Instituto Tecnológico
y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Department of Mathematics and Physics

Master in Data Science

Enhancing Cryptocurrency Transparency: A Graph Neural Network Approach for Bitcoin Address Classification

---

THESIS to obtain the **DEGREE** of
MASTER IN DATA SCIENCE

A thesis presented by: **Lic. Benito Tonatiuh Rojas Mayorquín**

Thesis Advisor: **M.Sc. Juan Francisco Muñoz Elguezabal**

Tlaquepaque, Jalisco, May, 2024

Enhancing Cryptocurrency Transparency: A Graph Neural Network Approach for Bitcoin Address Classification

Lic. Benito Tonatiuh Rojas Mayorquín

Abstract

Cryptocurrencies, notably Bitcoin, have catalyzed a significant shift in digital financial systems. The inherent pseudonymity of blockchain complicates efforts towards transparency and security, presenting a crucial problem that this thesis aims to resolve by enhancing address classification. The relevance of this problem lies in the increasing necessity for compliance with global financial regulations and ensuring the integrity of transactions.

Addressing this challenge involves overcoming significant difficulties such as the complexity of analyzing vast amounts of transaction data, the need for accurate data preprocessing, and the application of advanced machine learning techniques on non-traditional data structures like graphs. This research utilizes Graph Attention Networks (GATs) to classify Bitcoin addresses, a method chosen for its robustness in handling relational data and its capacity to focus selectively on the most informative parts of the transaction graph.

The efficacy of this approach is demonstrated through controlled experiments, where the GATs achieved an accuracy of 92.87%, a precision of 89.35%, a recall of 92.87%, and an F1 score of 90.17%. These results significantly improve upon previous internal benchmarks and confirm the model's capability to enhance transparency in Bitcoin transactions.

Furthermore, this work contributes a novel open-source Extract, Transform, Load (ETL) process tailored for blockchain data, fostering improved analytical transparency, and aiding regulatory and forensic analysis. The findings propose practical applications in financial technology, moving beyond theoretical discourse into actionable insights.

Keywords: Cryptocurrency, Blockchain, Bitcoin, Graph Neural Network, Financial Forensics.

Contents

List of Figures

List of Tables

Dedication

To my parents and siblings, who are the bedrock of my character, and the enduring foundation upon which I stand to reach my dreams—thank you for your unwavering support and love.

To Ana, the best partner on this journey of life, you inspire me to walk towards a future that we shape together.

1 Introduction

In the evolving landscape of digital finance, cryptocurrencies represent a paradigm shift, reshaping traditional monetary exchange and financial privacy frameworks. Among these, Bitcoin, as the progenitor, symbolizes the dawn of decentralized finance with its blockchain technology. However, the transparency of Bitcoin's ledger—while facilitating transaction traceability—also obscures the identities of participants, posing a dual-edged sword. This thesis endeavors to penetrate this veil of anonymity by applying data science techniques, thereby unveiling transactional behaviors within the network.

This research focuses on the utilization of Graph Neural Networks (GNNs), which merge graph theory with machine learning to effectively interpret complex data structures. Specifically, Graph Attention Networks (GATs) are employed to classify Bitcoin addresses by analyzing the transaction history encapsulated within a graph structure, thereby allowing the identification of patterns and address classification.

To address the problem, this study limits the transaction graph to data solely from 2022, justified by a comparative analysis over a decade, which shows that recent data more effectively reflects the dynamic nature of the network and improves classification performance. This limitation points to a broader discussion on the adaptability of GNNs to evolving data landscapes and their capacity to capture nuanced transactional behaviors in Bitcoin's rapidly changing environment.

In terms of industry standards and metrics, the model's performance is quantified through accuracy, precision, recall, and F1 score—metrics widely recognized in data science for evaluating classification models. This thesis also introduces an innovative Extract, Transform, Load (ETL) process using Bitcoin Core software, significantly enhancing the practicality and accessibility of blockchain data analysis for cryptocurrency analytics.

Subsequent chapters of this thesis will elaborate on these points:

- **Background and Literature Review:** This section sets the stage by discussing the theoretical underpinnings and related works, situating this study within the current research landscape.
- **Data and Methods:** Details on data acquisition, preprocessing, and the specifics of the GNN models used will be discussed, providing transparency and replicability of the research methods.
- **Experimental Setup:** A comprehensive description of the experimental design, including how data was curated and models were trained, to systematically verify the solution's effectiveness.

The final discussions and conclusions draw together empirical results and theoretical insights, framing them within the broader discourse on enhancing cryptocurrency transparency and regulatory frameworks. This organization not only chronicles the progression of the research but also outlines potential future avenues for further investigation.

By delineating these elements, this thesis serves as a rigorous inquiry into Bitcoin transaction analysis, aiming to enhance understanding and foster transparency in the burgeoning field of digital finance.

2 Problem Statement

The rapid advancement of digital currencies, with Bitcoin at the forefront, presents new challenges and opportunities in financial transparency and security. While the blockchain technology underpinning Bitcoin ensures that all transaction details are recorded, the pseudonymity of blockchain addresses permits users to conceal their identities. This scenario presents a critical problem: enhancing the transparency of Bitcoin transactions without undermining the anonymity that is fundamental to its user base.

Within the structured framework of a graph, we conceptualize the Bitcoin transaction network as follows:

$$G = (V, E) \tag{2.1}$$

where V represents the set of nodes, each corresponding to a unique Bitcoin address, and E denotes the set of edges, encapsulating transactions between these addresses. The main challenge lies in classifying each node $v \in V$ into predefined categories that reflect their transactional behavior, based on the complex and interconnected nature of the transaction network.

Mathematically, the problem is defined by the goal of learning a function $f : V \to C$, where C represents potential classes of transaction behaviors. This function is modeled using a Graph Attention Network (GAT), which employs parameters θ to dynamically weigh the influence of neighboring nodes, enhancing the precision of classification.

Key assumptions include:

- The graph G is a comprehensive yet static representation of the Bitcoin network for a specific timeframe, incorporating temporal elements through transaction timestamps.
- Feature vectors x_v are sufficiently informative of the nodes' transactional behaviors and are presumed accurate.
- Class labels within V_{train} and V_{test} are reliable and reflective of true behaviors, ensuring the model's applicability and generalizability.

This chapter outlines the foundational problem this thesis aims to tackle, setting the stage for a detailed exploration of the methodologies and technologies applied in subsequent chapters.

3 Background and Related Work

Contents

3.1 Background

Bitcoin, introduced in 2008 by Satoshi Nakamoto [1], revolutionized financial markets by introducing the concept of a decentralized cryptocurrency. Unlike traditional currencies, Bitcoin operates without central authority or banks, managing transactions collectively by the network. Nakamoto's whitepaper didn't just propose a digital currency but a foundational technology that created a new paradigm for money and financial transactions through its decentralized ledger technology, known as the blockchain.

[1] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system

The blockchain is a continuously growing list of records, called blocks, which are linked and secured using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data. This design inherently makes Bitcoin resistant to modification of the data; it is an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way. The introduction of this technology has not only spurred countless innovations but also raised significant regulatory and economic discussions globally.

3.1.1 Analysis of Illicit Activities and Regulatory Approaches

Despite the potential for democratizing finance, Bitcoin's pseudo-anonymity has also made it a favored medium for illicit activities. Bitcoin's design, which champions privacy and anonymity, has been exploited for transactions that elude traditional law enforcement and regulatory oversight. High-profile cases like the Silk Road marketplace highlighted the use of Bitcoin in black markets, where it was used to obfuscate the transfer of illicit goods [2]. The pseudo-anonymity of Bitcoin transactions complicates efforts to track and halt illegal activities, presenting both a technological challenge and a regulatory nightmare.

[2] Fleder, M., Kester, M. S., and Pillai, S. (2015). Bitcoin transaction graph analysis

The difficulty in tracking transactions has led to innovative approaches to "de-anonymize" the blockchain. One such effort is detailed by Yin et al., who utilized supervised machine learning to analyze the Bitcoin blockchain and identify patterns indicative of illegal use [3]. Their approach underscores the dual-use nature of machine learning in enhancing security and providing forensic tools for regulatory purposes, thereby contributing to the ongoing discourse on cryptocurrency regulation.

[3] Yin, H., Langenheldt, K., Harlev, M., Mukkamala, R. R., and Vatrapu, R. (2019). Regulating cryptocurrencies: A supervised machine learning approach to de-anonymizing the bitcoin blockchain. *Journal of Management Information Systems*, 36:37–73

3.1.2 *Emerging Forensic Techniques and Regulatory Efforts*

Further complicating the regulatory landscape are methods like "mixing" or "tumbling," which obscure the origins of Bitcoin to launder money from criminal activities. These methodologies pose significant challenges for financial regulators and law enforcement agencies in tracing illicit funds. The advent of graph theory applications in blockchain analysis, as demonstrated by Fleder et al. [4], provides a powerful tool for tracing transactions and understanding the complex web of transfers, enhancing the ability to track, monitor, and regulate the flow of Bitcoin across the network.

[4] Fleder, M., Kester, M. S., and Pillai, S. (2015). Bitcoin transaction graph analysis

Moreover, the work of Weber et al. highlights the application of Graph Convolutional Networks for financial forensics, using the Elliptic Data Set to distinguish between licit and illicit transactions [5]. This study exemplifies the practical applications of advanced machine learning techniques in regulatory frameworks, offering robust tools that can significantly improve the transparency and integrity of blockchain transactions.

[5] Weber, M., Domeniconi, G., Chen, J., Weidele, D. K. I., Bellei, C., Robinson, T., and Leiserson, C. E. (2019). Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics

3.2 *Related Work*

The application of Graph Neural Networks (GNNs) to the analysis of Bitcoin transactions has been a transformative development in cryptocurrency research. This section contrasts key contributions from leading studies with the methodology and findings of this thesis.

3.2.1 *Graph Neural Networks in Address Classification*

Huang et al. [6]: Huang and colleagues introduced the BAClassifier, a framework utilizing GNNs to categorize Bitcoin addresses into behavioral patterns such as exchanges, mining, and gambling. Their research is particularly noteworthy not only for its application of GNNs but also for its open and extensive dataset of over 2 million real-world Bitcoin addresses. This thesis utilizes the same dataset, enhancing the continuity and comparability of research findings in this domain. However, unlike Huang et al., who leverage a combination of GNNs with Long Short-Term Memory (LSTM) networks and Multilayer Perceptrons (MLP) for feature extraction and classification, this work employs a more streamlined model focusing solely on GATs to harness the dynamic and interconnected nature of transactional data, thus simplifying the computational process.

[6] Huang, Z., Huang, Y., Qian, P., Chen, J., and He, Q. (2022). Demystifying bitcoin address behavior via graph neural networks

Contrast with This Work: While Huang et al.'s work is foundational, the methodology in this thesis extends their dataset's utility by integrating it into a novel ETL process developed using Bitcoin Core software. This approach not only ensures a more transparent and reproducible data extraction process but also enhances the accessibility and integrity of the data used for

analysis. The ability to extract and process raw transaction data directly from a synchronized node presents a significant methodological improvement that democratizes data access and mitigates reliance on third-party data providers.

Also, as mentioned before, this work employs a more streamlined model focusing solely on GATs to harness the dynamic and interconnected nature of transactional data, thus simplifying the computational process.

3.2.2 *Benchmarking and Model Adaptation*

Pocher et al. [7]: Pocher et al.'s study explores the effectiveness of GCNs and GATs in classifying Bitcoin transactions for anti-money laundering and counter-financing of terrorism (AML/CFT). Their use of the Elliptic Data Set to classify transactions based on their licitness provides a valuable parallel to this thesis. Both studies underscore the enhanced capability of GATs over traditional machine learning methods in identifying complex relational patterns within transaction data.

[7] Pocher, N., Zichichi, M., Merizzi, F., Shafiq, M. Z., and Ferretti, S. (2023). Detecting anomalous cryptocurrency transactions: An aml/cft application of machine learning-based forensics

Contrast with This Work: This thesis differentiates itself by conducting a comparative year-over-year analysis, revealing that data from 2022 offers a more centralized and interconnected transaction network, which significantly boosts the model's performance. The inclusion of a benchmark analysis elucidates the reasons behind the varying effectiveness of GATs across different temporal datasets, a perspective not covered by Pocher et al. This insight is crucial for understanding the impact of network evolution on model efficacy and can guide future applications of GNNs in financial forensics.

3.2.3 *Technological and Methodological Innovations*

Veličković et al. [8]: The introduction of Graph Attention Networks by Veličković et al. provides the technical foundation for both the aforementioned studies and this thesis. Their development of GATs introduced a novel approach to node interaction within graphs, enabling dynamic weighting of node importance, which enhances the model's sensitivity to the subtleties of transactional relationships.

[8] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks

Innovation in This Work: Building on Veličković et al.'s work, this thesis applies GATs specifically tailored to the nuanced dynamics of the Bitcoin transaction network. By leveraging a simpler yet highly effective model configuration, this work achieves performance metrics that are comparable the works referenced in this section, especially when analyzing the more recent and structurally distinct 2022 data. The adaptability and performance of GATs, as demonstrated in this research, provide compelling evidence of their suitability for real-time monitoring and classification of Bitcoin transactions, enhancing both theoretical understanding and practical applications in cryptocurrency analytics.

4 Data and Methods

Contents

Our methodology is rooted in graph theory and machine learning, specifically, the application of Graph Neural Networks (GNNs) to transactional data represented as a graph.

4.1 Data Acquisition and Pre-Processing

4.1.1 ETL process

The Bitcoin Blockchain data is not readily available nor in an optimal format for analysis. There are two main problems to overcome: first, one needs to use client software to connect to the network directly and this program may have limited functionality regarding access to historical blockchain data; second, even with access to the data, this will be in a nested format from which a dataset should be built. Today, many services provide the data in any desired format, but apart from having a cost, which means is not possible to have truly open access to the full data for analysis without sufficient resources, this also means that there's no sure way to know what transformations had to be done the raw blockchain data to replicating them consistently.

To overcome the previous problems, we set up to build an end-to-end ETL process that will provide us with free and full access to the desired Bitcoin Blockchain data and set it up in the correct for a Data Science analysis. This process is one of the main contributions of this work.

The process built leverages the Bitcoin Core v25.0.0 client, one of the most popular clients, to synchronize a full node of the Bitcoin network. Then, through the client's RPC API, we extracted all the relevant information required to generate a dataset representation of the Bitcoin Blockchain; the complete ETL process is thoroughly documented in the following GitHub Repository.

The final dataset built from the genesis of the blockchain up to the cut-off date for this analysis (Decembrer-2022) has the following structure:

FILE(S)	COLUMN	DATA TYPE	DEFINITION
_blocks.parquet	block_hash	object	unique block identifier
768,333 rows	height	int32	ordered block index
1 file (44.28 MB)	time	int32	timestamp of block mining in UNIX
	tx_count	int32	transaction count per block
_transactions_parquets	txid	object	unique transaction identifier
789'789,238 rows	block_hash	object	unique block identifier
60 files (48.64 GB)	is_coinbase	bool	true if referring a coinbase transaction
_vin_parquets	txid	object	unique transaction identifier
2,082'564,677 rows	vin_txid	object	txid of input transactions
140 files (175.99 GB)	vout	int32	reference to output transaction
_vout_parquets	txid	object	unique transaction identifier
2,161'264,786 rows	value	float64	transaction value
134 files (134.14 GB)	n	int32	reference to output transaction
	addresses	object	list of receiving addresses

TABLE 4.1: Source dataset descriptions.

What is important to know about this dataset is the following:

- **Blocks**: in short, the Blockchain, as its name implies, will contain a series of cryptographically connected Blocks that in turn contain a verified list of transactions. These blocks will have two parts: the header, containing general information about the contents of the block, and the body, containing the list of transactions. This 'Blocks' table is intended to provide what the block header would, general information about the blocks within the blockchain.
- **Transactions**: this file contains the aforementioned body of the blocks, a list of the verified transactions existing in the blockchain.
- **Inputs**: nested into each transaction a list of inputs will provide us with the reference of the precedence of the bitcoins, which is the most relevant characteristic of having an open ledger such as the Bitcoin Blockchain. This table will only contain non-coinbase transactions, as coinbase blocks (newly mined blocks) will have no inputs.
- **Outputs**: the final destination of the bitcoins to be transferred. Pay-to-public-key transactions (P2PK) worked from the genesis of the blockchain (January 3, 2009) up to January 16 of the same year, and these transactions do not show the destination address on its outputs but its public key instead, which makes the blockchain less secure and harder to parse. This is why this table only shows Pay-to-public-key-hash transactions (P2PKH) and posterior protocols, which show the destination address as such.

Is important to note that going forward the input information does not contain the origin address of the bitcoins, which is a fundamental piece for the graph that we will build, so this

needs to be discovered by generating a data cross with the outputs table, as every input is the output of a previous transaction. This process will provide us with the correct mapping of the 'origin address' and 'destination address'. With this completed mapping we use the labeled dataset to safely add the labels to both the origin and destination addresses of each transaction and filter only the transactions of the 2 million labeled addresses.

The result of this mapping and filtering is a base file that will serve as the source of the graph construction:

COLUMN	DATA TYPE	DEFINITION
txid	object	unique transaction identifier
from_address	object	input address calculated from previous output
label_x	object	*from_address* label from labeled dataset
to_address	object	output address taken from current txid
label_y	object	*to_address* label from labeled dataset
value	float32	transaction value
date	datetime64[ns]	transaction date

TABLE 4.2: Base dataset for graph construction.

DATA DESCRIPTION	value	date (dd/mm/yy hh:mm:ss)
count	24 165 278.000 000 00	24,165,278
min	0.000 000 00	08/01/2011 20:35:49
25%	0.009 332 70	11/04/2013 04:23:50
50%	0.211 014 90	16/12/2015 12:59:20
75%	1.237 774 79	31/08/2018 17:29:08
max	87 318.710 937 50	21/12/2022 09:19:52
mean	8.076 344 91	NaN
std	104.655 464 17	NaN

TABLE 4.3: Statistical Summary of Dataset

4.1.2 *Graph Construction*

In the process of constructing the transaction graph for analysis, several pivotal decisions and preprocessing steps were taken to optimize both computational efficiency and the predictive relevance of the model. These steps can be formalized as follows:

- **Temporal Scope Restriction**: Given the computational challenges associated with processing the entire dataset and the observed predictive value of recent transactions, the graph $G = (V, E, W)$ is constructed using transactions from a specific, recent timeframe, specifically between 2021-12-21 and 2022-12-21. This approach ensures a focus on the most relevant and manageable subset of data.

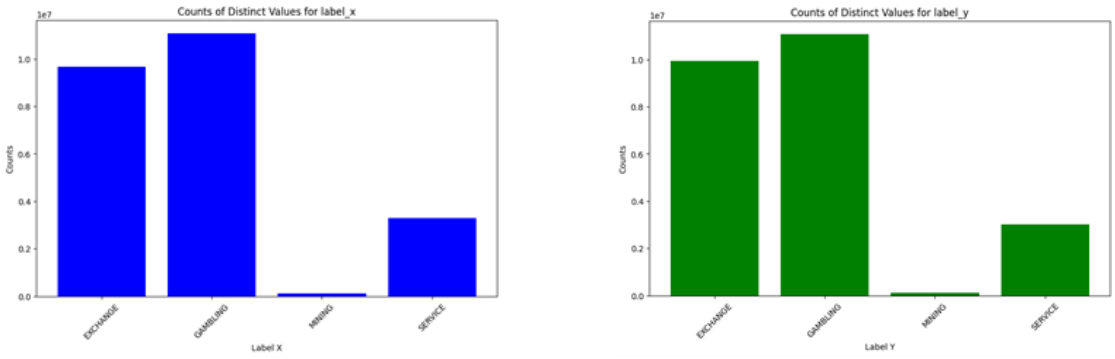

FIGURE 4.1: Counts of distinct label values

- **Label Encoding**: Let L be the set of all labels corresponding to Bitcoin address behaviors, and let $\phi : L \to \{0, 1, \ldots, |L| - 1\}$ be the label encoding function that maps each label to a unique integer.
- **Graph Definition**: The graph is defined as follows:
  - V is the set of nodes, each representing a unique Bitcoin address (7,990 nodes for the selected time-frame).
  - E is the set of edges, each corresponding to a transaction between addresses. An edge $e \in E$ is a tuple (u, v) where $u, v \in V$ and $u \neq v$ (86,789 nodes for the selected time-frame).
  - W is the set of edge weights, derived from a time-decay function to emphasize recent transactions, introducing a temporal dimension to the graph.
- **Time-Decay Weighting**: A key innovation in our approach is the introduction of a time-decay weight w_{uv} for each edge (u, v), calculated as $w_{uv} = e^{(-\delta \cdot t)}$, where $\delta = 0.01$ is the decay rate and t is the time elapsed since the transaction. This weighting scheme prioritizes recent transactions, reflecting their greater relevance to the address behaviors we seek to model.
- **Node Feature Preparation**: Each node $v \in V$ is associated with a feature vector $x_v \in \mathbb{R}^d$, summarizing transactional attributes, including the total value of all incoming and outgoing transactions for each address, providing a snapshot of the address's transactional behavior.
- **Node Label Assignment**: A subset of nodes $V' \subseteq V$ with known behaviors are labeled according to the encoding process, where for each $v \in V'$, a label y_v is assigned such that $y_v = \phi(l_v)$, with l_v being the label from the dataset.
- **Graph Data Representation**: The finalized graph data structure $G = (X, E, W, Y)$ incorporates:
  - $X \in \mathbb{R}^{|V| \times d}$ as the matrix of node features.
  - $E \subset V \times V$ as the set of transactional edges.
  - $W \in \mathbb{R}^{|E|}$ as the vector of time-decay weighted edge weights.
  - $Y \in \{0, 1, \ldots, |L| - 1\}^{|V'|}$ as the vector of encoded node labels.

This methodological approach to graph construction, particularly the strategic focus on a temporally bounded dataset and the incorporation of time-decay weighting, enables a nuanced representation that captures not only the static and topological features of the Bitcoin transaction network but also the dynamic aspects conferred by transaction temporality. Such a comprehensive representation facilitates the GNN model's ability to discern and classify based on a holistic view of transactional behaviors, enriched by the temporal dimension.

Moreover, the graph representation can be extended to tensor formalism to facilitate efficient computation in the GNN model. For instance, the node feature matrix $X \in \mathbb{R}^{|V| \times d}$ and the edge weight vector $W \in \mathbb{R}^{|E|}$ can be encapsulated within a higher-order tensor $\mathscr{T} \in \mathbb{R}^{|V| \times |V| \times d}$, where each element $\mathscr{T}i, j, :$ represents the feature vector of the node i if there is an edge from node i to node j, weighted by the edge weight wij. This tensorial representation is particularly beneficial when extending the model to include multiple types of relations or temporal dynamics where a third dimension could represent time steps or different relation types between nodes.

4.2 *Theoretical Framework*

The theoretical foundation of the proposed methodology is deeply rooted in the interdisciplinary confluence of graph theory, machine learning, and network analysis, each contributing indispensable perspectives and tools for the examination of complex systems such as the Bitcoin transaction network. Graph theory, with its rigorous mathematical framework, equips us to model the intricate web of transactions in Bitcoin. Through its constructs—nodes (representing addresses), edges (depicting transactions), and paths (illustrating the flow of currency)—we gain the capacity to abstract and analyze the transactional dynamics at play. Of particular relevance are centrality measures such as degree, closeness, and betweenness centrality, which serve to quantify the significance of nodes within the network. These metrics have found extensive application in scrutinizing financial networks, offering insights into patterns that may signify fraudulent activities.

Concurrently, the domain of machine learning—and neural networks, more specifically—has ushered in unprecedented capabilities for classification and prediction tasks across voluminous and complex datasets. Within this domain, Graph Neural Networks (GNNs) emerge as a specialized extension, adept at handling data structured in graph form. GNNs are distinguished by their ability to harness the relational information encoded in the connectivity patterns of nodes, thereby learning representations that encapsulate both the attributes of individual nodes and the overarching structure of the graph.

Although prior explorations in the literature have spotlighted Graph Convolutional Networks (GCNs) for their efficiency in neighborhood information aggregation and node embedding learning, our research pivots toward the utilization of Graph Attention Networks (GATs). This choice is motivated by GATs' innovative employment of attention mechanisms, which allow for the dynamic weighting of the significance of nodes' neighbors, thereby offering a more nuanced and adaptable approach to information processing within the graph.

4.2.1 Centrality Metrics

Centrality metrics are instrumental in network analysis, providing a quantitative basis to identify the most influential nodes within a network. These metrics elucidate the roles of various nodes in the network, highlighting their importance in terms of connectivity, influence, and control over the flow of transactions within the Bitcoin transaction network.

Three primary centrality metrics have been employed to analyze the Bitcoin transaction network: In-Degree Centrality, Out-Degree Centrality, and Betweenness Centrality. These metrics are the most relevant for a directed graph and our use case.

In-Degree Centrality quantifies the number of incoming edges to a node, indicating its popularity or receiver status within the network. Mathematically, it is defined for a node v as:

$$C_{\text{in}}(v) = \frac{|\{u \in V : (u, v) \in E\}|}{|V| - 1}, \tag{4.1}$$

where V is the set of nodes, and E is the set of edges in the graph.

Out-Degree Centrality measures the number of outgoing edges from a node, reflecting its influence or broadcaster role. It is given by:

$$C_{\text{out}}(v) = \frac{|\{w \in V : (v, w) \in E\}|}{|V| - 1}. \tag{4.2}$$

Betweenness Centrality captures the extent to which a node lies on the shortest paths between other nodes, serving as a bridge within the network. It is calculated as:

$$C_{\text{B}}(v) = \sum_{s,t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}, \tag{4.3}$$

where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths passing through node v.

4.2.2 GNN Model Architecture

The Graph Attention Network (GAT) employed in this investigation is meticulously designed to leverage the intrinsic graph structure of Bitcoin transaction data. The architecture reasoning aligns with the design goals of achieving efficient, flexible, and effective graph-based learning proposed by Velickovic et al. [1]. At the heart of GAT lies an attention mechanism that dynamically assigns significance to the features of neighboring nodes, a pivotal innovation that enhances the model's capacity to discern relevant transactional patterns. The mathematical formulation of the GAT is expressed as:

[1] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks

$$\alpha_{ij} = \text{softmax}_j \left(\text{LeakyReLU} \left(\mathbf{a}^T [Wh_i || Wh_j] \right) \right) \tag{4.4}$$

$$h_i' = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} W h_j \right) \tag{4.5}$$

where:

- h_i represents the feature vector of node i,

- α_{ij} denotes the attention coefficient between node i and node j,
- $\mathbf{a}$ is a learnable weight vector,
- W is a transformation weight matrix,
- σ represents the activation function, and
- $||$ signifies concatenation.

The softmax function is applied across all neighbors j of node i to ensure the comparability of the coefficients. The inclusion of the LeakyReLU nonlinearity, defined as

$$\text{LeakyReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{if } x \leq 0 \end{cases}, \tag{4.6}$$

where x is the input and α is a small constant, serves multiple purposes. Primarily, it prevents the occurrence of 'dead neurons' by allowing a small, non-zero gradient when the unit is inactive ($x \leq 0$), thus mitigating the dying ReLU problem. This characteristic is crucial for maintaining gradient flow during backpropagation, especially in the computation of attention scores e_{ij}, which are sensitive to the sign of their input. Furthermore, the LeakyReLU ensures that the attention mechanism remains responsive to both positive and negative inputs, enriching the model's ability to differentiate between various transactional relationships.

Softmax normalization, defined as

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})}, \tag{4.7}$$

applies to the attention coefficients, transforming them into a distribution over the neighbors of each node. This process not only makes the attention coefficients α_{ij} across different nodes' neighborhoods comparable but also accentuates the model's focus on the most relevant features by amplifying the differences among attention scores. Consequently, GATs can prioritize information from neighbors deemed most pertinent for the classification task at hand.

Through the iterative update of node representations, accentuated by the weighting of neighboring features as dictated by the attention coefficients, the GAT model demonstrates an unparalleled adeptness in learning which connections most significantly influence the feature representation of a node. This capability, rooted in the careful selection of LeakyReLU and softmax within the GAT framework, addresses the limitations inherent in prior graph neural network architectures and significantly augments the model's expressive power in analyzing graph-structured data.

4.2.3 *Model Training*

The training and evaluation methodology entails a structured sequence designed to optimize the model's performance on classifying Bitcoin transactions. The process unfolds as follows:

1. The graph data is partitioned into training, validation, and test subsets, adhering to a specified ratio to balance learning efficacy and evaluation integrity. This partitioning ensures diverse representation across the subsets for comprehensive learning and assessment.

2. Utilizing the GAT model, node features propagate through the network, leveraging the attention mechanism to dynamically weight the importance of neighboring nodes. This process is instrumental in refining the output predictions for each node, focusing on relevant transactional patterns.

3. The loss for nodes in the training subset is calculated using the negative log-likelihood loss function, formulated as:
$$L = -\sum_{i \in \text{train\_indices}} y_i \log(p_i), \tag{4.8}$$
where y_i denotes the true labels, and p_i represents the predicted probabilities for the nodes in the training set.

4. Backpropagation is employed to update the model parameters (θ), aiming to minimize the loss and enhance the model's predictive accuracy.

5. The model's efficacy is evaluated on the test subset by determining the accuracy, defined as the ratio of correctly predicted labels to the total number of test nodes.
$$\text{Accuracy} = \frac{\sum_{j \in \text{test\_indices}} 1(\hat{y}_j = y_j)}{|\text{test\_indices}|}, \tag{4.9}$$
where $\hat{y}_j$ is the predicted label, y_j is the true label, and 1 is the indicator function.

Additionally, the integration of the Optuna framework for hyperparameter optimization plays a pivotal role in fine-tuning the GAT model's configuration. This process involves defining a search space for key parameters, such as the number of hidden units, dropout rate, learning rate, and weight decay, and iteratively evaluating the model's performance across a range of trials to identify the optimal parameter set. The objective function, centered around maximizing the accuracy on a validation subset, guides the selection of hyperparameters that contribute to the model's generalizability and effectiveness.

It's worth noting that the Optuna study's specifics, including the selection of hyperparameters and the optimization strategy, are detailed within the experimental setup section. This approach underscores the synergy between the methodological rigor of the training and evaluation protocol and the adaptive nature of hyperparameter optimization, culminating in a robust framework for analyzing Bitcoin transactions through the lens of a GAT model.

4.2.4 *Evaluation Metrics*

The evaluation of the Graph Attention Network (GAT) model outlines three key aspects of model performance: the training loss and test accuracy, the precision, recall, and F1 score over epochs, and the average maximum probability of the model's predictions on the test set.

$$\text{Training Loss} = -\sum_{i \in \text{train}} y_i \cdot \log(p(y_i|x_i, \theta)), \tag{4.10}$$

$$\text{Test Accuracy} = \frac{\sum_{i \in \text{test}} 1(\hat{y}_i = y_i)}{|\text{test}|}. \tag{4.11}$$

Precision, recall, and the F1 score metrics highlight the model's balanced predictive performance:

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{4.12}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{4.13}$$

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{4.14}$$

The average maximum probability plot provides insight into the model's confidence in its predictions:

$$\text{Avg Max Probability} = \frac{1}{|\text{test}|} \sum_{i \in \text{test}} \max(p(y|x_i, \theta)), \tag{4.15}$$

where $\max(p(y|x_i, \theta))$ denotes the maximum predicted probability for the true label of each instance in the test set.

5 Experimental Setup

Contents

This chapter outlines the experimental setup used to evaluate the performance of a Graph Attention Network (GAT) model on the task of classifying Bitcoin addresses based on their transaction history. The setup encompasses the model architecture, hyperparameter optimization, data partitioning, training, and evaluation process, ensuring a comprehensive assessment of the model's predictive capabilities.

5.1 Open Source Repository

A significant challenge in cryptocurrency analytics has been the lack of free and accessible, open Bitcoin transaction data, particularly data that is transparently processed. Traditional datasets often lack clarity in their preprocessing steps, making reproducibility difficult. To address this gap, an extensive effort was made to develop a comprehensive Extract, Transform, Load (ETL) process leveraging Bitcoin Core v25 software [1].

[1] (2023). Bitcoin Core. https://bitcoin.org/en/releases/25.0/. [Computer software]

This ETL process, a pioneering endeavor in the field, utilizes the RPC API provided by Bitcoin Core to systematically extract transaction data from a synchronized node. The development of this process involved multiple methodologies to ensure robustness and reliability, ultimately deciding on the most efficient method to interface with Bitcoin's underlying data structures.

The resulting open-source repository can be accessed here: GitHub - Database From Bitcoin Core. This repository not only provides a blueprint for others seeking to analyze Bitcoin transaction data but also sets a foundation for ongoing improvements and updates, ensuring the dataset remains relevant for future research endeavors.

5.2 Computational Resources

The development and operation of the ETL process, along with the subsequent data analysis, demanded substantial computational resources. The primary system used in this research was equipped with an AMD Ryzen 7 5700X 8-core processor, operating at 3.40 GHz, and was enhanced with 128 GB of RAM to facilitate efficient handling of large datasets. This setup was crucial as it allowed for extensive data manipulation and analysis directly in memory, a necessity for processing the voluminous data generated by the Bitcoin blockchain.

Additionally, the setup included ample disk space to not only accommodate the large size of the blockchain data but also to maintain the obtained database and multiple backups of it (4TB of disk space would be advisable), ensuring data integrity throughout the research process. The use of Dask, a flexible parallel computing library for Python, proved instrumental. Dask's ability to emulate a distributed computing environment allowed for processing datasets larger than the machine's memory capacity, contrasting with other libraries like Pandas or Polars that require data to fit into memory. This capability was vital for managing the vast amounts of data involved in this study, demonstrating the utility of advanced computational tools in contemporary data science research.

5.3 Experiment Definition

5.3.1 Model Architecture

The model is defined as follows:

- Input feature size (*num_features*): Matches the dimensionality of the node attributes, incorporating transactional details of Bitcoin addresses.
- Hidden units (*num_hidden_units*): The number of neurons in the hidden layer, determined through hyperparameter optimization.
- Output layer size (*num_classes*): Corresponds to the number of categories defined by the label encoding, enabling multi-class classification of addresses.
- Dropout rate: Applied to prevent overfitting by randomly dropping units from the neural network during training.

The GAT model incorporates edge weights in the attention mechanism, allowing for the nuanced representation of transaction relationships.

5.3.2 Hyperparameter Optimization

To identify the optimal model configuration, we employ Optuna, an automated hyperparameter optimization framework. The study aims to maximize model accuracy on a validation set by tuning the following parameters:

- Learning rate (*lr*): Controls the step size at each iteration while moving toward a minimum of the loss function.

- Weight decay: Regularizes the model by penalizing large weights, preventing overfitting.
- Number of hidden units (*num_hidden_units*) and dropout rate: These parameters are crucial for model complexity and generalization ability.

Optuna's study involves executing a predefined number of trials, each with a unique combination of parameters, to find the set that yields the highest accuracy.

5.3.3 Data Partitioning

The dataset is split into training (70%), validation (10%), and test (20%) subsets, ensuring both effective learning and unbiased evaluation. We utilize a custom function to partition the graph data, adhering to specified ratios for each subset.

5.3.4 Training and Evaluation Process

Training is conducted over 500 epochs, with each epoch entailing a complete pass through the training data. We monitor the training loss and the model's performance on the validation set, focusing on precision, recall, and F1 score as key metrics. These metrics, calculated with a weighted average, address the class imbalance in the dataset.

Upon concluding the training, we evaluate the model on the test set to gauge its generalizability and overall predictive performance. The accuracy, precision, recall, and F1 score on the test data provide a comprehensive view of the model's effectiveness.

5.3.5 Implementation Details

The experimental procedures are implemented in Python 3.8, utilizing the PyTorch and PyTorch Geometric libraries for model construction and training. Optuna guides the hyperparameter optimization, with matplotlib employed for visualizing training progress and evaluation results.

This experimental setup delineates the systematic approach adopted for assessing the GAT model's capability to classify Bitcoin addresses based on transactional graphs. Through meticulous model configuration, training, and rigorous evaluation, this study aims to contribute valuable insights into the application of graph neural networks in analyzing cryptocurrency transactions.

6 Results and Discussion

Contents

6.1 Results

6.1.1 Centrality Metrics

The analysis of centrality metrics within the Bitcoin transaction graph has illuminated pivotal nodes in the network. The tables below enumerate the top three nodes according to each centrality metric. The focus on the top three nodes is due to their significantly higher centrality values, indicating a dominant role in the network's transaction dynamics.

Table 6.1: Top 3 Nodes by In-Degree Centrality

#	Address	In-Degree Centrality
1	12cgpFdJViXbwHbhrA3TuW1EGnL25Zqc3P	0.6515208411565903
2	17ac9tXHxu1nxdLgLu9WYk7vR8ggFN5GkH	0.07360120165227187
3	1FpTqAX7URD7akZcLvJQRRaXSm4NUbP7ng	0.044436099637000875

Table 6.2: Top 3 Nodes by Out-Degree Centrality

#	Address	Out-Degree Centrality
1	1HckjUpRGcrrRAtFaaCAUaGjsPx9oYmLaZ	0.07385154587557892
2	151zHjPneqsceawoDFf9sqDRBWU3pd4LgH	0.04656402553511078
3	1L1xSXttdsBAPVjVfyoyCg3RZbdHinT5G5	0.04218300162723745

TABLE 6.3: Top 3 Nodes by Betweenness Centrality

#	Address	Betweenness Centrality
1	12cgpFdJViXbwHbhrA3TuW1EGnL25Zqc3P	0.08708707071121141
2	151zHjPneqsceawoDFf9sqDRBWU3pd4LgH	0.08416928183613512
3	1HckjUpRGcrrRAtFaaCAUaGjsPx9oYmLaZ	0.008038056149048584

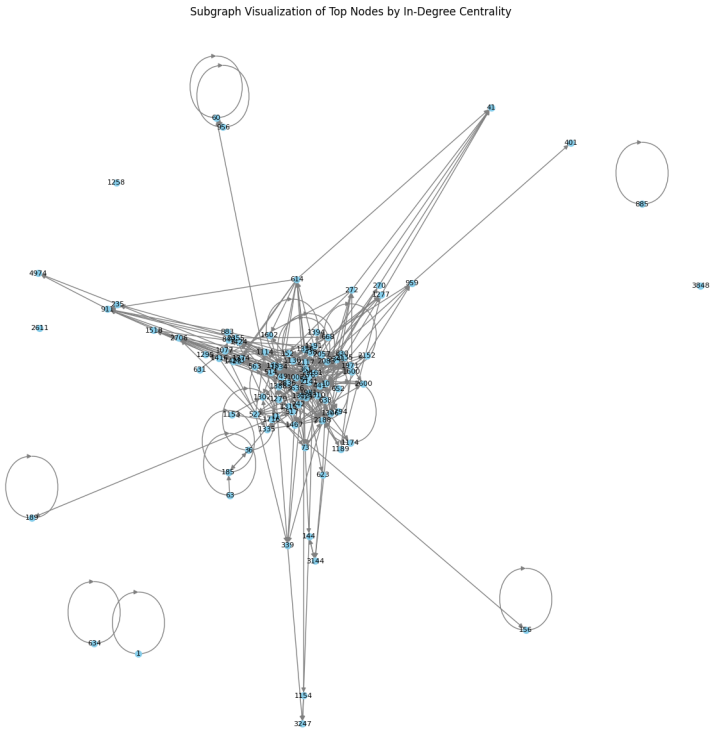

FIGURE 6.1: Graph visualization.

Additionally, these centrality metrics are utilized to refine the selection of nodes for graphical representation of the network. Specifically, Figure 6.1 incorporates the top 100 nodes based on In-Degree centrality to produce a coherent and manageable graph visualization. This approach avoids over-saturation of the visual representation, which is a standard practice in graph theory when dealing with large and interconnected networks.

6.1.2 *Model Performance*

The model's training was conducted using hyperparameters optimized via Optuna, resulting in the following configuration: 64 hidden units, a dropout rate of 0.1478, a learning rate of 0.0835, and a weight decay of 0.000156. The outcomes after training for 500 epochs are encapsulated in the composite image shown in Figure 6.2.

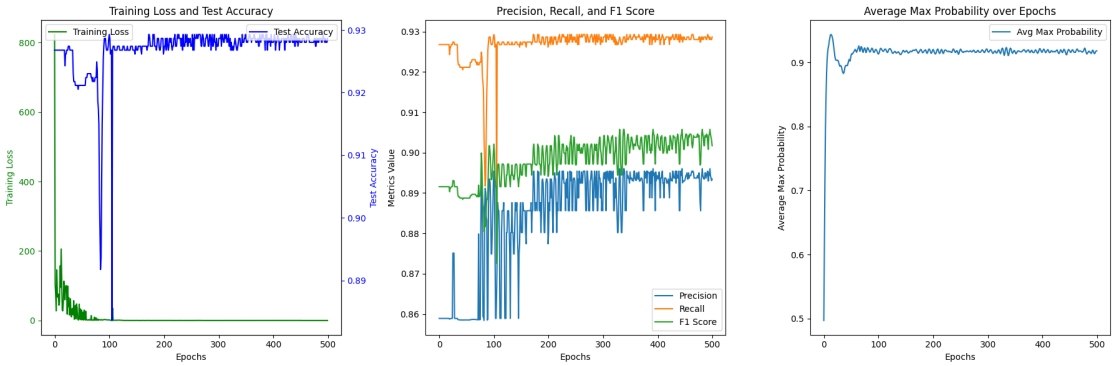

FIGURE 6.2: Training loss and test accuracy (left), precision, recall, and F1 score (center), and average maximum probability over epochs (right).

Training Metrics Overview:

- **Final Training Loss:** 0.3273
- **Test Accuracy:** 0.9287
- **Precision:** 0.8935
- **Recall:** 0.9287
- **F1 Score:** 0.9017
- **Average Maximum Probability:** 0.92

These metrics are visualized in Figure 6.2, which depicts the training loss and test accuracy on the left, precision, recall, and F1 score in the center, and the average maximum probability over epochs on the right. Each graph represents the progression of the respective metrics over the training period.

6.1.3 *Ten-year benchmark*

This section illustrates a decade-long analysis of the Bitcoin transaction network through various graphical representations. Each figure is detailed below to guide interpretation without drawing any conclusions.

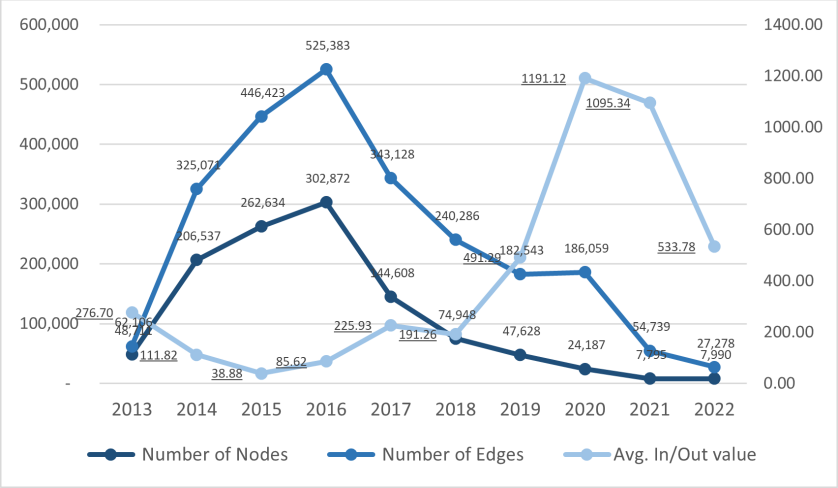

FIGURE 6.3: Evolution of the Bitcoin network size and average transaction values from 2013 to 2022.

Figure 6.3 displays two key metrics: the number of nodes and edges within the Bitcoin network over the years, and the average value of transactions per year. The lines indicate the growth or decline of these metrics, with scales provided on the left and right y-axes respectively.

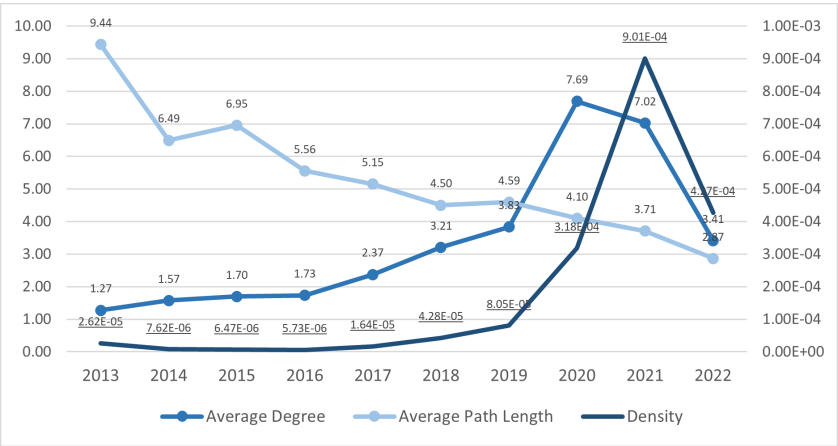

FIGURE 6.4: Network metrics highlighting changes in average degree, path length, and density from 2013 to 2022.

Figure 6.4 tracks three network characteristics: average degree, average path length, and density of the network each year. These lines represent the connectivity, efficiency, and compactness of the network respectively, with detailed yearly metrics shown on the graph.

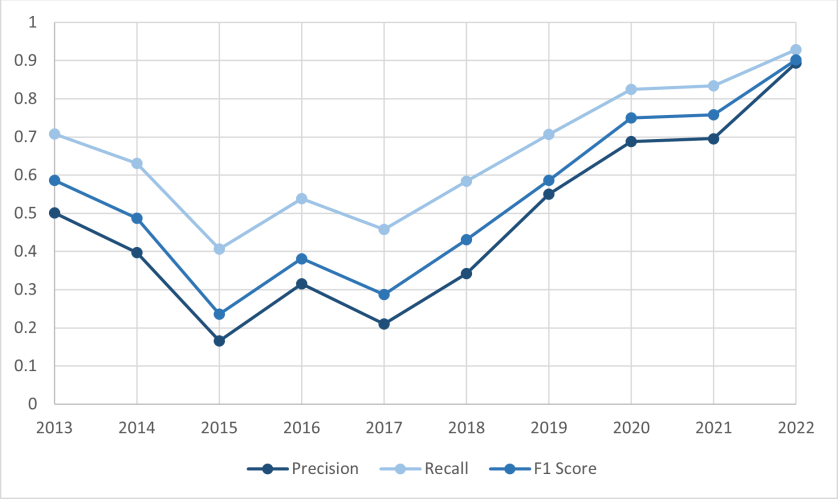

FIGURE 6.5: Performance metrics of the GAT model applied to yearly Bitcoin transaction networks.

Figure 6.5 presents the annual performance metrics—precision, recall, and F1 score—of the GAT model when applied to the Bitcoin transaction networks from 2013 to 2022. The graph plots each metric's progression over the years, providing a visual representation of the model's performance trends.

6.2 *Discussion*

6.2.1 *Centrality Metrics*

The centrality metrics within the Bitcoin transaction graph have highlighted the importance of several nodes that play pivotal roles in the network's dynamics. Notably, nodes identified as Node #1 and Node # 3 demonstrate significant influence across different centrality measures, indicating their critical roles within the transaction dynamics.

Node #1, ranking highest in both In-Degree and Betweenness Centrality, and identified as a major exchange, serves as a primary recipient of Bitcoin transfers. This suggests its crucial role in accumulating incoming transactions, likely reflecting its function as a deposit address within an exchange. The high In-Degree Centrality of Node #1 underlines its operational significance, attracting a substantial volume of transactions due to its foundational role within the exchange.

Node #3, prominent in Out-Degree Centrality, shows a high volume of outgoing transactions, indicative of its role in dispersing Bitcoin to various external addresses, which can be attributed to user withdrawals or transfers to other exchanges. This node's activity underscores its function in influencing liquidity and transaction flows within the Bitcoin ecosystem.

The marked centrality of these nodes, especially in facilitating significant volumes of Bitcoin transactions, underscores their structural importance but also points to potential vulnerabilities within the network. Such centralized nodes, while efficient, introduce risks related to security, transaction censorship, and market manipulation. Their centrality could pose systemic risks if

these nodes were to be compromised, suggesting a network that might be resilient in terms of transaction volume but fragile in terms of operational security.

This nuanced view of centrality within the Bitcoin transaction network reflects the evolving nature of its architecture and offers insights into both the operational dynamics and the associated strategic vulnerabilities. The continued monitoring of these metrics can provide valuable foresight into potential shifts in the network structure, guiding future enhancements in network design and governance.

6.2.2 *Model Performance*

The performance of our Graph Attention Network (GAT) model, as detailed in Table 6.4, offers insightful comparisons with both traditional ML models and advanced GNNs, reflecting its capabilities within graph-based representations of Bitcoin transactions.

TABLE 6.4: Performance comparison between different graph representation models.

Methods	Model	Precision	Recall	F1-score
GNNs	GFN (Huang et al.)	0.9815	0.9725	0.9769
	Diffpool	0.9218	0.9315	0.9299
	GCN	0.9534	0.9461	0.9514
	GAT (ours)	0.8935	0.9287	0.9017
MLs	LR	0.2208	0.3477	0.2684
	MLP	0.1011	0.2500	0.1440
	SVM	0.8787	0.5503	0.5574
	Bernoulli NB	0.5078	0.3434	0.3047
	Gaussian NB	0.5342	0.4418	0.3999
	KNN	0.8661	0.8553	0.8598
	Decision Tree	0.9298	0.9178	0.9236
	GBDT	0.9596	0.9575	0.9585
	XGBoost	0.9340	0.9321	0.9329

Our model's architecture and operational framework have been intentionally simplified and focused on a single year of data. This design choice contrasts with other GNNs that may utilize more complex or multi-year datasets. Despite its straightforward structure, our model achieves a high recall of 0.9287 and an F1-score of 0.9017, demonstrating effective transaction detection capabilities. This performance is particularly notable given the model's streamlined nature, which not only facilitates understanding and implementation but also enhances the replicability of our research.

The high recall indicates that our model effectively minimizes false negatives—critical for applications requiring stringent security measures such as financial transaction monitoring. Although the precision of 0.8935 does not surpass all other GNN models, it remains competitive, especially when considering the model's reduced complexity and narrower data scope.

Moreover, the consistent high average maximum probability across training epochs reinforces

the reliability of the model's outputs. This metric, crucial in a softmax output framework, confirms that the model's predictions are statistically robust, not merely the result of overfitting or underfitting.

One important thing to discuss here is the behavior observed in the training loss and test accuracy graphs reveals initial fluctuations, typical of the early stages of training where the model learns to navigate the complex landscape of high-dimensional data. However, as training progresses, these metrics stabilize, demonstrating the model's ability to adapt and learn from the data effectively. The convergence of these metrics towards the end of the training period, despite apparent early volatility, is indicative of the model reaching a stable solution.

Our approach prioritizes transparency and ease of replication, from the comprehensive ETL process to the clear delineation of the modeling steps. This not only supports the scientific validity of our findings but also provides a template for future research, encouraging other scholars and practitioners to replicate and extend our work. The model's adaptability, coupled with its robust performance, offers significant insights into the potential for using simplified GNN models in complex transaction networks.

While our GAT model may not redefine performance benchmarks when compared to the most advanced GNNs, it provides a robust framework for the accurate and reliable analysis of financial transactions within a constrained and comprehensible scope. This balance of simplicity, effectiveness, and transparency is particularly valuable in fields where data sensitivity and the need for reproducibility are paramount.

6.2.3 *Temporal Scope*

The decision to limit the graph construction to data from the year 2022 can be substantiated, and the results better understood, by an analytical study of the network's evolution over the past decade. The supporting evidence lies within the structural dynamics of the Bitcoin network, as revealed through a detailed analysis of network size and metrics.

A descending trend in the number of nodes and edges over the years was observed (figure 6.3), with the 2022 graph exhibiting a smaller yet more value-concentrated network. This trend indicates a move towards higher transaction values being processed through a reduced number of addresses, resulting in a denser and more interconnected network structure.

The 2022 network demonstrates an increase in average degree and network density (figure 6.4), indicating a more interconnected graph structure with nodes having more direct connections on average. This increased degree of interconnectivity within a smaller network aligns with the observed enhancement in model performance metrics, suggesting that the more recent and concentrated network structure provides a potent ground for the application of GAT.

The optimized model performance in the 2022 network is further corroborated by the re-training of the model using all previous yearly graphs and plotting precision, recall, and F1 score metrics. These metrics collectively suggest that the GAT model is particularly well-suited to a network that has evolved to be more transactionally value-dense and concentrated in influence.

The distilled analysis leads to a better explanation for the selection and results of the 2022 graph as the foundation for the GAT model application within this study. The pronounced

concentration of transactional value and connectivity in the latest graph provides an enriched dataset that likely contributes to the improved performance of the classification model. Future research may delve into the implications of these evolving network characteristics, particularly examining the impact of increasing concentration on the efficacy of transaction pattern analysis tools and their utility in monitoring and regulating digital currency flows.

7 Conclusions, Limitations and Future Work

Contents

7.1 Conclusions

This thesis has explored the application of Graph Attention Networks (GATs) in analyzing cryptocurrency transactions, with a focus on classifying Bitcoin addresses. The construction of a transaction network from a single year's data (2022) facilitated a detailed study of a network that was both concentrated and characterized by higher transaction values and increased average degrees. These factors notably enhanced the applicability and effectiveness of the GAT model within this specific context.

The GAT model demonstrated high performance across precision, recall, and F1 scores, affirming its robustness in identifying diverse behaviors associated with Bitcoin addresses. The significant centrality metrics of identified nodes validated their pivotal roles within the network, underscoring the model's practical relevance alongside its statistical soundness. These results not only showcase the strong predictive capabilities of the model but also its utility in detecting potentially illicit activities, thus offering valuable tools for regulatory oversight and financial security.

Theoretically, this work enriches the literature on the application of graph neural networks to financial data, illustrating the potential of GATs to uncover complex patterns inherent in transactional networks. Practically, the insights derived from this analysis provide actionable intelligence for tracking anomalous behaviors that could indicate fraudulent activities, benefiting both regulatory bodies and financial institutions.

Methodologically, the development of an open-source Extract, Transform, Load (ETL) process marks a significant stride towards enhancing accessibility to blockchain data. This initiative not only paves the way for greater transparency in cryptocurrency research but also promotes the democratization of data access, enabling a broader base of researchers to engage in this field without dependence on proprietary datasets.

Ultimately, this thesis bridges a crucial gap in the domain of cryptocurrency analytics

by leveraging advanced graph-based techniques to analyze the Bitcoin blockchain. The methodologies and findings discussed herein lay a robust foundation for future research and are poised to influence subsequent developments in digital finance analysis tools. As the cryptocurrency environment evolves, the approaches refined in this study will likely play a pivotal role in shaping the analytical strategies of the future, underscoring the enduring impact of this research.

7.2 *Limitations*

This study, while comprehensive in its approach to applying Graph Attention Networks to Bitcoin transaction data, encompasses several limitations that are intrinsic to the scope and design of the research.

Single-Year Data Focus: The decision to construct the transaction network using only one year's data (2022) was pivotal for managing the complexity and computational load of the analysis. However, this limitation may affect the generalizability of the findings across different temporal contexts. Bitcoin's transaction network can exhibit significant fluctuations in behavior and structure over time due to varying market conditions, regulatory changes, and technological advancements. Therefore, the conclusions drawn from this study might not fully encapsulate the dynamics that could emerge in different yearly datasets.

Model Simplicity: The GAT model's simplicity, while beneficial for ensuring clarity and manageability in analysis, potentially restricts the depth of insights that could be obtained from more complex models. By focusing on a streamlined model and feature set, certain nuanced interactions within the data might have been overlooked. This simplicity, though advantageous for interpretability and computational efficiency, might limit the detection of subtler patterns that could be critical for understanding more complex fraudulent behaviors or intricate transaction networks.

Scope of Cryptocurrency Coverage: The study is specifically tailored to the Bitcoin network and does not extend to other cryptocurrencies, which may have different transactional behaviors and network structures. This limitation restricts the applicability of the findings to other digital currencies, which might benefit from a similar analysis to uncover specific patterns relevant to their unique ecosystems.

Exclusion of Real-Time Data Analysis: The static nature of the dataset—reflecting transactions from a specific year without incorporating real-time data—limits the ability of the model to adapt to ongoing changes within the Bitcoin network. Real-time transaction analysis could provide more dynamic insights and enhance the model's applicability to current trends and emergent anomalies.

These limitations underscore the need for cautious interpretation of the study's outcomes and suggest areas for further research to enhance the robustness and applicability of the findings. Future studies could address these limitations by incorporating multi-year data analysis, exploring more complex models, and extending the research to include various cryptocurrencies and real-time data.

7.3 Future Work

The findings and limitations of this thesis open several promising avenues for future research in cryptocurrency analytics using Graph Neural Networks. The future work proposed here aims to extend the current research by addressing its limitations and exploring new opportunities that could further enhance the understanding and applicability of GNNs in this domain.

Multi-Year Data Analysis: To overcome the limitations posed by the use of a single-year dataset, future studies should consider the analysis of multi-year data. This approach would allow researchers to assess the temporal stability and robustness of the GAT model across different market conditions and regulatory environments. Such an analysis could provide deeper insights into the long-term patterns and shifts in the Bitcoin network, potentially leading to more generalized models that are effective across various temporal contexts.

Model Complexity and Feature Enrichment: Enhancing the model's complexity by incorporating more sophisticated GNN architectures or by integrating additional features could uncover more nuanced transactional behaviors within the network. Future research could explore the inclusion of node features that capture temporal dynamics or transaction contexts, which might improve the model's ability to detect complex fraudulent schemes or subtle anomalies.

Extension to Other Cryptocurrencies: Expanding the analysis to include other cryptocurrencies would provide a broader perspective on the applicability of GNNs across different blockchain technologies. Each cryptocurrency may exhibit unique transactional behaviors and network characteristics, and as such, tailored models could be developed to address specific challenges associated with each currency.

Real-Time Data Incorporation: Integrating real-time transaction data into the analysis could significantly enhance the model's relevance and applicability to current market conditions. Future work could focus on developing dynamic models that update their parameters in response to real-time data streams, thus providing timely insights that are critical for regulatory surveillance and fraud detection.

Improving Data Access and Transparency: Continuing to develop and refine open-source ETL processes that facilitate access to high-quality, comprehensive blockchain data would strengthen the research community's ability to conduct thorough and replicable studies. Further advancements in this area could include collaborations with blockchain networks to ensure data integrity and accessibility.

Interdisciplinary Approaches: Collaborating with experts in finance, cybersecurity, and regulatory fields could enhance the development of GNN models that are not only technically robust but also aligned with industry needs and compliance requirements. Such interdisciplinary research could lead to innovative solutions that bridge technical capabilities with practical financial technology applications.

By addressing these areas, future research can build on the foundational work presented in this thesis, driving forward the capabilities of machine learning in financial technology and contributing to the security and transparency of digital financial markets.

8 Appendix

Contents

This appendix provides references to the resources and tools utilized throughout the research presented in this thesis. These resources are essential for understanding the methodologies employed and for replicating the studies and experiments conducted.

8.1 Code Repository

The complete code for the Extract, Transform, Load (ETL) process as well as the modeling conducted in this study can be found in the following GitHub repository. This repository includes all scripts, data schema, and additional resources used to perform the analyses described in the thesis.

- **GitHub Repository:** https://github.com/benitotrm/database_from_Bitcoin_Core/tree/main

8.2 Bitcoin Core Software

Bitcoin Core is used for syncing the complete Bitcoin blockchain and accessing transaction data directly from the source. The software is central to the ETL process developed in this research, ensuring the accuracy and reliability of the data used.

- **Download Bitcoin Core:** https://bitcoin.org/en/download

8.3 Dask Library

The Dask library is utilized for handling large datasets that do not fit into memory, allowing for efficient parallel computations on single-machine setups. Dask is critical for processing the vast amounts of data involved in the Bitcoin blockchain efficiently.

- **Dask Documentation and Resources:** https://www.dask.org/

Each of these resources plays a vital role in the execution and success of the research, facilitating a deeper understanding and replication of the work. The code repository, in particular, is intended to be open and accessible to ensure that the scientific process is transparent and reproducible.

Bibliography

(2023). Bitcoin Core. https://bitcoin.org/en/releases/25.0/. [Computer software].

Fleder, M., Kester, M. S., and Pillai, S. (2015). Bitcoin transaction graph analysis.

Huang, Z., Huang, Y., Qian, P., Chen, J., and He, Q. (2022). Demystifying bitcoin address behavior via graph neural networks.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.

Pocher, N., Zichichi, M., Merizzi, F., Shafiq, M. Z., and Ferretti, S. (2023). Detecting anomalous cryptocurrency transactions: An aml/cft application of machine learning-based forensics.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks.

Weber, M., Domeniconi, G., Chen, J., Weidele, D. K. I., Bellei, C., Robinson, T., and Leiserson, C. E. (2019). Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics.

Yin, H., Langenheldt, K., Harlev, M., Mukkamala, R. R., and Vatrapu, R. (2019). Regulating cryptocurrencies: A supervised machine learning approach to de-anonymizing the bitcoin blockchain. *Journal of Management Information Systems*, 36:37–73.