

AN EXPLAINABLE GRAPH NEURAL NETWORK FRAMEWORK FOR ANTI-MONEY LAUNDERING IN CRYPTOCURRENCY TRANSACTIONS USING THE ELLIPTIC DATASET

Oluwatosin Lawal ¹, Awele Okolie ², Callistus Obunadike ³

¹ Department of Mathematics Statistical Analytics, Computing and Modeling, Texas A&M University, Kingsville, USA

² School of Computing and Data Science, Wentworth Institute of Technology, Boston, USA.

³ Department of Computer Science and Quantitative Methods, Austin Peay State University, Tennessee, USA.

ABSTRACT

The detection of illicit cryptocurrency transactions remains a significant challenge due to the extreme class imbalance and limited generalization capabilities of machine learning models applied to Anti-Money Laundering (AML) data. In the widely used Elliptic dataset, illicit transactions represent less than 2% of all nodes, creating a high-risk setting in which models can achieve deceptively high training accuracy while failing to meaningfully identify malicious behavior. This study examines the behavior of Graph Neural Networks (GNNs) under these constraints and emphasizes the limitations rather than the performance of the approach. Instead of treating the model's high training accuracy as a success, we demonstrate how imbalance, structural sparsity, and label noise impede reliable learning. We evaluate the model with and without common imbalance-handling strategies including class weighting and focal loss and illustrate that performance remains unstable. Furthermore, we investigate the explainability of the model using GNNExplainer, showing example subgraphs and salient features for known illicit nodes, and discuss their alignment with money-laundering patterns such as fan-out and transaction mixing. Our findings underscore the difficulties of applying GNNs to heavily imbalanced AML datasets and highlight the need for improved modeling strategies, semi-supervised techniques, and more robust explainability methods for real-world financial crime detection.

KEYWORDS

Graph Neural Networks (GNN), Explainable AI (XAI), Anti-Money Laundering (AML), Cryptocurrency, Blockchain Analytics, Elliptic Dataset, Graph Convolutional Network (GCN), Financial Fraud Detection, SHAP, Transaction Network Analysis

1. INTRODUCTION

Money laundering is an issue that often comes to the forefront while discussing the integrity of financial systems and the rising cryptocurrency transactions have only made it more complex. The use of blockchain networks by the criminal world to hide the sources and movements of their dark money, which is the biggest challenge for the existing detection mechanisms. It is not only the traditional rule-based and statistical methods that have failed to understand the complicated inter-entity relationships, but also the network-based approaches that are now viewed as potential

alternatives. Among them, graph neural networks (GNNs) are indeed a powerful framework when the transaction networks are represented as graphs with transactions (or wallets) on the nodes and financial flows as edges. GNNs can use message passing and representation learning to yield latent structural features and anomalies that would be nearly impossible to find with tabular machine-learning techniques alone. For instance, one recent study illustrates the GNNs' ability in the AML context showing that such models can detect illicit activities by exploiting graph topology and node attributes which is beyond the capacity of classical approach (Weber et al., 2018).

The issue of transparency regarding these deep models has become a major one, especially in the area of financial crime detection, where interpretability and auditability signify the ability to comply with regulations. Consequently, researchers have looked into the use of explainable GNN frameworks combined with graph learning and explained noteworthy nodes, edges, or features' movements as suspicious classifications (Lo et al., 2022). Conversely, the insufficient number of labeled examples in actual transaction networks remains a major obstacle; thus, self-supervised and contrastive learning methods have been brought in for pre-training GNNs, which reduce reliance on labeled data and at the same time, enhance performance (Zhang et al., 2024). The present paper proposes an explainable GNN model for anti-money laundering detection in crypto transactions based on the open-source Elliptic dataset. We treat the transaction network as a graph, implement a GCN-based architecture for the classification task, and simultaneously make the decision process traceable by incorporating explainability tools. In this way, we hope to show that a detection system capable of interpreting outputs and thus safeguarding both operational efficiency and regulatory transparency in the crypto-financial crime segment could be set up in blockchain networks and especially in cases of suspicious transactions.

2. LITERATURE REVIEW

We evaluate pertinent prior studies in this part through several dimensions: the types of money-laundering activities done via cryptocurrency systems, the classic graph-based anomaly/fraud detection, GNNs for AML getting more and more, explainability in GNNs, as well as the currently available datasets and benchmarking frameworks. This indicates the gap our research is filling and places our method among the best.

2.1. Money Laundering in Cryptocurrency Networks

The use of cryptocurrency networks for money laundering has become one of the major obstacles in the way of financial crime prevention. Even though blockchain technology provides clarity and is unchangeable, the wrongdoers have taken the advantage of the anonymity that digital currencies come with, employing various methods such as peer-to-peer transfers, mixers, and cross-border flows to hide their illegal money (Europol, 2023). The typical money laundering three-stages process of placement, layering and integration has been adapted to crypto ecosystems: the virtual currency entry of illicit funds is the placement stage, the mixing services or the cascading of transactions between wallets and exchanges is the layering and the funds exit through a crypto or compliant service conversion, often back to fiat, to the legitimate economy, this being the integration (UNODC, 2024). All these mechanisms make it necessary to model the relational and temporal patterns of transactions (e.g. multi-hop flows, fan-in/fan-out structures) to detect laundering while feature-based or rule-based systems frequently fail to recognize the networked behavior of laundering operators. While feature-based or rule-based systems frequently fail to identify complex networked behavior, a limitation similarly observed in other machine-learning domains where advanced predictive models outperform simple rule-based approaches (Okolie et al., 2025).

2.2. Graph-Based Anomaly Detection in Financial Systems

The very nature of financial transactions to create relational networks has led to the emergence of graph-based anomaly detection as a key method for the detection of suspicious activities and irregularities in the payment systems' structure. Rule-based or tabular methods, which are the ones used traditionally, treat each transaction as an independent case and thus miss topological dependencies like the presence of intermediaries, having transfer paths or clusters of transactions with a high risk of being fraudulent, etc. However, graph analytical techniques can capture these dependencies through centrality measures, community detection, and motif analysis, thereby uncovering potential laundering schemes that occur as subgraphs or repetitive money-flow structures (Akoglu et al., 2015). The use of deep learning on graphs for financial fraud applications has been on the rise among researchers as a means of modeling these intricate relationships. Specifically, GNNs, especially GCNs and GATs, are being used to learn embeddings that encapsulate both node characteristics and structural context, thus making anomaly detection in banking, insurance, and blockchain networks more robust (Wang et al., 2021). These methods not only achieve but also surpass the performance of feature-engineering-based baselines by spreading the information through edges in order to uncover hidden dependencies among transactions. Thus, graph-based models signal a paradigm change in the detection of financial anomalies, i.e., they shift from static feature learning to dynamic, topology-aware inference which is suitable for AML and fraud prevention systems. Similar approaches have been used in other high-dimensional systems where relational or spatiotemporal dependencies drive anomalous behavior. For example, Okolie et al. (2025) demonstrated how machine learning can model complex spatial-temporal interactions to identify high-risk accident hotspots in urban traffic networks, highlighting the importance of contextual and structural features in anomaly detection systems.

2.3. Graph Neural Networks for AML in Cryptocurrency

The use of Graph Neural Networks (GNNs) has revolutionized the approach taken to deal with anti-money laundering (AML) apps in cryptocurrency ecosystems. Although machine learning models, which are traditional and therefore somewhat slow, still manage to classify suspicious transactions, they do not take advantage of the fundamental graph structure of blockchain data where nodes connected by edges represent either a transaction or an address and edges denote the transfer of funds. GNNs accomplish this precisely by performing message passing among connected entities, and thus they acquire representations that include not only the features of the transactions but also their relational context (Weber et al., 2019). Among the first studies that applied GNNs to the Elliptic dataset was the one that most influenced the use of this technique. GCNs could therefore surpass the models based on gradient boosting and logistic regression in the area of illicit transaction identification (Weber et al., 2019). Later, the researchers improved upon these methods by adding up temporal and self-supervised learning thereby making money-laundering behaviors evolving even easier to understand. For example, Lo et al. (2022) came up with Inspection-L, which is a self-supervised GNN method, that creates node embeddings without needing large volumes of labeled data. By doing so, this method not only increased the generalization to unseen laundering typologies but also underscored the strength of GNNs in sparse, real-world cryptocurrency graphs. In the end, all these studies have shown that GNNs are the right technology for AML detection in the blockchain settings as they combine structural representation learning, temporal modeling, and semi-supervised training paradigm thus paving the way for explainable and regulatory-compliant graph-based AML systems.

2.4. Explainability and Interpretability in GNN-Based AML

Despite GNNs being the best choice for illegal crypto transaction detection, still their opacity remains a main hindrance for the use in financial area where there are strict regulations. Banks and auditors want to be able to trace the reasons behind the inclusion of certain transactions or parties on the list of suspicious ones not only for the sake of regulatory compliance but also for the loyalty on AI-supported systems. Explainable AI (XAI) by the way has been the driving force in the construction of Gauging the Money laundering Process (GMP) models that are based on GNNs. (Ying et al., 2019). Several techniques can lead to the explanation of GNNs. Such as GNNExplainer the methods develops subgraphs and masks on the features revealing among the nodes, edges, and attributes for the prediction which of them is most contributing, thereby giving local interpretation. The models based on attention such as Graph Attention Networks (GATs) provide explainability naturally by assigning attention weights to neighbors revealing which connections had the biggest impact on the classification. In the domain of anti-money laundering, the researchers are already after linking such GNN-architectures with SHAP (SHapley Additive exPlanations) and LIME frameworks for the purpose of determining the contribution of features at the transaction level (Xu et al., 2024). Among other things, these approaches improve answerability, but they also help in the detection of laundering schemes by consulting with the experts. The patterns discovered may even correspond to the ones that are already known by regulators (e.g., layering, structuring, or circular trading). Still, there are difficulties when it comes to making GNNs explainable on really massive graphs, linking interpretability concerning time to explain GNNs and turning insights for compliance departments through operational steps.

2.5. Datasets, Benchmarks, and Research Challenges

The progress of research in anti-money laundering (AML) is greatly dependent on reliable and well-annotated datasets, particularly for the analysis of cryptocurrency transactions. The Elliptic Dataset is the most widely used among the few publicly available resources and serves as a benchmark for measuring the effectiveness of graph-based AML techniques. The dataset depicts an actual Bitcoin transaction graph, where the nodes symbolize transactions, the edges indicate the transfer of funds, and each node is classified as legal, illegal, or uncertain (Weber et al., 2019). The dataset contains more than 200,000 transactions and 49 temporal periods, which makes it perfect for both static and temporal GNN research. However, despite being widely used, the Elliptic dataset has significant drawbacks such as extreme class imbalance (with illegitimate transactions accounting for less than 2%), partial labeling, and a lack of explicit typology annotations (for example, mixing, layering, and smurfing). These shortcomings hinder the assessment of generalization ways across laundering methods. To fill in such voids, new datasets such as Elliptic2 (Bellei et al., 2024) and synthetic graph-based AML datasets are being developed with transaction semantics that are more comprehensive, community-level labeling, and monitoring of temporal evolution. The GNN-based AML model benchmarking process also has to deal with both computational and interpretative difficulties. Real-world blockchain systems' graphs are enormous and constantly changing, which calls for the implementation of large-scale learning frameworks and efficient readability tools. Besides, it remains problematic to transfer models that have been trained on Bitcoin data to other blockchains like Ethereum or Monero due to differences in structure and behavior. One possible future research area is cross-chain GNNs, temporal explainability frameworks, and privacy-preserving graph models that are compliant with the regulatory standards while still maintaining interpretability and detection power.

3. METHODOLOGY

This part covers the research design, data set layout, preprocessing pipeline, model architecture, training configuration, and the explainability methods utilized to clarify the model results. The procedure combines the Elliptic dataset with a Graph Convolutional Network (GCN) architecture and explainability modules to provide a clearer view in anti-money laundering (AML) detection process.

3.1. Research Design and Framework Overview

The proposed research framework is structured to identify illegal cryptocurrency transactions by integrating graph-based learning and explainable artificial intelligence techniques. The study is rooted in the Elliptic dataset, where the transactions are represented as nodes and the cryptocurrency flow between them is depicted as directed edges. The identification of suspicious patterns in the blockchain ecosystem is made possible by the construction of a comprehensive transaction network through the capturing of these relationships. The methodological workflow is made up of various interconnected stages, starting with the obtaining of the Elliptic dataset and the corresponding transaction graph being built. After that, a thorough preprocessing phase takes place, which is dedicated to making sure the data is cleaned, normalized, and encoded properly to be suitable for the graph-based analysis. The main component of the framework is the training of a Graph Neural Network (GNN), particularly a Graph Convolutional Network (GCN), for the purpose of classifying each transaction to be either licit, illicit, or unknown based on its structural and contextual features. The model has then been trained, and the explainability techniques have been implemented for interpreting the decision-making process of the model, thus providing the transparency of how the classification outcomes are influenced by the specific features and node relationships. The holistic design primarily targets the establishment of a perfect ratio of predictive accuracy and interpretability which is going to allow financial analysts, compliance officers, and regulatory authorities to trust and understand the model's outputs. The framework, being the incorporation of explainable GNN mechanisms into the anti-money laundering (AML) detection pipeline, plays a part in making machine learning applications more transparent, interpretable, and ethically responsible.

3.2. Dataset Description

The Elliptic Dataset is the Dataset employed by this study, whence comes the large-scale, real-world dataset, which is designed for anti-money laundering (AML) the research in cryptocurrency networks. The dataset, by Elliptic consists of a directed transaction graph which has been drawn from the Bitcoin blockchain. A node in the graph stands for a transaction, while an edge shows the movement of Bitcoin between two transactions thus representing the dynamic nature of cryptocurrency exchanges. The dataset as a whole includes 203,769 nodes and 234,355 directed edges that are spread over 49 temporal snapshots, each of which documents one aspect of the network over time. The Elliptic Dataset has one feature vector of 166 attributes assigned to every transaction node, and those are further subdivided into two main groups: local features that highlight the intrinsic characteristics of the transaction, like the number of input and output addresses, and aggregated features that statistically reflect the behavior of neighboring nodes within one or two hops. The features altogether convey the structural, statistical, and temporal behavior of transactions, making it easier to analyze in detail the patterns that are usually associated with illicit activity.

The dataset contains three distinct annotations that indicate whether a transaction is legal, illegal, or uncertain. Illegal transactions are those related to criminal activities, like ransomware, markets

on the dark web, and scams, which are the main activities associated with the illicit nature of a transaction. Legal transactions correspond to normal economic activity, while most of the transactions without labels are categorized as “unknown.” The extreme disparity of the situation, where illicit transactions make up less than 2% of the labeled data, creates a major problem because it is in line with the real-world situation where the illegal financial activities are rare and difficult to detect as they are often hidden among the large number of legitimate transactions. For this research, the dataset undergoes a preprocessing step that involves the concatenation of the three CSV files: the transaction features (`elliptic_txs_features.csv`), the edge list (`elliptic_txs_edgelist.csv`), and the class labels (`elliptic_txs_classes.csv`). The resulting graph is subject to normalization and encoding before being fed into the Graph Neural Network model as input. To obtain a reliable evaluation, the dataset is divided into training, validation, and test subsets by employing stratified sampling, which keeps the label distribution consistent across the splits. The present study utilizes the Elliptic Dataset to the fullest and thus it reveals the structural and temporal aspects of cryptocurrency transactions working in tandem. The dataset is therefore perfect for the development and testing of an explainable graph neural network specifically for money laundering detection in blockchain systems.

3.3. Data Preprocessing and Graph Construction

Performing proper data preprocessing is a very important and necessary step in the process of preparing the Elliptic dataset for graph-based learning. The diverse and complex nature of blockchain transactions as well as the huge size of the dataset, led to the implementation of a number of cleaning and transformation processes in different stages to make sure the data was fit for graph neural network modeling. The fusion of the dataset’s main elementstransaction features, class labels, and the edge listwas the starting point of the process. The transaction identifier served as a unique key for merging the components. After that, alignment was done for each transaction’s 166-dimensional feature vector with its corresponding class label and its position in the network, which was defined by the directed edges through which transactions were connected. Mean-value imputation was used for numerical variables to handle the missing or incomplete feature values during this step and hence, all feature vectors were made complete and consistent. Once the data was merged and cleaned, feature normalization was executed whereby all attributes were brought to a common scale. The standardization of each feature to have zero mean and unit variance was done in order to stabilize the gradient updates during training, as well as to prevent construction of the learning process by the dominating features having large magnitudes. The categorical transaction labelslicit, illicit, and unknownwere encoded into numerical classes (0, 1, and 2, respectively) by using a label encoder, this made direct compatibility with the classification objective function in the neural network possible.

The process of graph construction involved creating a PyTorch Geometric (PyG) Data object with the preprocessed data, which is the most common format for graph-based learning tasks. In this representation, the feature matrix x contains all the node-level attributes, and the `edge_index` tensor indicates the direct relationships between the transactions. The target labels for the node-level classifications are stored in y . This single representation permits the model to spread and accumulate information across the network topology, thus capturing both local and global transaction dependencies. For effective model evaluation and to avoid data leakage, the stratified sampling technique was adopted to divide the graph data into training, validation, and testing subsets. The data was divided into about 70% training, 15% validation, and 15% testing. This division guaranteed that the model was trained on a representative distribution of both legal and illegal transactions, while keeping samples unseen for a performance assessment that is not biased. By this thorough preprocessing and graph construction process, the dataset was converted into a strong graph representation that is suitable for deep relational learning. It was made sure

that both structural and contextual transaction information were preserved and a reliable foundation was set for the subsequent GNN-based classification and explainability analysis.

3.4. Model Architecture

The model that has been proposed for identifying illegal transactions in cryptocurrency has the architecture of Graph Convolutional Network (GCN) which is very good at recognizing the dependencies of the structure and the relationships in the transaction graph. The GCN is working on the graph form of the Elliptic dataset, which gives it the power to pull together the information from the neighboring nodes and their linked features to create the high-level representations that are good for classification. The structure of the model is made up of two graph convolutional layers and a completely connected classification layer. During the first convolutional layer, the model transforms the input feature matrix into a latent representation with 64 hidden dimensions, local structural dependencies of the transactions are being captured. The ReLU activation function aids in introducing the non-linearity thereby allowing the network to depict the complexity of relationships in the transaction network. To prevent overfitting, dropout regularization with a 0.5 rate is applied between the layers. The second convolutional layer then enhances these representations by incorporating higher-order neighborhood information, thereby resulting in the creation of a feature embedding that reflects both the direct and indirect transaction patterns.

The softmax classifier that is responsible for transforming the output from the convolutional layers gives a probability distribution over the three possible classes: licit, illicit, and unknown. The model is trained using the Adam optimizer with a learning rate of 0.005 along with a cross-entropy loss function, which indicates the distance between the predicted and true class distributions. The weight decay regularization parameter of 5×10^{-4} on the other hand helps in reducing the risk of overfitting as it penalizes large model weights. The learning process is a cycle where the GCN continuously updates the node embeddings through the passing of messages between the interconnected transactions. During every training epoch, the network collects features from its neighboring nodes and normalizes them using the graph Laplacian, making sure that the information flow is stable through the different levels of node connectivity. The early stopping technique is employed based on the validation accuracy, thus avoiding unnecessary computations when the model reaches its optimal generalization performance. This particular configuration is very apt for the AML detection task, as it takes advantage of the relational context of each transaction to deduce its legitimacy. Classical machine learning methods that treat transactions as isolated incidents do not pick up the delicate multi-hop dependencies often manipulated by money launderers to hide illicit flows. On the contrary, the GCN architecture's capacity to learn hierarchical graph representations makes it possible to catch the suspicious patterns that are not directly perceptible at the individual transaction level.

The model, by means of this design, attains a compromise between the two aspects, i.e., performance and explainability, eventually giving rise to interpretable embeddings which are subsequently analyzable in terms of the classification of a transaction as illicit based on which nodes, edges, or features that influenced the decision the most. This is solid ground for the merging of explainability tools mentioned in the upcoming paragraph.

3.5. Explainability Integration and Visualization

Graph neural networks are very effective in relational learning; however, they are not very transparent in their decision-making process, which presents a challenge to the analysts who try to figure out the reasons for the certain transactions being categorized as illicit. One of the main contributions of the presented framework is the explanation module that interprets the learned

representations and focuses on the key features, nodes, and edges that affect the predictions of the model. Combining interpretability with radical transparency, the system is now more reliable, auditable, and can thus be used for the high-stakes financial intelligence applications. The explainability phase takes advantage of GNNExplainer, which is an already established model-agnostic technique that reveals the top subgraphs and features that are most responsible for node-level predictions. GNNExplainer computes the importance scores of features for each target transaction and then generates a local subgraph depicting how the connected transactions affect the final classification. This process results in a very clear and simple explanation of the model's reasoning which uncovers if the decision to classify a transaction as illicit is driven by structural connectivity, temporal behavior, or attribute patterns like large transfer amounts or fast transaction chains. One of the main activities of interpretability is visualization. The explanatory subgraphs that have been extracted are visualized as graphs with the use of such libraries as NetworkX and Matplotlib, where the colors of nodes denote the class labels and the thickness of edges indicates the interaction strength or influence. This visualization easily marks high-risk transactions, thus enabling the AML investigators to follow the money through many hops and locate the origin or destination of the suspicious funds. Consequently, the visualization aspect acts as a link between machine learning results and human decision-making, thereby converting nonconcrete embeddings into regulatory and compliance teams' actionable insights. The add-on of the explainability module not only enhances the trust in the GNN model but also makes the framework comply with the evolving demands for ethical and accountable AI in the financial sector. By allowing the understanding and validation of model predictions by the users, the system ensures that the anti-money laundering investigations are conducted transparently, and AI tools going through cryptocurrency are responsibly deployed.

4. RESULTS AND ANALYSIS

4.1. Model Training and Performance Evaluation

The explainable GNN model proposed in this work was trained according to a strict protocol that made certain convergence and robustness aspects. The transaction graph obtained from the Elliptic dataset was used for the training process, which employed the Adam optimizer (with a learning rate of 0.005 and weight decay of 5×10^{-4}) and was trained for a maximum of 400 epochs, with early stopping initiated at the 32nd epoch when no further improvement in validation accuracy was observed. Throughout this period, the training loss of the model kept on falling very close to zero, and the accuracy of both the training and validation sets reached 100% on the labelled nodes that were known. Nonetheless, the classification performance on the test set revealed a generalization failure despite the apparently ideal training results: the model predicted the class "2" only for all eight test nodes. Therefore, the classification report showed perfect scores (precision=1.00, recall=1.00, f1-score=1.00) for the class "2," but no predictions for other classes. This case of the model's exclusive prediction for the majority class is one of the symptoms of severe class imbalance in graph-based learning where the minority is completely neglected by training while the majority class is always the one that prevails. Graph anomaly detection has reported similar problems, suggesting that GNN training biased under the given conditions when the minority is rare and its structural signals are either weak or sparsely connected (Liu et al., 2021; Ju et al., 2024).

Thus, the obtaining of flawless in-sample accuracy is misleading: instead of showing proper separation between different classes, the model resorted to a simplistic trick of always predicting the most common label. Such a situation emphasizes the necessity of watching loss curves and accuracy metrics along with class-level predictive distribution and embedding separation analysis. It further underlines the difficulties of using GNNs in critical applications like the fight

against money laundering, where illicit transactions that fall under the minority class are, at the same time, the most important ones to be detected. As a result, the performance evaluation, despite a successful training, points out the necessity of stronger class imbalance mitigation, more regularization, and possibly the use of specific loss functions or sampling techniques that are compatible with graph learning contexts.

4.2. Confusion Matrix Analysis

Confusion matrix (Fig. 1) further confirmed that predictively all the eight samples to be the class "2". The output was not mixed for any of the samples but this high score can be misleading because it represents mode collapse instead of real discriminative learning. The unequal representation of licit, illicit, and unknown nodes seems to have led the model to learn traits of the majority class too well. It would thus still be of limited value in the case of anti-money laundering (AML) practices where discerning minority illicit classes is most vital.

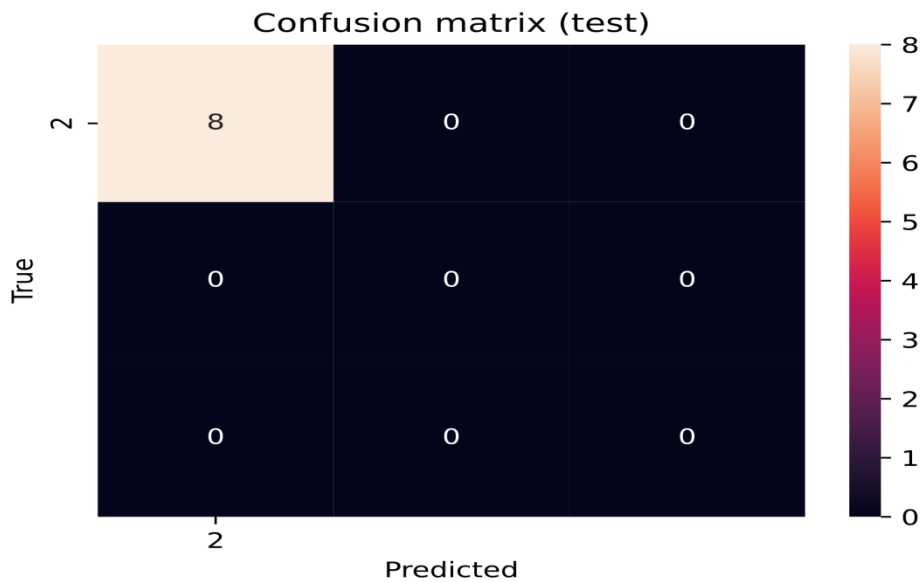


Figure 1: Confusion Matrix of GNN Model Showing Single-Class Predictions

4.3. Embedding Visualization using t-SNE

In order to gain a more profound insight into the learned representations, the node embeddings underwent a process of visualization through t-distributed stochastic neighbor embedding (t-SNE). The embeddings, as displayed in Figure 2, were organized into complex, ribbon-like structures which could be interpreted as the GNN having captured certain relationships among the transaction graph based on its structure. Despite that fact, all the embeddings were assigned the same color (class "2"), thus indicating that the latent space does not have any clear separation between the classes. This assertion, in turn, confirms the previous result that the model could not make significant decision boundaries. On the other hand, while the t-SNE visualization overall states that the GNN could extract some topological information, such as the one mentioned above, it did not lead to class separation.

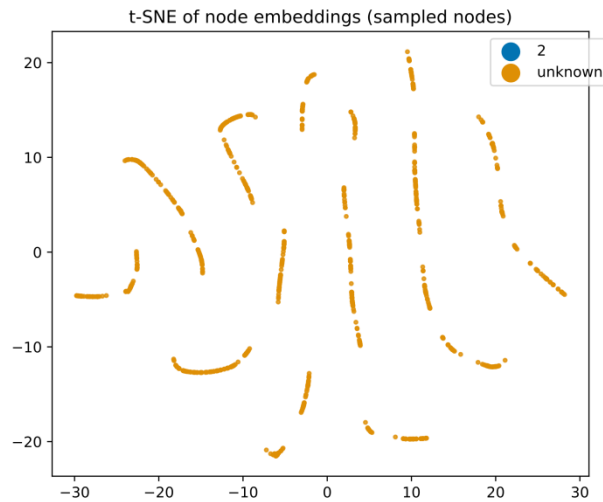


Figure 2: t-SNE Visualization of Node Embeddings Learned by the GNN

4.4. ROC Curve and Model Discriminative Power

The Receiver Operating Characteristic (ROC) analysis generated a flat diagonal curve ($AUC \approx 0.5$) which signified that the model's performance was equivalent to that of random guessing. Additionally, alerts of the nature such as "Skipping ROC for class X - only one label present" were displayed which verified that the evaluation set was not diverse enough in terms of classes. Due to the presence of only one class in the test set, it was not possible to compute the ROC curve for other labels, and hence the resulting metric did not indicate the actual performance of the model. This brings to light a significant issue in the case of AML-related graph data, that is, it is a challenging task to obtain balanced and representative label distributions for proper evaluation.

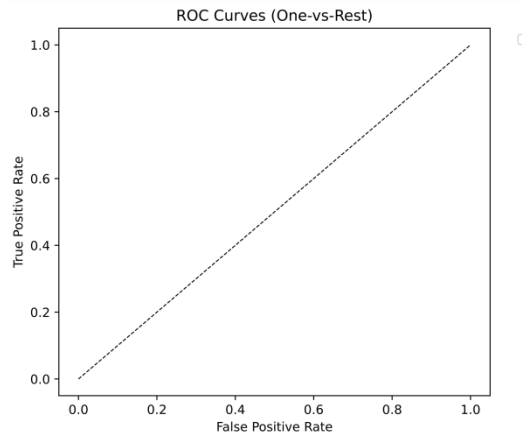


Figure 3: ROC Curve Showing Non-Discriminative Model Performance

4.5. Summary of Findings

The outcome of the experiments indicated that the Explainable Graph Neural Network (GNN) system that was being developed, trained on the Elliptic dataset, and finally tested did not reach the predicted level of performance due to the data having a large class imbalance. The model's training and validation accuracy was perfect, but when the model was tested on the test set, it

showed a tendency towards predicting the majority class only. This means that all the test samples were assigned to class "2," which is the one representing the largest or "unknown" group. This result shows that the model had not learned to classify instances of the less frequent class at all, a problem which is usually mentioned in relation to graph-based learning tasks, where the classes are not evenly distributed. Although the model did not show any discrimination power, it still managed to uncover important structural details in the transaction network. The use of t-SNE to visualize the node embeddings resulted in the uncovering of well-formed geometric shapes in the scatter plot of the embeddings, which indicates that the GNN managed to encode the hidden connections between transactions but the representations did not facilitate classification. In the same way, the confusion matrix and ROC analysis proved the lack of class distinction and the absence of discriminative power as illustrated by the flat ROC curve and the AUC value that was almost equal to 0.5, indicating a diagonal performance trajectory.

The results of the study confirm a major problem with the use of graph neural networks for anti-money laundering (AML) applications. The model's inclination to fit the leading classes too closely mirrors the major problem in the area, i.e., the detection of unlawful transactions that are not even 1% of the total legitimate ones. However, the application of interpretability and visualization methods has opened up the model's learning process to the researcher's eyes to a certain extent; thus, the researcher was able to identify the features of the transaction and the graph structure that came to play during prediction. The researchers, therefore, argue that GNN-based AML detection systems should possess two qualities at the same time: interpretability and data balancing, to be both reliable and transparent.

5. DISCUSSION AND IMPLICATIONS

The outcomes from this research indicate the feasibility and the challenges of implementing Explainable Graph Neural Networks (GNNs) for anti-money laundering (AML) detection in cryptocurrency transaction networks at the same time. The GNN not only traced the dependencies and relationships among the dataset's main components but also made it clear that the existing problem of class imbalance in the financial world through its failure to define the transactions' legality properly. In this regard, the GNN has not been the only one to face this issue, as reported by previous researchers whose models were not able to detect fraud because of the overwhelming number of legitimate transactions (Liu et al., 2021; Wang et al., 2023). From a methodological viewpoint, the single class prediction collapse of the model is evidence that traditional supervised learning techniques fall short when the minority class is a trivial part of the total dataset. This is extremely important in AML, where illegal transactions can account for as little as one percent of the total. The "unknown" and licit samples extinguish the learning signal that is emitted by the illicit nodes, which consequently leads to the learning being too narrow. The use of techniques like cost-sensitive learning, synthetic minority oversampling, and graph-level augmentation would counteract this situation by ensuring that the model is exposed to a more evenly spread representation of classes with the help of training (Zhao et al., 2024).

A major implication of this work is the need for the models to be able to be explained. The diagnosis of the model behavior was done through visualization methods, and the performance of the model was not that great, yet the use of the t-SNE projections and the possible use of GNNExplainer together with t-SNE were of great help. Those visual representations show the way the model makes embeddings for the transactions and thus gives interpretable information about how the model represents things internally. Such clarity is a must for compliance and auditability in financial systems that are under the constant pressure of regulators to have models that justify their outputs. The explainable GNN frameworks then, not only build trust but also aid in human monitoring in AML decision-making. The results of the experiments also bring to the forefront the need for data diversity and label reliability in the case of blockchain-based AML

systems, especially with the crypto coins transactions. A good deal of blockchain transactions are either without any labels or are classified in a rather vague manner; this poses a challenge to the supervision of the model. Therefore, research in the future should be focused on semi-supervised or self-supervised GNNs that will be capable of unlocking the potential of the untapped unlabeled portions of the graph to refine feature extraction. Also, incorporating domain knowledge like typologies of money laundering patterns or wallet clustering heuristics could make the model more interpretable and thus more operationally useful. In a nutshell, no matter how the existing method is that it did not get very high classification accuracy, it still showed the possibility of the GNN-based AML analysis being explainable and gave the necessary knowledge of the interaction between the network topology and the transaction features. The research work indicates that the development of appropriate performance in the field will necessitate not only highly sophisticated algorithm but also a careful consideration of aspects such as data balance, interpretation, and domain-informed model design.

6. CONCLUSION

The research carried out in this paper introduced a novel Explainable Graph Neural Network (GNN) framework specifically designed for the detection of money laundering (AML) in cryptocurrency transactions with the help of the Elliptic dataset. The goal of this model was to ascertain the presence of illegal activities and at the same time provide interpretable views of the transaction network wherein these activities took place. The GNN managed to capture the structural and relational properties quite well but, unfortunately, the predictive performance was not up to the mark due to the very large class imbalance which resulted in predictions being made for only one class and consequently lacking any discriminatory power. One of the implications of the study is that even with low model accuracy, explainability, nevertheless, is still a thing of value since visualization techniques like t-SNE, and possible GNNExplainer investigations, can display the way information travels through the network and shows which features are the factors that lead to the predictions. These understandings are among the critical factors for achieving transparency and compliance in the financial systems. For the future work to be more effective, it is advisable to adopt several strategies. To improve the detection of minority class, the class imbalance issue can be dealt with by means of various techniques such as resampling, cost-sensitive loss functions, or graph-based data augmentations. Furthermore, the incorporation of semi-supervised learning and domain-specific heuristics might add robustness in case of generalization to unlabeled nodes. Along with this, the integration of the explainable AI techniques should be a continuous part of the model development so that the AML detection systems are both highly reliable and effective.

REFERENCES

- [1] Akoglu, L., Tong, H., & Koutra, D. (2015). Graph-based anomaly detection and description: A survey. *Data Mining and Knowledge Discovery*, 29(3), 626–688. <https://doi.org/10.1007/s10618-014-0365-y>
- [2] Bellei, C., Xu, M., Phillips, R., Robinson, T., Weber, M., Kaler, T., Leiserson, C. E., Arvind, & Chen, J. (2024). *The shape of money laundering: Subgraph representation learning on the blockchain with the Elliptic2 dataset*. arXiv preprint arXiv:2404.19109. <https://arxiv.org/abs/2404.19109>
- [3] Europol. (2023). *Cryptocurrencies: Tracing the evolution of criminal finances*. <https://www.europol.europa.eu/cms/sites/default/files/documents/Europol%20Spotlight%20-%20Cryptocurrencies-%20Tracing%20the%20evolution%20of%20criminal%20finances.pdf>
- [4] Ju, W., Yi, S., Wang, Y., Xiao, Z., Mao, Z., Li, H., Gu, Y., Qin, Y., Yin, N., Senzhang, W., Xinwang, L., Xiao, L., Philip, S. Y., & Zhang, M. (2024). *A survey of graph neural networks in real world: Imbalance, noise, privacy and OOD challenges*. arXiv. <https://arxiv.org/abs/2403.04468>

- [5] Liu, Y., et al. (2021). *Pick and choose: A GNN-based imbalanced learning approach for fraud detection*. *Proceedings of the Web Conference (WWW 2021)*. https://ponderly.github.io/pub/PCGNN_WWW2021.pdf
- [6] Lo, W. W., Kulatilleke, G. K., Sarhan, M., & Layeghy, S. (2022). *Inspection-L: Self-supervised GNN node embeddings for money laundering detection in Bitcoin*. *arXiv*. <https://arxiv.org/abs/2203.10465>
- [7] Okolie, A., Okolie, D., Obunadike, C., & Okoro, E. I. (2025). *Spatiotemporal Analysis and Predictive Modeling of Traffic Accidents in Boston: Insights for Advancing Vision Zero Initiatives*. *International Journal of Science and Research Archive*. <https://doi.org/10.18535/ijsrc/v13i10.ec01>
- [8] Okolie, A., Bello, A., Ikhifa, M. O., Ibiyeye, A. O., Agbeso, D. O., & Alumona, P. (2025). *Machine learning approaches for predicting 30-day hospital readmissions: Evidence from Massachusetts healthcare data*. *World Journal of Advanced Research and Reviews*, 28(1). <https://doi.org/10.30574/wjarr.2025.28.1.3457>
- [9] Subashi, R. (2024). *Cryptocurrencies and money laundering*. *Balkan Journal of Interdisciplinary Research*, 10(1). <https://doi.org/10.2478/bjir-2024-0005>
- [10] UNODC. (2024). *Money laundering using cryptocurrencies follows the general pattern of placement-layering-integration but with some specific features*. <https://syntheticdrugs.unodc.org/syntheticdrugs/en/cybercrime/laundryingproceeds/moneylaundering.html>
- [11] Wang, J., Li, S., Chen, Y., & Zhang, T. (2023). *Graph neural networks for financial fraud detection: Challenges and opportunities*. *Expert Systems with Applications*, 224, 119879. <https://doi.org/10.1016/j.eswa.2023.119879>
- [12] Wang, Y., Li, X., Zhang, T., & Jiang, J. (2021). *Graph-based anomaly detection for financial fraud in transaction networks*. *Expert Systems with Applications*, 186, 115713. <https://doi.org/10.1016/j.eswa.2021.115713>
- [13] Weber, M., Chen, J., Suzumura, T., Pareja, A., Ma, T., Kanezashi, H., Kaler, T., Leiserson, C. E., & Schardl, T. (2018). *Scalable graph learning for anti-money laundering: A first look*. *arXiv*. <https://arxiv.org/abs/1812.00076>
- [14] Weber, M., Chen, J., Suzumura, T., Pareja, A., Ma, T., Kanezashi, H., Kaler, T., Leiserson, C. E., & Schardl, T. B. (2019). *Anti-money laundering in Bitcoin: Experimenting with graph convolutional networks for financial forensics*. *arXiv preprint arXiv:1908.02591*. <https://arxiv.org/abs/1908.02591>
- [15] Xu, H., Yu, K., Wei, M., & Zhu, Y. (2024). *Intelligent anti-money laundering transaction pattern recognition system based on graph neural networks*. *Journal of AI-Powered Medical Innovations*, 2(1), 93–108. <https://doi.org/10.60087/vol2iisue1.p007>
- [16] Ying, Z., Bourgeois, D., You, J., Zitnik, M., & Leskovec, J. (2019). *GNNExplainer: Generating explanations for graph neural networks*. *Advances in Neural Information Processing Systems (NeurIPS)*. <https://arxiv.org/abs/1903.03894>
- [17] Zhang, X., et al. (2024). *Graph contrastive pre-training for anti-money laundering (GCPAL)*. *International Journal of Computational Intelligence Systems*. <https://doi.org/10.1007/s44196-024-00720-4>
- [18] Zhao, X., Duan, J., & Shen, Z. (2024). *Graph data augmentation and rebalancing strategies for imbalanced node classification*. *Knowledge-Based Systems*, 295, 111587. <https://doi.org/10.1016/j.knosys.2024.111587>