National College of Ireland

Detecting Ransomware Payments in the Bitcoin Network: A Comprehensive Analysis and Classification Using Bitcoin Heist Ransomware Address Dataset

MSc Research Project
MSc in Data Analytics

Mani Maran Rajendran
Student ID: 23204711

School of Computing
National College of Ireland

Supervisor: Prof. Christian Horn

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Mani Maran Rajendran

Student ID: 23204711

Programme: MSc in Data Analytics **Year:** 2024-2025

Module: MSc Research Project

Supervisor: Christian Horn

Submission Due Date: 12-12-2024

Project Title: Detecting Ransomware Payments in the Bitcoin Network: A Comprehensive Analysis and Classification Using Bitcoin Heist Ransomware Address Dataset

Word Count: 8694 words **Page Count:** 23 pages

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Mani Maran Rajendran

Date: 12-12-2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Detecting Ransomware Payments in the Bitcoin Network: A Comprehensive Analysis and Classification Using Bitcoin Heist Ransomware Address Dataset

Mani Maran Rajendran
23204711

Abstract

Ransomware attacks pose a significant threat to global cybersecurity, causing substantial financial and operational losses. The anonymity inherent in Bitcoin transactions further exacerbates this issue, making it exceedingly difficult to trace and detect ransom payments associated with ransomware attacks. This study addresses the challenge by leveraging machine learning and deep learning models to detect ransomware transactions within the Bitcoin Heist Ransomware Address (BHRAD) dataset. The preprocessing pipeline included feature scaling, Synthetic Minority Oversampling Technique (SMOTE) to address class imbalances, and graph-based feature construction to capture relational data effectively. Five models—Random Forest, XGBoost, CNN, GCN, and GIN—were evaluated based on metrics such as accuracy, precision, recall, and F1 score. Random Forest achieved the highest accuracy (94.64%), demonstrating its effectiveness in handling structured data for ransomware detection. XGBoost also performed well but slightly lagged behind in recall. For graph-based data, GCN and GIN exhibited lower performance overall, with GCN achieving an F1 score of 78.09% and GIN struggling with an accuracy of 33.85% and an F1 score of 49.33%. The CNN model, designed for recognising patterns, showed moderate performance with an accuracy of 79.17% and an F1 score of 88.32%. These findings underscore the feasibility of combining ensemble methods with graph-based approaches for detecting ransomware transactions, offering valuable insights for enhancing cybersecurity frameworks and promoting transparency in blockchain transactions.

1 Introduction

Ransomware attacks have become one of the most frequent and profitable cyberattacks in the world in recent years, increasing its estimated damage to more than $ 30 billion per year by 2025 (Akcora et al., 2019; Anderson et al., 2019). These attacks use viruses to encrypt a victim's data, making it impossible to access it unless the attacker is paid a specified amount of money in cryptocurrencies such as bitcoins, which provide anonymity and decentralisation (Raheem et al., 2021; Berrueta et al., 2022). Ironically, the high rates of anonymity in blockchain technology make the identification and classification of ransomware payments equally a challenging affair; this is because auditing Bitcoin transactions does not easily link them to their source (Huang et al., 2018; Ampel et al., 2023).

This is compounded by the intelligence ransom actors employ, such as network layering and wallet mix services that hide transaction blocks (Solanki, 2019; Zakaria et al., 2022). In previous works, the authors investigated using various machine learning techniques to detect malicious transactions, achieving satisfactory performance. For example, Random forest and Xgboost models have established high accuracy when classifying transactions (Blanco & Tallón-Ballesteros, 2021; Alsaif, 2023). CNNs and GCNs are other sophisticated enhancements of these capabilities, analysing data patterns for revealing complex transaction flow structures (Manavi & Hamzeh, 2020; Sallout & Ashour, 2023).

Nevertheless, the issue remains unsolved since existing solutions do not adapt well to learning across multiple ransomware families and transaction types (Turner et al., 2020). This research seeks to fill this gap by using a novel set of advanced machine learning and deep learning models on the Bitcoin Heist Ransomware Address Dataset (BHRAD), which is a unique ransomware transaction detection dataset. The work does not only intend to increase the efficacy of ransomware classification for enhanced detection; it also aims to make a breakthrough in cybersecurity by establishing a clearer understanding of the behavioural characteristics of ransomware transactions.

1.1 Research Motivation

The latest advances in ransomware attacks evidenced its growing frequency and added to the existing needs for proper detection means. Ransomware attacks netted cybercriminals $11.5 billion in 2021 and have increased by 25% in 2022; 70% of victims reportedly pay the ransom in bitcoin (Anderson et al., 2019; Huang et al., 2018). The anonymity of Bitcoin transactions also makes it possible for cybercriminals to ask for and get ransoms and other currencies, avoiding regular tracking tools (Raheem et al., 2021). A heuristic and rule-based approach of analyzing Bitcoin transaction is a very slow and hence cannot match the pace of the current rapidly changing attack pattern.

This study is informed by the belief that limitations of this nature can be handled through the use of machine learning and deep learning models. Thus, it becomes apparent that recent developments in AI are the optimal approach for identifying valuable patterns related to ransomware transactions because the variety of datasets is increasing (Blanco & Tallón-Ballesteros, 2021, p. 32). Random Forest and XGBoost work well both in terms of interpretability and accuracy when tested on the structured data, as for more complex relationships between the nodes of the Bitcoin network, the model with a CNN or GCN approach will be required (Manavi & Hamzeh, 2020, Sallout & Ashour, 2023).

In addition, the aim of the proposed research is to fill a significant gap in the available literature by presenting a comparison of the cited models using a real ransomware dataset. This is important because no single model can offer a strong solution for ransomware detection; multiple models need to be employed for improved accuracy and to learn from a set of different models that best suit the scenario (Turner et al. 2020, Nayyer et al 2023). The findings obtained in the framework of this research may significantly improve the efficiency of cybersecurity

frameworks, and help law enforcement agencies and financial organizations to penetrate ransomware groups' networks more effectively.

1.2 Research Question and Objectives

The primary research question guiding this study is:

What is the comparative effectiveness of Random Forest, XGBoost, Convolutional Neural Networks (CNN), and Graph-based models like Graph Convolutional Networks (GCN) and Graph Isomorphism Networks (GIN) in detecting and classifying ransomware payments in the Bitcoin network using the Bitcoin Heist Ransomware Address Dataset?

The objectives of this research are as follows:

1. **To Analyse the Bitcoin Heist Ransomware Address Dataset:** Conduct a comprehensive analysis of the dataset to understand its structure, identify key features, and explore trends in ransomware-related transactions.

2. **To Develop Machine and Deep Learning Models:** Implement Random Forest, XGBoost, CNNs, GCNs, and GINs to classify transactions as ransomware-related or benign.

3. **To Evaluate Model Performance:** Compare the performance of the proposed models using metrics such as accuracy, precision, recall, and F1-score to determine their effectiveness.

This research contributes to and assesses a comparative framework of Machine Learning and Deep Learning Models that can expose the effectiveness of models in ransomware detection. Finally, through exploiting the BHRAD dataset, the study identifies six features in ransomware by providing insights and findings regarding the characteristics and behaviours of ransomware attackers. The studies aim to enhance cybersecurity plans because the methods provided have predictive power and are computationally efficient in real-time identifying and preventing ransomware payments.

The rest of the thesis is structured as follows: Section 2 covers the Related Work with respect to the study, Section 3 presents the Methodology in detail, Section 4 details the Design Specification, and the Study Implementation is discussed in Section 5. The results obtained for the implementation are discussed in Section 6, and the thesis ends with Section 7, that discusses the Conclusion and presents avenues for Future Work.

2 Related Work

In addition to the general review of ransomware and the possible consequences sampled in the previous chapter, this chapter provides insight into the approaches and academic research related to ransomware detection and prevention within the Bitcoin ecosystem. It synthesises prominent papers to offer an integrated analysis of ransomware modi operandi, as well as

detection advancements and transaction categorisation approaches. The chapter is structured into three sections:

Understanding Ransomware Mechanisms and the Role of Cryptocurrencies: This section presents the effectiveness of Bitcoin in ransomware payments and its anonymity to enable laundering and define trends.

Advances in Ransomware Detection Techniques: The emphasis now is made on modern approaches to detection, such as machine learning, behavior analysis, and deep learning frameworks and their performance indicators.

Classification and Prevention of Ransomware Transactions: The last discusses identification and prevention of ransomware related transactions by means of clustering, supervised learning and cryptographic methods.

2.1 Understanding Ransomware Mechanisms and the Role of Cryptocurrencies

The element of decentralization and the fact that nobody knows the real identity of the owner of the Bitcoin makes it the favorite of ransomware. The fraudulent use of Bitcoin in ransomware attacks and employability of the cryptocurrency by the WannaCry perpetrators, CryptoDefense, and NotPetya have been explored by Turner et al. (2020). As the research of the authors pointed out, the actors engaged in ransomware-related transactions demonstrate the different patterns of cash-in and cash-out activities, with speedy accumulation and dispersion of cash highlights their activities from the legitimates ones. In their work, Turner et al. (2020) use Ransomware–Bitcoin Intelligence–Forensic Continuum to show that the work of analysing specimens of new Ransomware is to identify specimens that can disaggregate and disrupt illicit financial networks as well as present the key role of graph-based analysis in dealing with this kind of crime.

In a similar vein, Wang et al. (2021) further this viewpoint in a massive empirical analysis of 63 ransomware types from 2012 to 2021. Their study follows more than 41 000 cases of ransom payments that exceeded $176 million. When over 92% of Bitcoin addresses are clustered, the study reveals how ransomware actors employ exchanges and mixers to launder their proceeds. It is evident that the evaluation of ransom payments proves difficult, as well as identifying the essence of multi-industry cooperation in the fight against such incidents.

Raheem et al. (2021) follow our previous work to extend the understanding of how ransom payments are made by studying ten ransomware families and the Bitcoin Heist dataset. The study captures typical payment features like ransom size and ransom deadline and shows how the anonymity of the blockchain is leveraged by the attackers. These results can serve as a basis for constructing heuristic algorithms for linking Bitcoin addresses to illegal transactions, which will deepen the knowledge of ransomware environments.

2.2 Advances in Ransomware Detection Techniques

Modern development trends in relation to ransomware detection have been devoted to using machine learning and behavioural analysis since ransomware is not a static phenomenon. In

this paper, Goyal et al. (2020), the authors highlight the system-level behavioural analysis, and specifically entropy fluctuations and recurrent file operations as biomarkers of ransomware. As for detection adaptation, they use a combination of supervised machine learning with feature engineering leading to a dynamic detection solution for the new ransomware variants.

Kok et al. (2022) present the proactive method of the Pre-Encryption Detection Algorithm (PEDA) that identifies the ransomware during the pre-encryption phase. In this way, PEDA uses both the SHA-256 signature comparison and API behavioural analysis to reach the training 100% recall and 99.9% during the cross-validation while decreasing the likelihood of data encryption. This first contingency measure demonstrates the need for early ransomware identification.

To overcome the drawbacks of entropy-based techniques, Kim et al. (2022) propose byte frequency-based measures EntropySA and DistSA. These new features can successfully distinguish between normal and ransomware-encrypted files with a true positive rate of 99.46% and an average F1- score of 0.994. The method also reduces computational expenses that work best in processing large sets for ransomware identification in cryptocurrency networks.

The work of Zakaria et al. (2022) introduces RENTAKA, a machine learning framework for pre-encryption ransomware detection. There are five classifiers used in RENTAKA; amongst that the best performer is Support Vector Machines queries with 97.05% accuracy. This proactive framework profiles API calls during the early stages of ransomware lifecycle thus allowing for an interdiction that minimizes data loss during encryption.

It has also been found that ransomware can be detected using deep learning effectively. Similar to Jemal (2023) Bi-LSTM and CNN architectures are used to explore the shared network directories with an average accuracy of greater than 98% for distinguishing ransomware from benign samples and even for the Zero-day attacks. This approach showns the compound use of the behavioural analysis and NN for real time and dynamic perception.

In an attempt to extend the detection methodologies even further, Kok et al. (2020) identified areas that lacked evaluation methodologies for ransomware detection and put forward six new evaluation criteria. In doing so, their work will contribute to enhancing the practical applicability of detection frameworks, and narrowing down the previously mentioned shortcomings mainly because the key performance indicators presented allow for more accurate and consistent identification of ransomware incidents.

2.3 Classification and Prevention of Ransomware Transactions

Classification of ransomware related Bitcoin transactions is now considered as essential to combating ransomware business. Alsaif (2023) creates an effective classification model of high performance with the help of supervised learning approaches, including Random Forest and XGBoost. Our model tested on the Bitcoin Heist dataset yields a classification performance of 99.08% demonstrating higher feature extraction and data balancing in improving on the detection of fraudulent transactions.

There are frameworks in deep learning that has now set the bar for classification. To achieve the objective of this study, Abu Sallout and Ashour (2023) developed a neural network model in order to detect ransomware-related Bitcoin addresses. This model proves to have great generalization to new ransomware patterns as opposed to those used in training the model. Raising the problem of detecting ransomware threats based on metadata and transaction behaviours, the study demonstrates high effectiveness and capacity of deep learning approaches.

Another area in which rule-based approaches have made significant contribution in ransomware transaction classification. Talabani and Abdulhadi (2022) contrast Decision Table and Partial Decision Tree (PART) algorithms where PART obtained the (96.01%). This research highlights the usefulness of combining rule-based approaches with learning algorithms to handle the challenges of ransomware associated Bitcoin datasets. Likewise, Nayyer et al. (2023) also present an ensemble stacking model of classifiers, such as Random Forest and Decision Tree; they record an F1-score of 97% and AUC-ROC of 99%. This reduces precision as well as recall to an equal level, making it a favourable method for identifying fraudulent blockchain transactions.

Thus, we only get beyond the level of simple detection and classification with the analysis of preventive measures. Manavi and Hamzeh (2020) present a method to detect ransomware through CNN on the PE header with a detection accuracy of 93.33%. Herein, we simplify the PE headers into grayscale images to lower computational load while enhancing the features' detection rate. In their research, Karlphana et al., (2024) employed deep learning alongside hybrid cryptographic approaches to a higher accuracy of 99.89% in identifying Android ransomware. It improved its classified identification capabilities as well as security mechanisms when it comes to storage and encoding in the cloud capacity.

By using clustering and feature extraction, the classification and prevention of ransomware-related transactions have been taken one step forward. In the same way, De Lima (2021) performed K-Means clustering to classify the transactions in the given Bitcoin Heist dataset with 0.855 Adjusted Rand Index. Thus, this study shows the effectiveness of the use of unsupervised learning for separating the ransomware-related transactions from the normal ones; this research also highlights the need for clustering algorithms in ransomware analysis.

2.4 Summary

In this paper, we focus on the latest published literature in the identification and categorization of ransomware transactions and the development of methods to mitigate the occurrence of such activities in the blockchain network of Bitcoin. It is important to stress that significant progress in understanding ransomware mechanisms and combating this threat has been made Indeed, the work of researchers in recent years has covered almost all the issues mentioned above and presented them systematically in this article. PEDA, RENTA,AKA and deep neural network models have high accuracy, recall and have shown faster computation and thus are potential models for early detection and remedy.

These classifications and clusters of ransomware related transactions have expanded the knowledge about ransomware's functioning within the Bitcoin environment. This has been evident from the high evaluation metrics arrived by forms like supervised learning, unsupervised clustering and ensemble models for the analysis of malice.

Nevertheless, there are issues such as limited availability of standard datasets, the ability for real-time detection of supply chain risks, and framework that checks the multidimensional supply chain across the industries. More research must be conducted to overcome these limitations; increased collaboration between researchers in different disciplines and the formation of evidence-based public policies to curb ransomware operations in the cryptocurrency space must be deemed a high priority.

3 Methodology

The methodology adopted for the study is discussed in detail within this chapter. The detection of the ransomware in Bitcoin Heist Ransomware Dataset is a classification problem addressed through a detailed data analysis using machine and deep learning algorithms. The general flow of the methodology followed in the study is given in Figure 1 below.

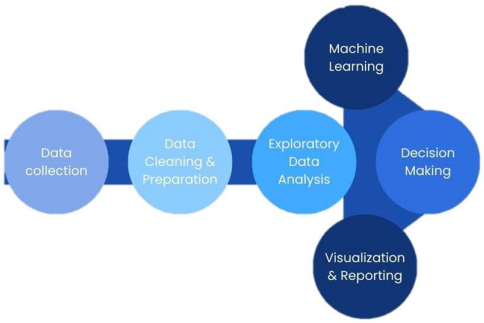

Figure 1: Methodology Flow

3.1 Bitcoin Heist Ransomware Dataset

The Bitcoin Heist Ransomware Address Dataset is a comprehensive resource designed for analysing and detecting illicit financial activities within the Bitcoin blockchain, particularly focusing on ransomware transactions. Below is an overview of the dataset:

3.1.1 Dataset Composition

- **Total Instances**: 2,916,697
- **Total Features**: 10
- **Data Collection Period**: January 2009 to December 2018

3.1.2 Features Description

1. **Address**: The unique Bitcoin address identifier.
2. **Year**: The year in which the transaction took place.

3. **Day**: The day of the year, ranging from 1 to 365.
4. **Length**: A metric capturing behavioural patterns of the address, such as the depth of transaction chains.
5. **Weight**: Reflects transaction merge behaviour, quantifying the relationship between input and output addresses.
6. **Count**: The total number of transactions linked to the address.
7. **Looped**: The count of transactions that involve the same address both as input and output.
8. **Neighbours**: The number of adjacent addresses in the transaction graph, representing connectivity.
9. **Income**: The total amount received by the address, expressed in satoshis (1 bitcoin = 100 million satoshis).
10. **Label**: A categorical label indicating whether the address is associated with a ransomware family (e.g., Cryptolocker, Cryptxxx) or is 'white' (non-ransomware).

3.1.3 Target Variable

- The **Label** feature serves as the target for classification tasks. It categorises addresses into ransomware families or as legitimate (non-ransomware) activities.

3.1.4 Dataframe Summary

	year	day	length	weight	count	looped	neighbors	income
count	2.916697e+06	2.916697e+06	2.916697e+06	2.916697e+06	2.916697e+06	2.916697e+06	2.916697e+06	2.916697e+06
mean	2.014475e+03	1.814572e+02	4.500859e+01	5.455192e-01	7.216446e+02	2.385067e+02	2.206516e+00	4.464889e+09
std	2.257398e+00	1.040118e+02	5.898236e+01	3.674255e+00	1.689676e+03	9.663217e+02	1.791877e+01	1.626860e+11
min	2.011000e+03	1.000000e+00	0.000000e+00	3.606469e-94	1.000000e+00	0.000000e+00	1.000000e+00	3.000000e+07
25%	2.013000e+03	9.200000e+01	2.000000e+00	2.148438e-02	1.000000e+00	0.000000e+00	1.000000e+00	7.428559e+07
50%	2.014000e+03	1.810000e+02	8.000000e+00	2.500000e-01	1.000000e+00	0.000000e+00	2.000000e+00	1.999985e+08
75%	2.016000e+03	2.710000e+02	1.080000e+02	8.819482e-01	5.600000e+01	0.000000e+00	2.000000e+00	9.940000e+08
max	2.018000e+03	3.650000e+02	1.440000e+02	1.943749e+03	1.449700e+04	1.449600e+04	1.292000e+04	4.996440e+13

Table 1: DataFrame Summary

3.2 Exploratory Data Analysis

The initial examination of the distribution of Bitcoin Heist Ransomware Address Dataset revealed its structure and distribution. Another area in which extraordinary effort was made was the use of visualizations in order to understand if there are any trends between features.

First, to show the distribution of ransomware families in the dataset, count plots were created (Figure 2). These plots show that there are fewer counts for some of the ransomware families compared to others. Barplot (Figure 3), Scatter Plot (Figure 4), and Histogram (Figure 5) also convey important information about the features in the dataset.

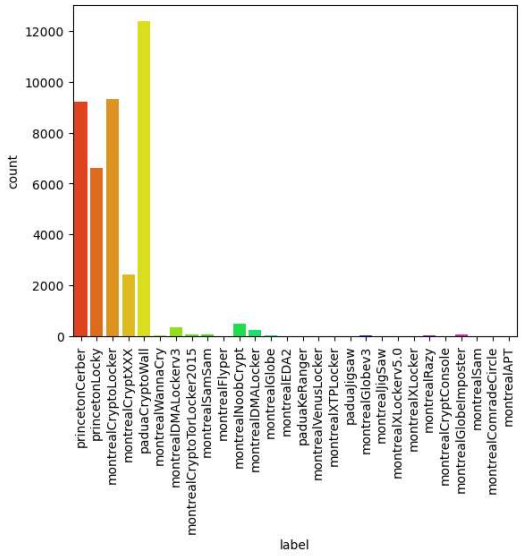

Figure 2: Count plot for the distribution of different ransomware attacks

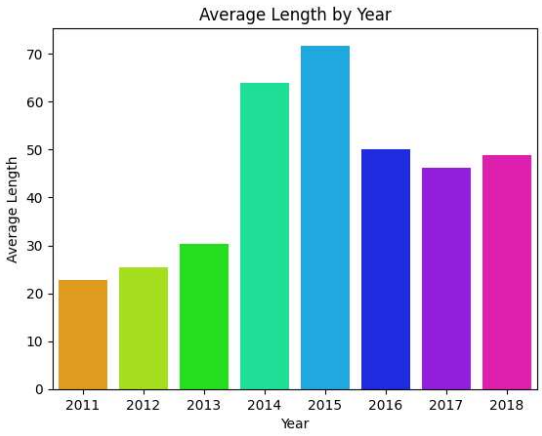

Figure 3: Bar Plot for Average Length by Year

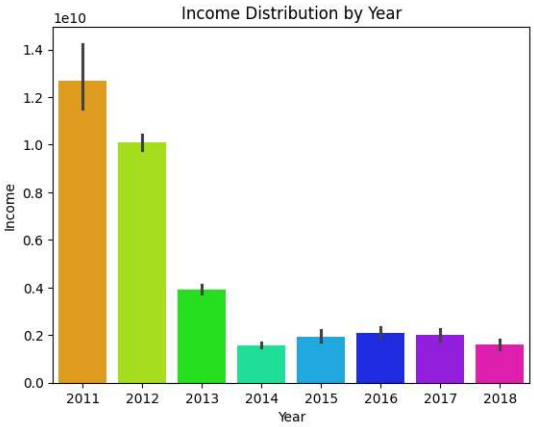

Figure 4: Barplot for Income Distribution by Year

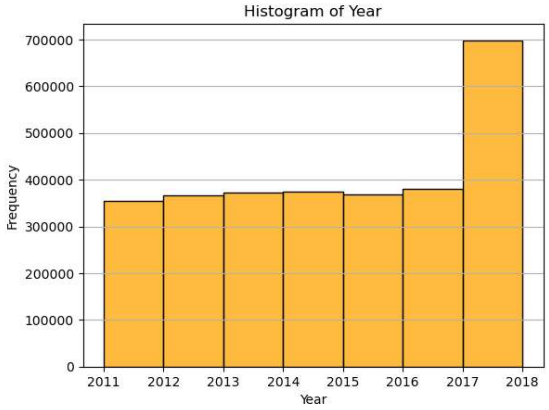

Figure 5: Number of Transactions according to Year

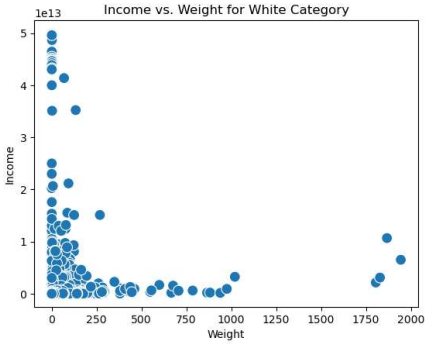

Figure 6: Scatter Plot for Income vs. Weight for Benign Transactions

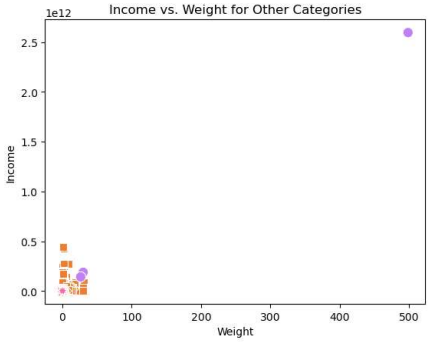

Figure 7: Scatter Plot for Income vs. Weight for Ransomware Transactions

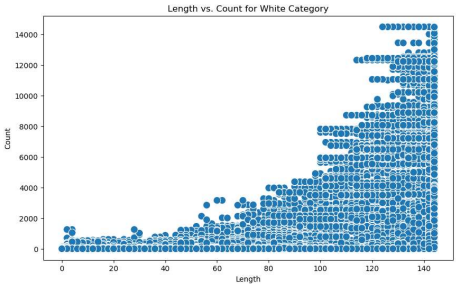

Figure 8: Scatter Plot for Length vs. Count for Benign Transactions

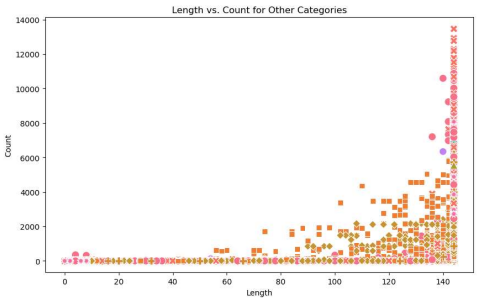

Figure 9: Scatter Plot for Length vs. Count for Ransomware Transactions

Scatter plots were used to compare relationships, such as weight (Figure 6), income (Figure 7), and length (Figure 8 and Figure 9), to discover variations between the transaction labels. Further, violin plots and bar charts are depicting the distribution of the features of the transactions with the details highlighting their possible use for predictions.

To analyse pairwise connections between numerical attributes a correlation heatmap was constructed which is represented in Figure 11 below. But in this analysis, specific attributes having high correlation values were determined with the aid of which features needed to be selected for improving the model had been chosen.

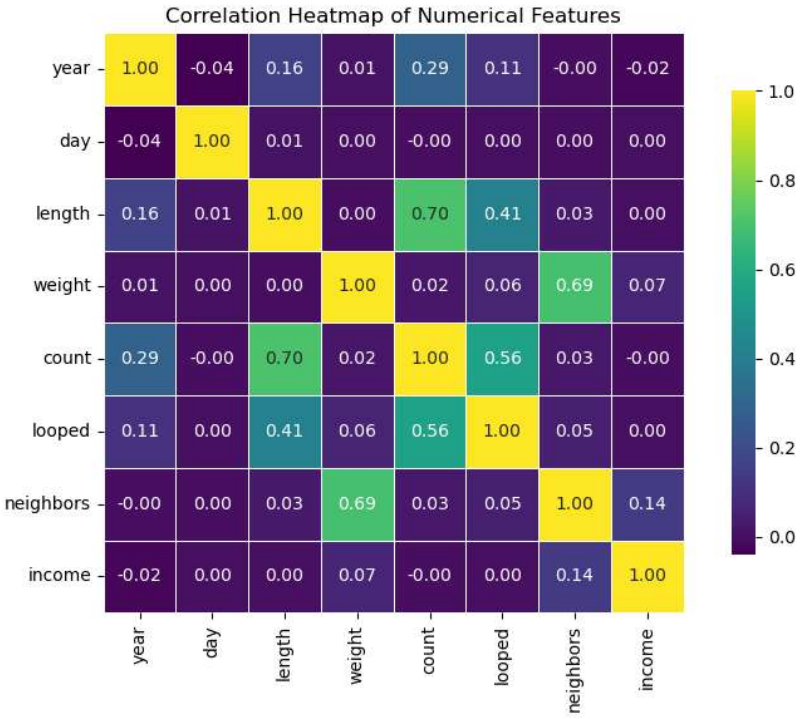

Figure 10: Correlation Matrix for the Features

3.3 Data Preprocessing

The data provided for model training was cleaned and normalized to ensure that data fed to the model would be both correct and relevant. Outliers and unnecessary values were further dealt

with appropriately in order to enhance the dataset. Irrelevant columns which include address, year, and day were removed.

Label encoding was crucial among preprocessing techniques as it helps in transforming the non-homogeneous data type to a form that concessions with machine learning algorithms. The last column, which was labeled as benign or ransomware families, was encoded to numerical data by using LabelEncoder. In order to downplay the complexity of the task at hand, all types of ransomware were classified under the umbrella term – ransomware, which led to a binary classification problem.

A problem prevalent in cybersecurity datasets is class imbalance, and the SMOTE algorithm was used to solve this. The dataset was balanced using SMOTE, which created new instances of the minority class in order to enhance their representation in the modeling process. Continuous variables including weight and income, which can affect features at large amounts more than others, were scaled via when using StandardScaler to standardise the values.

3.4 Feature Engineering

Feature engineering was one of the methods that formed the backbone of the methodology, aiming at identifying the peculiarities of the ransomware transactions. Features were derived across three categories: domain, time and network oriented.

Weight, income and length features were kept in the domain-specific features list as characteristics of ransomware-related transactions. Specific temporal dimensions like number of transactions during a certain period and duration were incorporated to capture behavioural characteristics of ransomware actions. Network features were designed while building k-nearest neighbour (KNN) graphs where the nodes were the Bitcoin addresses, and the links were the transactions. As such, graph metrics, including node centrality and clustering coefficients, added another layer of transactional information to the final dataset.

3.5 Data Splitting

The dataset is divided into two parts for model training. First part known as Training Set consists of 80% of the total data whereas the remaining 20% data acting as unseen data to the model makes up the Testing Set. The models are trained on the training set and they are tested on the test set for further evaluation.

3.6 Model Training

There are a number of machine learning and deep learning models which were trained and tested and different evaluation measures such as accuracy, precision or recall, F1 – score was used. The idea was to evaluate the performance of these categories in identifying ransomware related transactions in the Bitcoin network. The models selected encompasses both basic machine learning and deep learning techniques, offering a good way to compare the best methods for the classification of this paper.

3.6.1 Random Forest

The Random Forest classifier was some of the first models utilized in this investigation. As one of the ensemble learning, Random Forest involves the use of many decision trees in order to make one prediction to make it more accurate and reliable. For these reasons, it is particularly useful when dealing with high-dimensional data and which helps it avoid data overfitting, which makes it very useful for binary classification (Blanco & Tallón-Ballesteros, 2021). Other parameters including the number of trees(n_estimators) and the maximum depth of the trees (max_depth) were fine tuned from the Grid Search.

3.6.2 XGBoost

XGBoost algorithm, which is among the most optimized gradient boosting algorithm in the recent past, was used. XGBoost constructs a model of weak learning with each learner correcting the mistakes of its previous counterpart. A common approach to this phenomenon increases the predictive ability, however, minimizes vulnerability to overfitting (Nayyer et al., 2023). Hyperparameters of high significance in regulating model learning were adjusted to their optimal levels, composite of learning rate (learning_rate) and the number of boosting rounds (n_estimators) for best accuracy.

3.6.3 Convolutional Neural Networks (CNNs)

CNNs were employed because they capture local patterns in the transactional data better than other solution components. As CNNs were originated from image data the network was successfully changed to be applied to sequential and structured data (Solanki, 2019). Framework for this study was to have three layers of convolution followed by ReLU activation, max-pooling and fully connected dense layers. To mitigate overfitting dropout layers were used, and to train the model, the Adam optimiser with the learning rate tuned by a grid search was employed.

3.6.4 Graph Convolutional Networks

Graph Convolutional Networks were employed to analyse the graph structure of Bitcoin transactions, where nodes represented Bitcoin addresses and edges denoted transactions. GCNs perform particularly well on the analysis of relations inside graph-based data, which makes them suitable for this work (Ampel et al., 2023). The model in the study used two GCN layers with ReLU activation and cross-entropy loss as the optimisation function. Transaction connectivity was captured by k-nearest neighbour (k-NN) graphs to build edge indices.

3.6.5 Graph Isomorphism Networks (GINs)

GINs were added to serve as an instance of graph neural network of high complexity level that can capture complex graph representations. Thus, due to the sequential flow of information acquisition from neighbouring nodes, GINs include comprehensive inter-transaction affiliations. For this study, GINs were trained with specific optimized hidden dimensions and learning rates aimed at mimicking high accuracy classification (Berrueta et al., 2022).

3.7 Evaluation Metrics

The performance of all models was assessed using the following metrics:

- **Accuracy**: The proportion of correctly classified transactions.

- **Precision**: The ratio of true positive classifications to all positive predictions, indicating the model's ability to avoid false positives.
- **Recall**: The ratio of true positives to all actual positives, highlighting the model's ability to detect ransomware transactions.
- **F1-Score**: The harmonic mean of precision and recall, balancing the two metrics for a comprehensive performance measure.

According to these measurements, the models' accurate classification of ransomware associated transactions in the binary target variable is compared. This systematic thinking style enables successful data preprocessing, model training, features extraction, and performance assessment and gives a definite structure to ransomware detection and classification.

4 Design Specifications

The models for this study were designed to utilise machine learning and deep learning approaches to increase the/efficiency of detecting ransomware transactions within the Bitcoin network. The selected models include machine learning models, including Random Forest, XGBoost; deep learning models, including CNNs and GCNs or GINs. Among these models, the ones selected were suitable for dealing with both tabular and graph-based data sets. Figure 12 shows the design specification flow.

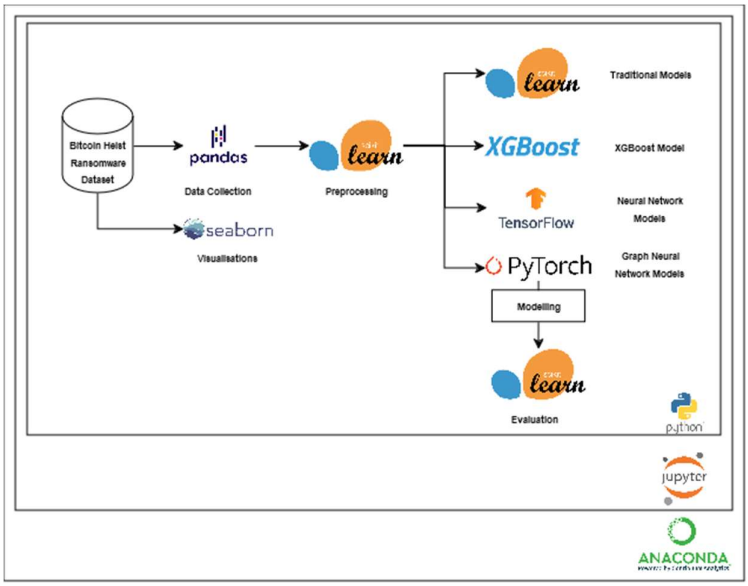

Figure 11: System Architecture

5 Implementation

This chapter discusses the experimental setup for the study, starting with a description of the hardware and software requirements for the study after which the in-depth discussion of the modelling part of the study is presented and discussed with a focus on model training and hyperparameter tuning.

5.1 Experimental Setup

The experimental setup for the study involves the installation of anaconda software which will facilitate the implementation of the experiment using Jupyter notebook. Libraries such as Tensorflow with Keras, Torch, and XGBoost.

5.1.1 Hardware Configuration

Table 2 below shows the hardware configuration of the system over which the study has been implemented.

Component	Specification
RAM	8GB
ROM (Internal)	512GB
ROM (External)	512GB
OS	Windows 10

Table 2: Hardware Configuration for the Implementation

5.1.2 Software Configuration

For software, Python 3.8 was used as the main coding language with Jupyter Notebook in use for interactivity as well as implementation of visualisations. Data undersampling was achieved by using SMOTE, a technique that was supported by imblearn. In contrast, the data was preprocessed and trained using sci-kit learn and numpy and pandas libraries for data manipulation. Regarding the visualisation, the datasets were analysed using matplotlib and seaborn to construct meaningful plots such as count plots, scatter plots, and heatmaps. Random Forst and XGBoost were implemented using sci-kit learn and xgboost libraries; CNNs, GCNs, and GINs were implemented using TensorFlow and Keras and PyTorch and PyTorch Geometric, respectively. Hyperparameters were optimised via GridSearchCV from scikit-learn for traditional models, and scikeras was used to connect keras and GridSearchCV. This configuration proved more efficient and created a scalable structure to facilitate the experiments in relation to the research objectives that were laid down.

5.2 Model Training

The hyperparameters are tuned for the models to identify the most suitable model architectures. Table 3 below lists the hyperparameters that have been tuned along with their values that were tested to obtain the best accuracy.

Model	Hyperparameters	Values Tested
Random Forest	n_estimators, max_depth	n_estimators: [50, 100], max_depth: [None, 10]
XGBoost	learning_rate, n_estimators	learning_rate: [0.01, 0.1], n_estimators: [50, 100]
CNN	batch_size, epochs	batch_size: [32, 64], epochs: [10]
GCN	hidden_dimensions, learning_rate	hidden_dimensions: [16, 32], learning_rate: [0.01, 0.001]
GIN	hidden_dimensions, learning_rate	hidden_dimensions: [16, 32], learning_rate: [0.01, 0.001]

Table 3: List of Hyperparameters Tuned for the Models

- **Random Forest**: Random Forest was used in this study as a regular ensemble of -based model of machine learning, which is well understood for its efficiency at dealing with highly dimensional data. We split the data into two training sets and used 50 and 100 estimators (n_estimators) which are the number of trees in the ensemble, depths of None and 10 for each tree (max_depth). These parameters were optimized by means of grid search in order to obtain the best results. In addition, Random Forest minimises overfitting and increases generality by combining the results of numerous decision trees. The model's performance was also examined on the test set through accuracy measure, precision, recall and F1-score. It showed quite a good level of stability with highly structured low value transactional datasets which formed the basis for comparison in most studies.
- **XGBoost**: XGBoost which is a modern gradient boosting algorithm used as it works well with large and imbalanced datasets. The model was further improved, by hyperparameters tuning, including n_estimators = 50, 100 and learning_rate= 0.01, 0.1. These parameters were optimized through the grid search which allowed the model to develop weak learners one at a time and gradually reduce errors. The missing value handling of XGBoost along with the features of regularisation was advantageous for the classification task. Evaluation results showed that the predictive accuracy achieved impressive values, especially for detecting ransomware transactions compared with non-ransomware ones.
- **CNN**: The CNN model was designed with the input of structured transactional data, and they work in a way that converts the input data into a 1D input format to extract hierarchical patterns from them. The architecture adopted three convolutional layers with ReLU activation functions followed by max-pooling layers to diminish dimensionality. The extracted features were passed to the fully connected dense layers and the final output layer with sigmoid activation function for binary classification. Some of the experiments conducted during the hyperparameter tuning included the following batch sizes: 32 and 64, while the model was trained for 10 epochs. The dropout layers were incorporated into the model to avoid overfitting, while the Adam optimiser was employed to improve convergence. The CNN was particularly effective in identifying local patterns and played a key role in the classification process.
- **GCN**: To enhance the performance of the model by tapping graph-structured data, the GCN model was applied with Bitcoin addresses as nodes and transactions as edges. The architecture contained two convolution layers that pool information from adjacent nodes over the network using ReLU nonlinearity. The edge indices were generated using a k-nearest neighbour (k-NN) graph. The hyperparameters like hidden dimensions @ 16, 32 and learning rate @ 0.001, 0.01 were also optimised for the model. The model was trained on the current epoch using the Adam optimiser for 10 epochs and the results on the test set are presented below. GCN was shown to outperform other networks regarding relational information and transaction dependency hence suitable for ransomware detection.
- **GIN**: GIN was chosen because of its optimal capacity to produce high-density node features, which are important for many graph computations. This architecture entailed

two GIN convolutional layers with linear transformations- A capability allowing the model to understand the sophisticated patterns within the transactional network. As in GCN, a k-NN graph was used to define edge indices. Hyperparameters were tuned to the hidden dimensions equal to 16 and 32, and selected learning rates were 0.001 and 0.01, with training being done for 10 cycles. According to the GIN model results, graph embedding learning performance was excellent in terms of accuracy, precision, recall, and F1, and much higher than other models for the block of inter-transaction structure.

This chapter discusses the implementation of the system employed for the study in depth. It put forth the hardware and software requirements for the system implementation, followed by the in-depth presentation of the model training with hyperparameter tuning. The models implemented in the study were evaluated as five different case studies which are discussed in great detail in the coming evaluation chapter.

6 Evaluation

In this section, the focus will be to offer a comprehensive analysis of the findings of the conducted study and put forward the relative implications in light of research findings at the theoretical and practical levels. It only specifically reports on the most relevant outcomes related to the study question and purpose and provides a comprehensive and critical discussion.

6.1 Results for the Random Forest Model

In the tuning of hyperparameters, the Random Forest model was assessed using the three-fold cross-validation method. Several experiments were conducted to determine the best parameters, and it was discovered that a n_estimator parameter of 100 along with a max_depth of None has the best accuracy. These settings allowed the model to navigate complexity and generalisation with reasonable success.

For instance, the model's ability to classify ransomware-related transactions was established by checking its accuracy on the test set. Thus, within the Random Forest model, 94.64% of cases were correctly classified. The high precision score of 98.85% indicated its capability of reducing the number of false positives and thus allowing few benign transactions to be misclassified as ransomware. Also, the recall score of 95.68% showed that the model detected true positive transactions of ransomware since 95.68% of the sample was correctly identified. A F1 score of 97.24% further supported overall accuracy.

6.2 Results for the XGBoost Model

Evaluation of the XGBoost model was conducted using the three fold cross validation technique in an attempt to tune the hyperparameters of the model better. As for the hyper-parameters, the best parameters were determined to be learning_rate=0.1, n_estimators=100. This setup enabled the model to combine the best of both worlds, specifically the learning speed or iterations per data epoch and the model accuracy or cumulative gradient boosting .

Using the test data, the model demonstrated high accuracy in predicting the properties of molecules. The model efficiency was further confirmed by the results of the test done where

the efficiency of the model was 91.29%, meaning that most of the transactions had been correctly classified. The low false positive rate of 98.97 of % further affirmed its efficiency in distinguished between benign activities from real ransomware. The recall score of the model was found to be 92.14% which clearly means that it was capable of covering a large number of true ransomware transactions from the positive class. About the F1 score of 95.43%, involving both precision and recall into the evaluation demonstrated that the model was well tested and capable of weighting both precision and recall.

6.3 Results for the CNN Model

The used CNN model was tuned using the grid search to define the best hyperparameters for ransomware identification. The highest performance was achieved when a batch size of 32 was used together with training up to 10 epochs. The above parameters were selected so that a model achieves a reasonable level of accuracy in the computation and does not over-fit or under-fit.

When proceeding with the ability of the CNN model, moderate performance on the test set was achieved. Overall, the model had an accuracy of 79.17%, a show that out of the many transactions the model was correct for the most part though the was poor in handling of the intricate patterns within the data set. The total percent of accuracy it obtained was 98.93% signifying its effectiveness to filter out maximum number of false positive and classify them as non-ransomware transactions. However, the recall score of 79.76% indicate that there was a problem in identifying all ransomware related Transactions and the system has 12.14% false negatives. The F1 score of 88.32 was obtained from the model that combines precision and recall of the model and shows that there is room for increasing an effective identification of true positives.

6.4 Results for the GCN Model

To compare the performance of the Graph Convolutional Network (GCN) model, hyperparameters for the model were chosen using grid search optimization. The best configuration found was the hidden dimension size of 16 and the learning rate (lr) = 0.01 allowed to learn graph embeddings while keeping the optimisation stable.

On the test set, the evaluation metrics painted somewhat mixed picture. According to the assessment outcome, the distinction between different transactions was done with moderate success and the mode of preference was giving an accuracy of 65.55%. The precision of 97.35% showed high effectiveness of the model in excluding false positive, or, in other words, the ability to recognize true negative or, in the context of transactional data, benign transactions. However, when it comes to specifics of ransomware-related transactions, the recall score of 65.55% showed the model's significant weaknesses due to the number of true positives that were completely overlooked. With an F1 score of 78.09%, recall, it was revealed that, while the model successfully distinguished non-ransomware transactions, its identification of ransomware activity was limited.

This study shows that, even though the GCN model has higher accuracy when compared to other performance measures, difficulties persist in using graph-structured data for this

classification task where recall has a relatively low value. Practical applications of the existing framework may reap benefits from additional feature engineering or optimization of the graph construction phase to increase the model's capacity for detecting ransomware transactions comprehensively.

6.5 Results for the GIN Model

The GIN model was tested in relation to ransomware detection. Hyperparameter tuning was performed through grid search for optimisation, and the best configuration of the model was found to include a hidden dimension size of 32 and a learning rate (lr) of 0.01. This configuration aimed to enable the model to learn graph embeddings and capture intricate transactional relationships effectively.

The evaluation metrics on the test set, however, revealed significant limitations in the model's performance. The GIN model achieved an accuracy of 33.85%, indicating a considerable failure to correctly identify ransomware transactions. Despite attaining a precision of 97.39%, which demonstrates the model's ability to minimise false positives and classify benign transactions accurately, its recall score of 33.85% reflects its inability to detect a substantial portion of ransomware transactions. The F1 score of 49.33% highlights an overall imbalance between precision and recall, further indicating the model's poor detection capability.

The results indicate that while the GIN model effectively reduces false positives, it struggles to identify ransomware transactions accurately. This suggests that while graph-based methods hold potential for addressing problems involving complex transactional interactions, the GIN model in its current configuration is not a strong candidate for ransomware detection. Further optimisation and enhancements, such as additional feature engineering or advanced architectures, would be required to improve its effectiveness in this domain.

6.6 Discussion

The results of this research are vital as the study aims to present an analysis of various machine learning and deep learning algorithms for ransomware detection in the Bitcoin context. Every model showcased its advantages and risks, which makes choosing algorithms for complex classification problems comprehensible. Finally, the results are presented in the light of theory and practice. Table 4 below provides the comparative results of the models.

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.9464	0.9885	0.9568	0.9724
XGBoost	0.9129	0.9897	0.9214	0.9543
CNN	0.7917	0.9893	0.7976	0.8832
GCN	0.6555	0.9735	0.6555	0.7809
GIN	0.3385	0.9739	0.3385	0.4933

Table 4: Results for Modelling

6.6.1 Random Forest Model

The Random Forest model demonstrated exceptional performance, achieving an accuracy of 94.64% and an F1 score of 97.24%. Its precision of 98.85% highlights the model's ability to minimise false positives, ensuring that benign transactions are rarely misclassified as ransomware. The recall score of 95.68% confirms the model's capability to detect the majority of ransomware transactions, making it highly effective in scenarios where both precision and recall are critical. These results align with the established strengths of ensemble methods, particularly their ability to balance bias and variance in structured data classification. Moreover, the Random Forest model's interpretability and computational efficiency make it practical for deployment in real-world cybersecurity frameworks.

6.6.2 XGBoost Model

The XGBoost model exhibited strong performance with an accuracy of 91.29% and an F1 score of 95.43%. Its precision of 98.97% underscores its effectiveness in distinguishing benign transactions from ransomware. However, the recall of 92.14% is slightly lower than that of the Random Forest model, indicating that it may miss some ransomware transactions. These findings demonstrate XGBoost's suitability for datasets with intricate patterns and class imbalances due to its iterative boosting mechanism, which improves predictions with each iteration. Nonetheless, the relatively lower recall suggests that additional feature engineering or hyperparameter tuning could further enhance its performance, particularly for applications demanding high recall.

6.6.3 Convolutional Neural Network (CNN)

The CNN model achieved an accuracy of 79.17% and an F1 score of 88.32%, indicating moderate performance. Its precision of 98.93% shows that it effectively reduces false positives, but the recall of 79.76% highlights its inability to capture a significant portion of ransomware-related transactions. This asymmetry suggests that the model struggles to identify complex transactional patterns, particularly when applied to structured data without extensive preprocessing or feature extraction. While CNNs excel at recognising patterns in image-based data, their application to transactional datasets is less effective. Nevertheless, the model's ability to adapt to local features makes it a viable component in hybrid solutions combined with graph-based algorithms for improved detection.

6.6.4 Graph Convolutional Network (GCN)

The GCN model achieved an accuracy of 65.55%, with precision and recall scores both at 65.55%, resulting in an F1 score of 78.09%. Although its precision indicates a low rate of false positives, the model's low recall reflects significant challenges in identifying actual ransomware transactions. These results suggest that processing graph-structured data for this classification task remains complex and requires improvements in feature extraction and graph construction. Despite its limitations, GCN offers potential for analysing relational data and could benefit from further refinements, such as advanced graph embeddings, to enhance its suitability for detecting ransomware transactions.

6.6.5 Graph Isomorphism Network (GIN) Model

The GIN model achieved the lowest performance among the tested models, with an accuracy of 33.85%, a precision of 97.39%, and a recall of 33.85%, resulting in an F1 score of 49.33%.

While its high precision highlights its ability to minimise false positives, its poor recall indicates a failure to capture most ransomware transactions. These results reveal that the model struggles to generalise effectively and may not be well-suited for this classification task in its current configuration. Despite these limitations, the GIN model's capability to process intricate graph-based features suggests potential for improvement through better graph construction or enhanced feature engineering.

6.6.6 Implications and Recommendation

Analysis of the models points out to the discrepancies between precision and recall ratios, which are essential in real-world ransomware detection scenarios. Thus, although Random Forest and XGBoost outperform the models when given highly structured data, Graph Isomorphism Network link features, and therefore provide more suitable representation of relational characteristics, making them complementary. Based on these observations, we hypothesize that a synergistic integration of insights from traditional supervised/unsupervised learning and graph-based deep learning might produce even more promising results.

From the theoretical point of view, the results confirm the relevance of ensemble techniques and graph neural networks for cybersecurity purposes. In practical terms, the work shows how the proposed models can be effectively adopted for the purpose of ransomware detection, and how relevant real-world challenges can be overcome mostly through realistic design considerations and practical performance optimizations. Further studies should focus on the integration of the architecture, and other complex feature engineering beyond what was explained above, and optimizing graph structures and algorithms to improve the detection rates and robustness of the proposed graph-based approaches.

7 Conclusion and Future Work

Ransomware detection using machine learning and deep learning models was investigated in this work in the context of detecting ransomware transactions in the Bitcoin system. Using data from the Bitcoin Heist Ransomware Address Dataset, the study aimed to increase the understanding and classification of ransomware payments successfully through analysis. Random Forest, XGBoost, CNN, GCN, and GIN are the five models tested with accuracies, precisions of the five models, recalls, and F1-scores calculated.

Key Findings:

- **Random Forest**: Random Forest emerged as the most robust model, achieving the highest accuracy (94.64%) and a well-balanced F1-score (97.24%). Its ensemble method effectively handled high-dimensional structured data, demonstrating its reliability for ransomware detection tasks.
- **XGBoost**: XGBoost exhibited strong performance with an accuracy of 91.29% and an F1-score of 95.43%. Its iterative boosting mechanism proved effective for datasets with intricate patterns and class imbalances, making it a competitive option for ransomware detection, though slightly less effective than Random Forest in recall.
- **CNN**: The Convolutional Neural Network (CNN) model showed moderate performance, achieving an accuracy of 79.17% and an F1-score of 88.32%. While its

high precision (98.93%) highlights its ability to reduce false positives, its relatively low recall (79.76%) underscores challenges in identifying ransomware transactions comprehensively within structured data.

- **GCN**: The Graph Convolutional Network (GCN) demonstrated precision at 97.35%, effectively minimising false positives. However, its accuracy (65.55%) and recall (65.55%) revealed limitations in capturing complex ransomware patterns within graph-structured data, highlighting the need for further optimisation.
- **GIN**: Despite its high precision (97.39%), the Graph Isomorphism Network (GIN) struggled overall, achieving an accuracy of 33.85% and an F1-score of 49.33%. While it shows potential for processing graph-based features, significant improvements are needed for effective ransomware detection.

These results suggest trade-offs between precision and recall and various measures of cost for the models under comparison. The study also reveals the significance of integrating various machine learning and deep learning techniques to solve the problems of ransomware detection in blockchain systems.

Implications

The results validate the continuation of ensemble methods and graph-based neural networks as significant for cybersecurity. From the log–level point of view, Random Forest and XGBoost are suitable for structured data, whereas GCN and GIN are efficient for relational and graph data. It will be useful in formulating even better ways of identifying ransomware and or formulating better cybersecurity measures.

Limitations

There were several constraints in the execution of this study as follows. It was also noted that the problem of class imbalance in the given set of data examples affected model performance even when SMOTE approach was implemented. Random Forest and XGBoost accuracy were high, but CNN and GCN deep learning models performed poorly with structured data and low recall. GIN delivered slightly better results, though at a cost of higher computation time.

Future Work

Building on the findings of this study, future research can explore the following directions:

Hybrid Models: It is possible that the integration of ensemble methods like Random Forest and XGBoost and graph-based models including GCN and GIN can further improve detection since they have unique advantages each possesses.

Feature Engineering: The new temporal and contextual features in Cubic live and in ADABT prove that new feature engineering can help increase the accuracy of the models and their resilience.

Graph Optimization: Proposing better graph structures construction, for example, the dynamic graph representations may help the construction of graph neural networks.

Deep Learning Architectures: Improving the architectures that have been discovered, for example, Transformers or attention mechanisms can enhance the opportunity to embrace relational characteristics of the transactional data.

This research offers a basis for improving ransomware detection and explains the prospects of machine learning and deep learning models for solving multifaceted cybersecurity issues. In the next generation, updated hybrid ive and novel architectures present themselves as potentially more effective in addressing rider ransomware within next generation environments.

8 References

Akcora, C.G., Li, Y., Gel, Y.R. and Kantarcioglu, M., 2019. Bitcoinheist: Topological data analysis for ransomware detection on the bitcoin blockchain. *arXiv preprint arXiv:1906.07852*.

Alsaif, S.A., 2023. Machine Learning-Based Ransomware Classification of Bitcoin Transactions. *Applied Computational Intelligence and Soft Computing*, *2023*(1), p.6274260.

Ampel, B., Otto, K., Samtani, S. and Chen, H., 2023, October. Disrupting ransomware actors on the bitcoin blockchain: A graph embedding approach. In *2023 IEEE International Conference on Intelligence and Security Informatics (ISI)* (pp. 1-6). IEEE.

Anderson, R., Barton, C., Böhme, R., Clayton, R., Ganán, C., Grasso, T., Levi, M., Moore, T. and Vasek, M., 2019, June. Measuring the changing cost of cybercrime. In *The 18th Annual Workshop on the Economics of Information Security (WEIS 2019)*.

Berrueta, E., Morato, D., Magaña, E. and Izal, M., 2022. Crypto-ransomware detection using machine learning models in file-sharing network scenarios with encrypted traffic. *Expert Systems with Applications*, *209*, p.118299.

Blanco, J.A. and Tallón-Ballesteros, A.J., 2022. Supervised Machine Learning Techniques in the Bitcoin Transactions. A Case of Ransomware Classification. In *16th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2021)* (pp. 803-810). Springer International Publishing.

de Lima, T.R., 2021. Bitcoin Blockchain Clustering Analysis for Ransomware Detection. *Authorea Preprints*.

Goyal, P.S., Kakkar, A., Vinod, G. and Joseph, G., 2020. Crypto-ransomware detection using behavioural analysis. In *Reliability, Safety and Hazard Assessment for Risk-Based Technologies: Proceedings of ICRESH 2019* (pp. 239-251). Springer Singapore.

Huang, D.Y., Aliapoulios, M.M., Li, V.G., Invernizzi, L., Bursztein, E., McRoberts, K., Levin, J., Levchenko, K., Snoeren, A.C. and McCoy, D., 2018, May. Tracking ransomware end-to-end. In *2018 IEEE Symposium on Security and Privacy (SP)* (pp. 618-631). IEEE.

Jemal, M., 2023. Detection of crypto-ransomware attack using deep learning.

Kalphana, K.R., Aanjankumar, S., Surya, M., Ramadevi, M.S., Ramela, K.R., Anitha, T., Nagaprasad, N. and Krishnaraj, R., 2024. Prediction of android ransomware with deep learning model using hybrid cryptography. *Scientific Reports*, *14*(1), p.22351.

Kim, G.Y., Paik, J.Y., Kim, Y. and Cho, E.S., 2022. Byte frequency based indicators for crypto-ransomware detection from empirical analysis. *Journal of Computer Science and Technology*, *37*(2), pp.423-442.

Kok, S.H., Abdullah, A. and Jhanjhi, N.Z., 2022. Early detection of crypto-ransomware using pre-encryption detection algorithm. *Journal of King Saud University-Computer and Information Sciences*, *34*(5), pp.1984-1999.

Manavi, F. and Hamzeh, A., 2020, September. A new method for ransomware detection based on PE header using convolutional neural networks. In *2020 17th international ISC conference on information security and cryptology (ISCISC)* (pp. 82-87). IEEE.

Nayyer, N., Javaid, N., Akbar, M., Aldegheishem, A., Alrajeh, N. and Jamil, M., 2023. A new framework for fraud detection in bitcoin transactions through ensemble stacking model in smart cities. *IEEE Access*.

Raheem, A., Raheem, R., Chen, T.M. and Alkhayyat, A., 2021, September. Estimation of ransomware payments in bitcoin ecosystem. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (pp. 1667-1674). IEEE.

Sallout, B.W.A., Gaza, P. and Ashour, H.A., Detecting Ransomware in Bitcoin Transaction Using Machine Learning.

Solanki Y., Panchal M., 2019. Detection and Prevention for Ransomware using Machine Learning. *International Journal for Research in Applied Science & Engineering Technology*, 7(5), pp. 455-463.

Talabani, H.S. and Abdulhadi, H.M.T., 2022. Bitcoin ransomware detection employing rule-based algorithms. *Science Journal of University of Zakho*, *10*(1), pp.5-10.

Turner, A.B., McCombie, S. and Uhlmann, A.J., 2020. Discerning payment patterns in Bitcoin from ransomware attacks. *Journal of Money Laundering Control*, *23*(3), pp.545-589.

Turner, A., McCombie, S. and Uhlmann, A., 2021. Follow the money: Revealing risky nodes in a Ransomware-Bitcoin network.

Wang, K., Pang, J., Chen, D., Zhao, Y., Huang, D., Chen, C. and Han, W., 2021. A large-scale empirical analysis of ransomware activities in bitcoin. *ACM Transactions on the Web (TWEB)*, *16*(2), pp.1-29.

Zakaria, W.Z., Abdollah, M.F., Mohd, O., Yassin, S.W.M.S.M. and Ariffin, A., 2022. Rentaka: A novel machine learning framework for crypto-ransomware pre-encryption detection. *International Journal of Advanced Computer Science and Applications*, *13*(5).