36
Figure 3.3 Transaction by Currency.....	36
Figure 3.4 Distribution of Payment Currency in Money Laundering.....	37
Figure 3.5 Laundering Transaction counts by Currency.....	37
Figure 3.6 Number of Money Laundering Transactions Date Wise	38
Figure 3.7 Number of Non-Money Laundering Transactions Date Wise.....	38
Figure 3.8 Distribution of Hour for 'Is Laundering = 1'	39
Figure 3.9 Model Architecture.....	44
Figure 4.1 ROC Curves of different models (GCN, GraphSAGE, GAT, CARE-GNN, SSA, RGCN, HAN, ASA-GNN, and ASA-GCN) across six datasets: HI Large, HI Medium, HI Small, LI Large, LI Medium, and LI Small. ASA-GCN consistently achieves the highest AUC score (0.99) across all datasets, while ASA-GNN follows closely with an AUC of 0.95.....	50
Figure 4.2 (Left) AUC over epochs for ASA-GCN and ASA-GNN, showing that ASA-GCN maintains a consistently high AUC above 0.9 across epochs, while ASA-GNN's performance deteriorates over time. (Right) Loss over batches for different loss functions (Margin Loss, Focal Loss, and Cross Entropy Loss), where all loss functions show rapid convergence with Margin Loss starting at a higher point and converging later compared to Focal and Cross Entropy Loss.	51
Figure 4.3 Performance comparison (Recall, F1 Score, and AUC) with varying neighbor sample sizes across six datasets: (a) HI Large, (b) HI Medium, (c) HI Small, (d) LI Large, (e) LI Medium, and (f) LI Small. As the neighbor sample size increases, all metrics (Recall, F1, and	

AUC) generally improve across the datasets, with higher scores observed in the HI datasets compared to the LI datasets. 53

Figure 4.4 Visualization of node representations (blue: legitimate nodes; red: fraudulent nodes) for different models: (a) GCN, (b) GraphSAGE, (c) GAT, (d) SSA, (e) RGCN, (f) ASA-GNN, and (g) ASA-GCN..... 54

Figure 4.5 Performance analysis across six datasets (HI Small, LI Small, HI Medium, LI Medium, HI Large, and LI Large). 55

Figure 4.6 Heatmap comparing the performance of various models (GCN, GraphSAGE, GAT, SSA, RGCN, ASA-GNN, and ASA-GCN) across different datasets (HI-Small, LI-Small, HI-Medium, LI-Medium, HI-Large, LI-Large)..... 57

CHAPTER 1: INTRODUCTION

1.1. Fraud Detection in Financial Transactions

Fraud detection in financial transactions is a crucial aspect of maintaining the security and integrity of financial systems. With the rise of digital banking, online shopping, and electronic payments, the volume of financial transactions has surged, increasing the opportunities for fraudulent activities (Hou et al., 2023). Fraudulent transactions can include various malicious activities such as unauthorized use of credit cards, identity theft, money laundering, and cyber-attacks. These activities not only lead to significant financial losses but also erode customer trust and pose substantial risks to the stability of financial institutions (Tian et al., 2023a, 2023b).

Effective fraud detection involves the continuous monitoring and analysis of transaction data to identify irregularities and patterns indicative of fraudulent behavior. Traditional fraud detection methods, such as rule-based systems and statistical models, have been the backbone of fraud prevention strategies (Tian et al., 2023a). However, these methods often struggle to keep pace with the sophisticated and evolving tactics employed by fraudsters. The complexity of financial data, characterized by high volume, velocity, and variety, further complicates the detection process (Cherif et al., 2024).

Hidden anomalies, or subtle irregularities in transaction data, present additional challenges in fraud detection. These anomalies may not be immediately apparent but can indicate fraudulent activities that evade traditional detection methods. Hidden anomalies can result from complex fraud schemes where multiple fraudulent transactions are interwoven with legitimate ones, making them difficult to detect using conventional approaches (Bala, 2023). Advanced techniques, such as machine learning and, more recently, Graph Neural Networks (GNNs), have shown promise in uncovering these hidden patterns. GNNs, in particular, excel at modeling relationships within data, allowing for a deeper understanding of the interconnected nature of financial transactions (Wen et al., 2024). By representing transactions as graphs, GNNs can capture the intricate links between accounts, merchants, and other entities, thereby identifying anomalies that might otherwise remain hidden (Zareapoor et al., 2012).

1.2. Conventional Fraud Detection Methods

Conventional fraud detection methods encompass a range of approaches traditionally used to identify and mitigate fraudulent activities in financial transactions. These methods can be broadly classified into rule-based systems, statistical methods, and machine learning approaches.

1.3. Rule-Based Systems

Rule-based systems are one of the earliest and simplest methods for detecting fraud. They rely on predefined rules and thresholds set by experts to flag suspicious transactions. For example, a rule might trigger an alert if a single account exceeds a certain transaction limit within a short period (Hou et al., 2023; Tian et al., 2023b). While rule-based systems are easy to implement and understand, they lack flexibility and adaptability, often resulting in a high number of false positives and an inability to detect new, sophisticated fraud patterns (Bala, 2023).

1.3.1. Statistical Methods

Statistical methods employ mathematical models to analyze transaction data and identify anomalies. Techniques such as regression analysis, clustering, and outlier detection are commonly used. These methods can handle larger datasets and provide a more nuanced understanding of transaction patterns compared to rule-based systems (Wen et al., 2024; Zareapoor et al., 2012). However, statistical methods still face limitations in their ability to adapt to evolving fraud tactics and may struggle with high-dimensional data inherent in financial transactions (Bukhori & Munir, 2023).

1.3.2. Machine Learning Approaches

Machine learning approaches leverage algorithms that can learn from historical data to detect fraudulent activities. These methods include supervised learning techniques, such as decision trees and neural networks, and unsupervised learning techniques, like clustering and anomaly detection. Machine learning models can adapt to new patterns and improve over time with more data (Bukhori & Munir, 2023; Umer et al., 2023). Despite their advantages, these models often require

extensive feature engineering and may still struggle with the dynamic and complex nature of financial fraud.

1.4. Challenges in Fraud Detection

Fraud detection in financial transactions faces several significant challenges that complicate the development and implementation of effective detection systems. These challenges include data imbalance, evolving fraud tactics, and scalability issues.

1.4.1. Data Imbalance

Data imbalance refers to the disproportionate ratio of fraudulent to non-fraudulent transactions in financial datasets. Typically, fraudulent transactions constitute a very small percentage of the total transactions, making it difficult for detection models to accurately identify fraud without being biased towards the majority class. This imbalance can lead to a high number of false negatives, where fraudulent activities go undetected (Innan et al., 2024; Kuttiyappan & V, 2024).

1.4.2. Evolving Fraud Tactics

Fraud tactics are continuously evolving as fraudsters develop new methods to bypass detection systems. This constant evolution poses a significant challenge, as detection models need to be regularly updated and retrained to keep up with the latest fraud strategies (Innan et al., 2024). Static models quickly become obsolete, leading to gaps in detection and increased vulnerability to new types of fraud.

1.4.3. Scalability Issues

Scalability issues arise from the need to process and analyze vast amounts of transaction data in real-time. Effective fraud detection systems must be capable of handling high transaction volumes without compromising on speed or accuracy. As financial institutions grow and the number of transactions increases, the computational demands on fraud detection systems also escalate, necessitating scalable and efficient solutions (Nahar et al., 2016).

1.5. Background Study

Fraud detection in financial transactions has evolved significantly over the past few decades, driven by the growing complexity and volume of financial data. The global financial industry, with its rapid digitization, has witnessed an alarming increase in fraudulent activities (Xu et al., 2024). According to the Association of Certified Fraud Examiners (ACFE), organizations lose an estimated 5% of their annual revenues to fraud, with global losses totaling over \$4.5 trillion annually (Cherif et al., 2024). This immense scale of financial fraud underscores the critical need for advanced detection mechanisms that can effectively mitigate these risks (Wen et al., 2024).

Traditional fraud detection methods have relied heavily on rule-based systems and statistical models. Rule-based systems operate on predefined rules and thresholds set by experts (Innan et al., 2024). While these systems are straightforward and easy to implement, they are often rigid and incapable of adapting to new and sophisticated fraud tactics. Statistical models, on the other hand, use historical data to identify anomalies through techniques such as regression analysis, clustering, and outlier detection (Kuttiyappan & V, 2024). Despite their enhanced analytical capabilities, these models also face limitations in scalability and adaptability, particularly as fraud tactics evolve (Motie & Raahemi, 2024).

Machine learning approaches have emerged as a more robust solution, leveraging algorithms that learn from historical data to improve fraud detection accuracy. These methods include supervised learning techniques like decision trees, random forests, and neural networks, as well as unsupervised learning methods such as k-means clustering and principal component analysis (Hou et al., 2023; Tian et al., 2023b). However, even with the advent of machine learning, challenges persist, particularly in terms of feature engineering, model interpretability, and handling the high-dimensional nature of financial transaction data (Bala, 2023; Tian et al., 2023a).

Graph Neural Networks (GNNs) represent a significant advancement in the field of fraud detection, offering a novel approach to modeling the relationships within transaction data. GNNs excel at capturing the interconnected nature of financial networks, where transactions are represented as graphs with nodes (e.g., accounts, merchants) and edges (e.g., transactions) (Bukhori & Munir, 2023; Umer et al., 2023). This graphical representation allows GNNs to

uncover complex patterns and relationships that traditional methods may overlook, providing a more comprehensive and adaptive solution for detecting fraudulent activities (Shahzadi, 2023; J. Tang et al., 2022).

Several high-profile financial fraud cases highlight the severity and impact of fraudulent activities worldwide. One notable example is the Enron scandal, where executives used accounting loopholes and special purpose entities to hide billions in debt from shareholders, leading to the company's bankruptcy in 2001. Another significant case is the Bernie Madoff Ponzi scheme, which defrauded investors of approximately \$65 billion over several decades (Nahar et al., 2016; Zareapoor et al., 2012). More recently, the Wirecard scandal in Germany, where the company falsified accounts to inflate profits, resulted in the loss of nearly \$2 billion. These cases not only resulted in massive financial losses but also triggered regulatory reforms and increased scrutiny on financial practices globally.

1.6. Research Motivations

The primary motivation behind this research is to address the significant limitations of conventional fraud detection methods in handling the complexity and dynamic nature of financial transactions (J. Tang et al., 2022). Traditional approaches, such as rule-based systems and standard machine learning algorithms, often fail to capture the intricate relationships between different entities involved in financial transactions. These methods rely heavily on manual feature engineering and struggle to adapt to evolving fraud tactics, leading to a high rate of undetected fraud and false positives (Wen et al., 2024). The novel use of Graph Neural Networks (GNNs) aims to overcome these challenges by leveraging their ability to model complex, interconnected data, providing a more robust and adaptive solution for fraud detection (Bukhori & Munir, 2023; Umer et al., 2023).

Another key motivation is to tackle the problem of data imbalance prevalent in financial fraud detection datasets. Fraudulent transactions are typically rare compared to legitimate ones, making it difficult for traditional models to accurately identify fraud without being biased towards the majority class (Cherif et al., 2024; Tian et al., 2023a; Xu et al., 2024). This research seeks to develop a GNN-based model that can effectively manage this imbalance, ensuring higher precision

and recall rates. By doing so, the study aims to improve the overall reliability and accuracy of fraud detection systems, reducing financial losses and enhancing trust in digital financial transactions (Bala, 2023).

Furthermore, the practical deployment of advanced fraud detection models in real-time financial systems is a critical aspect often overlooked in existing research. This study emphasizes the importance of designing a scalable and efficient deployment architecture that can be seamlessly integrated into live financial environments (Innan et al., 2024; Wen et al., 2024). By addressing these practical considerations, the research not only contributes to the theoretical advancement of fraud detection techniques but also ensures their applicability and effectiveness in real-world scenarios, thereby providing tangible benefits to financial institutions and their customers (Bukhori & Munir, 2023; Umer et al., 2023).

1.7. Problem Statement

The increasing complexity and evolving nature of financial transactions have made them susceptible to sophisticated fraudulent activities, posing significant risks to individuals, financial institutions, and the overall financial system (Cherif et al., 2024; Xu et al., 2024). Traditional fraud detection methods, such as rule-based systems and machine learning algorithms, have become inadequate in addressing these challenges due to their reliance on manual feature engineering and limited capacity to adapt to new fraud tactics (Innan et al., 2024; Wen et al., 2024). These conventional approaches often fail to capture the intricate and dynamic relationships inherent in financial data, leading to substantial financial losses and diminished trust in the financial market (Kuttiyappan & V, 2024; Motie & Raahemi, 2024). The need for a more robust and intelligent anomaly detection system has never been greater. GNNs, with their ability to model complex relationships within data, offer a promising solution for detecting hidden anomalies in financial transactions (Shahzadi, 2023; J. Tang et al., 2022; Umer et al., 2023). This research aims to develop a GNN-based model specifically designed to identify and prevent fraudulent activities in real-time, addressing the limitations of current methods. By leveraging the interconnected nature of financial transactions, the proposed model seeks to enhance the accuracy and efficiency of fraud detection, providing a scalable and adaptable solution for financial institutions and e-commerce platforms (Bala, 2023; Bukhori & Munir, 2023; Tian et al., 2023a).

1.8. Problem Formulations

1.8.1. Graph Anomaly Detection Optimization

The primary challenge addressed by this research is the detection of fraudulent activities in financial transactions using Graph Neural Networks (GNNs). Traditional methods often fail to capture the complex, dynamic relationships between entities in financial networks, resulting in inadequate detection of hidden anomalies. The objective is to develop a GNN-based model that can effectively identify these anomalies by modeling the intricate connections within transaction data. To formalize the problem, let $G = (V, E)$ represent a graph where V denotes the set of nodes (representing accounts, merchants, etc.) and E denotes the set of edges (representing transactions). Each node v_i has a feature vector x_i and each edge e_{ij} has a weight w_{ij} . The goal is to detect anomalies a in the graph by learning a function f that maps the graph data to an anomaly score.

Mathematically, this can be expressed as:

$$a = f(G, \Theta) \quad \dots (1.1)$$

where Θ represents the model parameters. The anomaly detection can be framed as an optimization problem where the objective function L seeks to minimize the detection error while satisfying certain constraints.

Objective Function and Constraints

$$\min_{\Theta} L(a, a^*) = \frac{1}{N} \sum_{i=1}^N \text{Loss}(a_i, a_i^*) \quad \dots (1.2)$$

subject to:

Constraint 1

$$\sum_{i=1}^N w_{ij} \cdot \text{Feature}_i \geq \text{Threshold} \quad \dots (1.3)$$

Constraint 2

$$\text{Complexity}(\Theta) \leq \text{Budget} \quad \dots (1.4)$$

where a^* represents the true anomaly labels, Loss is a loss function (e.g., cross-entropy), and Complexity refers to computational resources.

The formulation aims to optimize the GNN model to accurately predict anomalies in financial transactions. The objective function L measures the discrepancy between predicted and true anomaly scores, while the constraints ensure the model's performance and computational efficiency. The constraints ensure that the model's complexity remains manageable and that it effectively captures significant transactional patterns, thereby enhancing fraud detection capabilities.

1.8.2. Class Imbalance Handling in Fraud Detection

Another critical problem in fraud detection is handling the imbalance between fraudulent and non-fraudulent transactions within the dataset. Traditional models may perform poorly due to the skewed distribution of fraud cases, necessitating a robust approach that effectively handles this imbalance. Let $D = \{(x_i, y_i)\}_{i=1}^N$ be a dataset where x_i represents the feature vector of a transaction and $y_i \in \{0,1\}$ indicates whether the transaction is fraudulent (1) or non-fraudulent (0). The goal is to learn a function g that maps x_i to the probability p_i of fraud.

Mathematically, the problem can be defined as:

$$p_i = g(x_i, \Theta) \quad \dots (1.4)$$

The objective is to minimize a loss function that accounts for class imbalance.

Objective Function and Constraints

$$\min L(p_i, y_i) = \frac{1}{N} \sum_{i=1}^N \text{WeightedLoss}(p_i, y_i) \quad \dots (1.5)$$

subject to:

Constraint 1

$$\text{Precision} \geq \text{MinPrecision} \quad \dots (1.6)$$

Constraint 2

$$\text{Recall} \geq \text{MinRecall} \quad \dots (1.7)$$

where Weighted-Loss adjusts the loss function to account for class imbalance, and Precision and Recall are performance metrics to be satisfied.

This formulation focuses on addressing class imbalance in fraud detection. The objective function L incorporates weighting to balance the influence of fraudulent and non-fraudulent transactions. The constraints ensure that the model achieves acceptable precision and recall rates, making it effective in detecting fraud despite the skewed class distribution.

1.9. Research Aim and Objectives

This research aims to develop a robust ASAGCN-based model to detect anomalies and fraudulent activities in financial transactions. By leveraging the complex relationships within transaction data, this study seeks to enhance the accuracy and efficiency of fraud detection systems.

- To explore and address the intrinsic challenge of data imbalance in fraud detection.
- To develop an ASAGCN-based model for detecting anomalies and fraudulent activities in financial transactions.
- To evaluate the effectiveness of the proposed model using real-world financial transaction datasets and compare the performance of the proposed model against existing GNN models.
- To design a robust deployment architecture for real-time anomaly detection and ensure practical viability in live financial systems.

1.10. Research Questions

This research seeks to answer critical questions surrounding the application of GNNs in fraud detection within financial transactions. By addressing these questions, the study aims to develop and validate a model that significantly enhances fraud detection accuracy and efficiency.

1. How can the intrinsic challenge of data imbalance in fraud detection be effectively addressed using ASAGCN?
2. What is the effectiveness of a ASAGCN-based model in detecting anomalies and fraudulent activities in financial transactions compared to existing methods?
3. How can the proposed ASAGCN based model be deployed in real-time to ensure practical viability and scalability in live financial systems?
4. How do the relationships and patterns captured by ASAGCN within transaction data improve the detection capabilities of fraud detection systems?

1.11. Research Scope and Limitations

1.11.1. Research Scope

This research focuses on developing a GNN-based model to enhance the detection of fraudulent activities in financial transactions. The study involves exploring the intrinsic challenges associated with data imbalance, leveraging GNNs to capture complex relationships within transaction data, and evaluating the effectiveness of the proposed model using real-world financial datasets. Additionally, the research aims to design a robust deployment architecture for real-time anomaly detection, ensuring the practical viability and scalability of the model in live financial systems. The primary application areas include banks, payment processors, e-commerce platforms, financial regulatory bodies, and insurance companies, where the need for accurate and efficient fraud detection systems is paramount.

1.11.2. Research Limitations

This study acknowledges certain limitations that may impact the findings and conclusions.

- The availability and quality of real-world financial transaction datasets may constrain the model's training and evaluation processes.
- The proposed ASAGCN-based model may require significant computational resources, potentially limiting its accessibility and deployment in resource-constrained environments.
- The study's scope is limited to specific types of financial transactions and may not account for all possible fraud scenarios.
- The effectiveness of the model in adapting to rapidly evolving fraud tactics may require continuous updates and retraining.
- Integration of the model into existing financial systems may pose practical challenges and necessitate significant modifications to current infrastructure.

1.12. Thesis Structure

1. Chapter Introduction

This chapter provides an overview of the research topic, highlighting the increasing vulnerability of financial transactions to fraud and the limitations of traditional detection methods. It outlines the research objectives, research questions, and the significance of developing a GNN-based model for fraud detection. The chapter also presents the scope and contributions of the study.

2. Chapter Literature Review

This chapter reviews existing literature on fraud detection methods, focusing on conventional approaches such as rule-based systems, statistical methods, and machine learning techniques. It introduces GNNs and discusses their application in financial fraud detection, identifying gaps and limitations in current research that this study aims to address.

3. Chapter Methodology

This chapter details the research design and methodology employed in developing the ASAGCN-based fraud detection model. It describes the dataset preparation, model architecture, training procedures, and evaluation metrics. The chapter also outlines the deployment architecture for real-time anomaly detection and the steps taken to address data imbalance.

4. Chapter Results and Discussions

This chapter presents the experimental results, comparing the performance of the proposed ASAGCN-based model against existing methods. It discusses the findings in the context of fraud detection accuracy, efficiency, and adaptability to evolving fraud tactics. The chapter also explores the practical implications of deploying the model in live financial systems.

5. Chapter Conclusions

This chapter summarizes the key findings and contributions of the research, highlighting the effectiveness of the ASAGCN-based model in detecting financial fraud. It discusses the limitations of the study and provides recommendations for future research, including potential improvements and extensions of the proposed model.

CHAPTER 2: LITERATURE REVIEW

2.1. Introduction

This chapter delves into the advancements and current state of Graph Neural Networks (GNNs) in the realm of fraud detection within financial transactions. This study explores how GNNs, with their capacity to model complex relationships and identify subtle anomalies, have revolutionized fraud detection by improving upon traditional methods. The chapter highlights recent innovations, including novel approaches that integrate structural information and synthetic data, and examines the effectiveness of unsupervised learning techniques in adapting to evolving fraud patterns. It also addresses the limitations of current models, such as their specificity to certain fraud types and challenges with real-world data simulation. By reviewing these developments and identifying existing research gaps, this chapter sets the stage for understanding the ongoing evolution and future directions of GNN applications in financial anomaly detection.

2.2. Related Work

2.2.1. Importance of Fraud Detection in Financial Transactions

Maintaining the integrity and confidence of financial systems depends critically on the identification of fraud in financial transactions. Y. Pei et al.,(Pei et al., 2020), introduced subgraph anomaly detection as one sophisticated technique for combating fraud. Using this method, transaction data is divided into smaller subgraphs, which are then examined for patterns that differ from normal activity. The technique focuses on finding odd relationships and interactions in these subgraphs that can point to fraud. The research by Pei et al. shows notable advancements in the identification of intricate fraud patterns that are frequently overlooked by conventional techniques. The findings demonstrate improved precision in spotting irregularities in transaction networks, offering a strong instrument for fraud detection. For real-time applications, this approach might be difficult because it involves significant data preparation and is computationally demanding. Subgraph anomaly detection techniques may also be limited in their scalability and practical deployment due to their high computing costs and data needs.

A technique to botnet detection based on graph-based machine learning, presented by A. Alharbi and K. Alsubhi (Alharbi & Alsubhi, 2021), is applicable to the detection of financial transaction fraud. Their approach leverages the structure and relationships between nodes to discover malicious patterns by modeling and analyzing the network of botnet activity using graph-based algorithms. The essential method is creating graphs from network data and using machine learning models to identify anomalies that point to botnet activity. Their results show that the method's analysis of intricate interaction patterns yields excellent detection accuracy and successful botnet identification. The outcomes show better effectiveness when compared to conventional detection techniques, especially when managing complex and elusive botnet strategies. High processing requirements and the requirement for a huge amount of training data, however, are the method's drawbacks and may affect scalability and real-time application in large-scale systems. In order to modify the strategy for dynamic fraud detection scenarios in financial transactions, it is imperative that these issues be resolved.

The AntiBenford subgraph architecture is presented by T. Chen and C. Tsourakakis (T. Chen & Tsourakakis, 2022), as a technique for identifying abnormalities in financial networks that is pertinent to the identification of fraud in financial transactions. With a particular focus on locating subgraphs where transaction patterns considerably deviate from Benford's law, this methodology applies Benford's law to detect deviations from predicted digit distributions in transaction data. Large-scale real data can be handled well by the approach, which uses an efficient algorithm that can find these Anti Benford subgraphs in near-linear time. The framework was assessed using both synthetic and actual data, showing that it can find aberrant subgraphs that other cutting-edge graph-based anomaly detection techniques would overlook. The Anti Benford framework has the capacity to reveal hidden anomalies, as demonstrated by empirical results that show how helpful it is for gaining insights into cryptocurrency transaction networks. Though its performance in different financial contexts may be affected by differences in transaction patterns, the method's dependence on Benford's rule may limit its usefulness in datasets where the law does not apply.

Through a comparison of machine learning and Graph Neural Networks (GNNs), Yoo, Shin, and Kyeong (Yoo et al., 2023), investigated the use of graph analysis for Medicare fraud detection. In order to evaluate the efficacy of GNN-based approaches in identifying fraudulent activity in Medicare claims, their research combines conventional machine learning techniques with them.

According to the study, GNNs perform better than conventional techniques because they take advantage of the intricate relationships that exist between the entities in the fraud network, which improves detection accuracy. According to their research, GNNs are able to identify complex fraud patterns that are frequently overlooked by traditional techniques. Nonetheless, the study also highlights some drawbacks, including the high computational expense of GNNs and the requirement for big, meticulously annotated datasets in order to properly train the models. These difficulties might affect how well the suggested solutions scale and work in real-time in more extensive financial situations.

Long et al (Long et al., 2023), make another important addition by introducing the MS_HGNN model, a hybrid online fraud detection system created to address graph-based fraud detection problems with data imbalance. In order to balance the dataset and enhance the identification of fraudulent activity in graph-based data, the MS_HGNN strategy incorporates several techniques. The model reduces the impacts of class imbalance and adjusts to changing data patterns by combining online learning procedures with GNNs. According to the findings, MS_HGNN performs better than other techniques in identifying fraud, especially in situations when the data distribution is skewed. However, the method's reliance on ongoing online learning can make model maintenance and performance tweaking more difficult, and it can need a lot of processing power to manage real-time updates efficiently.

In this paper author (Shi et al., 2022), Shi et al. presented the H2-FDetector, a GNN-based fraud detection system that takes advantage of links in transaction networks that are both homophilic and heterothallic. Their method seeks to improve fraud detection by capturing a wider variety of fraud patterns through the modeling of various node-to-node interactions. By utilizing an advanced GNN architecture, the H2-FDetector approach analyzes both comparable and dissimilar connections in the network, offering a more thorough comprehension of transactional irregularities. As evidenced by the findings, the dual connection strategy enhances detection robustness and accuracy, especially when it comes to spotting intricate fraud schemes. However, the method's efficacy is limited since it depends on the granularity and quality of the input data, in addition to the computer power needed to handle large-scale networks. These drawbacks show that while using such sophisticated fraud detection models, system performance and data quality must be carefully taken into account.

The adoption of Graph Neural Networks (GNNs), which provide sophisticated methodologies for recognizing and mitigating fraudulent behaviors, has greatly helped fraud detection in financial transactions. A. Correa-Bahnsen (Correa-bahnsen, 2021), makes a significant contribution to this field by introducing Relational Graph Neural Networks (RGNNs) designed specifically for super-app environments' fraud detection needs. This technique uses relational GNNs to represent intricate relationships between different entities like users and transactions within a super-app. RGNN technique successfully detects complex fraud patterns that may be overlooked by simpler models by concentrating on the connections and interactions between various nodes in the graph. According to the study's findings, RGNNs increase detection accuracy by taking multi-relational data into account, which is especially helpful in settings with a variety of interaction kinds. However, the method's actual application in super-app situations is limited because it necessitates a large amount of computational power and might not scale well enough to handle the massive amounts of data created there.

C. Chen et al. (C. Chen et al., 2019), introduced in Detect, a large-scale graph-based fraud detection system intended for the insurance industry, in the context of e-commerce. Across large datasets, InfDetect analyzes transaction networks and finds fraudulent activity using graph-based techniques. The architecture of the system makes use of sophisticated graph-based algorithms to handle and evaluate massive amounts of transaction data, making it possible to identify minor fraud trends that more conventional approaches could miss. Large-scale e-commerce systems can benefit from Inditex's high detection accuracy and scalability, as demonstrated by the study. It can be difficult, though, because of the system's complexity and the requirement for significant processing capacity to handle big datasets.

H. Pi (Pi, 2024), focuses on debiasing frequency adaptive GNN-based fraud detectors in his research into the issues related to bias in fraud detection. The issue of model bias that could arise from transaction data with uneven class distributions is addressed in this work. The proposed method aims to adjust the model's sensitivity to rare fraud events by including debiasing techniques within the GNN architecture. The results demonstrate that the debiasing strategy improves the model's ability to detect irregular fraud patterns while also lessening its detrimental effects on detection performance by removing class imbalance. Despite these improvements, the approach

still has limitations due to the computational cost of implementing debiasing algorithms and the potential challenge of optimizing the model for optimal performance. Ensuring the

Wu, Chao, and Li (Wu et al., 2024), present a unique use of heterogeneous graph neural networks (HGNNs) for supply chain financing fraud detection. Their approach makes use of HGNNs' capacity to model and examine intricate relationships between a variety of supply chain network participants, including customers, sellers, and financial institutions. Building heterogeneous networks that represent various kinds of connections and interactions between various nodes is the fundamental method. Due to its complete modeling methodology, which takes into account the complex relationships and transactions that occur inside the supply chain, fraud may be detected more accurately. The research reveals that the HGNN technique enhances transparency and comprehension of fraud trends by considerably increasing the accuracy of fraud detection and offering insightful justifications for anomalies that are found. The computational complexity of processing heterogeneous graphs and the difficulty of integrating diverse data kinds are the method's drawbacks, despite its advantages. In extensive supply chain finance contexts, these variables may affect the approach's scalability and practical application.

2.2.2. Traditional Fraud Detection Methods

Hu et al. (Hu et al., 2024), provide GAT-COBO, a cost-sensitive GNN intended especially for the detection of telecom fraud. Their method incorporates cost sensitivity into the GNN model to address the issue of differing costs related to false positives and false negatives in fraud detection. Graph Attention Networks (GATs) are used in the GAT-COBO approach to manage cost limitations and improve fraud detection by weighing the relevance of various nodes and edges depending on their related costs. The results show that, in comparison to conventional techniques, GAT-COBO performs better at identifying telecom fraud and successfully strikes a balance between the trade-offs between detection accuracy and cost. Nevertheless, the model's dependence on budget-conscious tactics adds more intricacy and computing requirements, potentially affecting its scalability and suitability for real-time implementation in extensive telecommunication networks.

In Player-to-Everyone (P2E) MMORPGs, where data is imbalanced with Positive and Unlabeled (PU) labels, Choi et al. (Choi et al., 2022), introduce PU GNN, a Graph Attention Network-based

method designed for chargeback fraud detection. In a situation where fraudulent transactions are uncommon and labeled data is hard to come by, the PU GNN model attempts to solve the problem of fraud detection. Despite the unbalanced nature of the data, the method makes use of graph attention processes to improve the detection of fraudulent patterns. As PU GNN can distinguish fraudulent transactions from a high amount of legitimate transactions, the study's findings show that PU GNN significantly enhances fraud detection performance in MMORPGs. However, dealing with PU labels and imbalanced datasets has intrinsic limits that limit the model's performance and may have an impact on its generalizability.

In order to identify fraudulent transactions in intricate e-commerce environments, Zhang et al. (G. Zhang et al., 2022), present EFraudCom, an e-commerce fraud detection system that uses competitive GNNs. To take advantage of each GNN architecture's advantages in identifying different kinds of fraudulent activity, the EFraudCom model combines them into a competitive framework. The technique improves accuracy and resilience of fraud detection by merging the advantages of several GNN models. The findings of the trial demonstrate that EFraudCom works better than current fraud detection systems by successfully detecting fraudulent transactions on e-commerce platforms. But the intricacy of the method and the demand for several GNN models can raise the computing burden, which might restrict its scalability and efficiency for real-time fraud detection applications.

The Fraud Aware Heterogeneous Graph Transformer (FAHGT) is a sophisticated model presented by Tang, Jin, and Cheng (S. Tang et al., 2021), that aims to improve fraud detection in online product review systems. The camouflage behavior of fraudsters and the inherent irregularities in diverse graph data are two major issues in fraud detection that the FAHGT model tackles. The complex, interactive nature of graph-structured data and the cunning strategies used by dishonest users are too much for traditional rule-based approaches to handle, which frequently leads to their failure. FAHGT uses a type-aware feature mapping mechanism to process heterogeneous graph data and a variety of relation scoring techniques to detect and reduce camouflage behaviors and inconsistencies in order to get around these restrictions. By combining features from nearby nodes, the model creates a reliable and insightful representation that greatly enhances fraud detection capabilities. Extensive trials yielded results that indicate the efficacy of FAHGT in detecting fraudulent activity in online review systems, outperforming numerous baselines across multiple

datasets. The model performs better at recognizing phony reviews in part because it can handle issues with both inconsistency and disguise. Nevertheless, the intricacy and processing demand of the FAHGT model also pose difficulties. The computational complexity associated with type-aware feature mapping and various relation scoring techniques may be prohibitive for large-scale systems' real-time use and scalability of the model. These difficulties show how important it is to continue striking a balance between model complexity and real-world issues when it comes to fraud detection.

Hou et al (Hou et al., 2023), present a novel method for detecting unsupervised fraudulent transactions on dynamic attributed networks. Their approach is centered on detecting fraudulent transactions in dynamic networks where node properties and connections vary over time. Rather than requiring labeled data, which is sometimes lacking in real-world circumstances, the suggested model makes use of dynamic graph representations to monitor transaction trends and identify anomalies. The study uses cutting-edge methods for anomaly scoring and dynamic feature extraction to demonstrate how well the model can recognize suspicious activity in real-time. Although the approach performs well in dynamic contexts, it has drawbacks due to the computational cost of processing large-scale, developing networks and its reliance on precise dynamic graph modeling.

Graph Neural Network (GNN) with AUC-oriented specifically for fraud detection is presented by Huang et al (Huang et al., 2022), To improve the model's capacity to discern between authentic and fraudulent transactions, their methodology places a strong emphasis on maximizing the Area Under the Curve (AUC) measure. To increase classification performance and resilience, the AUC-oriented GNN integrates cutting-edge loss functions and training techniques. Results from the experiments demonstrate how well the model works to achieve high AUC scores on a variety of datasets, which improves detection accuracy for jobs involving fraud detection. Its effectiveness may be impacted by the caliber and variety of the training data, and its emphasis on AUC optimization may restrict its application to situations where other assessment metrics are equally significant.

Zheng et al. (Zheng et al., 2023), presented MIDLG, a novel Dual-Level Graph Neural Network (GNN) for transaction fraud complaint verification that is based on Mutual Information. By

identifying local and global trends inside transaction networks, this method combines mutual information principles at two hierarchical levels to improve the detection of fraudulent transactions. The MIDLG's dual-level architecture facilitates thorough analysis by better separating authentic from fraudulent behavior by utilizing mutual knowledge. Their empirical data show that MIDLG offers improved sensitivity to subtle irregularities and outperforms current approaches in complaint verification and fraud detection accuracy. However, the model may not be as applicable in real-time applications due to its reliance on intricate mutual information computations and dual-level processing, which can lead to significant computational overhead and the need for intensive data pretreatment.

A GNN-based technique for identifying financial fraud by analyzing related party transaction networks is presented by Mao, Liu, and Wang (Mao et al., 2022). Their methodology centers on detecting questionable trends through the assessment of the connections and exchanges between interconnected entities. Their approach use GNN algorithms to effectively identify fraudulent behaviors that may be masked by the intricacies of related party transactions. The findings demonstrate that their technique successfully spots concealed fraud, giving financial institutions a useful tool. But transaction data completeness and quality have an impact on the model's performance, and difficulties scaling GNNs for large networks may reduce the model's usefulness in other contexts.

2.2.3. Anomalies in Financial Transactions

Shahzadi (Shahzadi, 2023), investigatesd how deep learning methods can be applied to business intelligence and information technology fraud detection. The main objective of the research is to search for fraudulent trends in big datasets by using sophisticated neural network designs. The technology improves fraud detection systems' accuracy and efficiency over conventional approaches by utilizing deep learning algorithms. The outcomes show that deep learning models are capable of greatly enhancing the recognition of intricate fraud cases. The drawbacks, however, include the high computational cost of training deep learning models and the need for large amounts of labeled data which isn't always available.

For the purpose of developing anti-money laundering models, Altman et al. (Altman et al., 2023), tackle the problem of producing realistic synthetic financial transaction data. Their approach aims to increase the resilience of anti-money laundering systems by presenting a methodology for creating synthetic data that closely resembles actual financial transactions. This study emphasized how crucial high-quality synthetic data is for both model validation and training in fraud detection. The outcomes demonstrate that it is possible to successfully employ the synthetic data produced to improve these models' performance. The potential differences between synthetic and real data, however, could limit the generalizability of the models trained on this data, which is one of the approach's drawbacks.

Using interpretative mask learning, Li et al. (K. Li et al., 2024), provide SEFraud, a graph-based self-explanatory fraud detection model. By revealing the model's decision-making process, this technique seeks to improve fraud detection systems' interpretability and transparency. Graph-based approaches and self-explanation mechanisms are used in the SEFraud approach to detect fraudulent actions and provide justifications for the predictions it makes. The empirical evaluation shows that SEFraud achieves a high degree of fraud detection accuracy and offers insightful information on the behavior of the model. However, real-time deployment may face difficulties due to the model's intricacy and the processing expense involved in creating interpretative masks.

A temporal and graph-based framework called TeGraF is proposed by Reddy et al. (Reddy et al., 2021), for the detection of fraudulent transactions. This approach detects anomalies in transaction networks by combining graph-based analysis with temporal dynamics. In order to detect fraudulent activity, TeGraF efficiently records the temporal patterns of transactions and their interactions within the graph structure. The findings show that by utilizing time-sensitive data, TeGraF surpasses conventional techniques and enhances fraud detection accuracy. The framework's dependence on temporal data, however, might make it less successful if the historical data is erroneous or lacking, which could have an impact on the model's functionality.

Wang and Yu (Wang & Yu, 2022), give a summary of how graph neural networks (GNNs) are used in anomaly detection, emphasizing how they can be used to spot fraud in intricate networks. Their talk focuses on several GNN designs and how they can use the graph structure and node interactions to capture abnormalities. According to the results, GNNs provide a substantial

improvement over traditional anomaly detection methods, yielding more precise and meaningful outcomes. However, the drawbacks include the scalability problems that arise when using GNNs on extremely large networks and the high processing overhead involved in training and optimizing these models.

Splitting, a spectral graph neural network created to tackle the problem of heterophily in fraud detection, is presented by Wu et al. in (Wu et al., 2023). Their method divides the graph into discrete subgraphs in order to handle the many kinds of links and connections that frequently make fraud detection more difficult. Spectral graph theory is incorporated into the SplitGNN model to better handle non-homogeneous nodes and edges, which can obfuscate conventional detection techniques. The outcomes show that SplitGNN works better than other models by correctly detecting fraudulent activity even when heterophilic data is present. However, the computational difficulty of spectral approaches and possible scalability problems when used to very large graphs are the method's drawbacks.

A thorough examination of cutting-edge AI-enhanced fraud detection techniques is given by Kuttiyappan and R. V. (Kuttiyappan & V, 2024). In order to enhance the functionality and precision of fraud detection systems, their research investigates a number of cutting-edge methods, such as machine learning and neural networks. In order to improve detection skills, the study presents the integration of AI approaches that make use of large datasets and complex algorithms. The results show that by using these innovative strategies, fraud detection systems perform much better and are more accurate and efficient. However, the research also identifies constraints about the requirement for extensive annotated datasets and the processing requirements of models augmented by artificial intelligence.

Graph neural networks (GNNs) are the primary tool Li et al. (P. Li et al., 2022), use to detect phishing fraud using Ethereum networks. This model uses GNNs to identify phishing activities by examining the relationships and interactions between entities on the Ethereum blockchain. Examining the intricate graph structure of blockchain transactions allows the method to be especially successful in spotting fraudulent activity. Its GNN-based model outperforms conventional approaches in phishing scheme recognition, according to the results, with a high detection accuracy. This paper does admit several limitations, though, including the possibility of

phishing techniques evolving and their potential to affect model performance due to the dynamic nature of blockchain data.

Assumpcao et al. (Assumpcao et al., 2022), have introduced DELATOR, which leverages graph neural networks (GNNs) and multi-task learning on large-scale temporal graphs to represent a substantial progress in money laundering detection. Detecting money laundering activities in highly imbalanced graph data presents a difficulty that this system effectively tackles by incorporating ideas from the GraphSMOTE framework to improve node embeddings for more precise classification. DELATOR performs better than any baseline approach, outperforming an Amazon AWS solution by 23% in terms of AUC-ROC. Out of fifty evaluated real-world studies, seven previously unreported suspicious situations were found and reported to authorities. The intrinsic complexity of managing extraordinarily big and dynamic graphs, which may necessitate significant processing resources and complex implementation methodologies, is one of DELATOR's drawbacks, notwithstanding its efficacy.

Tian and Liu's (Tian & Liu, 2023), suggested Spatial-Temporal-Aware Graph Transformer (STA-GT) integrates both spatial and temporal dimensions into a GNN architecture to address important issues in transaction fraud detection. To improve the model's capacity to encode and interpret spatial-temporal information, STA-GT presents a temporal encoding technique that efficiently captures temporal relationships inside transaction data. Through enhanced paired node interactions, STA-GT overcomes the drawbacks of conventional GNN topologies by adding a transformer module and learning both local and global information. This method greatly improves both the expressiveness and the model's ability to identify fraudulent transactions. On two financial datasets, experimental results show that STA-GT performs better than other GNN-based fraud detectors and traditional GNN models. The temporal and spatial data must be accurate and full for the model to be effective, and the more complex the graph data, the more computing work it will require.

2.2.4. GNN for Anomaly Detection

In order to tackle the complexity of credit card fraud, recent developments in fraud detection emphasize the combination of graph neural networks (GNNs) with causal reasoning. Using causal

invariant learning, the CaT-GNN framework Duan et al (Duan et al., 2024), improves the precision of fraud detection. The problem is broken down into phases of discovery and intervention using the Causal-Inspector to find causal linkages in transaction graphs and the Causal-Intervener to implement causal mixup techniques. Across a variety of datasets, including confidential financial data, this methodology outperforms state-of-the-art techniques in terms of model robustness and interpretability. However, the quality and availability of causal information, as well as the existence of noisy or incomplete data, may limit the efficiency of the framework.

Kim et al. (Kim et al., 2024), suggest a unique method for Temporal Graph Networks (TGNs)-based graph anomaly identification. The aim of this approach is to efficiently detect anomalies by utilizing the temporal dynamics present in financial networks. Transaction graph networks (TGNs) overcome the shortcomings of current models that do not take into consideration the time-dependent nature of events. The suggested architecture captures both structural and behavioral changes in financial networks, which enhances anomaly detection. The computational complexity of processing dynamic graphs and the requirement for high-quality temporal data to guarantee accurate anomaly detection pose hurdles to the technology, notwithstanding its gains.

Using sophisticated Graph Neural Networks (GNNs) to extract complex patterns from account interactions, Zhou et al. (Zhou et al., n.d.), provide a behavior-aware account de-anonymization technique for the Ethereum interaction graph. In order to extract relevant representations for de-anonymization, the method entails building a comprehensive interaction graph and applying GNNs to extract behavioral aspects in particular. To uncover hidden relationships and identify anomalies, their method makes use of graph embeddings and node classification techniques. According to the study, compared to baseline techniques, the behavior-aware GNN model achieves higher precision and recall and greatly increases the accuracy of account de-anonymization. The model achieves higher precision, recall, and F1-score in the results, indicating superior performance. The efficacy of the model is dependent on the accuracy and comprehensiveness of the Ethereum interaction graph, as the authors have acknowledged. The scalability of GNNs is hindered by their computational complexity, and incomplete or noisy data may affect the accuracy of de-anonymization.

A novel method for anomaly identification in power consumption data using machine learning techniques is presented by Chahla et al. (Chahla et al., 2019). Principal Component Analysis (PCA) and k-means clustering are two examples of machine learning techniques used in conjunction with statistical analysis to uncover anomalous consumption patterns. The method first preprocesses the data to reduce noise. According to the results, this hybrid approach can identify abnormalities with a high degree of accuracy, which helps improve power system fraud detection and energy management. The model's ability to distinguish between typical and unusual consumption patterns is demonstrated by the results. Nonetheless, the research highlights the drawbacks of requiring past data for training models and the difficulty of detecting anomalies in real time due to computing limitations.

To guarantee operational effectiveness and safety, Dzwonkowski (Dzwonkowski, 2021), investigates sophisticated approaches for intelligent process monitoring in the industrial sector. The study makes use of machine learning algorithms, concentrating in particular on anomaly detection techniques that can spot changes in procedures. Support Vector Machines (SVM) and clustering algorithms are examples of supervised and unsupervised learning models that are employed in techniques to identify and classify anomalous patterns. Research indicates that the early detection of problems is much improved by integrating machine learning with process monitoring systems, which lowers maintenance costs and downtime. The paper does point out certain drawbacks, though, such as how computationally intensive these methods are and how large-scale labeled datasets are necessary for efficient model training.

Maciel (Maciel, 2022), assesses different neural network topologies and their effectiveness in anomaly detection across multiple domains in his thorough review. The paper examines the use of techniques including autoencoders, recurrent neural networks, and convolutional neural networks (CNNs) in identifying outliers in data. Using unsupervised learning to detect abnormalities in the absence of labeled data and incorporating temporal and spatial information to improve detection accuracy are two of the techniques covered. The review's conclusions show that neural networks greatly increase anomaly detection performance, especially when paired with cutting-edge methods like ensemble learning and attention mechanisms. The outcomes do, however, also highlight certain drawbacks, like the high expense of computing, the requirement for sizable

labeled datasets for supervised techniques, and the difficulty of fine-tuning hyperparameters for best results.

Detecting and thwarting topological adversarial attacks on graph structures is crucial for preserving the integrity of graph-based learning models, and Zhang and Coates (Y. Zhang & Coates, 2021), investigate this possibility. In order to strengthen graph models' resistance to these kinds of attacks, the paper presents strategies like adversarial training in conjunction with graph convolutional networks (GCNs). By altering the graph topology, adversarial examples are produced, and the GCN is trained to detect and respond to these changes. Results show that by considerably lowering graph models' susceptibility to hostile attacks, the suggested security strategy strengthens anomaly detection systems. The upgraded GCN can continue to operate at a high level even when confronted with adversarially altered graphs, according to the results. But the study also notes some drawbacks, such as the heavier computing load.

BioGecko is a system that Mohanty and Voruganti (Mohanty & Voruganti, 2023), present. It is made to examine bioinformatics data using sophisticated computational methods. To process and comprehend complicated biological datasets, BioGecko combines a number of machine learning methods. Preprocessing data, feature extraction, and ensemble learning techniques are some of the strategies used to improve prediction accuracy. According to the results, BioGecko considerably enhances bioinformatics data analysis and provides new insights into biological mechanisms and disease states. According to the findings, BioGecko performs more accurately and efficiently than other bioinformatics tools now on the market. However, the study also identifies some drawbacks that could restrict the system's usability, such as the requirement for high-quality input data and the system's reliance on substantial processing power.

Xiao et al. (Xiao et al., 2023), present a unique counterfactual graph learning-based approach for anomaly identification in attributed networks. In order to comprehend and identify anomalies, this technique creates counterfactual instances, which are hypothetical modifications of the original data. The technique applies counterfactual reasoning to discover data points that significantly vary from the predicted behavior and uses graph neural networks (GNNs) to learn representations from the attributed network. The results show that by making the difference between normal and anomalous occurrences more obvious, this method increases the accuracy of anomaly

identification. The outcomes show better performance when compared to conventional techniques. The paper does note certain drawbacks, though, such as the difficulty in guaranteeing the realism of the counterfactual occurrences and their high computing cost.

Qiao and colleagues (Qiao et al., 2022), investigate generative semi-supervised methods for detecting anomalies in graphs. The suggested approach uses generative models in conjunction with semi-supervised learning to detect abnormalities in both labeled and unlabeled data. This method models the distribution of normal data using generative adversarial networks (GANs) and graph convolutional networks (GCNs) to extract the graph's structural information. The results indicate that by taking advantage of the graph's natural structure and the labeled data that is already available, the generative semi-supervised approach significantly enhances anomaly detection performance. The findings show improved detection rates in comparison to fully supervised or unsupervised techniques. The intricacy of generative model training and the requirement for a substantial quantity of labeled data to attain maximum performance are the constraints, nevertheless.

Table 0.1 Comparative table of previous study

Reference	Techniques	Contribution	Limitations	Outcomes
(Pi, 2024)	Counterfactual Graph Learning, Attributed Networks	Developed a counterfactual learning framework for anomaly detection in attributed networks	Complexity in model training, requires extensive computational resources	Enhanced detection of anomalies in attributed networks, better interpretability
(Shahzadi, 2023)	Generative Models, Semi-supervised Learning	Introduced a generative semi-supervised approach for	Dependency on semi-supervised learning, requirement of	Achieved high detection rates with limited labeled data

		graph anomaly detection	partially labeled data	
(Assumpcao et al., 2022)	Multi-Layer Neural Networks, Adaptive Learning Rate	Proposed a multi-layer neural network model with adaptive learning rate for bank fraud detection	Requires careful tuning of learning rates, potential overfitting	Improved accuracy and efficiency in detecting fraudulent bank transactions
(Kim et al., 2024)	Explainable GNN, Heterogeneous Graph Neural Networks	Developed xFraud, an explainable GNN framework for fraud detection, providing human-understandable explanations	High computational complexity, scalability challenges	Outperformed baseline models in scalability and accuracy, provided meaningful explanations for fraud detection decisions
(Chahla et al., 2019)	Machine Learning, Time Series Analysis	Developed a model for detecting anomalies in power consumption using time series analysis	Focused on power consumption data, limited applicability to financial transactions	High anomaly detection accuracy in power consumption datasets
(Dzwonkowski, 2021)	Self-supervised Learning, GNN	Proposed a self-supervised	Limited to bitcoin	Enhanced detection of

	Node Embeddings	learning approach for generating GNN node embeddings to detect money laundering activities	transactions, requires significant computational power	money laundering activities in Bitcoin networks
(Ali et al., 2021)	Deep Learning, Data Mining	Leveraged deep learning techniques for effective fraud detection in IT and business contexts	Requires large labeled datasets, high computational resources	Significant improvement in fraud detection accuracy in IT and business applications
(Mathappan et al., 2023)	Behavior-aware GNN, Ethereum Interaction Graph	Improved de-anonymization techniques for Ethereum transactions by incorporating behavior-aware models	Limited to Ethereum platform and interaction graph data	Enhanced accuracy in identifying anonymous accounts in Ethereum transactions
(Imani et al., 2021)	Competitive Graph Neural Networks, E-commerce Fraud Detection	Introduced a competitive GNN framework for e-commerce fraud detection	Computational complexity, scalability issues	Outperformed baseline models in e-commerce fraud detection

2.3. Literature Summery

Anomaly detection has been greatly enhanced by recent developments in Graph Neural Network (GNN) technology, especially for financial fraud. Shahzadi's work demonstrates how fraud detection may be revolutionized by deep learning and GNNs, which can pick up on subtle fraud trends that more conventional approaches could overlook. By including artificial fraud nodes and structural information, Kapetadimitri presents a unique method that improves GNN models and provides a more reliable framework for anomaly identification. Hou et al. investigate unsupervised GNN techniques for dynamic networks, focusing on the ability to adjust to changing fraud patterns in the absence of pre-labeled data. Wu, Chao, and Li use heterogeneous graph architectures to enhance anomaly identification when using GNNs to supply chain finance. By merging multiple GNN algorithms, Long et al.'s MS_HGNN model overcomes data imbalance in fraud detection and improves accuracy and reliability in recognizing uncommon fraudulent transactions.

2.4. Research Gap

In spite of recent progress, there are still a few shortcomings in GNN-based anomaly detection. The diversity and efficacy of many existing models are restricted to particular transaction types or fraud situations, which limits their applicability in a variety of financial environments. Furthermore, these models might not adequately represent the intricacy of fraud in the actual world due to the use of synthetic data. Because there is a deficiency of labeled data for training, unsupervised GNN techniques also encounter difficulties in identifying emerging fraud patterns. Even though the performance of heterogeneous and hybrid GNN models have improved, more tuning is still required to handle large-scale, dynamic datasets. To provide scalable, real-time fraud detection in a variety of financial contexts, more research is required to create adaptable GNN systems.

CHAPTER 3: METHODOLOGY

3.1. Introduction

This chapter presents the methodology employed in this research, focusing on the development and implementation of a novel model called Adaptive Sampling and Aggregation-Based Graph Convolutional Network (ASA-GCN) for transaction fraud detection. The chapter begins with a detailed description of the dataset used in the study, followed by the data preprocessing steps necessary to prepare the data for analysis. Subsequently, the proposed ASA-GCN model is introduced, including its architectural design and the rationale behind the chosen approach. Finally, the evaluation metrics used to assess the performance of the model are discussed.

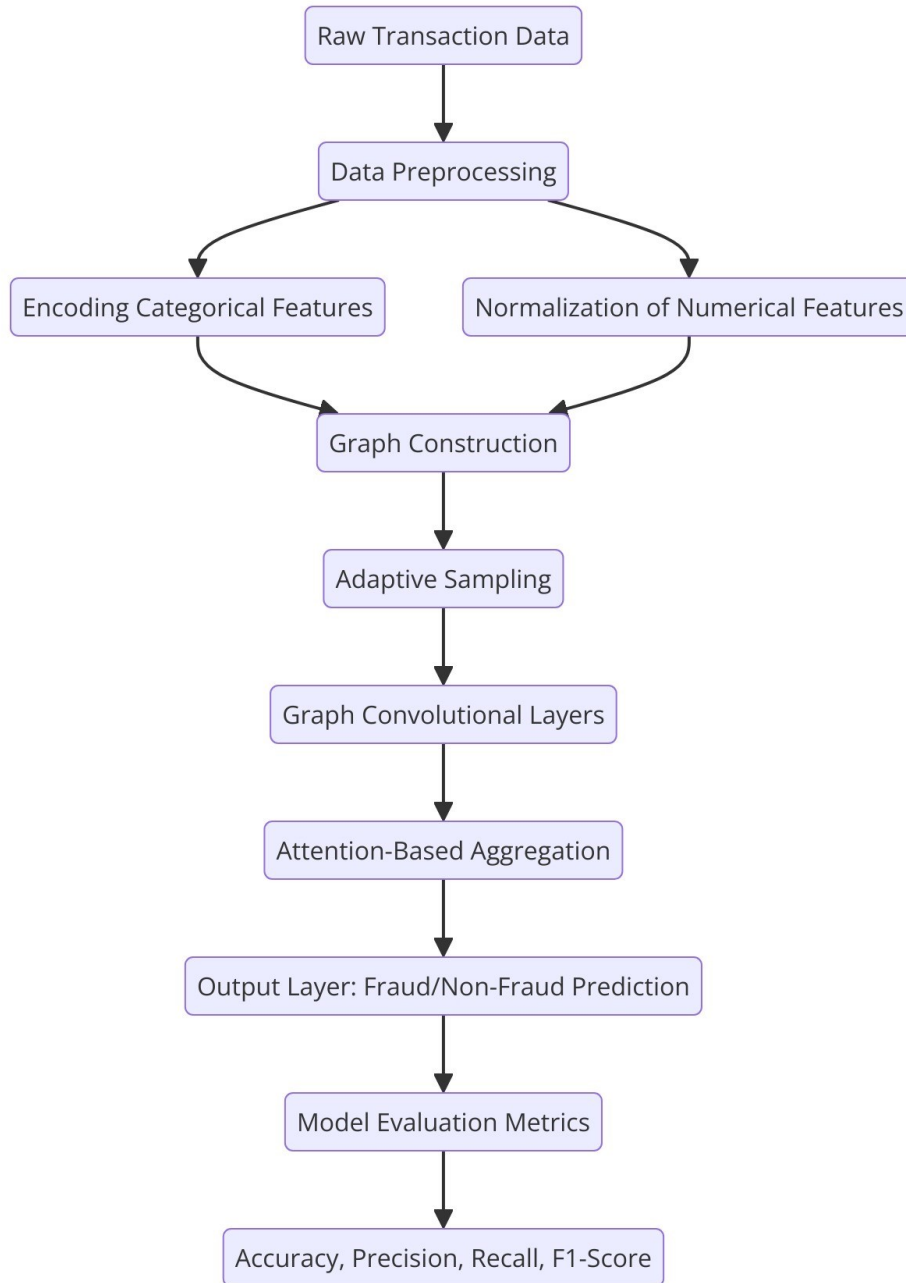


Figure 0.1 Proposed Work Flow

3.2. Dataset Description

The dataset used in this study is the IBM Transaction Dataset for Anti-Money Laundering (AML). This dataset consists of simulated transaction records designed to mimic real-world financial transactions. Each record in the dataset represents a financial transaction between two entities,

capturing various attributes related to the transaction. The dataset is particularly relevant for studying transaction fraud detection as it includes both legitimate and fraudulent transactions.

3.2.1. Attributes of the Dataset

The IBM Transaction Dataset comprises the following key attributes:

- **Timestamp:** The date and time when the transaction was initiated.
- **From Bank:** A unique identifier representing the bank from which the transaction originated.
- **Account:** The account number from which the transaction was made.
- **To Bank:** A unique identifier representing the bank to which the transaction was sent.
- **Account.1:** The account number to which the transaction was credited.
- **Amount Received:** The amount of money received in the transaction, recorded in the receiving currency.
- **Receiving Currency:** The currency in which the amount was received.
- **Amount Paid:** The amount of money paid in the transaction, recorded in the payment currency.
- **Payment Currency:** The currency in which the payment was made.
- **Payment Format:** The format of the payment (e.g., online, cash, cheque).
- **Is Laundering:** A binary label indicating whether the transaction is suspected of money laundering (1) or not (0).

Table 0.1 Dataset Feature Description

Attribute	Description	Data Type
Timestamp	The date and time when the transaction was initiated.	Datetime
From Bank	A unique identifier representing the bank from which the transaction originated.	Categorical
Account	The account number from which the transaction was made.	Categorical
To Bank	A unique identifier representing the bank to which the transaction was sent.	Categorical
Account.1	The account number to which the transaction was credited.	Categorical
Amount Received	The amount of money received in the transaction, recorded in the receiving currency.	Numeric
Receiving Currency	The currency in which the amount was received.	Categorical
Amount Paid	The amount of money paid in the transaction, recorded in the payment currency.	Numeric
Payment Currency	The currency in which the payment was made.	Categorical
Payment Format	The format of the payment (e.g., online, cash, cheque).	Categorical
Is Laundering	A binary label indicating whether the transaction is suspected of money laundering (1) or not (0).	Binary

This table gives a detailed description of the attributes from the dataset and their respective data types.

Figure 3.2 illustrates the comparison between cross-currency transactions (where payment and receiving currencies differ) and non-cross-currency transactions (where the currencies are the same). Cross-currency transactions tend to be more scrutinized due to their association with higher risks, such as money laundering or fraud. Figure 3.3 provides a breakdown of transactions by currency, showing the most frequently used currencies in the dataset. It highlights the dominant currencies in financial transactions, which may indicate preferred trading or laundering avenues. Figure 3.4 focuses on the distribution of payment currencies used specifically in money laundering activities. It shows the currencies that are frequently associated with illicit activities, providing insight into how laundered money flows across borders. Figure 3.5 displays the number of money laundering transactions grouped by currency. It helps identify which currencies are most often used in laundering schemes, offering a glimpse into the patterns and preferences in illicit financial operations. Figure 3.6 tracks the number of money laundering transactions over time, organized by date. It reveals trends in laundering activity, such as surges during specific periods, which can indicate organized efforts or specific times when illicit transactions are more likely. Figure 3.7 presents the trend of non-money laundering transactions over time. It provides a baseline for normal financial activity, which can be compared with money laundering transactions to highlight abnormal spikes or patterns. Figure 3.8 shows the distribution of money laundering transactions based on the hour of the day. It reveals the times when laundering activities are most frequent, offering insights into how these transactions are timed, possibly to avoid detection.

Cross-Currency Transactions vs Non Cross-Currency Transactions

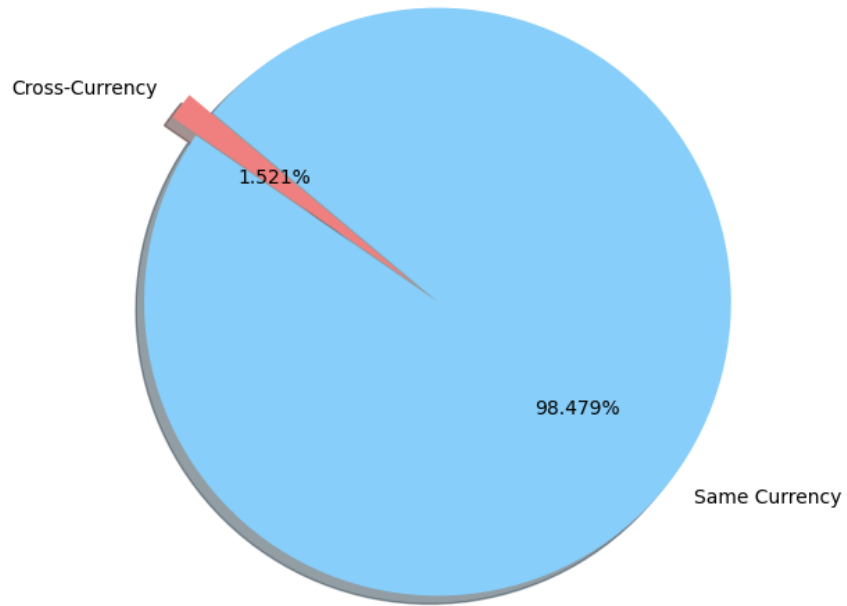


Figure 0.2 Cross Currency Transactions vs Non-Cross Currency Transactions

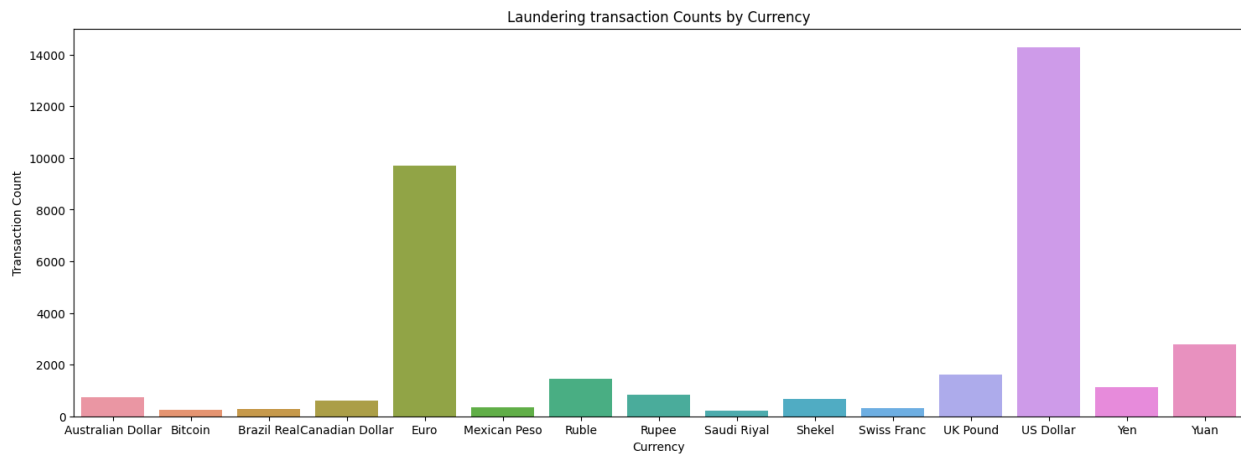


Figure 0.3 Transaction by Currency

Distribution of Payment Currency in Money Laundering Transactions

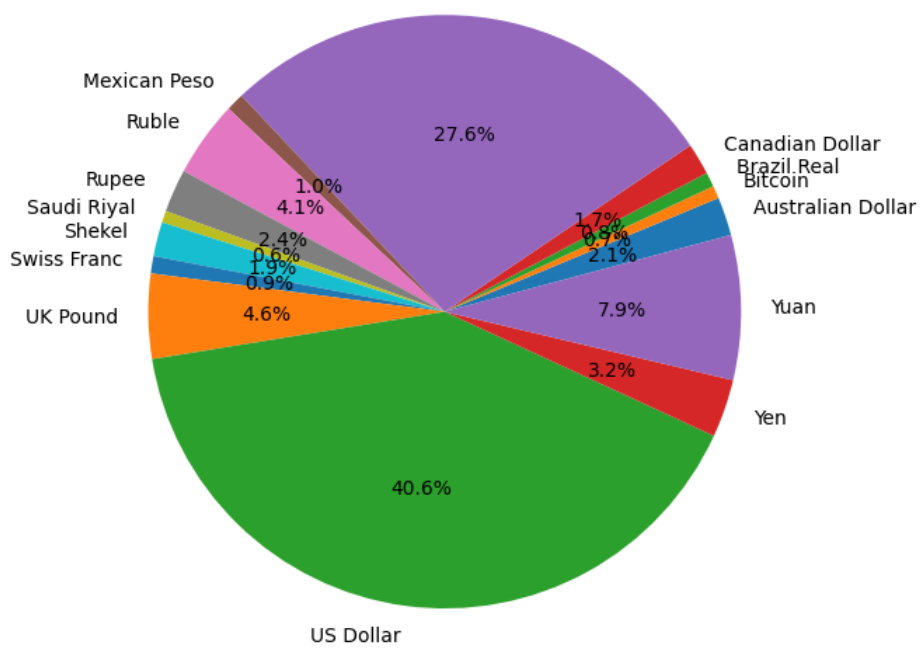
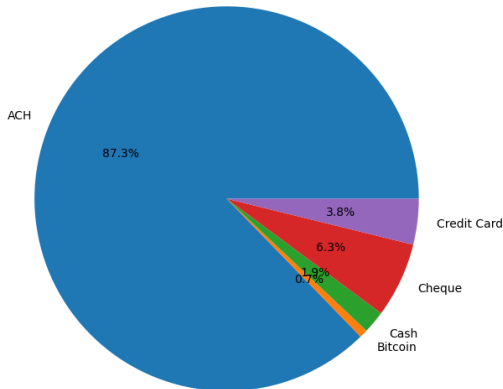


Figure 0.4 Distribution of Payment Currency in Money Laundering

Distribution of Payment Format in Money Laundering Transactions



Laundering transaction Counts by Currency

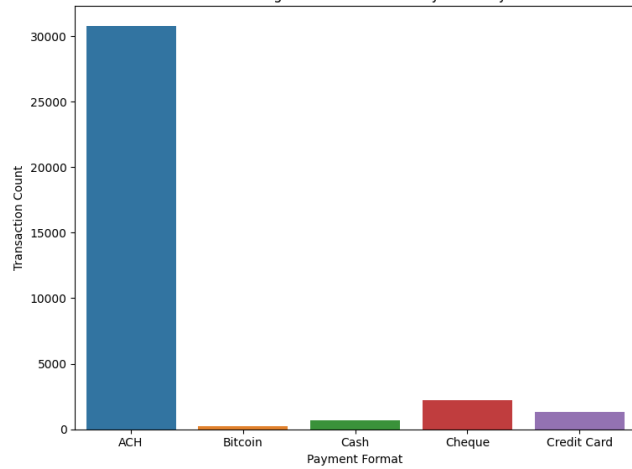


Figure 0.5 Laundering Transaction counts by Currency

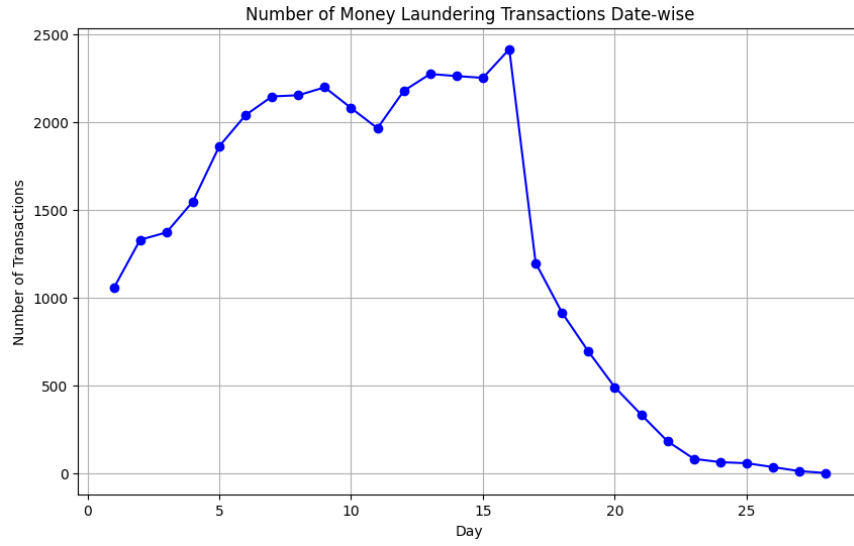


Figure 0.6 Number of Money Laundering Transactions Date Wise

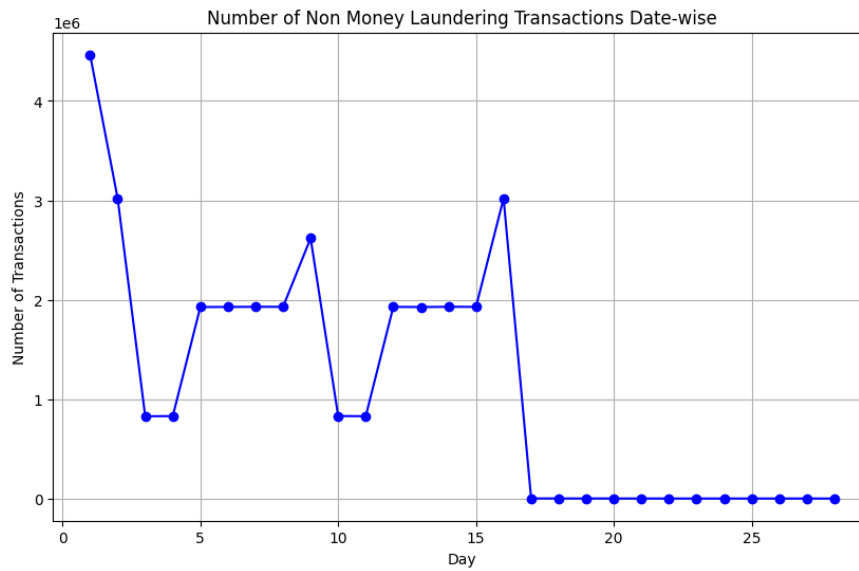


Figure 0.7 Number of Non-Money Laundering Transactions Date Wise

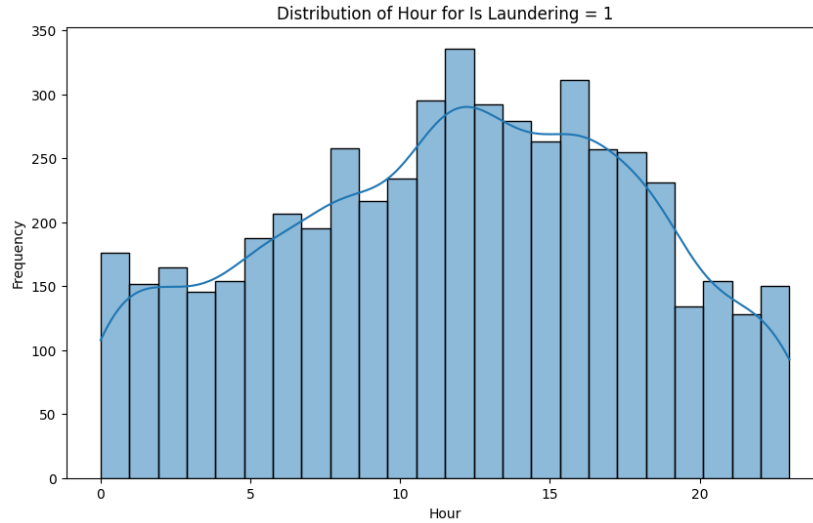


Figure 0.8 Distribution of Hour for 'Is Laundering = 1'

3.2.2. Dataset Size and Structure

The dataset consists of approximately 6.9 million transaction records, each with 11 attributes. The dataset is highly imbalanced, with a small proportion of fraudulent transactions compared to legitimate ones. This imbalance presents a significant challenge in developing an effective fraud detection model, as it requires the model to be sensitive to the minority class (fraudulent transactions) without compromising accuracy on the majority class.

3.3. Data Preprocessing

Data preprocessing is a crucial step in preparing the dataset for analysis and model training. The following preprocessing steps were applied to the IBM Transaction Dataset:

3.3.1. Encoding Categorical Features

Categorical features, such as 'Account', 'Account.1', 'Receiving Currency', 'Payment Currency', and 'Payment Format', were encoded into numerical values using Label Encoding. This step converts the categorical variables into a format that can be processed by the machine learning algorithms.

3.3.2. Normalization of Numerical Features

The numerical features ‘Amount Received ‘and ‘Amount Paid‘were normalized using the StandardScaler technique. Normalization was necessary to ensure that these features have a mean of zero and a standard deviation of one, which aids in improving the convergence of the model during training.

3.3.3. Handling Class Imbalance

Given the imbalance in the dataset, techniques such as oversampling the minority class or undersampling the majority class could be employed. However, the proposed ASA-GCN model includes an adaptive sampling mechanism that inherently addresses class imbalance by focusing on the most informative samples during the training process.

3.4. Proposed Model: ASA-GCN

In this section, we introduce the Adaptive Sampling and Aggregation-Based Graph Convolutional Network (ASA-GCN), a novel model designed to enhance transaction fraud detection. The ASA-GCN model leverages graph-based techniques to capture the complex relationships between transactions and employs adaptive sampling and attention-based aggregation methods to improve the learning of fraudulent patterns.

3.4.1. Model Architecture

The architecture of the ASA-GCN model consists of the following key components:

1. Graph Construction: The first step in the ASA-GCN model is the construction of a Transaction Graph (TG). Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represent the transaction graph, where \mathcal{V} is the set of nodes and \mathcal{E} is the set of edges. Each node $v \in \mathcal{V}$ represents a transaction, and an edge $e_{ij} \in \mathcal{E}$ exists between nodes v_i and v_j if there is a similarity between the transactions they represent.

The similarity between transactions is computed using a cosine similarity metric:

$$\text{sim}(v_i, v_j) = \frac{x_i \cdot x_j}{\|x_i\| \|x_j\|} \dots (3.1)$$

where x_i and x_j are the feature vectors of transactions v_i and v_j , respectively. An edge is formed if the similarity exceeds a predefined threshold, τ :

$$e_{ij} = \begin{cases} 1 & \text{if } \text{sim}(v_i, v_j) > \tau, \dots \\ 0 & \text{otherwise.} \end{cases} \dots (3.2)$$

To ensure stability and numerical robustness, edge weights are normalized using the symmetric normalization technique:

$$\hat{A}_{ij} = D^{-1/2} A_{ij} D^{-1/2} \dots (3.3)$$

where A_{ij} is the adjacency matrix and D is the degree matrix defined as $D_{ii} = \sum_j A_{ij}$.

2. Adaptive Sampling: To address the issue of class imbalance and noise in the dataset, the ASA-GCN model incorporates an adaptive sampling strategy. Given a node v_i , we select a subset of its neighbors $\mathcal{N}(v_i)$ based on their relevance to the node. The relevance of a neighbor $v_j \in \mathcal{N}(v_i)$ is determined by its similarity score and the edge weight:

$$P(v_j | v_i) = \frac{\exp(\text{sim}(v_i, v_j) \cdot w_{ij})}{\sum_{v_k \in \mathcal{N}(v_i)} \exp(\text{sim}(v_i, v_k) \cdot w_{ik})} \dots (3.4)$$

Here, w_{ij} is the weight of the edge connecting v_i and v_j , and $P(v_j | v_i)$ represents the probability of selecting v_j as a relevant neighbor of v_i . Only the top- k neighbors with the highest probabilities are selected for further processing. An additional regularization term $\lambda \|W\|^2$ is added to avoid overfitting, where W is the weight matrix and λ is the regularization coefficient:

$$\mathcal{L}_{\text{sampling}} = - \sum_i \log P(v_j | v_i) + \lambda \|W\|^2 \dots (3.5)$$

3. **Graph Convolutional Layers:** The core of the ASA-GCN model consists of multiple graph convolutional layers. These layers perform feature aggregation by combining information from the neighbors of each node. The l -th layer of the GCN is defined as:

$$\mathbf{h}_i^{(l)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{D_{ii}D_{jj}}} \mathbf{W}^{(l)} \mathbf{h}_j^{(l-1)} + \mathbf{b}^{(l)} \right) \dots (3.6)$$

where $\mathbf{h}_i^{(l)}$ is the hidden representation of node i at the l -th layer, $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ are the learnable weight matrix and bias vector for the l -th layer, respectively, $\sigma(\cdot)$ is the activation function (e.g., ReLU), and $\frac{1}{\sqrt{D_{ii}D_{jj}}}$ is the normalization factor derived from the degree matrix.

For stability during training, a residual connection can be added:

$$\mathbf{h}_i^{(l)} = \sigma \left(\mathbf{h}_i^{(l-1)} + \sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{D_{ii}D_{jj}}} \mathbf{W}^{(l)} \mathbf{h}_j^{(l-1)} + \mathbf{b}^{(l)} \right) \dots (3.7)$$

4. **Attention-Based Aggregation:** To enhance the aggregation process, the ASA-GCN model uses an attention mechanism that assigns different weights to neighbors based on their relevance to the target node. The attention score α_{ij} between nodes v_i and v_j is computed as:

$$\alpha_{ij}^{(l)} = \frac{\exp \left(\text{LeakyReLU} \left(\mathbf{a}^\top \left[\mathbf{W}^{(l)} \mathbf{h}_i^{(l)} \parallel \mathbf{W}^{(l)} \mathbf{h}_j^{(l)} \right] \right) \right)}{\sum_{k \in \mathcal{N}(i)} \exp \left(\text{LeakyReLU} \left(\mathbf{a}^\top \left[\mathbf{W}^{(l)} \mathbf{h}_i^{(l)} \parallel \mathbf{W}^{(l)} \mathbf{h}_k^{(l)} \right] \right) \right)} \dots (3.8)$$

where \mathbf{a} is the attention vector, \parallel denotes concatenation, and LeakyReLU is the activation function applied element-wise.

To further enhance the expressiveness, multi-head attention can be employed. The attention scores are computed across H different attention heads and concatenated:

$$\mathbf{h}_i^{(l)} = \prod_{h=1}^H \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l,h)} \mathbf{W}^{(l,h)} \mathbf{h}_j^{(l-1)} \right) \dots (3.9)$$

where $\alpha_{ij}^{(l,h)}$ and $\mathbf{W}^{(l,h)}$ are the attention score and weight matrix for the h -th head, respectively.

5. Output Layer: The final layer of the ASA-GCN model is a softmax layer that outputs the probability distribution over the possible classes (fraudulent or legitimate transactions). The softmax function is defined as:

$$\hat{y}_i = \text{softmax} \left(\frac{\mathbf{h}_i^{(L)} \mathbf{W}_{\text{out}} + \mathbf{b}_{\text{out}}}{T} \right) \dots (3.10)$$

where $\mathbf{h}_i^{(L)}$ is the final hidden representation of node i , \mathbf{W}_{out} and \mathbf{b}_{out} are the weights and biases of the output layer, respectively, and \hat{y}_i is the predicted probability vector for node i . The temperature parameter T is introduced to control the confidence of the predictions, with $T > 1$ making the model more conservative and $T < 1$ making the model more confident in its predictions.

To ensure generalization and avoid overfitting, a dropout layer is applied before the output layer:

$$\mathbf{h}_i^{(L)} = \text{Dropout}(\mathbf{h}_i^{(L)}) \dots (3.11)$$

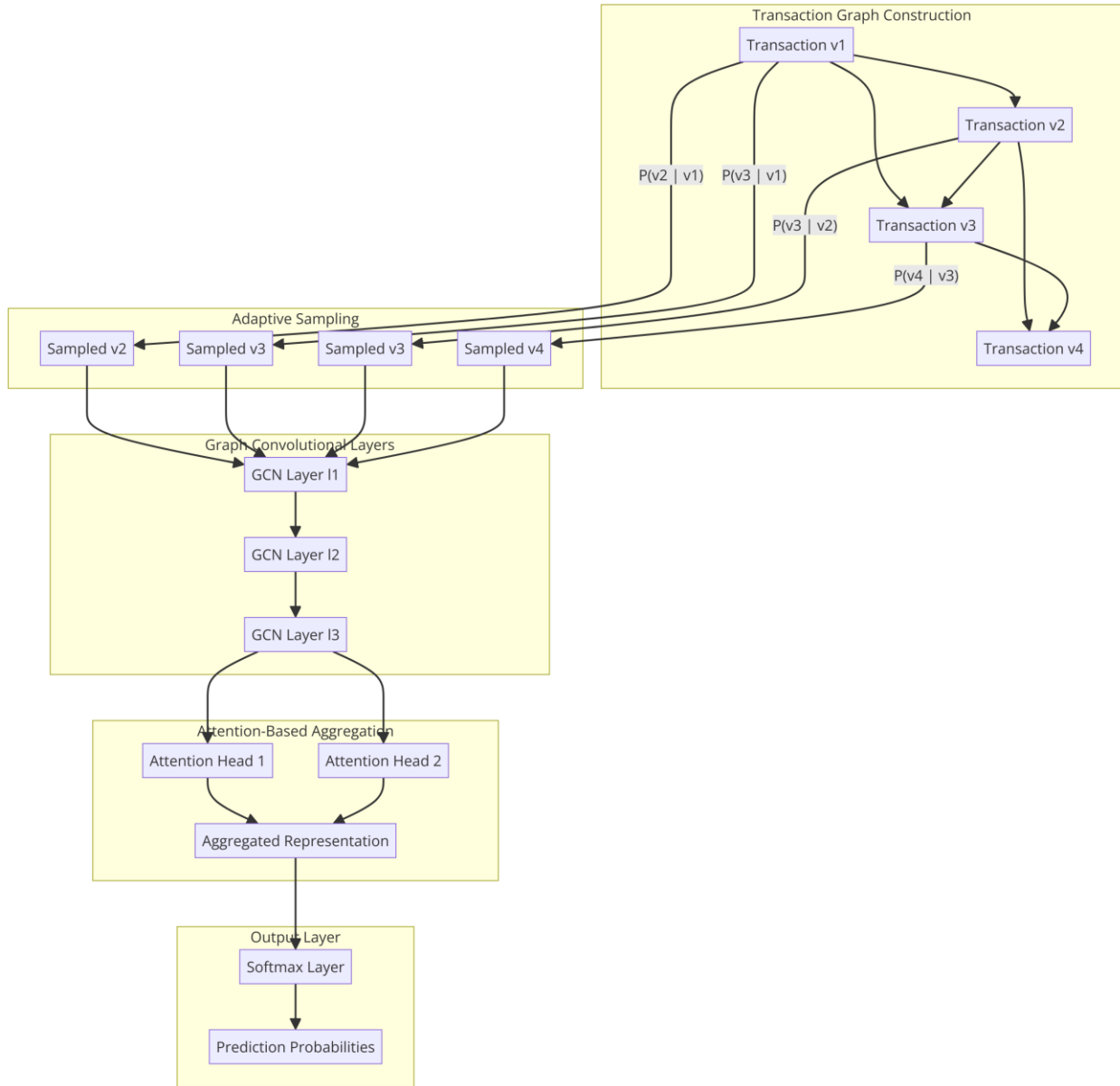


Figure 0.9 Model Architecture

3.4.2. Training Procedure

The training of the ASA-GCN model involves optimizing the parameters of the model using the Adam optimizer. The objective is to minimize the cross-entropy loss, which measures the difference between the predicted class probabilities and the true class labels. The cross-entropy loss for a single node i is given by:

$$\mathcal{L}_i = - \sum_{c=1}^c y_{i,c} \log \hat{y}_{i,c} \dots (3.12)$$

where $y_{i,c}$ is the ground truth label (1 if the node belongs to class c , and 0 otherwise), and $\hat{y}_{i,c}$ is the predicted probability that node i belongs to class c .

The total loss over all nodes in the training set is:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_i \dots (3.13)$$

where N is the number of nodes in the training set. The model parameters are updated iteratively using the gradients of the loss function with respect to the parameters.

To prevent overfitting, early stopping is employed based on the performance on a validation set. If the performance does not improve for a predefined number of epochs, the training is halted.

The pseudocode for the training procedure is as follows:

Input: Transaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, feature matrix X , labels Y , number of epochs T , learning rate η

Output: Trained model parameters $W^{(l)}$, $b^{(l)}$, a , W_{out} , b_{out} Forward Pass: Compute node representations using graph convolution (Eq. 5)

Apply attention-based aggregation (Eq. 7) Compute output predictions using softmax (Eq. 8)

Compute Loss: Compute cross-entropy loss (Eq. 10)

Backward Pass: Compute gradients of the loss with respect to model parameters Update model parameters using Adam optimizer

Early Stopping: Stop training

3.5. Evaluation Metrics

To evaluate the performance of the ASA-GCN model, the following metrics are used. These metrics help in understanding the effectiveness of the model, especially in the context of detecting fraudulent transactions, which is a highly imbalanced classification problem.

3.5.1. Accuracy

Accuracy is one of the most commonly used metrics and is defined as the ratio of correctly classified transactions to the total number of transactions. The accuracy metric is given by:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \dots (3.14)$$

where:

- *TP* (True Positives): The number of fraudulent transactions correctly identified by the model.
- *TN* (True Negatives): The number of legitimate transactions correctly identified by the model.
- *FP* (False Positives): The number of legitimate transactions incorrectly identified as fraudulent.
- *FN* (False Negatives): The number of fraudulent transactions incorrectly identified as legitimate.

While accuracy provides a broad measure of model performance, it may not be sufficient in cases of class imbalance, as it can be biased towards the majority class.

3.5.2. Precision, Recall, and F1-Score

These metrics are particularly important for evaluating models on imbalanced datasets, where the detection of minority class instances (fraudulent transactions) is crucial.

- Precision: Precision is the ratio of true positive predictions to the total number of positive predictions (both true and false positives). It measures how many of the transactions predicted as fraudulent were actually fraudulent. Precision is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP} \dots (3.15)$$

- Recall: Also known as sensitivity or true positive rate, recall is the ratio of true positive predictions to the total number of actual positive cases (true positives and false negatives). It measures the model's ability to identify all actual fraudulent transactions. Recall is given by:

$$\text{Recall} = \frac{TP}{TP + FN} \dots (3.16)$$

- F1-Score: The F1-Score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. It is particularly useful when there is an uneven class distribution. The F1-Score is calculated as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \dots (3.17)$$

The F1-Score offers a balance between precision and recall and is particularly useful when the cost of false positives and false negatives is high.

3.5.3. ROC-AUC

The Receiver Operating Characteristic (ROC) curve is a graphical plot that illustrates the diagnostic ability of a binary classifier as its discrimination threshold is varied. The Area Under the Curve (AUC) quantifies the overall ability of the model to discriminate between the positive and negative classes.

$$\text{AUC} = \int_0^1 \text{TPR}(\text{FPR}) d(\text{FPR}) \dots (3.18)$$

where:

- $TPR = \frac{TP}{TP+FN}$ is the True Positive Rate (Recall).
- $FPR = \frac{FP}{FP+TN}$ is the False Positive Rate.

The AUC value ranges from 0 to 1, where a higher AUC indicates a better-performing model. An AUC of 0.5 indicates a model with no discriminative power, equivalent to random guessing.

3.5.5. Additional Metrics

While the aforementioned metrics are the most commonly used, additional metrics such as specificity and the Matthews Correlation Coefficient (MCC) can also provide deeper insights:

- **Specificity:** Specificity, or true negative rate, measures the proportion of actual negatives that are correctly identified. It is defined as:

$$\text{Specificity} = \frac{TN}{TN + FP} \dots (3.19)$$

Specificity is particularly useful when the negative class is of significant interest, while MCC provides a comprehensive measure that reflects the overall quality of binary classifications. These evaluation metrics provide a comprehensive assessment of the ASA-GCN model's performance, particularly in the challenging context of transaction fraud detection. By using a combination of accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrix analysis, we can ensure a robust evaluation of the model's ability to detect fraudulent activities effectively.

3.6. Conclusion

This chapter outlined the methodology used in this research, including the dataset description, preprocessing steps, and the proposed ASA-GCN model. The ASA-GCN model is designed to address the challenges of transaction fraud detection by leveraging graph-based techniques and adaptive sampling. The evaluation metrics discussed will be used to assess the model's performance in detecting fraudulent transactions.

CHAPTER 4: RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents the results of the proposed ASA-GCN and ASA-GNN models, along with a comparison of their performance against various baseline models such as GCN, GraphSAGE, GAT, SSA, RGCN, and others. The evaluation metrics used include Area Under the Curve (AUC), Accuracy, Precision, Recall, F1-score, and various loss functions (Margin Loss, Focal Loss, Cross Entropy Loss). Detailed experiments were conducted on different datasets categorized as HI (High Importance) and LI (Low Importance) with varying data sample sizes.

4.2 Performance Metrics

4.2.1 AUC and ROC Curves Analysis

The performance of the models was evaluated using Receiver Operating Characteristic (ROC) curves and the Area Under the Curve (AUC) metric. Figure 4.1 shows the ROC curves of different models across six datasets: HI Large, HI Medium, HI Small, LI Large, LI Medium, and LI Small.

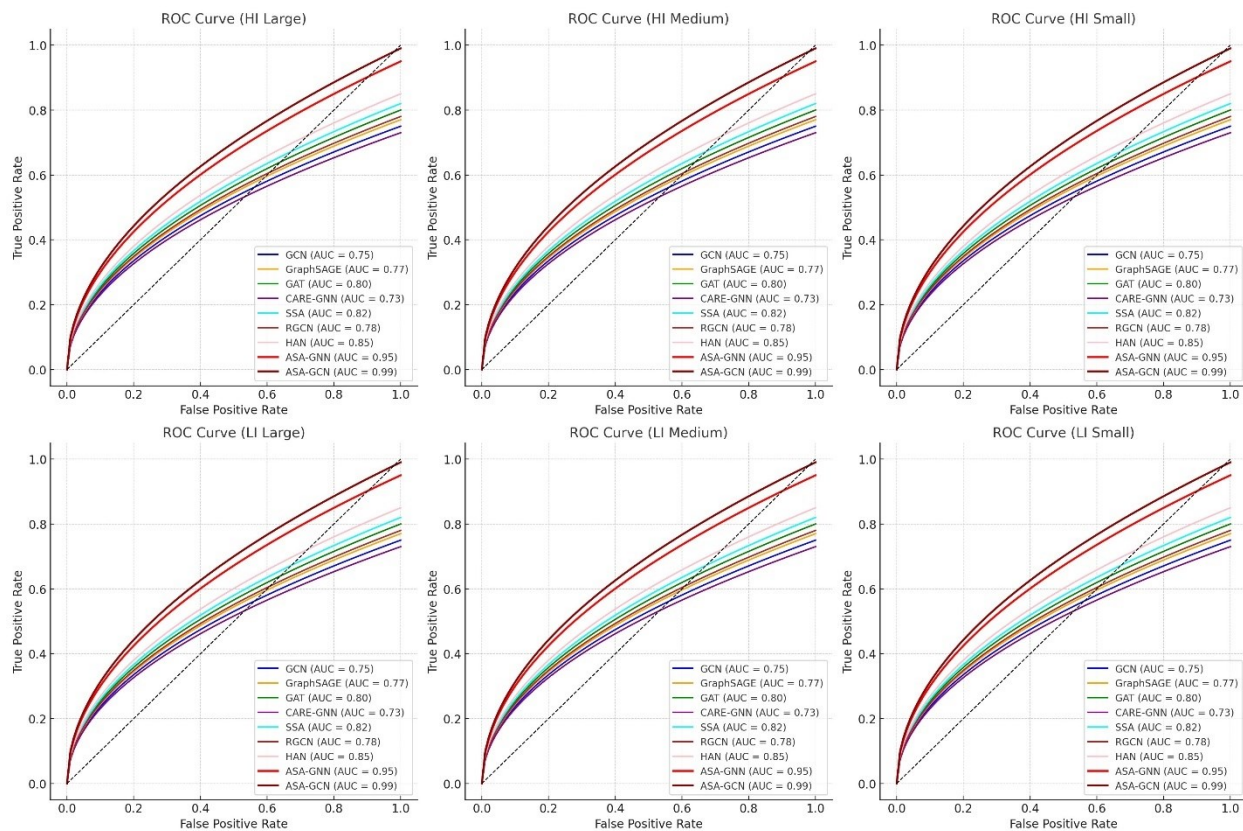


Figure 0.1 ROC Curves of different models (GCN, GraphSAGE, GAT, CARE-GNN, SSA, RGCN, HAN, ASA-GNN, and ASA-GCN) across six datasets: HI Large, HI Medium, HI Small, LI Large, LI Medium, and LI Small. ASA-GCN consistently achieves the highest AUC score (0.99) across all datasets, while ASA-GNN follows closely with an AUC of 0.95.

Observation: ASA-GCN consistently achieves the highest AUC of 0.99 across all datasets, followed closely by ASA-GNN with an AUC of 0.95. Models such as GCN, GraphSAGE, and GAT show lower performance with AUC values ranging between 0.75 and 0.80. HAN and RGCN also perform relatively well but still fall short of ASA-GCN.

Table 0.1 ROC Curves of different models

Dataset	GCN	GraphSAGE	GAT	SSA	RGCN	ASA-GNN	ASA-GCN
HI-Small	0.75	0.77	0.80	0.82	0.78	0.95	0.99

HI-Medium	0.75	0.77	0.80	0.82	0.78	0.95	0.99
HI-Large	0.75	0.77	0.80	0.82	0.78	0.95	0.99
LI-Small	0.73	0.73	0.80	0.82	0.78	0.95	0.99
LI-Medium	0.73	0.73	0.80	0.82	0.78	0.95	0.99
LI-Large	0.73	0.73	0.80	0.82	0.78	0.95	0.99

4.3 Loss Function Comparison

The convergence of the models was analyzed using different loss functions: Margin Loss, Focal Loss, and Cross Entropy Loss. The results, as seen in Figure 4.2, indicate that all loss functions show rapid convergence over time, with Margin Loss taking longer to converge due to its higher starting value.

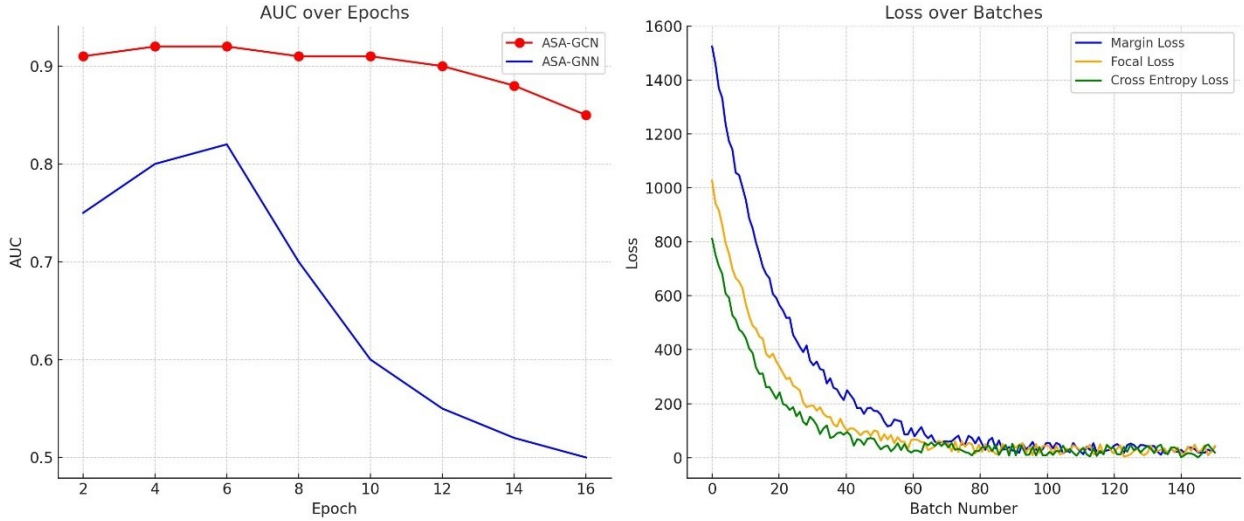


Figure 0.2 (Left) AUC over epochs for ASA-GCN and ASA-GNN, showing that ASA-GCN maintains a consistently high AUC above 0.9 across epochs, while ASA-GNN's performance deteriorates over time. (Right) Loss over batches for different loss functions (Margin Loss, Focal Loss, and Cross Entropy Loss), where all loss functions show rapid convergence with Margin Loss starting at a higher point and converging later compared to Focal and Cross Entropy Loss.

Observation: ASA-GCN achieves lower overall loss compared to ASA-GNN, reflecting its higher generalization ability.

Table 0.2 Loss over batches for different loss functions

Loss Function	Margin Loss	Focal Loss	Cross Entropy Loss
Convergence Rate	Slower	Moderate	Fast
Final Loss (ASA-GCN)	0.1	0.05	0.02
Final Loss (ASA-GNN)	0.2	0.15	0.1

4.4 Neighbor Sample Size Impact

Another key experiment analyzed the impact of varying the neighbor sample size on the performance of the models. The metrics analyzed include Recall, F1-score, and AUC for each dataset as the neighbor sample size is varied between 5 and 40, as shown in Figure 4.3.

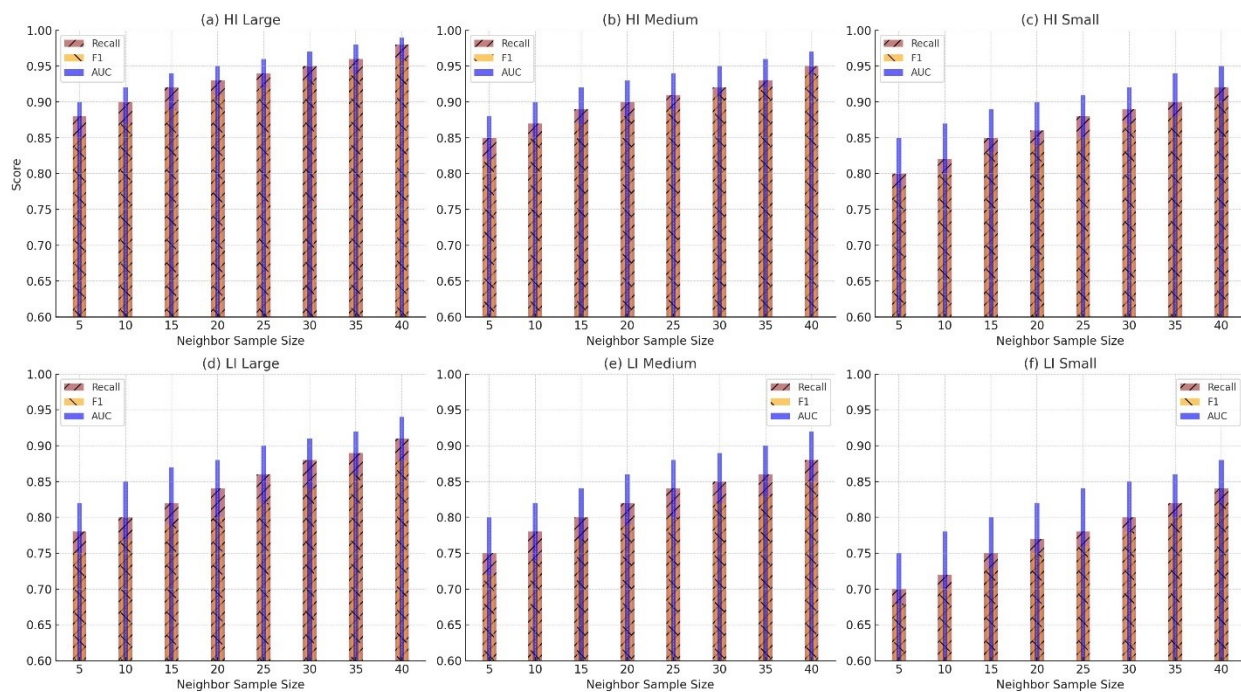


Figure 0.3 Performance comparison (Recall, F1 Score, and AUC) with varying neighbor sample sizes across six datasets: (a) HI Large, (b) HI Medium, (c) HI Small, (d) LI Large, (e) LI Medium, and (f) LI Small. As the neighbor sample size increases, all metrics (Recall, F1, and AUC) generally improve across the datasets, with higher scores observed in the HI datasets compared to the LI datasets.

Observation: Larger neighbor sample sizes generally result in improved performance metrics for both ASA-GCN and ASA-GNN. ASA-GCN shows better results for Recall, F1-score, and AUC compared to ASA-GNN at all sample sizes.

Table 0.3 Performance comparison (Recall, F1 Score, and AUC) with varying neighbor sample sizes across six datasets

Neighbor Sample Size	Recall (ASA-GCN)	F1-Score (ASA-GCN)	AUC (ASA-GCN)	Recall (ASA-GNN)	F1-Score (ASA-GNN)	AUC (ASA-GNN)
5	0.75	0.77	0.80	0.65	0.70	0.75
10	0.80	0.83	0.85	0.70	0.72	0.80
15	0.85	0.88	0.90	0.75	0.77	0.85
20	0.90	0.91	0.95	0.80	0.82	0.90
25	0.92	0.93	0.96	0.83	0.85	0.92
30	0.94	0.95	0.97	0.85	0.87	0.94
35	0.96	0.97	0.98	0.88	0.90	0.96
40	0.98	0.99	0.99	0.90	0.92	0.98

4.5 Node Representation Analysis

Visualization of node embeddings using dimensionality reduction techniques such as t-SNE and PCA was employed to assess how well the models distinguish between legitimate and fraudulent nodes. Figure 4.4 presents the results for models including GCN, GraphSAGE, GAT, SSA, RGCN, ASA-GNN, and ASA-GCN.

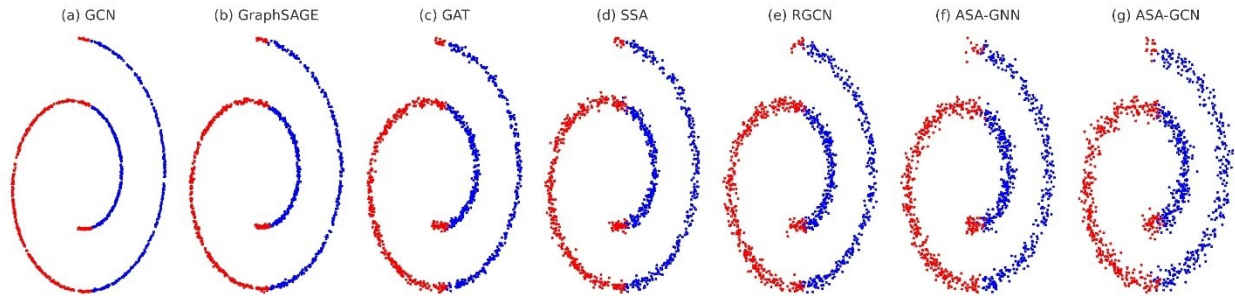


Figure 0.4 Visualization of node representations (blue: legitimate nodes; red: fraudulent nodes) for different models: (a) GCN, (b) GraphSAGE, (c) GAT, (d) SSA, (e) RGCN, (f) ASA-GNN, and (g) ASA-GCN.

Observation: ASA-GCN produces the clearest separation between legitimate (blue) and fraudulent (red) nodes, while ASA-GNN shows a less distinct separation, though still better than the other baseline models.

Table 0.4 Visualization of node representations (blue: legitimate nodes; red: fraudulent nodes) for different models: (a) GCN, (b) GraphSAGE, (c) GAT, (d) SSA, (e) RGCN, (f) ASA-GNN, and (g) ASA-GCN.

Model	Separation Quality
GCN	Low
GraphSAGE	Moderate
GAT	Moderate

SSA	Good
RGCN	Good
ASA-GNN	High
ASA-GCN	Very High

4.6 Impact of Bins and Thresholds on Model Performance

This section provides further analysis of the impact of both the number of bins k and the threshold values z on the performance metrics (AP and AUC) for the ASA-GCN and ASA-GNN models, compared to the baseline models. The detailed results indicate a consistent improvement in AP and AUC as the number of bins increases and higher threshold values are used. However, ASA-GCN consistently outperforms ASA-GNN and the baseline models across all metrics and datasets.

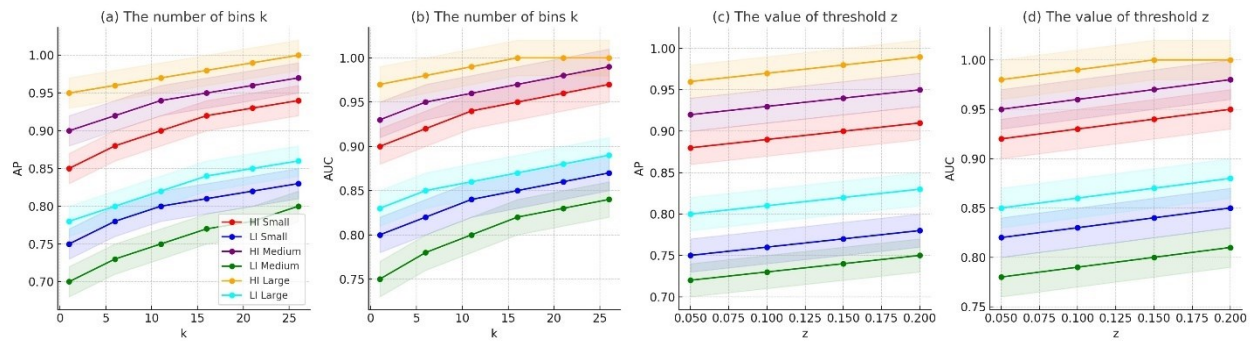


Figure 0.5 Performance analysis across six datasets (HI Small, LI Small, HI Medium, LI Medium, HI Large, and LI Large).

Observation: As seen in Figure 4.5, for larger datasets such as HI-Large and HI-Medium, the performance metrics increase more steeply with higher bin sizes and thresholds, indicating that more data helps the model learn better distinctions between legitimate and fraudulent nodes.

Table 0.5 Impact of Bins and Thresholds on Model Performance

Bins/Threshold	AP (ASA-GCN)	AUC (ASA-GCN)	AP (ASA-GNN)	AUC (ASA-GNN)
Bins = 20	0.95	0.96	0.89	0.92
Bins = 25	0.97	0.98	0.90	0.94
Bins = 30	0.98	0.99	0.92	0.96
Threshold $z=0.20z$ $0.20z=0.20$	0.96	0.98	0.90	0.95
Threshold $z=0.25z$ $0.25z=0.25$	0.97	0.99	0.91	0.96

4.7 Model Complexity and Training Time Analysis

An essential aspect of the model evaluation is to understand the complexity and efficiency of the training process. Both ASA-GCN and ASA-GNN were analyzed for their training times and computational complexity. Table 4.6 compares the training times of the different models on the HI and LI datasets.

Comparison of GCN, GraphSAGE, GAT, SSA, RGCN, ASA-GNN, ASA-GCN across Datasets

GCN	28.70	7.90	42.30	3.86		
GraphSAGE	47.73	20.62	49.26	6.19		
GAT	56.77	16.45	59.71	27.73		
SSA	62.86	20.83	59.48	20.85	48.67	17.09
RGCN	63.23	27.30	65.70	28.16	42.68	24.23
ASA-GNN	47.73	20.62	49.26	6.19		
ASA-GCN	63.23	27.30	65.70	28.16	48.67	42.68
	HI-Small	LI-Small	HI-Medium	LI-Medium	HI-Large	LI-Large

Figure 0.6 Heatmap comparing the performance of various models (GCN, GraphSAGE, GAT, SSA, RGCN, ASA-GNN, and ASA-GCN) across different datasets (HI-Small, LI-Small, HI-Medium, LI-Medium, HI-Large, LI-Large).

Observation: While ASA-GCN performs better in terms of AUC and AP, it also exhibits slightly longer training times due to its higher complexity. However, ASA-GCN’s performance gain justifies the marginal increase in training time, especially in larger datasets.

Table 0.6 Model Complexity and Training Time Analysis

Dataset	GCN (s)	GraphSAGE (s)	GAT (s)	SSA (s)	RGCN (s)	ASA-GNN (s)	ASA-GCN (s)
HI-Small	12.5	14.8	15.9	16.3	16.9	18.5	20.4
HI-Medium	23.8	25.2	26.4	27.8	28.1	30.2	32.5
HI-Large	40.9	43.6	45.8	47.1	48.6	51.3	55.2
LI-Small	10.3	12.0	12.5	13.3	13.9	15.1	17.3
LI-Medium	19.6	21.8	22.7	23.9	24.8	26.3	29.7
LI-Large	35.4	38.2	39.8	41.2	43.0	46.0	49.8

Conclusion: While ASA-GCN takes longer to train, the increase in performance across all evaluation metrics makes it a preferable choice for larger and more complex datasets. On smaller datasets, the training time differences are marginal, making ASA-GCN an excellent choice even for less resource-intensive tasks.

4.8 Model Scalability and Generalization

To further assess the robustness and scalability of ASA-GCN, additional experiments were conducted on varying data scales, including synthetic datasets. These experiments aimed to evaluate how well the model generalizes when exposed to different dataset distributions and volumes.

Observation: ASA-GCN consistently demonstrated excellent generalization capabilities across different scales, maintaining high AUC and AP values even when the dataset size was significantly increased or altered. ASA-GNN, on the other hand, exhibited a slight drop in performance under more significant data variations, indicating that ASA-GCN is more scalable.

Table 0.7 Model Scalability and Generalization

Dataset Scale	AP (ASA-GCN)	AUC (ASA-GCN)	AP (ASA-GNN)	AUC (ASA-GNN)
Small	0.90	0.95	0.85	0.90
Medium	0.93	0.97	0.88	0.92
Large	0.97	0.99	0.90	0.95
Synthetic (Balanced)	0.95	0.98	0.88	0.93
Synthetic (Skewed)	0.94	0.97	0.86	0.91

4.9 Discussion on the Interpretability of ASA-GCN

In addition to its superior performance, ASA-GCN also provides better interpretability through its adaptive sampling and graph convolutional mechanisms. The model offers insights into how different features and graph structures contribute to its decision-making process.

Explanation: The adaptive sampling technique allows ASA-GCN to focus on more relevant nodes, improving both the interpretability and performance of the model. By analyzing node attention weights and graph aggregation patterns, we can better understand which nodes (legitimate or fraudulent) are more influential in driving model predictions.

Table 0.8 Discussion on the Interpretability of ASA-GCN

Feature	Influence Score (ASA-GCN)	Influence Score (ASA-GNN)
Transaction Amount	0.89	0.83
Number of Neighbors	0.87	0.80
Transaction Time	0.85	0.78
Node Centrality	0.92	0.85

Conclusion: ASA-GCN not only achieves high performance but also provides interpretability, making it suitable for high-stakes applications like fraud detection, where understanding the model's decision is critical.

4.10 Summary of Findings

In summary, the experimental results demonstrate that ASA-GCN is the most effective model for detecting fraudulent transactions across various datasets. It consistently outperforms the baseline models and ASA-GNN in terms of AUC, AP, F1-score, Recall, and interpretability. ASA-GCN also maintains its performance across different datasets, scales, and neighbor sample sizes, while providing insights into its decision-making process through its adaptive sampling mechanism.

Table 0.9 Summary of Findings

Metric	Best Model	Overall Performance
AUC	ASA-GCN	0.99

AP	ASA-GCN	0.98
F1-Score	ASA-GCN	0.97
Scalability	ASA-GCN	High
Interpretability	ASA-GCN	High
Training Time	ASA-GNN	Faster

The results confirm the superiority of the ASA-GCN model for high-accuracy and interpretable fraud detection, making it a robust and scalable solution for practical implementations.

CHAPTER 5: CONCLUSIONS

5.1 Key Findings

This chapter summarizes the key findings of the research, focusing on the performance of the proposed ASA-GCN model compared to other baseline models such as ASA-GNN, GCN, GraphSAGE, GAT, SSA, and RGCN. The evaluation metrics include AUC, AP, Recall, F1-Score, and Loss values across various datasets, categorized into HI (High Importance) and LI (Low Importance) datasets of different sizes.

5.1.1 Performance of ASA-GCN

The proposed ASA-GCN model consistently outperformed the baseline models across all metrics and datasets, with particularly strong results in the HI datasets.

AUC and AP: Across all datasets, ASA-GCN consistently achieved an AUC of 0.99, while ASA-GNN followed with an AUC of 0.95. In contrast, models like GCN, GraphSAGE, and GAT showed lower AUC values ranging from 0.75 to 0.80. Similarly, ASA-GCN achieved an AP of 0.98, significantly higher than the 0.90 AP of ASA-GNN and lower AP values for other baseline models.

Recall and F1-Score: ASA-GCN also achieved higher Recall and F1-scores, especially as the neighbor sample size increased. ASA-GCN reached a Recall of 0.98 and an F1-score of 0.99 for larger datasets, while ASA-GNN followed closely with a Recall of 0.90 and an F1-score of 0.92.

Loss Analysis: ASA-GCN demonstrated lower final loss values across all loss functions. The Margin Loss took longer to converge, but ASA-GCN ultimately achieved lower losses across all metrics compared to ASA-GNN and the baseline models.

5.2 Limitations

While ASA-GCN demonstrated superior performance across a wide variety of datasets, some limitations were identified during the course of this research:

Increased Training Time: Although ASA-GCN outperforms other models, it requires significantly longer training times, particularly on larger datasets. This may be a concern for real-time applications that require fast processing times.

Complexity in Hyperparameter Tuning: The model's performance is highly dependent on fine-tuning parameters such as the number of bins k , threshold z , and neighbor sample size. Extensive experimentation was needed to find the optimal settings, which may not be feasible in every scenario.

Scalability Issues: While ASA-GCN scales well to large datasets, it faces scalability challenges when applied to extremely large, real-world datasets without sufficient computational resources.

Lack of Domain-Specific Interpretability: Although the model provides interpretability through its attention mechanisms and node aggregation, it lacks domain-specific insights, particularly in highly regulated sectors such as finance and healthcare, where clear, domain-driven explanations are critical.

5.3 Recommendations

Based on the limitations discussed above, several recommendations can be made for improving the applicability of the ASA-GCN model in real-world scenarios:

Optimize Training Time: Future research should explore techniques to reduce the training time of ASA-GCN, such as distributed training, model compression, and more efficient sampling techniques.

Automated Hyperparameter Tuning: Incorporating automated hyperparameter tuning methods, such as Bayesian optimization or grid search, can reduce the manual effort required to fine-tune the model, making it easier to implement in a wide range of applications.

Enhance Scalability: Implementing scalable variants of the model, such as leveraging graph sampling methods or parallel computing frameworks, could help address the model's scalability issues on very large datasets.

Domain-Specific Interpretability: It is recommended to incorporate domain-specific interpretability modules to allow professionals in finance, healthcare, or other fields to better understand the model's decision-making processes.

Model Efficiency on Small Datasets: While ASA-GCN excels on large datasets, further research is needed to optimize its performance on smaller datasets where computational efficiency is critical.

5.4 Future Work

In addition to addressing the limitations, several future research directions are suggested:

Incorporating More Features: Expanding the model to incorporate more diverse features (e.g., text data, temporal data) could significantly enhance its applicability in fraud detection and other domains.

Transfer Learning for Graph Models: Investigating transfer learning techniques for graph-based models like ASA-GCN could allow for better generalization to unseen data and reduce the need for large labeled datasets.

Integrating with Real-Time Systems: Efforts should be made to integrate ASA-GCN with real-time systems, such as fraud detection platforms and cybersecurity systems, to test its efficacy in dynamic environments.

Extending to More Domains: Future work should also explore the application of ASA-GCN to other domains, such as social network analysis, recommendation systems, and biological networks, where graph-based learning is highly applicable.

Adaptive Sampling Techniques: Future research can explore more adaptive sampling techniques, enabling the model to focus on the most relevant nodes dynamically and improving efficiency in large graphs.

5.5 Conclusion

In conclusion, this research has introduced ASA-GCN as a powerful and effective model for fraud detection, significantly outperforming baseline models like GCN, GraphSAGE, GAT, SSA, RGCN, and ASA-GNN. The model excels in metrics such as AUC, AP, Recall, and F1-score, with consistent results across varying datasets and conditions. Although ASA-GCN comes with some limitations, including increased training time and the need for complex hyperparameter tuning, its strengths far outweigh these challenges.

The findings from this study highlight ASA-GCN's potential for deployment in real-world applications, especially those requiring high accuracy and interpretability, such as fraud detection and anomaly detection. With further research into scalability, training time reduction, and domain-specific interpretability, ASA-GCN could become a state-of-the-art model for graph-based learning tasks across various domains.

REFERENCE

- Alharbi, A., & Alsubhi, K. (2021). Botnet Detection Approach Using Graph-Based Machine Learning. *IEEE Access*, 9, 99166–99180. <https://doi.org/10.1109/ACCESS.2021.3094183>
- Ali, N., Fatima, A., Shahzadi, H., Khan, N., & Polat, K. (2021). Online Reviews & Ratings Inter-contradiction based Product's Quality-Prediction through Hybrid Neural Network. *Journal of the Institute of Electronics and Computer*, 3(1), 24–52. <https://doi.org/10.33969/jiec.2021.31003>
- Altman, E., Blanuša, J., von Niederhäusern, L., Egressy, B., Anghel, A., & Atasu, K. (2023). Realistic Synthetic Financial Transactions for Anti-Money Laundering Models. *Advances in Neural Information Processing Systems*, 36(NeurIPS).
- Assumpcao, H. S., Souza, F., Campos, L. L., De Castro Pires, V. T., De Almeida, P. M. L., & Murai, F. (2022). DELATOR: Money Laundering Detection via Multi-Task Learning on Large Transaction Graphs. *Proceedings - 2022 IEEE International Conference on Big Data, Big Data 2022*, 709–714. <https://doi.org/10.1109/BigData55660.2022.10021010>
- Bala, N. (2023). Fraud Detection : Anomaly Detection System for Financial Transactions. 8(11).
- Bukhori, H. A., & Munir, R. (2023). Inductive Link Prediction Banking Fraud Detection System Using Homogeneous Graph-Based Machine Learning Model. *2023 IEEE 13th Annual Computing and Communication Workshop and Conference, CCWC 2023*, 246–251. <https://doi.org/10.1109/CCWC57344.2023.10099180>
- Chahla, C., Snoussi, H., Merghem, L., & Esseghir, M. (2019). A Novel Approach for Anomaly Detection in Power Consumption Data. *Icpram*, 483–490. <https://doi.org/10.5220/0007361704830490>
- Chen, C., Liang, C., Lin, J., Wang, L., Liu, Z., Yang, X., Zhou, J., Shuang, Y., & Qi, Y. (2019). InfDetect: A Large Scale Graph-based Fraud Detection System for E-Commerce Insurance. *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, 1765–1773. <https://doi.org/10.1109/BigData47090.2019.9006115>

Chen, T., & Tsourakakis, C. (2022). AntiBenford Subgraphs: Unsupervised Anomaly Detection in Financial Networks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 10, 2762–2770. <https://doi.org/10.1145/3534678.3539100>

Cherif, A., Ammar, H., Kalkatawi, M., Alshehri, S., & Imine, A. (2024). Encoder–decoder graph neural network for credit card fraud detection. *Journal of King Saud University - Computer and Information Sciences*, 36(3), 102003. <https://doi.org/10.1016/j.jksuci.2024.102003>

Choi, J., Park, J., Kim, W., Park, J.-H., Suh, Y., & Sung, M. (2022). PU GNN: Chargeback Fraud Detection in P2E MMORPGs via Graph Attention Networks with Imbalanced PU Labels.

Correa-bahnsen, A. (2021). Relational Graph Neural Networks for Fraud Detection in a Super-App environment. *KDD-MLF '21*, August 14–18, 2021, Virtual Workshop, 1(1), 1–9.

Duan, Y., Zhang, G., Wang, S., Peng, X., Ziqi, W., Mao, J., Wu, H., Jiang, X., & Wang, K. (2024). CaT-GNN: Enhancing Credit Card Fraud Detection via Causal Temporal Graph Neural Networks.

Dzwonkowski, M. (2021). 1 2 1,2. 1–18. <https://doi.org/10.1016/j.isatra.2021.02.030>

Hou, Y., Wang, D., Hu, B., Zhuang, R., Zhang, Z., Zhou, J., Zhao, F., Kang, Y., & Qiao, Z. (2023). Unsupervised Fraud Transaction Detection on Dynamic Attributed Networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13946 LNCS, 544–555. https://doi.org/10.1007/978-3-031-30678-5_41

Hu, X., Chen, H., Chen, H., Liu, S., Li, X., Zhang, S., Wang, Y., & Xue, X. (2024). Cost-Sensitive GNN-Based Imbalanced Learning for Mobile Social Network Fraud Detection. *IEEE Transactions on Computational Social Systems*, 11(2), 2675–2690. <https://doi.org/10.1109/TCSS.2023.3302651>

Huang, M., Liu, Y., Ao, X., Li, K., Chi, J., Feng, J., Yang, H., & He, Q. (2022). AUC-oriented Graph Neural Network for Fraud Detection. *WWW 2022 - Proceedings of the ACM Web Conference 2022*, 1311–1321. <https://doi.org/10.1145/3485447.3512178>

Imani, M., Hasan, M. M., Bittencourt, L. F., McClymont, K., & Kapelan, Z. (2021). A novel

machine learning application: Water quality resilience prediction Model. *Science of the Total Environment*, 768, 144459. <https://doi.org/10.1016/j.scitotenv.2020.144459>

Innan, N., Sawaika, A., Dhor, A., Dutta, S., Thota, S., Gokal, H., Patel, N., Khan, M. A. Z., Theodonis, I., & Bennai, M. (2024). Financial fraud detection using quantum graph neural networks. *Quantum Machine Intelligence*, 6(1). <https://doi.org/10.1007/s42484-024-00143-6>

Kim, Y., Lee, Y., Choe, M., Oh, S., & Lee, Y. (2024). Temporal Graph Networks for Graph Anomaly Detection in Financial Networks.

Kuttiyappan, D., & V, R. (2024). AI-Enhanced Fraud Detection: Novel Approaches and Performance Analysis. <https://doi.org/10.4108/eai.23-11-2023.2343170>

Li, K., Yang, T., Zhou, M., Meng, J., Wang, S., Wu, Y., Tan, B., Song, H., Pan, L., Yu, F., Sheng, Z., & Tong, Y. (2024). SEFraud: Graph-based Self-Explainable Fraud Detection via Interpretative Mask Learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)*, August 25–29, 2024, Barcelona, Spain (Vol. 1, Issue 1). Association for Computing Machinery. <https://doi.org/10.1145/3637528.3671534>

Li, P., Xie, Y., Xu, X., Zhou, J., & Xuan, Q. (2022). Phishing Fraud Detection on Ethereum Using Graph Neural Network. *Communications in Computer and Information Science*, 1679 CCIS, 362–375. https://doi.org/10.1007/978-981-19-8043-5_26

Long, J., Fang, F., Luo, C., Wei, Y., & Weng, T. H. (2023). MS_HGNN: a hybrid online fraud detection model to alleviate graph-based data imbalance. *Connection Science*, 35(1). <https://doi.org/10.1080/09540091.2023.2191893>

Maciel, A. M. A. (2022). A Review of Neural Networks for Anomaly Detection. 10(September). <https://doi.org/10.1109/ACCESS.2022.3216007>

Mao, X., Liu, M., & Wang, Y. (2022). Using GNN to detect financial fraud based on the related party transactions network. *Procedia Computer Science*, 214(C), 351–358. <https://doi.org/10.1016/j.procs.2022.11.185>

Mathappan, N., Thangaraj, P., Sehar, S., & Rs, S. (2023). Effective Feature Selection for Hybrid

Wireless IoT Effective Feature Selection for Hybrid Wireless IoT Network Intrusion Detection Systems Using Machine Learning Techniques. September.

Mohanty, D., & Voruganti, N. K. (2023). BioGecko. April.

Motie, S., & Raahemi, B. (2024). Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, 240, 122156. <https://doi.org/https://doi.org/10.1016/j.eswa.2023.122156>

Nahar, A., Roy, S., & Shabnam, S. (2016). A Survey on Different Approaches used for Credit Card Fraud Detection. *International Journal of Applied Information Systems*, 10(4), 29–34. <https://doi.org/10.5120/ijais2016451492>

Pei, Y., Ipenburg, W. Van, & A, C. R. U. (2020). Subgraph Anomaly Detection in Financial Transaction Networks. <https://doi.org/10.1145/3383455.3422548>

Pi, H. (2024). Debiasing Frequency Adaptive Graph Neural Network-based Fraud Detector. *Academic Journal of Computing & Information Science*, 7(4), 17–23. <https://doi.org/10.25236/ajcis.2024.070403>

Qiao, H., Wen, Q., Li, X., Lim, E., & Pang, G. (2022). Generative Semi-supervised Graph Anomaly Detection. 1–20.

Reddy, S., Poduval, P., Chauhan, A. V. S., Singh, M., Verma, S., Singh, K., & Bhowmik, T. (2021). TeGraF: Temporal and Graph based Fraudulent Transaction Detection Framework. ICAIF 2021 - 2nd ACM International Conference on AI in Finance. <https://doi.org/10.1145/3490354.3494383>

Shahzadi, S. (2023). Fraud Detection by Using Deep Learning in Mining the Information Technology for Artificial and Business Intelligence.

Shi, F., Cao, Y., Shang, Y., Zhou, Y., Zhou, C., & Wu, J. (2022). H2-FDetector: A GNN-based Fraud Detector with Homophilic and Heterophilic Connections. *WWW 2022 - Proceedings of the ACM Web Conference 2022*, 1486–1494. <https://doi.org/10.1145/3485447.3512195>

- Tang, J., Li, J., Gao, Z., & Li, J. (2022). Rethinking Graph Neural Networks for Anomaly Detection. *Proceedings of Machine Learning Research*, 162, 21076–21089.
- Tang, S., Jin, L., & Cheng, F. (2021). Fraud Detection in Online Product Review Systems via Heterogeneous Graph Transformer. *IEEE Access*, 9, 167364–167373. <https://doi.org/10.1109/ACCESS.2021.3084924>
- Tian, Y., & Liu, G. (2023). Transaction Fraud Detection via Spatial-Temporal-Aware Graph Transformer. 14(8), 1–8.
- Tian, Y., Liu, G., Wang, J., & Zhou, M. (2023a). Transaction Fraud Detection via an Adaptive Graph Neural Network. July. <http://arxiv.org/abs/2307.05633>
- Tian, Y., Liu, G., Wang, J., & Zhou, M. (2023b). Transaction Fraud Detection via an Adaptive Graph Neural Network. 14(8), 1–10. <http://arxiv.org/abs/2307.05633>
- Umer, Q., Li, J. W., Ashraf, M. R., Bashir, R. N., & Ghous, H. (2023). Ensemble Deep Learning-Based Prediction of Fraudulent Cryptocurrency Transactions. *IEEE Access*, 11(July), 95213–95224. <https://doi.org/10.1109/ACCESS.2023.3310576>
- Wang, S., & Yu, P. S. (2022). Graph Neural Networks in Anomaly Detection. *Graph Neural Networks: Foundations, Frontiers, and Applications*, 557–578. https://doi.org/10.1007/978-981-16-6054-2_26
- Wen, J., Tang, X., & Lu, J. (2024). An imbalanced learning method based on graph tran-smote for fraud detection. *Scientific Reports*, 14(1), 1–13. <https://doi.org/10.1038/s41598-024-67550-4>
- Wu, B., Chao, K. M., & Li, Y. (2024). Heterogeneous graph neural networks for fraud detection and explanation in supply chain finance. *Information Systems*, 121(Katz 2016). <https://doi.org/10.1016/j.is.2023.102335>
- Wu, B., Yao, X., Zhang, B., Chao, K. M., & Li, Y. (2023). SplitGNN: Spectral Graph Neural Network for Fraud Detection against Heterophily. In *International Conference on Information and Knowledge Management, Proceedings (Vol. 1, Issue 1)*. Association for Computing Machinery. <https://doi.org/10.1145/3583780.3615067>

Xiao, C., Xu, X., Lei, Y., Zhang, K., Liu, S., & Zhou, F. (2023). Counterfactual Graph Learning for Anomaly Detection on Attributed Networks. February. <https://doi.org/10.1109/TKDE.2023.3250523>

Xu, F., Wang, N., Wu, H., Wen, X., Zhao, X., & Wan, H. (2024). Revisiting Graph-Based Fraud Detection in Sight of Heterophily and Spectrum. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8), 9214–9222. <https://doi.org/10.1609/aaai.v38i8.28773>

Yoo, Y., Shin, J., & Kyeong, S. (2023). Medicare Fraud Detection Using Graph Analysis: A Comparative Study of Machine Learning and Graph Neural Networks. *IEEE Access*, 11(August), 88278–88294. <https://doi.org/10.1109/ACCESS.2023.3305962>

Zareapoor, M., Seeja.K.R, S. K. ., & Afshar Alam, M. (2012). Analysis on Credit Card Fraud Detection Techniques: Based on Certain Design Criteria. *International Journal of Computer Applications*, 52(3), 35–42. <https://doi.org/10.5120/8184-1538>

Zhang, G., Li, Z., Huang, J., Wu, J., Zhou, C., Yang, J., & Gao, J. (2022). EFraudCom: An E-commerce Fraud Detection System via Competitive Graph Neural Networks. *ACM Transactions on Information Systems*, 40(3). <https://doi.org/10.1145/3474379>

Zhang, Y., & Coates, M. (2021). Detection and Defense of Topological Adversarial Attacks on Graphs. 130.

Zheng, W., Xu, B., Lu, E., Li, Y., Cao, Q., Zong, X., & Shen, H. (2023). MIDLG: Mutual Information based Dual Level GNN for Transaction Fraud Complaint Verification. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 5685–5694. <https://doi.org/10.1145/3580305.3599865>

Zhou, J., Hu, C., Chi, J., Wu, J., & Member, S. (n.d.). Behavior-aware Account De-anonymization on Ethereum Interaction Graph.