Detecting Illicit Entities in Bitcoin using Supervised Learning of Ensemble Decision Trees

Pranav Nerurkar, Yann Busnel, Romaric Ludinard, Kunjal Shah, Sunil Bhirud, Dhiren Patel

Detecting Illicit Entities in Bitcoin using Supervised Learning of Ensemble Decision Trees

Pranav Nerurkar
panerurkar_p16@ce.vjti.ac.in
pranav.n@nmims.edu
Dept. of CE&IT, VJTI
MPSTME, NMIMS University
Mumbai, India

Yann Busnel
yann.busnel@imt-atlantique.fr
IMT Atlantique, IRISA
Cesson-Sévigné, France

Romaric Ludinard
romaric.ludinard@imt-atlantique.fr
IMT Atlantique, IRISA
Cesson-Sévigné, France

Kunjal Shah
kunjal1999@gmail.com
VJTI Mumbai
Mumbai, India

Sunil Bhirud
sgbhirud@ce.vjti.ac.in
VJTI Mumbai
Mumbai, India

Dhiren Patel
dhiren29p@gmail.com
VJTI Mumbai
Mumbai, India

Abstract

Since its inception in 2009, Bitcoin has been mired in controversies for providing a haven for illegal activities. Several types of illicit users hide behind the blanket of anonymity. Uncovering these entities is key for forensic investigations. Current methods utilize machine learning for identifying these illicit entities. However, the existing approaches only focus on a limited category of illicit users. The current paper proposes to address the issue by implementing an ensemble of decision trees for supervised learning. More parameters allow the ensemble model to learn discriminating features that can categorize multiple groups of illicit users from licit users. To evaluate the model, a dataset of 2059 real-life entities on Bitcoin was extracted from the Blockchain. Nine features were engineered to train the model for segregating 28 different licit-illicit categories of users. The proposed model provided a reliable tool for forensic study. Empirical evaluation of the proposed model vis-a-vis three existing benchmark models was performed to highlight its efficacy. Experiments showed that the specificity and sensitivity of the proposed model were comparable to other models.

CCS Concepts: • **Computing methodologies** → **Artificial intelligence**; **Boosting**.

Keywords: Bitcoin, Fraud detection, Supervised Learning, Data Mining

1 Introduction

Bitcoin [1] platform has attracted anti-social elements [1] as it creates hurdles for law enforcement to trace suspicious transactions due to the anonymity and privacy [2]. As bitcoin became financially significant, the emergence of Ponzi schemes, money laundering, frauds, embezzlements, extortion, and tax evasion [3] practices were seen. These businesses used the blanket of secrecy afforded by Bitcoin to mislead the audit trail. It was speculated that in 2017, BTCs worth $770 million were traded for illicit activities [4], a quarter of bitcoin users were malicious and 46% of all bitcoin activity was illegal [5].

Due to voluminous data generated about bitcoin transactions on the Blockchain, machine learning became a popular technique for tracking and scrutinizing illicit users or transactions. Existing literature surveyed on detecting illegal activities using Machine Learning (ML) had focused on illegal transactions, identifying suspicious bitcoin users (extortionists, ponzi scams, darknet markets, ransomware, human traffickers, frauds), detecting money laundering, identifying mixing services, identifying bitcoin exchanges, identifying illegal transactions, identifying bitcoin wallets and bitcoin miners.

The issues faced in the application of machine learning in identifying illegal activities are lack of benchmark, public datasets (see Table 2), full information of Blockchain, and lack of ground truth information on the identities of bitcoin users. Apart from these issues, cryptocurrencies offer their users pseudo-anonymity by allowing users to transact with each other through hash address. These addresses can be created and discarded countless times, complicating the task of linking a transaction to a user. The target of interest was restricted to limited categories of illicit users to reduce the computational complexity of machine learning models. Additionally, the time interval for which data was collected from the Blockchain for feature engineering was restricted to shorter spans. Due to these, the models obtained after training were not generalized.

1.1 Motivation

The current paper aimed to build upon and extend work in detecting illegal entities in Bitcoin. Features of bitcoin users were derived by scrutinizing the Blockchain from 03

[1]In this paper, Bitcoin refers to the system, and bitcoin or BTC refers to the digital currency

Preprint to ICICM2020 '20, August 12-14, 2020, Paris, France
2020.

Jan 2009 12:45:05 GMT to 08 May 2020 at 13:21:33 GMT. This was to provide the model with features suitable for generalized learning of entity behavior. Additionally, this avoided a model trained for recognizing a limited category of entities.

1.2 Contributions

Contribution of the current paper are as follows:

- A public dataset of addresses and features of illicit Bitcoin entities [2]
- Empirical analysis of different learning strategies for classifying illicit Bitcoin entities;
- Implementing a supervised learning approach that estimated the most discriminating features for detecting categories of illicit Bitcoin users.

1.3 Novelty

An extensive literature survey could find studies focusing on only a subset of illicit activities viz. botnets, extortionists, ponzi scams, darknet markets, ransomware, human traffickers, frauds, money laundering, and mixing services. At the time of writing (May 2020), there has not been any research focusing on a broad spectrum of illegal activities.

1.4 Outline

The rest of the paper is organized into four sections. Section 2 provides the preliminaries needed for the paper, along with a critique of the current literature. Materials and methods detail the data collection and preparation strategy in Section 3. The proposed work is described in Section 4 followed by Experimental study in Section 5 and Conclusion and future works in Section 6.

2 Related work

Fundamental concepts of Bitcoin are described in Sections 2.1 and 2.2. Followed by critical analysis of published studies on detecting illegal users (see Section 2.3), issues in available datasets (see Table 2) and popular ML models used in published studies (see Table 3).

2.1 Description of Bitcoin system

Bitcoin transactions are added to "Blocks" and recorded into a distributed public ledger "Blockchain." Each transaction has several inputs (senders) and outputs (receivers). The metadata [3] associated with blocks, transactions, inputs, and outputs provides scope for analysis.

[2]https://github.com/pranavn91/blockchain/blob/master/datasetjune2.RData
[3]https://github.com/blockchain-etl

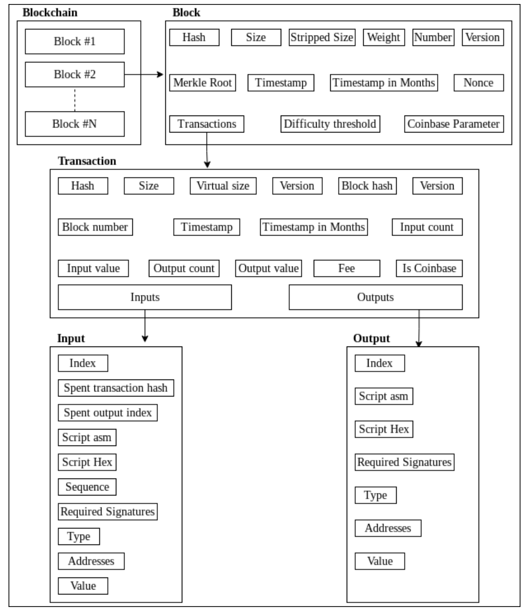

Figure 1. Anatomy of Bitcoin system

2.2 Common types of services on Bitcoin

- Gambling: Allow placing of bets using BTCs
- Darknet markets: Selling and buying goods using BTCs
- Mixers: Remove traceability of BTCs from source
- Cyber-criminals: Blacklisted by governments
- Ponzi: High yield investment scams

Other illicit users are ransom calls for extortions, bombs threats, sextortionists, scams, and blackmail.

2.3 Studies on detecting illegal activities in Bitcoin

Table 1. Summary of published bitcoin studies

Authors	Description	Features extracted
B Zarpelao *et al.* [6]	Detection of botnets to launch DDoS attacks	Transaction features
C Lee *et al.* [4]	Detecting Illegal Transactions	Transaction features
Y Wu *et al.* [7]	Tracing suspicious bitcoin entities	Transaction features
M Weber *et al.* [8]	Identifying illicit bitcoin users	Transaction features
Y Hu *et al.* [9]	Detecting Money Laundering	Graph embeddings
H Yin *et al.* [10]	Identifying illicit bitcoin users	Transaction features
L Nan *et al.* [11]	Mixing service detection	Graph embeddings

Majority researchers have focused on limited categories of illicit users and shorter periods.

Table 2. Datasets used in published bitcoin studies

Dataset	Accessibility	Features	Categories	Size
Chainanalysis [12]	Private	9	exchange, gambling, hosted wallet, merchant services, miningpool, mixing, ransomware, scam, tor market or other	198,097,356
UIUC [13]	Public	0	0	37,450,461
BitcoinPonzi [14]	Public	11	Ponzi, Non-ponzi	6432
R Portnoff *et al.* [15]	Private	2	Sex offender, Ordinary	753,929
D Ermilov *et al.* [16]	Private	238	Service, gambling, mixer, exchange, pool, darknet	244,030,115

Table 3 gives the popular ML models for bitcoin studies.

Table 3. ML classifier used in published bitcoin studies

ML models	Research Paper
Decision Trees (DT)	[17]
Bagging Classifier	[18]
Gradient Boosting	[12]
AdaBoost	[19]
Logistic regression (LogReg)	[17]

From Table 1 in literature, it is evident that the implementation of a reliable and secure illegal user detection system is a major concern for privacy and security in Bitcoin. Existing works have not focused on a broad spectrum of illegal activities that are conducted on Bitcoin. Additionally, existing datasets to are unsuitable for machine learning as their focus is narrow or are proprietary. In this respect, in Section 3, describes the data collection methodology to overcome the issue of data availability in public datasets. Section 4 discusses the proposed methodology for a classifier that could identify a broad spectrum of illicit users on Bitcoin.

3 Materials and Methods

As available datasets in literature (see Table 2) have shortcomings, procedure described in Section 3.1 was used to extract features mentioned in Section 3.2 of Bitcoin entities. Hardware and software configuration used for data collection is given in Section 3.3.

3.1 Data collection and preprocessing

Bitcoin blockchain dataset in raw form was obtained from VJTI Blockchain lab [4]. This raw data was then converted to CSV files using the blockchain parser [5]. Table 4 shows the three ".csv" files of the processed dataset.

3.2 Feature extraction

Based on the structure of Bitcoin (see Figure 1), features to train the classifier were extracted (see Table 5).

[4]https://www.vjti-bct.in/

[5]https://github.com/pranavn91/blockchain

Table 4. Description of processed dataset

Relation	Attributes		
Output	tx_hash:ID	receiver_address	amount
Inputs	sender_address	tx_hash:ID	amount
Transactions	tx_hash:ID	timestamp	

Table 5. List of Features

Feature symbol	Feature description
T_x	Total transactions in which wallet has participated
B	Current BTC present in the wallet
$T_x{}^{in}$	Total incoming transactions to the wallet
$T_x{}^{out}$	Total outgoing transactions from the wallet
L	Total active life of the wallet
A_w	Total addresses of the wallet
A_v	Average number of incoming transactions received by an address of a wallet
T	Total number of addresses sending BTC to the wallet
R	Ratio of Transaction count and address count. Gives the average number of times an address of the wallet was reused for a transaction.

Features extracted for each entity would allow a classifier to learn the categorization of each wallet into one of the 28 categories. Bitcoin entities were identified using an API [6] [20]. The flowchart for the proposed work is illustrated in Figure 2.

Table 6. Types of Bitcoin entities in dataset

affiliatemarketing	**blackmail**	**bomb**	**bond**
2	53	1	1
criminals	**cybersec**	**darkmarket**	**donations**
1	2	16	48
exchange	**explorer**	**faucet**	**gambling**
88	2	2	35
laundering	**microworker**	**miner**	**mixer**
1	1	3	53
p2plender	**p2pmarket**	**paymentgateway**	**ponzi**
6	1	6	28
pools	**ransomwares**	**scams**	**sextortionist**
8	11	23	57
trading	**Unclassified**	**videosharing**	**wallets**
9	1592	1	8

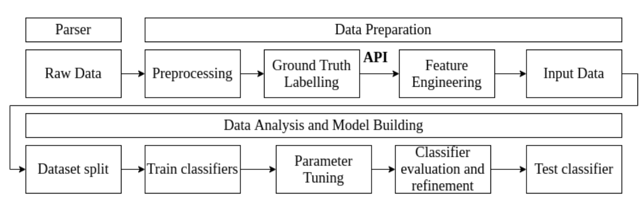

Figure 2. Flowchart of proposed work

3.3 Experimental setup

Experiments were performed on a single core 1 TB Intel(R) Xeon(R) Silver 4114 CPU@2.20GHz.

[6]https://github.com/pranavn91/blockchain/blob/master/walletexplorer-api

4 Classification of Illicit entities in Bitcoin

The mathematical model of the proposed classifier is given in Section 4.1 with steps used for training it listed in Section 5.3.

4.1 Mathematical model

Each training sample (x_i, y_i) of dataset $\mathcal{D}$ is with m features and total n samples are present in the dataset. Hence, $\mathcal{D} = \{(x_i, y_i)\}$ $(|\mathcal{D}| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$. The proposed tree ensemble model uses K additive functions to predict the output.

$$\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \quad f_k \in \mathcal{F}, \tag{1}$$

where

- $\mathcal{F} = \{f(\mathbf{x}) = w_{q(\mathbf{x})}\}(q : \mathbb{R}^m \to T, w \in \mathbb{R}^T)$: set of regression trees
- q: tree structure mapping an $x^{(i)}$ to its leaf index
- T: leaves count in the tree
- f_k: q having leaf weights w
- w_i: score on i^{th} leaf

For each $x^{(i)}$ the decision rules of q classify it into the leaf nodes and calculate the final prediction by $\sum w$ i.e. summing up the score in the corresponding leaves. The obtain the optimal model ϕ, the loss $\mathcal{L}(\phi)$ is minimized by following *regularized* objective.

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k)$$
$$\text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{2}$$

where,

- l: loss function
- $\hat{y}_i$: prediction
- y_i: target
- Ω: regularization term

4.1.1 Optimization.

As the loss function given in Eq. 2 cannot be optimized using standard optimization techniques, the model is trained in an additive manner specified in [21]. Eq. 2 is modified as Eq 3.

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) \tag{3}$$

where,

- $\hat{y}_i^{(t)}$: prediction of the i-th instance at the t-th iteration
- f_t: q having leaf weights w

5 Experimental study

The proposed model in Section 4 was evaluated on dataset described in Table 6 for the experimental study with metrics (Section 5.2).

5.1 Description of experiment

Comparative study was performed of popular non-parametric ML models in literature - Decision trees (DT) and Random Forest (RF) (see Section 3) with proposed model for evaluating classification accuracy on dataset (Table 6).

5.2 Metrics

Given the true positives t_p, true negatives t_n, type I error f_p and type II error f_n obtained from observing $(\hat{y}_i, y_i)$, following metrics were used.

$$Sensitivity(S) = \frac{t_p}{t_p + f_n} \tag{4}$$

$$Specificity(S_p) = \frac{t_n}{t_n + f_p} \tag{5}$$

$$Accuracy(A) = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \tag{6}$$

$$Prevalence(P) = \frac{t_p + f_n}{t_p + t_n + f_p + f_n} \tag{7}$$

5.3 Experimental results and discussion

Dataset was split in ratio 4:1 for training and evaluation. Optimal classifier parameters were identified using random grid search (caret package in R) [k-fold cross-validation, up sampling]. Table 7, and 8 give performance of the classifiers on the train set and Table 9 gives performance of the classifier on the Test set.

Table 7. Evaluating classifier performance on train set

Model	Train									
	logloss		AUC		prAUC		Accuracy		Kappa	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Decision Trees	4.64	0.06	0.72	0.04	0.11	0.02	0.75	0.01	0.42	0.001
Random Forest	1.62	0.07	0.85	0.03	0.20	0.01	0.75	0.33	0.43	0.21
Proposed Model	0.94	0.12	0.90	0.04	0.21	0.01	0.69	0.07	0.4	0.11

Table 8. Evaluating classifier performance on train set

Model	Train			
	Detection rate		Mean Specificity	
	Mean	SD	Mean	SD
Decision Trees	0.02	0.0004	0.98	0.01
Random Forest	0.02	0.01	0.98	0.001
Proposed Model	0.02	0.001	0.98	0.01

Table 9. Evaluating classifier performance on test set

Model	Test				
	Accuracy	95% CI	No Information rate	P-value	Kappa
Decision Trees	0.77	0.7263, 0.8108	0.793	0.874	0.43
Random Forest	0.77	0.7289, 0.8132	0.79	0.85	0.4517
Proposed Model	0.66	0.6122, 0.7071	0.793	1	0.35

Table 10. Time taken by classifiers for training

Model	Type of Timing	User	System	Elapsed
Decision Tree	Everything	357.2	0.29	357.5
	Final	3.66	0	3.66
RF	Everything	2011.6	65.4	20382.2
	Final	17.14	0.38	17.5
Proposed model	**Everything**	130928.1	50.6	16529.4
	Final	821.3	0.3	102.8

The proposed model outperforms other popular tree-based classifiers on 13 out of 14 metrics with the exception of metric "accuracy". The performance of the test set is slightly below par, albeit with a lower standard deviation than other models (see Table 9). As the proposed model has a higher number of parameters, it needs additional training time (see Table 10).

6 Conclusion and Future works

Various services and entities have become active in the Bitcoin space. Although Bitcoin has created new avenues for business models, the pseudo-anonymity has attracted even illegal operations to operate using Bitcoins. Individual differences between licit-illicit are observed in their transactions viz. amount transferred/received, and so on. Using Blockchain features were engineering and extracted to train an ensemble tree-based model to classify licit from illicit. Additionally, a dataset was created of 2059 real-life entities using Bitcoin. These included 28 categories of users who were involved in legal and illegal businesses on Bitcoin. The proposed tree-based classifier could identify 66% of users in the correct category. Additional features need to be engineered to improve accuracy and training time, which are tasks earmarked for future work.

References

[1] Farida Sabry, Wadha Labda, Aiman Erbad, Husam Al Jawaheri, and Qutaibah Malluhi. Anonymity and privacy in bitcoin escrow trades. In *Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society*, pages 211–220, 2019.

[2] Mohamed Rahouti, Kaiqi Xiong, and Nasir Ghani. Bitcoin concepts, threats, and machine-learning security solutions. *IEEE Access*, 6:67189–67205, 2018.

[3] Kentaroh Toyoda, P Takis Mathiopoulos, and Tomoaki Ohtsuki. A novel methodology for hyip operatorsâĂŹ bitcoin addresses identification. *IEEE Access*, 7:74835–74848, 2019.

[4] Chaehyeon Lee, Sajan Maharjan, Kyungchan Ko, and James Won-Ki Hong. Toward detecting illegal transactions on bitcoin using machine-learning methods. In Zibin Zheng, Hong-Ning Dai, Mingdong Tang, and Xiangping Chen, editors, *Blockchain and Trustworthy Systems*, pages 520–533, Singapore, 2020. Springer Singapore.

[5] Sean Foley, Jonathan R Karlsen, and Tālis J Putniņš. Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies? *The Review of Financial Studies*, 32(5):1798–1853, 2019.

[6] Bruno Bogaz Zarpelão, Rodrigo Sanches Miani, and Muttukrishnan Rajarajan. Detection of bitcoin-based botnets using a one-class classifier. In Olivier Blazy and Chan Yeob Yeun, editors, *Information Security Theory and Practice*, pages 174–189, Cham, 2019. Springer International Publishing.

[7] Yan Wu, Anthony Luo, and Dianxiang Xu. Identifying suspicious addresses in bitcoin thefts. *Digital Investigation*, 31:200895, 12 2019.

[8] Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I Weidele, Claudio Bellei, Tom Robinson, and Charles E Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv preprint arXiv:1908.02591*, 2019.

[9] Yining Hu, Suranga Seneviratne, Kanchana Thilakarathna, Kensuke Fukuda, and Aruna Seneviratne. Characterizing and detecting money laundering activities on the bitcoin network. *arXiv preprint arXiv:1912.12060*, 2019.

[10] Hao Hua Sun Yin, Klaus Langenheldt, Mikkel Harlev, Raghava Rao Mukkamala, and Ravi Vatrapu. Regulating cryptocurrencies: a supervised machine learning approach to de-anonymizing the bitcoin blockchain. *Journal of Management Information Systems*, 36(1):37–73, 2019.

[11] L. Nan and D. Tao. Bitcoin mixing detection using deep autoencoder. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pages 280–287, 2018.

[12] Mikkel Alexander Harlev, Haohua Sun Yin, Klaus Christian Langenheldt, Raghava Mukkamala, and Ravi Vatrapu. Breaking bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning. In *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[13] Thai Pham and Steven Lee. Anomaly detection in bitcoin network using unsupervised learning methods. *arXiv preprint arXiv:1611.03941*, 2016.

[14] M. Bartoletti, B. Pes, and S. Serusi. Data mining for detecting bitcoin ponzi schemes. In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 75–84, 2018.

[15] Rebecca S. Portnoff, Danny Yuxing Huang, Periwinkle Doerfler, Sadia Afroz, and Damon McCoy. Backpage and bitcoin: Uncovering human traffickers. In *KDD '17*, 2017.

[16] Dmitry Ermilov, Maxim Panov, and Yury Yanovich. Automatic bitcoin address clustering. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 461–466, 2017.

[17] Jiaqi Liang, Linjing Li, Shu Luan, Lu Gan, and Daniel Zeng. Bitcoin exchange addresses identification and its application in online drug trading regulation. 2019.

[18] Jordi Zayuelas Muñoz. Detection of bitcoin miners from network measurements. B.S. thesis, Universitat Politècnica de Catalunya, 2019.

[19] Francesco Zola, Maria Eguimendia, Jan Lukas Bruse, and Raul Orduna Urrutia. Cascading machine learning to attack bitcoin anonymity. In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 10–17. IEEE, 2019.

[20] A Janda. Walletexplorer. com: Smart bicoin block explorer, 2016.

[21] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, New York, NY, USA, 2016. Association for Computing Machinery.