Supervised Machine Learning Framework for Deanonymization of Transactional Entities in Decentralized Infrastructure Environment

Rohit Saxena (rohit.saxenacse@gmail.com)
Amity University Uttar Pradesh
Deepak Arora
Amity University Uttar Pradesh
Vishal Nagar
Pranveer Singh Institute of Technology

Research Article

Keywords: Blockchain, supervised machine learning, ensemble learning, hyperparameter tuning, randomized search, grid search

Posted Date: December 13th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3734345/v1

Additional Declarations: No competing interests reported.

Supervised Machine Learning Framework for Deanonymization of Transactional Entities in Decentralized Infrastructure Environment

Rohit Saxena[1], Deepak Arora[2], Vishal Nagar[3]

[1,2]Amity University Uttar Pradesh, Lucknow, India
[3]Pranveer Singh Institute of Technology, Kanpur, UP, India

Abstract: A public ledger is used by Bitcoin, a digital currency, to keep track of transactions. The owner of the Bitcoin keeps their identity secret and is identified only by their unique address. This indicates that because Bitcoin offers anonymity, it may be utilized for illicit purposes on a regular basis. This study presents a supervised machine learning approach for predicting anonymous user activities on the Bitcoin Blockchain. As a training dataset to facilitate the user activities classification, we created a labelled dataset with over 4 million samples from exchanges, gambling, pools, and services whose identities and types were disclosed. The primary goal is to classify transactions on the blockchain in order to deanonymize them and distinguish between legitimate and illegitimate ones. On the class imbalanced dataset, we obtained impressive cross-validation (CV) accuracy using the Gradient Boosting, Random Forest, and eXtreme Gradient Boosting with default parameters and hyperparameters. Using Random Forest helped achieve the best cross-validation accuracy on default parameters and hyperparameters obtained using grid search on the class-balanced dataset using the Synthetic Minority Oversampling Technique, while Bagging and eXtreme Gradient Boosting were used on hyperparameters obtained using randomized search. Empirical results show that the recommended model is up to 98% accurate.

Keywords: Blockchain, supervised machine learning, ensemble learning, hyperparameter tuning, randomized search, grid search.

1 Introduction

Cryptocurrency, digital payments, contactless payments, and *e*-commerce have become more popular as a result of the COVID-19 crisis. Cryptocurrencies such as Bitcoin, Tether, Ethereum, Dogecoin, etc. tend to be digital assets that offer secure and verified transactions and the creation of new assets through the use of a decentralised control system and encryption [1]. The cryptocurrency known as Bitcoin first came to light in 2008 [2], [3]. Due to their distinctive features, such as the lack of centralised control, guarantee against ambiguity, and substantial level of anonymity, cryptocurrencies at large and specifically Bitcoin, have recently attracted greater interest from scientists from diversified disciplines of academia [4], [5] as well as practitioners. Bitcoin has been stated to be the perfect payment mechanism for illegitimate activities because of its comparatively high degree of anonymity. A well-known illustration in this context is the closure of the illicit drug marketplace Silk Road [6]. In addition, several publications [7], [8] allege that Bitcoin has previously been used for ransomware, theft, scams, and the funding of terrorism. Financial regulators, law enforcement organisations, intelligence agencies, and businesses that use the Bitcoin blockchain for transactions have developed a watchful eye towards technical advancements, business challenges, and social acceptance of Bitcoin [4], [9]. This research seeks to offer an enhanced comprehension of the diverse Bitcoin transactions in order to more effectively educate administrative and institutional factors connected to regulatory and compliance with laws. We achieve this by utilizing supervised machine learning's potential to de-anonymize the Bitcoin ecosystem in order to assist in the identification of high-risk counterparties and likely cybercriminal activities [10]. Legal constraints (such as those related to safeguarding against money laundering protocols) or reputational risk considerations may result in adverse outcomes for organizations when they communicate with counterparties who pose a high risk on the Bitcoin network. Governments face a significant challenge when it comes to the illegitimate adoption of Bitcoin for cybercrime, terrorism financing, and money laundering. In such circumstances, disclosing the true identities of the people in question would be ethically acceptable and permitted by law, but it could prove technically difficult, according to a common

misconception about how resilient anonymity is in the Bitcoin ecosystem. Nevertheless, earlier studies [11], [12] have proven that it is possible to classify Bitcoin addresses based on user activities and connect these classes to real-world individuals. The widely held belief that the identities of users and hence their addresses are secured when using Bitcoin is refuted by these results of research. Here, we refer entities by Bitcoin user addresses. In this work, we have examined the blockchain's transactions & deanonymized them using supervised ML approaches.

1.1 Motivation

The primary intent of this research is to enhance transparency in the Blockchain ecosystem and encourage consumers to accept Bitcoin as a legitimate mode of payment. This research will contribute towards economic growth without resorting to illegal tactics. The results will also be useful to lawmakers seeking data-driven sources for projections of the Bitcoin landscape, enterprises seeking compliance and effective risk evaluation of Bitcoin transactions, and law enforcement agencies looking to analyze and investigate Bitcoin addresses associated with illegal activities. Additionally, this study may assist in identifying illicit user addresses by linking such addresses to suspicious activities.

1.2 Contribution

The goal of the research is to use supervised ML clasiifiers to achieve deanonymization of blockchain transactions. The noteworthy additions to this work are as follows:

- Scrapping the dataset samples from various repositories, preparing labelled dataset samples, cleaning them, and then normalizing them.
- Resolve class imbalance problem using SMOTE and Weighted Mean (via, Weight of the User Activities).
- Assessing the percentage of illegitimate and legitimate transactions in the Bitcoin Blockchain.
- Optimize the classification accuracy by utilizing hyperparameters.
- Use supervised machine learning techniques to deanonymize Bitcoin transactions.

1.3 Organisation of the Paper

The remainder of this paper is structured as follows: There is related work in Section 2. Section 3 presents a formulation of the problem. Section 4 deals with the preliminaries for the proposed work. The proposed framework is presented in Section 5, and the analysis and results are presented in Section 6. In Section 7, the conclusion and the future scope are presented.

2 Related Work

This section provides a literature review of numerous approaches. In relation to current research into deanonymizing blockchain transactions in the broader context of cryptocurrencies, we will assess the state of the art at the moment. Starting with a quick assessment of related studies from the perspective of information systems, we'll go on to a summary of various legal groups' efforts to establish a regulatory framework for cryptocurrencies. We will also briefly outline the most recent advancements in de-anonymizing cryptocurrency entities.

2.1 Information Systems & Cyber Threat Intelligence

The current research that is pertinent to our work can be separated into two groups from the viewpoint of Information Systems (IS). The first section is empirical and relates to *Cyber Threat Intelligence*, and covers the literature that has been published on the subject; the second is conceptual and presents the literature in the context of *Blockchain and Cryptocurrencies*.

The studies on anonymity, identifying dishonest traders, identifying cybercrime activities, and identifying financial fraud in relation to electronic markets and commerce channels are reviewed in [12], [13], [14]. A meta-learning framework that improved financial fraud detection is presented in [15] using a design science approach. [16] established the Writeprint technique for identifying anonymous traders, and they proposed the use of stylometric analysis to detect traders on the internet based on the writing style traces present in the posted feedback comments. With the goal of discovering possible long-term and important members, Benjamin et *al*. [13] suggested using a computational method to analyse the Internet relay chat (IRC) groups of cybercriminals. To examine IRC participation by hackers and better understand the key behaviours they display, the authors applied the extended Cox model. By utilising an automated and ethical web, data, and text mining approach for gathering and analysing massive volumes of dangerous hacker tools from significant, global underground hacker networks, Samtani et *al.* [14] made a significant contribution to the development of a cyber threat intelligence framework. The authors discovered numerous openly accessible harmful elements employing this framework, including keyloggers, crypters, web attacks, and database attacks. Recent breaches against companies like the Office of Personnel Management may have been brought on by some of these technologies. Abbasi et *al*. [16] suggested a novel method for identifying phishing websites employing a design science approach. The suggested genre tree kernel technique uses fraud cues connected to differences in intent between genuine and phishing websites, displayed via genre composition and design structure, leading to increased anti-phishing capabilities through the use of a genre theoretic perspective. Several tests were run on a testbed made up of numerous genuine and phishing websites in order to assess the genre tree kernel approach. Abbasi et *al.* [17] suggested the establishment of a new class of statistical learning theory-based fraudulent website identification systems in response to these shortcomings. They created a prototype system to show the potential utility of this class of technologies using a design-science approach. On a test bed of 900 websites, the authors ran a number of trials evaluating the suggested approach against several other fake website identification techniques.

2.2 Blockchain and Cryptocurrencies

The research regarding blockchain-based technologies has been carried out by Beck et *al*. [15], who predicted that in the near future, distributed ledger technology would be made available to organizations, enabling them to adopt solutions. These technologies will make it easier for decentralized autonomous organizations to emerge because they will allow organizations to manage contracts and transactions independently of one another without the need for separate legal bodies [18].

Without having to provide their personal information, end users may generate pseudo-anonymous financial transactions using Bitcoin. This is accomplished by creating a user-generated pseudonym, often known as an “address”. On the one hand, users who respect their privacy were drawn to the seeming anonymity and convenience of setting up pseudo-anonymous financial transactions; on the other hand, hackers who wish to exploit it for ransomware and other illicit activities were drawn to it as well [19]. This study showed that mapping Bitcoin addresses to IP data allows for the identification of the address owners through real-time transaction relay traffic tracking. By simulating user actions and transactions on the Bitcoin Blockchain, simulation experiments were used to analyse the privacy guarantees of the cryptocurrency and reveal that, even when users take the privacy precautions that Bitcoin recommends, it is still possible to discover almost 40% of users' profiles [20]. Numerous researchers also emphasized the shortcomings of the Bitcoin Blockchain and looked forward to a few of the alternative cryptocurrencies in addition to ideas for enhancements and/or brand-new approaches to provide users with anonymity. A protocol allowing anonymous transactions in Bitcoin and other cryptocurrencies and depends on technology widely used by mixing services has been disclosed by some of the research's in-depth examinations of Bitcoin's technological workings. These analyses highlighted technical faults in the system and offered suggestions for how to repair them. [21]. A noteworthy scientific effort in this

field is the development of Zerocash, a Bitcoin substitute with zero-knowledge proofs, and other ZKP uses for IMoT [22], as well as theoretically feasible privacy-enhancing overlays for Bitcoin [23].

Existing literature has very few studies of anonymity attacks on blockchain transactions. An ML-based approach to attacking blockchain bitcoin transactions is addressed in [24]. The authors employed an entity characterization strategy to challenge Bitcoin anonymity using an ML model with a suitable number of input attributes that were directly extracted from Bitcoin blockchain data, such as entity and address data, as well as developed *via* first motif and second motif principles. For several crucial Bitcoin entity classes, this model showed remarkable categorization results.

By utilising supervised ML to forecast the characteristics of as-yet-unidentified entities, Harlev et *al*. [25] developed a way to decrease the anonymity of the Bitcoin blockchain. They developed classifiers that could distinguish between ten categories using a training set of 434 entities (with 200 million transactions) whose identity and kind had been made public. Their main finding was an estimation of the type of unknown creature.

In the Bitcoin blockchain dataset, authors [26] performed the classification and prediction of the proportion of user activities that are lawful and unlawful. Approximately 27 billion samples, separated into nine user behaviours, five of which were unlawful while the other four were lawful, made up the dataset. To predict CV accuracy, the authors employed ensemble learning. Hyperparameter tuning was done to determine the ideal parameters for the most effective classification and prediction, which helped to increase the cross-validation accuracy. The classification of Ethereum blockchain addresses [27] using supervised machine learning models, encompassing linear, non-linear, and ensemble learning models that take into account non-malicious and non-malicious activities. The outcomes also demonstrate that it is easy to identify malicious users' Ethereum blockchain addresses. However, accuracy and efficiency are issues with ML-based approaches to deanonymizing blockchain transactions.

3. Problem Formulation

Significant challenges arise due to the illegitimate utilization of Bitcoin for cybercrime, terror funding, etc. A widely believed notion regarding the resilience of anonymity in the Bitcoin ecosystem states that while it would be ethically and legally right to publicly disclose the identities of the participants in these circumstances, nevertheless might be practically impossible. Additionally, the great majority of clusters across the Bitcoin Blockchain are still unclassified. Hence, it's required to classify and deanonymize the Bitcoin addresses used in illegitimate transactions based on user activities.

4. Preliminaries

This section outlines the primary concepts that motivate our research on utilising supervised ML to deanonymize the transaction carried out using the Bitcoin blockchain. In the context of decentralised networks like blockchain, we first provide fundamental principles and a brief discussion on the anonymity and deanonymity of blockchain transactions. We first briefly explore the fundamental concepts underlying blockchain technologies. Lastly, the fundamental principles behind the numerous supervised ML techniques used in this study will be covered.

4.1 Anonymity

Anonymity is a feature that has probably been crucial to the widespread adoption of cryptocurrencies. The ability for users to generate a limitless number of anonymous Bitcoin addresses for use in their Bitcoin transactions provides the foundation for anonymity on the Bitcoin network. The Bitcoin ecosystem is under surveillance, and the adversary can have access to the transactions coming from that address or its pseudonym. The Bitcoin ecosystem is an anonymity zone $\mathcal{B}$.

The level of anonymity of a transaction is the inability of the adversary to pinpoint the source address of the transaction $\mathcal{T}$ (the anonymity set).

This anonymity set $\mathcal{T} \subseteq \mathcal{T}_{total}$ with $\mathcal{T}_{total}$ being the total number of transactions in $\mathcal{B}$. The entropy of the anonymity set is the measure of the anonymity of a transaction in the set.

If all the addresses can be the source of transactions with equal probability, then the probability p_i that the address $\mathbb{A}_i$ under observation is the target,

$$p_i = P_r(\mathbb{A}_i = \mathbb{A}),\ \forall\ i\ \in\ \mathcal{B}\ i\ and\ \sum p_i$$

The entropy [28], [29] of the distribution of the anonymity set is:

$$H(p) = -\sum p_i \log_2 p_i$$

For a transaction to be completely anonymous, all addresses involved should have an equal chance of being identified as the source. This means that the probability of any specific address, $\mathbb{A}_i$, being the source should be the same, represented by the variable p_i

$$p_i = \tfrac{1}{\mathbb{A}}$$

Following the definition of the level of anonymity given by Wu and Bertino [30], we have

$$\mathbb{A}_i = 1 - {}^{1}/_{|\mathbb{A}|}$$

With entropy $H(p) = -\log_2 \mathbb{A}$

In the context of the Internet and electronic communication, Froomkin [31], [32] proposed traceable anonymity, untraceable anonymity, traceable pseudonymity, and untraceable pseudonymity as the four unique kinds of anonymity/pseudonymity that may be used.

Blockchain transaction deanonymity may also be referred to as traceable anonymity. When information is conveyed via traceable anonymity, the sender's identity is hidden from the recipient. The sender's information is only accessible to the agent or system acting as the communication's intermediary.

4.2 Supervised Machine Learning

The labelled dataset has been subjected to supervised ML techniques in order to identify patterns in the Bitcoin transaction data. This section provides a basic introduction to the numerous algorithms employed in our method as well as a brief explanation of the main notion underlying the machine learning algorithm. In the statistical literature, responsive variables—often referred to as outcomes or targets—are computed to fit a prediction model, and the underlying function that describes the relationship between explanatory variables—often referred to as predictors—is computed to fit a prediction model[33]. Consider that there are *n* number of training examples(as shown in(1)):

$$(x_1, x_2, x_3, \ldots x_n) \qquad \text{----- 1}$$

where *x* represents the feature vector, x_i for each individual feature component, and y stands for the responsive variable. The algorithm used in supervised ML seeks a function in which the input is *x* while the output is *y*. A multi-class classification issue, which is what the learning problem at hand recommends, can categorize Bitcoin user addresses in accordance with the actions of four different classes. This function(2) can be represented as[34]:

$$y = f(x) + e \qquad \text{----2}$$

In the unlabelled dataset, supervised ML techniques are highly effective at discovering patterns and making predictions about categories that have not yet been assigned labels [34]. In this study, we employed supervised ML techniques to predict a subject's category. Using this method, we also wish to determine the scope of unethical and cybercriminal activities within the Bitcoin blockchain ecosystem.

4.3 Blockchain Transactions

In the Bitcoin system, transactions amongst Bitcoin accounts are used to make payments. The movement of bitcoin from source addresses to destination addresses is indicated by a transaction. In a transaction, source addresses are called input addresses, while destination addresses are called output addresses. There can be one or more input addresses and one or more output addresses in a single transaction. For every input address, a transaction specifies how much bitcoin is to be transferred exactly. The output addresses, which show the total quantity of bitcoins that would be transferred to each account, are equivalent in this regard. The sum of the input addresses (the money's source) must be larger than or equal to the sum of the output addresses (the money's destination) in order for consistency. Additionally, the Bitcoin protocol mandates that input addresses must spend the exact amount of a prior received transaction; as a result, any input address can unmistakably identify the transaction index in which the bitcoins were received while participating in a transaction. In order to establish his legitimacy as the account owner, the owner of the input addresses should lastly use his private keys to execute a digital signature[35]. This paper presents a concrete instance of employing supervised learning for the deanonymization of blockchain transactions.

5 Proposed Framework

This section presents the proposed supervised machine learning framework for the classification and deanonymization of blockchain transaction. The proposed framework comprises data collection & preparation, data preprocessing, class balancing, obtaining hyperparameters using randomized search and grid search, classification using default parameters and hyperparameters, and result analysis for finding the best model for classification and deanonymization as shown in Fig. 1.

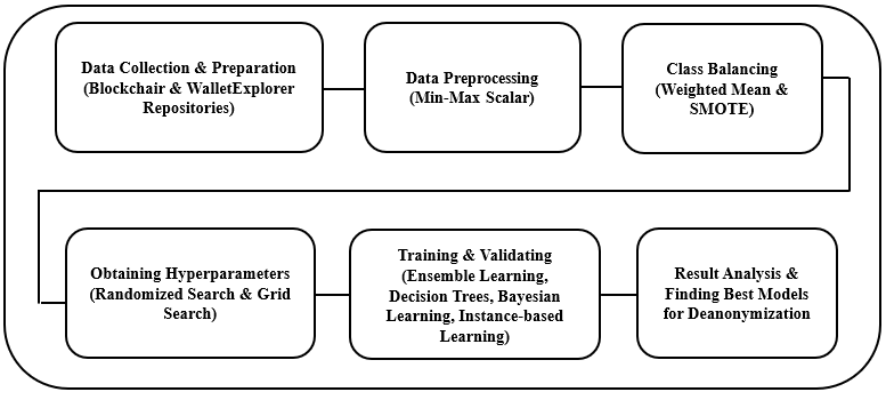


Fig. 1 Proposed Framework for Classification and Deanonymization

The methodology, which illustrates how each phase of the proposed framework progress, is displayed in Fig.2.

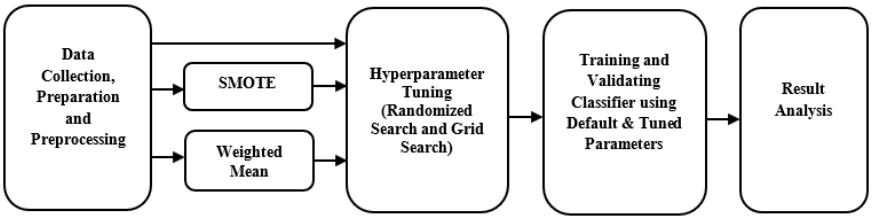


Fig. 2 Progression of Proposed Framework

5.1 Data Collection, Preparation and Preprocessing

The process of data collection, preparation and preprocessing to obtain the processed dataset samples ready for training and validation is depicted in Fig 3.

Data Collection: The dataset employed for the research has been collected from the *Blockchair* [36] and *WalletExplorer*[37]. repositories *Blockchair* is a blockchain explorer that serves as a search engine for numerous different blockchains, such as Bitcoin, Ethereum, Litecoin, Ripple, etc. In addition to carrying out exhaustive searches on the blockchains, one can also filter and arrange blocks, transactions, and data within them using a variety of other parameters. There are a total of 22 features in every transaction obtained from this repository. Some of them are transaction hash, block_id, timestamp, size, weight, version, fee_used, etc. *WalletExplorer* is a Bitcoin blockchain explorer that offers an easy way to view public blockchain data, *i.e*., Bitcoin transactions corresponding to wallets. This repository provides sample datasets of the wallets mapped to the transactions for exchanges, pools, services, and gambling. It has a total of 7 features, and some of them are transaction hash, timestamp, received amount, wallet address of the sender, sent amount, wallet address of the receiver, and wallet balance.

***Data Preparation*:** In this phase, we have prepared the dataset with features, followed by preprocessing the dataset [36], [37]. We collected the Bitcoin blockchain's transaction history for the months of January, February, and March of the year 2023 from the Blockchair repository. The dataset that is available at *Blockchair* is unlabelled; however, those available at WalletExplorer are *labelled*. The transaction hash is a common feature in both datasets. Therefore, it has been used to merge the two datasets. As a result, a feature-enriched, labelled dataset with 29 features is obtained.

***Data Preprocessing*:** In data preprocessing, the dataset is then cleaned to eliminate the samples comprising *null* and *inf* values because the ML models cannot handle such values. String values are transformed into integer values by applying an encoder library in order to make the data suitable for ML models. The dataset samples are then normalised to the same scale using the feature-based min-max scaler. *Min-max* normalisation is one of the most frequently used data normalisation strategies. The minimum and maximum values for each feature are both set to 0, and all other values are set to a decimal between 0 and 1. The min-max normalisation technique is shown below[38]:

$$v' = \frac{[v - min(p)] * [new\_max(p) - new\_min\ (p)]}{[max(p) - min(p)]} + new\_min\ (p) \tag{3}$$

where min(p) = minimum value of the attribute p, max(p) = maximum value the of attribute p. The new_max(p) and new_min(p) denote the maximum and minimum values of the range, or the needed boundary values, respectively.

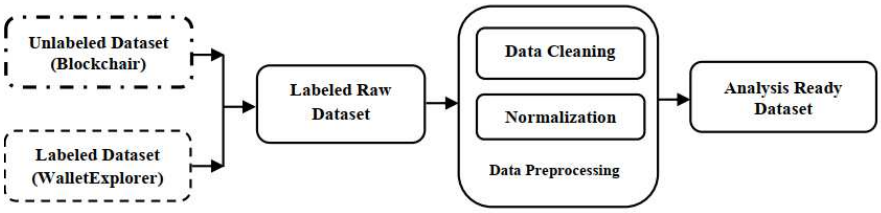


Fig. 3 Data Collection, Preparation and Preprocessing

5.2 Class Balancing

The data preparation method produced 427,625 transactions grouped among four categories, including *exchange, pool, services,* and *gaming*, which are included in the analysis-ready dataset samples that were obtained. It is evident, from Table 1, that the dataset samples' proportion of user activity is unbalanced. As shown in Table 1, the dataset samples of user activities, *pool, services,* and *gambling* account for 12.95%, 7.98%, and 0.53% of the total dataset samples, respectively, while *exchange* accounts for 78.53%. A class imbalance issue resulted from the dataset samples where *gambling, services,* and *pools* remained unidentified and under-sampled. Due to the unexplained nature of their behaviour, which encourages the deployment of privacy-enhancing measures, many classes continue to be undersampled. They use peeling chain mixing, which combines a customer's payments into a single address, to hide transactions. The remaining coins (change) are then sent to an address for recent changes, and the system begins sending extremely small amounts of money from that address to various other services. This procedure is repeated unless the last coin is expended. This generates a large number of change addresses, which makes it nearly impossible to identify and group addresses and hides the true source of a transaction. Performance can be attained by the prediction model; it has been demonstrated by Chawla et *al*. [39] by improving the sensitivity of the classifiers to the minority classes by increasing the sample size. The categories of the dataset samples must be balanced to a great extent. ML algorithms frequently create subpar classifications if faced with unbalanced datasets. The classification result is unexpected if the event that was predicted falls to the majority class or the minority class in any unbalanced data set. Samples from the training and testing datasets are distributed at 60% and 40%, respectively. Only the samples from the training dataset were used for class balancing, and the samples from the testing dataset were retained separately. The class balancing issue is resolved using Weighted Mean[41] and the Synthetic Minority Oversampling Technique(SMOTE)[39].

Table 1. Categorization of Samples

User Activities	**No. of Transactions per User Activities**	**Percentage-wise share of Activities (%)**
Exchanges	335,847	78.53
Pool	55,390	12.95
Services	34,124	7.98
Gambling	2,254	0.53
Total	**427,625**	

- *Synthetic Minority Oversampling Technique (SMOTE)*

The fundamental concept is to interpolate between a number of nearby minority class examples to create new minority class samples. As a result, the overfitting issue is avoided, and the boundaries of decision-making over the minority class are expanded into the space of the majority class [40]. This method operates in "feature space" rather than "data space," generating synthetic instances of samples in a less application-specific manner. By taking each minority class sample and inserting synthetic samples along the line segments connecting any or all of the k minority class nearest neighbours, the minority class is oversampled. Randomly selected neighbours from the k-nearest neighbours are determined by the volume of oversampling necessary [39].

- *Weighted Mean*

The Weight of user activities is taken into account when calculating the Weighted Mean[41]. It is utilised to decide whether to over- or under-sample the dataset as needed. The resulting dataset samples have a balanced distribution of user activities.

5.3 Hyperparameter Tunning

Non-parametric models' corresponding hyperparameters need to be optimized in order to achieve stable performance results. Additional focus on this crucial stage should be given because default hyperparameter settings cannot provide the best performance of machine learning techniques [42]. Grid search is one of the most basic techniques since it evaluates each potential combination of the discrete parameter spaces that are provided. Continuous parameters have to first be discretized. Another method is randomized search, which selects hyperparameter values at random (for example, from a uniform distribution) from a predetermined hyperparameter space [43].

- *Randomized Search*
  The randomized search approach evaluates the hyperparameters while selecting the best results [44], [45]. It then randomly attempts a number of specified combinations. Randomized search is effective and effectively handles data with several dimensions [46].

- *Grid Search*
  Grid search, in actuality, is an in-depth search based on subsets, whose hyperparameters are established by employing a lower limit, an upper limit, and the number of steps [47]. The grid technique will thoroughly investigate all alternatives by creating a grid, which will then be assessed to determine which grid offers the best value [48]. Data execution correctness is a benefit of the grid search approach [49].

5.4 Classification using Default Parameters & Hyperparameters

This phases includes ensemble learning (adaptive boosting, bagging, extra trees, gradient boosting, extreme gradient boosting), bayesian learning(gaussian naïve bayes), decision tree(classification and regression tree) and instance-based learning(k-nearest neighbour), has been employed for the classification in the research work. The classifiers are trained and tested using Scikit-learn libraries. The user activities are used as the y-axis and the remaining features as the x-axis for training the model and predicting the user activities, *i.e.*, y.

6. Experimental Evaluation

This section evaluates the efficacy of the proposed supervised machine learning framework used to deanonymize Bitcoin blockchain transactions. The experiments were conducted in Python 9.10.2 and Visual Studio Code 1.79.2. The processor employed for this work is the Intel(R) Core(TM) I9-10900, 2.81 GHz, with 32 GB of RAM. The user activity exchange dominated, with the highest share among the available classes in the dataset samples, followed by the pool, services, and gambling. A total of 4.0 million of data are used for the experiment. Fig. 4 depicts the proportion of various user activities from the dataset gathered.

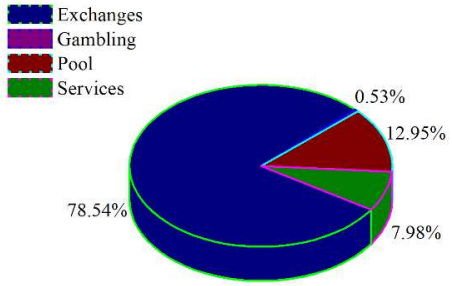


Fig. 4 Percentage-wise Share of Dataset Samples of Blockchain Transactions

This uneven proportion of dataset samples led to the issue of class imbalance. To overcome this issue, we employed, SMOTE [39], and Weighted Mean [41]. The dataset samples after the balancing of classes of user activities are given in Table 2.

Table 2. Balanced Dataset

User Activities	***Generated Dataset Samples for Training***		
	Unbalanced	**SMOTE**	**Weighted Mean**
Exchanges	335847	201580	64143
Gambling	55390	201580	64143
Pool	34124	201580	64143
Services	2254	201580	64143

6.1 Results

We have evaluated the classification models from *ensemble learning*, *bayesian learning*, *instance-based learning* and *decision trees* that were initially trained and tested using a 60:40 ratio, respectively. The models are *AdaBoost (AB), Bagging (BG), Extra Trees (ET), Gradient Boosting(GB), Random Forest (RF), eXtreme Gradient Boosting (XGB), Classification And Regression Trees(CART), Gaussian Naive Bayes (GNB), K-Nearest Neighbour (KNN)*. We have evaluated the proposed model in three scenarios: an imbalanced dataset, balanced dataset using SMOTE, and balanced dataset using Weighted Mean. The metrics considered is Cross Validation(CV) Accuracy with non-parameteric classification, and hyperparameteric classification, *i.e.*, randomized search and grid search. The hyperparameters for the supervised machine learning classifiers employed in the research are shown in table 3 as those found using randomized search, and table 4 as those obtained using grid search.

Table 3. Hyperparameters Obtained using Randomized Search

Supervised ML Classifiers	Hyperparameters Obtained using Randomized Search
AB	{'n_estimators': 100, 'learning_rate': 1.03, 'algorithm': 'SAMME'}
BG	{'n_estimators': 100, 'max_samples': 20, 'max_features': 10}
ET	{'max_depth': 9, 'max_features': 'sqrt', 'min_samples_split': 6, 'n_estimators': 10}
GB	{'learning_rate': 0.456, 'max_depth': 7, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 50, 'subsample': 0.701}
RF	{'n_estimators': 140, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': None, 'bootstrap': False}
XGB	{'min_child_weight': 1, 'max_depth': 5, 'learning_rate': 1, 'gamma': 0.3, 'colsample_bytree': 0.4}
GNB	{'var_smoothing': 1e-06}
CART	{'max_depth': 9, 'min_samples_split': 6}
KNN	{'n_neighbors': 1}

Table 4. Hyperparameters Obtained using Grid Search

Supervised ML Classifiers	Hyperparameters Obtained using Grid Search
AB	{'learning_rate': 0.01, 'n_estimators': 200}
BG	{'base_estimator__max_depth': 8, 'base_estimator__min_samples_leaf': 2, 'max_features': 1.0, 'max_samples': 1.0, 'n_estimators': 50}
ET	{'max_depth': None, 'max_features': 'auto', 'min_samples_split': 2, 'n_estimators': 100}
GB	{'learning_rate': 0.1, 'max_depth': 7, 'n_estimators': 100}
RF	{'bootstrap': True, 'max_depth': None, 'max_features': 'auto', 'n_estimators': 14}
XGB	{'subsample': 0.978, 'n_estimators': 500, 'min_child_weight': 9, 'max_depth': 4, 'learning_rate': 0.360, 'colsample_bytree': 0.845}
GNB	{'var_smoothing': 1e-07}
CART	{'max_depth': None, 'min_samples_split': 2}
KNN	{'n_neighbors': 1}

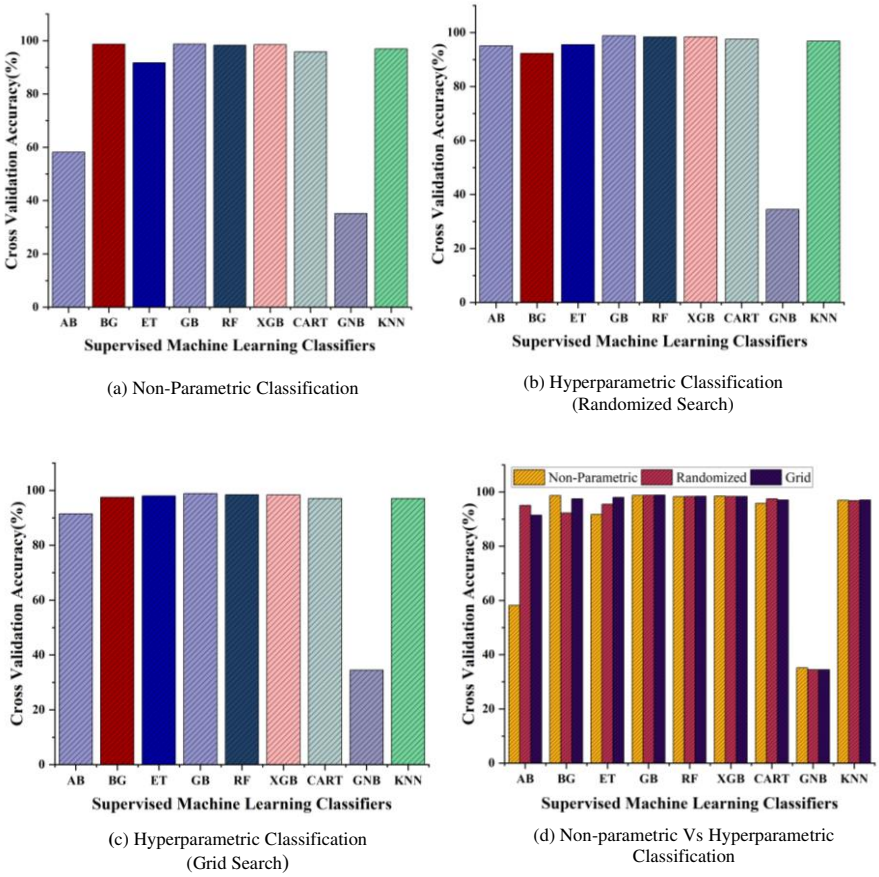
Scenario 1: Performance over Dataset Samples with Imbalanced Classes

Table 5 displays the cross-validation accuracy-based classification report for the supervised machine learning classifiers employed in this research. It contrasts the classifiers with and without the usage of hyperparameters across the samples from an class imbalanced dataset.

Table 5: Comparison of Classifiers over Class Imbalanced Dataset Samples

Supervised ML Classifiers	Cross Validation Accuracy (%)		
	Non-Parametric	Randomized Search	Grid Search
AB	58.20	95.10	91.50
BG	98.70	92.30	97.49
ET	91.70	95.50	98.00
GB	98.73	98.79	98.88
RF	98.30	98.40	98.47
XGB	98.50	98.39	98.40
CART	95.85	97.50	97.10
GNB	35.16	34.50	34.50
KNN	96.92	96.80	97.10

The graphs in Fig. 5 displays the performance of classifiers on class imbalanced dataset samples.



(a) Non-Parametric Classification

(b) Hyperparametric Classification (Randomized Search)

(c) Hyperparametric Classification (Grid Search)

(d) Non-parametric Vs Hyperparametric Classification

Fig. 5 Performance over the Dataset Samples with Class Imbalanced Dataset Samples

A detailed description of accuracy is as follows:

- ***Non-Parametric Classification*:** As shown in Fig. 5a, the best and most accurate classifiers are *GB*, and *BG* with a CV accuracy of 98.73%, and 98.70%. *XGB* and *RF* are next, with CV accuracies of 98.50% and 98.30%, respectively. With CV accuracy of 58.20% and 35.16%, *AB* and *GNB* are the least accurate classifiers.

- ***Hyperparametric Classification using Randomized Search*:** Using the hyperparameters that have been obtained through the randomized search technique, the classification accuracy of *AB*, *ET*, and *CART* has been optimized from 58.2% to 95.10%, 91.8% to 95.48%, and 95.83% to 98.37%, respectively. There has been a modest improvement in *RF* and *XGB*. As depicted in Fig. 5b, the *GB*, *RF*, and *XGB* are the most accurate with CV accuracy of over 98%, while the *GNB* is the least accurate.

- ***Hyperparametric Classification using Grid Search***: It can be clearly observed from Fig. 5c that the grid search technique yielded hyperparameters that were used to optimize the classification accuracy of *AB*, *ET*, and *CART*. These values increased from 58.20% to 91.50%, 91.8% to 98.33%, and 95.85% to 97.10%, respectively. As before, *GB, RF, XGB, and ET* are the most accurate with CV accuracy over 98%, and *GNB* is the least.

- ***Randomized Search Vs Grid Search:*** Based on the results shown in Figure 5d, the classification performance of *AB*, *ET*, and *CART* has been optimized using hyperparameters obtained using randomized search and grid search. The cross-validation accuracy of *AB* and *CART* is significantly better with hyperparameters obtained using randomized search, while the cross-validation accuracy of *ET* is more optimized with hyperparameters obtained using grid search. Slight improvements and declines can be observed with the other classifiers.

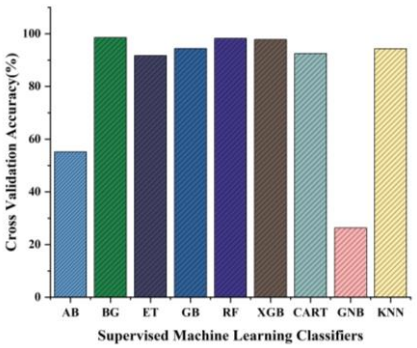
Scenario 2: Performance over Class Balanced Dataset Samples Oversampled using SMOTE

Table 6 presents the classification report for the supervised machine learning classifiers used in this research, comparing the classification with and without the employment of hyperparameters on class balanced dataset samples obtained by oversampling performed using SMOTE.

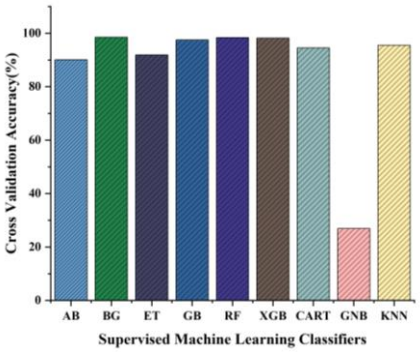
Table 6: Comparison of Classifiers over Class Balanced Dataset Samples Oversampled using SMOTE

Supervised ML Classifiers	Cross Validation Accuracy (%)		
	Non-Parametric	Randomized Search	Grid Search
AB	55.20	90.14	59.53
BG	98.51	98.53	94.57
ET	91.79	91.99	98.25
GB	94.32	97.57	98.83
RF	98.23	98.37	99.31
XGB	97.84	98.22	96.36
CART	92.18	94.53	93.66
GNB	26.66	27.92	26.75
KNN	94.32	95.53	95.55

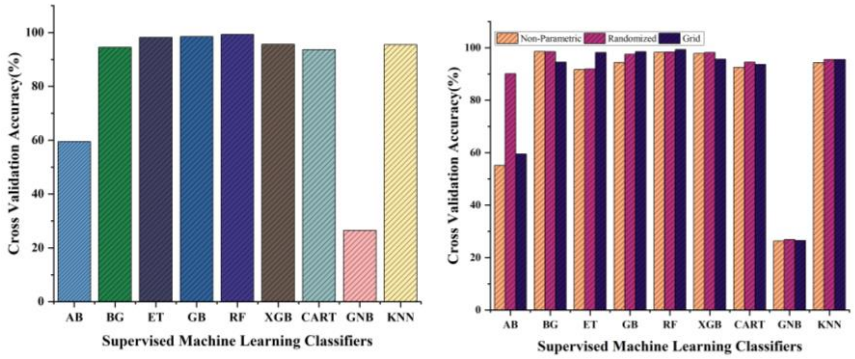
The performance of classifiers on over the dataset samples oversampled using SMOTE to balance the classes is displayed in Fig 6.



(a) Non-parametric Classification



(b) Hyperparametric Classification (Randomized Search)



(c) Hyperparametric Classification (Grid Search)

(d) Non-parametric Vs Hyperparametric Classification

Fig. 6 Performance over Class Balanced Dataset Samples Oversampled using SMOTE

A detailed description of accuracy is as follows:

- ***Non-Parametric Classification*:** With a CV accuracy of 98.51%, *BG* is the most accurate classifier, as seen in Fig. 6a. With CV accuracies of 98.23%, 97.84%, 94.32%, and 94.32%, respectively, *RF*, *XGB*, *GB*, and *KNN* are the next in order. Classifiers *AB* and *GNB* have the lowest CV accuracy, at 55.20% and 26.66% respectively.

- ***Hyperparametric Classification using Randomized Search***: The CV accuracy of *AB*, *GB*, *CART*, and *KNN* has been optimized from 55.2% to 98.53%, 94.32% to 97.57%, 92.18% to 94.53%, and 94.32% to 95.53%, respectively, using the hyperparameters that were obtained by the randomized search technique. For *RF* and *XGB*, there has been a little improvement. The most accurate are the *BG*, *RF*, and *XGB*, with CV accuracy above 98%, while the least accurate is the *GNB*, with CV accuracy under 30%, as shown in Fig. 6b.

- ***Hyperparametric Classification using Grid Search***: Fig. 6c makes this evident: the grid search method generated hyperparameters that were utilized to optimize the classification accuracy of *AB, ET, GB, RF, CART*, and *KNN*. From 55.20% to 91.50%, 91.79% to 98.25%, 94.32% to 98.33%, 98.23% to 99.31%, and 92.18% to 93.66% and 94.32% to 95.55%, respectively, were improvements in the accuracies. *GNB* has the lowest accuracy- below 30%, whereas *RF* has the highest accuracy—over 99%.

- ***Randomized Search Vs Grid Search**:* Based on the findings displayed in Figure 6d, hyperparameters derived from grid search and randomized search have been used to optimize the classification performance of *AB*, *ET*, *GB*, *RF*, *CART*, and *KNN*. While the cross-validation accuracy of *ET*, *GB*, and *RF* is more optimal with hyperparameters obtained using grid search, the cross-validation accuracy of *AB*, *XGB*, and *CART* is much better with hyperparameters obtained using randomized search.

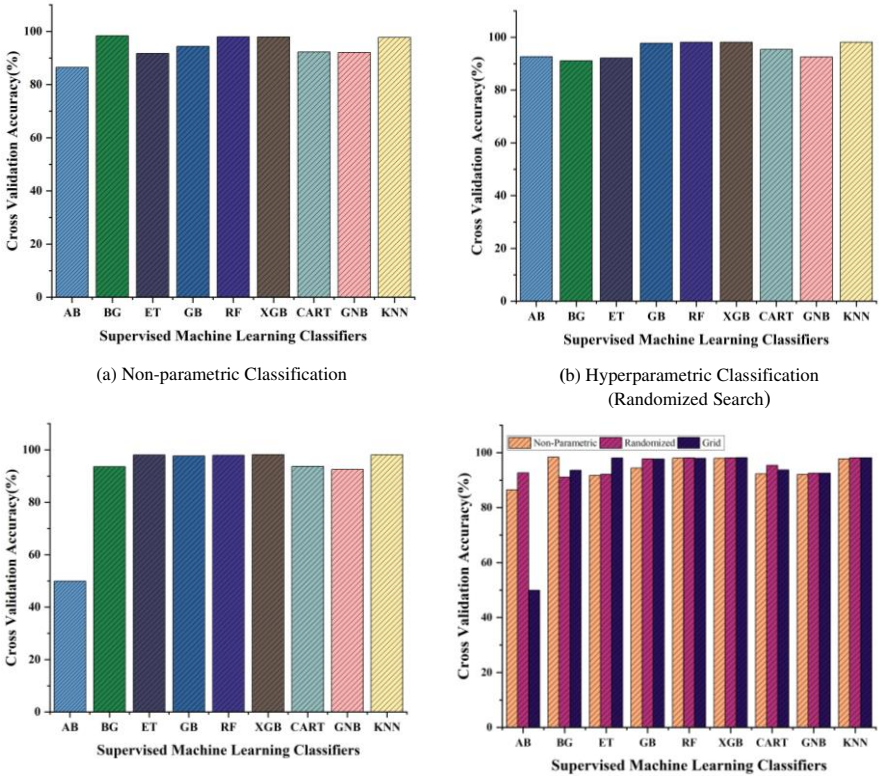
Scenario 3: Performance over Class Balanced Dataset Samples Oversampled and Undersampled using Weight of the User Activities(Weighted Mean)

The classification report for the supervised machine learning classifiers utilized in this study is shown in Table 7, which contrasts the classification on class-balanced dataset samples produced by oversampling and undersampling using Weight Mean with and without the use of hyperparameters.

Table 7: Comparison of Classifiers over Class Balanced Dataset Samples Oversampled and Undersampled using Weighted Mean

Supervised ML Classifiers	Cross Validation Accuracy (%)		
	Non-Parametric	Randomized Search	Grid Search
AB	86.53	92.67	49.89
BG	98.37	91.10	93.58
ET	91.72	92.16	98.08
GB	94.41	97.78	97.68
RF	97.99	98.42	97.98
XGB	97.97	98.12	98.22
CART	92.30	95.39	93.74
GNB	92.10	92.52	92.52
KNN	97.75	98.12	98.12

The performance of classifiers on over the dataset samples oversampled and undersampled using the Weighted Mean to balance the classes is displayed in Fig. 7.



(a) Non-parametric Classification

(b) Hyperparametric Classification (Randomized Search)

(c) Hyperparametric Classification (Grid Search)

(d) Non-parametric Vs Hyperparametric Classification

Fig. 8 Performance over Class Balanced Dataset Samples Oversampled & Undersampled using Weight of the User Activies

A detailed description of accuracy is as follows:

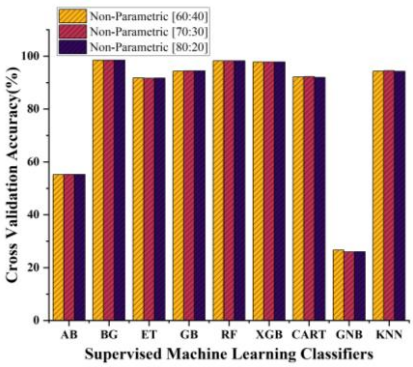
- ***Non-Parametric Classification*:** As shown in Fig. 7a, BG is the most accurate classifier with a CV accuracy of 98.37%. Next in order are *RF*, *XGB*, and *KNN*, with respective CV accuracies of 97.98%, 97.96%, and 97.75%. With 86.52%, *AB* has the lowest CV accuracy.

- ***Hyperparametric Classification using Randomized Search*:** With slight optimization observed for *ET*, *RF*, *XGB*, and *KNN*, the most accurate are *KNN*, *RF*, and *XGB*, with CV accuracy above 98%, as illustrated in Fig. 7b. The CV accuracy of *AB* and *CART* has optimized from 86.52% to 92.29%, and 92.29% to 95.39%, respectively.

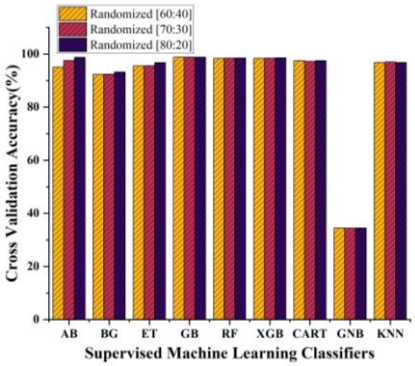
- ***Hyperparametric Classification using Grid Search***: This is clearly shown in Fig. 7c, where the classification accuracy of *ET*, *GB*, and *KNN* was optimized from 91.72% to 98.08%, 94.40% to 97.67%, and 97.75% to 98.11%, respectively, using the hyperparameters that were generated by the grid search approach. While *KNN* and *ET* have the highest accuracy—more than 98%—*AB* has the lowest accuracy—less than 50%.

- ***Randomized Search Vs Grid Search**:* The classification performance of *AB*, *ET*, *GB*, *CART*, and *KNN* has been optimized through the utilization of hyperparameters obtained from grid search and randomized search, as indicated by the results presented in Figure 7d. While the cross-validation accuracy of ET is better when the hyperparameters are obtained through grid search, *AB*, *XGB*, and *CART* perform significantly better when the hyperparameters are obtained through randomized search. Both *GB* and *KNN* have the same accuracy.

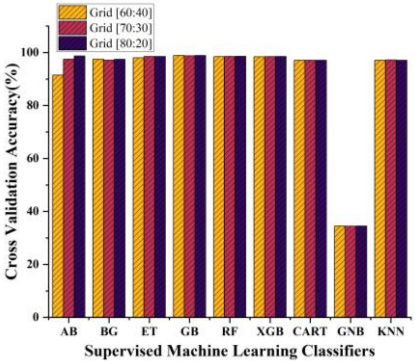
Figs. 5-7 provide an illustration of the final results of the proposed methodology for de-anonymization using supervised machine learning. The results demonstrate that for the class balanced dataset samples, the has significantly improved accuracy, i.e. upto 98%. For the randomized search, the hyperparameter *algorithm* which uses *Stagewise Additive Modeling using a Multi-class Exponential(SAMME)* has tuned the accuracy of *Adaptive Boosting(AB).* It is boosting algorithm that is used to improve the accuracy of machine learning models. For *Guassian Naive Bayes(GNB),* the variance of the distribution is artificially increased by the hyperparameter, *var_smoothing*, whose default value is taken from the training data set, by a user-defined value. In essence, this "smooths" out the curve and allows for a greater number of samples that deviate from the distribution mean. Hence, optimizing the accuracy in the context of grid search. Additionally, the effectiveness of the classifiers has been assessed using training to test ratios of 70:30 and 80:20.



(a) Non-parametric Classification



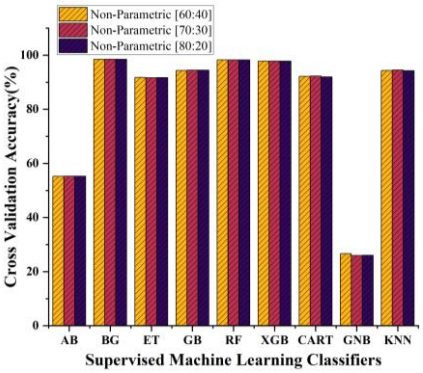
(b) Hyperparametric Classification (Randomized Search)



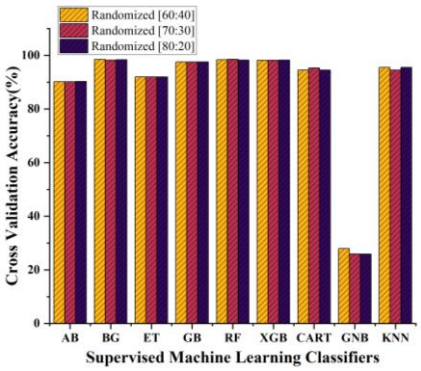
(c) Hyperparametric Classification
(Grid Search)

Fig 8. Comparative Analysis of Classifiers across Different Training: Testing Ratios over Unbalanced Dataset Samples

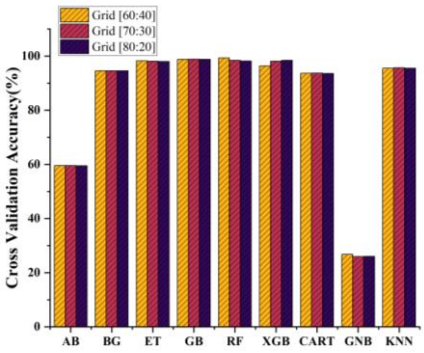
The experimental results depicted in Fig. 8 show that accuracy of classification carried out over the class imbalanced dataset samples is roughly the same for above said ratios is nearly same and is more than 98% for *BG, GB, RF,* and *XGB.*



(a) Non-parametric Classification



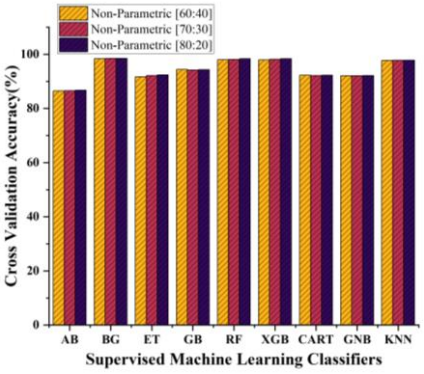
(b) Hyperparametric Classification
(Randomized Search)



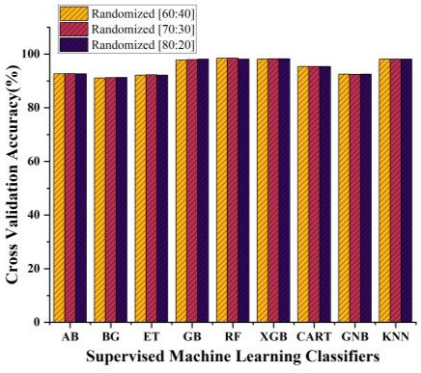
(c) Hyperparametric Classification
(Grid Search)

Fig 9. Comparative Analysis of Classifiers across Different Training: Testing Ratios over Class Balanced Dataset Samples(SMOTE)

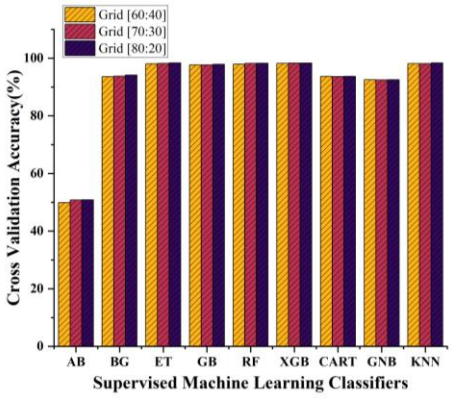
The experimental results, which are shown in Fig. 9, demonstrate that the classification accuracy over the class balanced dataset samples obtained using SMOTE, is approximately the same for the aforementioned ratios and is greater than 98% for *RF*.



(a) Non-parametric Classification



(b) Hyperparametric Classification
(Randomized Search)



(c) Hyperparametric Classification
(Grid Search)

Fig 10. Comparative Analysis of Classifiers across Different Training: Testing Ratios over Class Balanced Dataset Samples(Weighted Mean)

According to the experimental results, which are displayed in Fig. 10, the classification accuracy over the class balanced dataset samples generated by Weighted Mean is almost the same for the previously indicated ratios and is more than 97% for *RF*, *XGB*, and *KNN*.

7. Conclusion and Futures Work

This paper uses supervised machine learning for the deanonymization of blockchain transactions. We have performed a thorough multi-class classification in order to deanonymize transactions conducted on the Bitcoin blockchain. The Blockchair [36] and WalletExplorer [37] repositories were the sources of the dataset samples that were scraped. By applying the supervised machine learning, an average cross-validation accuracy of 83.23% was attained. Furthermore, the work employs Weighted Mean and SMOTE which has enhanced the classification performance, particularly that of Adaptive Boosting and Gaussian Naive Bayes. Additionally, this study uses Grid Search and Randomized Search to improve the accuracy of optimization cross-validation. The classifiers' accuracy has increased thanks to these hyperparameter tuning techniques. A notable improvement in accuracy is evident from the data.

The outcomes demonstrate that it is feasible to classify the user addresses based on their activities carried out on Bitcoin Blockchain which is a decentralized infrastructure environment. This opposes the notion that Bitcoin is truly anonymous, as it allows for the disclosure of the class of a substantial number of user addresses on the blockchain. Potentially, the proposed framenwork could help with both criminal investigations and regulatory compliance. The unavailability of labelled datasets is the biggest challenge to carrying out the research. An organization using the Bitcoin Blockchain for transactions may have to demonstrate that the funds it received were not purposefully used for illegal purposes. Our research opens the door to identifying and detecting high-risk transactions for such compliance activities, allowing organizations to protect their reputation and adhere to local laws.

In the future, we will propose hybrid models and other approaches to improve results, along with collecting datasets from multiple sources and preparing the labelled dataset, which may have imbalanced classes.

Compliance with Ethical Standards

This article excludes any research that any of the authors conducted on either humans or animals.

Competing Interests

There's no conflict of interest since it is not submitted to any other journal.

Research Data Policy and Data availability statement

The corresponding author may provide the dataset created and/or analyzed throughout this research upon acceptable request.

References

1. Nayyer N, Javaid N, Akbar Ma, Aldegheıshem A, Alrajeh N, Jamil M (2023) A New Framework for Fraud Detection in Bitcoin Transactions Through Ensemble Stacking Model in Smart Cities. In IEEE Access 11:90916- 90938. https://doi.org/ 10.1109/ACCESS.2023.3308298
2. Nicholls J, Kuppa A, Le-Khac NA (2023) SoK: The next phase of identifying illicit activity in Bitcoin. In Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC) 1–10. https://doi.org/10.1109/ICBC56567.2023.10174963
3. Nakamoto S (2008) Bitcoin: A peer-to-peer electronic cash system. Available at SSRN 3440802. Accessed 11 September 2023.
4. Bohme R, Christin N, Edelman B, Moore T. (2015) Bitcoin: Economics, technology, and governance. Journal of economic Perspectives. 29(2):213-38. https://doi.org/10.1257/jep.29.2.213
5. Rahouti M, Xiong K, Ghani N (2018) Bitcoin Concepts, Threats, and Machine-Learning Security Solutions. In IEEE Access 6: 67189- 67205. https://doi.org/10.1109/ACCESS.2018.2874539
6. Christin N (2013) Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace. In Proceedings of the 22nd International Conference on World Wide Web. 213-224. https://doi.org/10.1145/2488388.2488408
7. Hout MCV, Bingham T (2013) 'Silk Road', the virtual drug marketplace: A single case study of user experiences. International Journal of Drug Policy. 24(5):385-91. https://doi.org/10.1016/j.drugpo.2013.01.005
8. Martin J (2014) Lost on the Silk Road: Online drug distribution and the 'cryptomarket'. Criminology & Criminal Justice 14(3):351-67. https://doi.org/10.1177/1748895813505234
9. Karlstrøm H (2014) Do libertarians dream of electric coins? The material embeddedness of Bitcoin. Distinktion: Scandinavian journal of social theory 15(1):23-36. https://doi.org/10.1080/1600910X.2013.870083
10. Nouman M, Qasim U, Nasir H, Almasoud A, Imran M, Javaid N (2023) Malicious Node Detection Using Machine Learning and Distributed Data Storage Using Blockchain in WSNs. In IEEE Access 11: 6106- 6121. https://doi.org/10.1109/ACCESS.2023.3236983
11. Meiklejohn S, Pomarole M, Jordan G, Levchenko K, McCoy D, Voelker GM, Savage S. A fistful of bitcoins: characterizing payments among men with no names (2016) Communications of the ACM 59 (4): 86–93. https://doi.org/10.1145/2896384
12. Chaurasia BK, Verma S (2010) Maximising Anonymity of a Vehicle," In International Journal of Autonomous and Adaptive Communications Systems (IJAACS), Special Issue on Security, Trust, and Privacy in DTN and Vehicular Communications, Inderscience 3(2): 198-216. https://doi.org/10.1504/IJAACS.2010.031091
13. Benjamin V, Zhang B, Nunamaker Jr JF, Chen H (2016) Examining hacker participation length

in cybercriminal internet-relay-chat communities. Journal of Management Information Systems 33(2):482-510.
http://dx.doi.org/10.1080/07421222.2016.1205918

14. Samtani S, Chinn R, Chen H, Nunamaker Jr JF (2017) Exploring emerging hacker assets and key hackers for proactive cyber threat intelligence. Journal of Management Information Systems 34(4):1023-53.
https://doi.org/10.1080/07421222.2017.1394049
15. Beck R (2018) Beyond bitcoin: The rise of blockchain world. In Computer 51(2):54-8. https://doi:10.1109/mc.2018.1451660
16. Abbasi A, Zahedi FM, Zeng D, Chen Y, Chen H, Nunamaker Jr JF (2015) Enhancing predictive analytics for anti-phishing by exploiting website genre information. Journal of Management Information Systems 31(4):109-57.
https://doi.org/10.1080/07421222.2014.1001260
17. Abbasi A, Hsinchun Chen (2005). Applying Authorship Analysis to Extremist-Group Web Forum Messages. IEEE Intelligent Systems 20(5): 67–75. https://doi.org/10.1109/mis.2005.81 .
18. Beck R, Czepluch JS, Lollike N, Malone S. Blockchain–the gateway to trust-free cryptographic transactions (2016) In Twenty-Fourth European Conference on Information Systems (ECIS), 1-14.
19. Koshy P, Koshy D, McDaniel P (2014) An analysis of anonymity in bitcoin using p2p network traffic. In: Christin, N., Safavi-Naini, R. (eds) Financial Cryptography and Data Security. FC 2014. Lecture Notes in Computer Science, 8437:469-485. https://doi.org/10.1007/978-3-662-45472-5_30
20. Androulaki E, Karame GO, Roeschlin M, Scherer T, Capkun S. Evaluating user privacy in bitcoin (2013) In: Sadeghi, AR. (eds) Financial Cryptography and Data Security 7859: 34-51.
https://doi.org/10.1007/978-3-642-39884-1_4
21. Bonneau J, Narayanan A, Miller A, Clark J, Kroll JA, Felten EW (2014) Mixcoin: Anonymity for bitcoin with accountable mixes. In: Christin, N., Safavi-Naini, R. (eds) Financial Cryptography and Data Security. FC 2014. Lecture Notes in Computer Science 8437: 486-504.
https://doi.org/10.1007/978-3-662-45472-5_31
22. Misra G, Hazela B, Chaurasia BK (2013) Zero Knowledge based Authentication for Internet of Medical Things. In 14th International Conference on Computing, Communication And Networking Technologies (ICCCNT), 1-6.
23. Meiklejohn S, Orlandi C (2015) Privacy-enhancing overlays in bitcoin. In International Conference on Financial Cryptography and Data Security, 127- 141.
https://doi.org/10.1007/978-3-662-48051-9_10
24. Zola F, Eguimendia M, Bruse JL, Urrutia RO (1029) Cascading machine learning to attack bitcoin anonymity. In IEEE International Conference on Blockchain (Blockchain), 10-17.
https://doi.org/10.1109/Blockchain.2019.00011.
25. Harlev MA, Sun Yin H, Langenheldt KC, Mukkamala R, Vatrapu R (2018) 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, 3497- 3506.
26. Saxena R, Arora D, Nagar V. (2023) Efficient blockchain addresses classification through cascading ensemble learning approach. International Journal of Electronic Security and Digital Forensics 15(2):195-210.
https://doi.org/10.1504/IJESDF.2023.129278
27. Saxena R, Arora D, Nagar V (2023) Classifying Transactional Addresses using Supervised Learning Approaches over Ethereum Blockchain. In Procedia Computer Science 218:2018-2025. https://doi.org/10.1016/j.procs.2023.01.178
28. Reiter MK, Rubin AD (1998) Crowds: anonymity for Web transactions. ACM Transactions on Information and System Security (TISSEC) 1(1): 66–92 (1998). https://doi.org/10.1145/290163.290168
29. Chaurasia BK, Verma S, Tomar GS (2013), "Intersection Attack on Anonymity in VANET. In the M.L. Gavrilova and C.J.K. Tan (Eds.): Transactions on Computational Science XVII, Springer-Verlag Berlin Heidelberg 7420:133-149. https://doi.org/10.1007/978-3-642-35840-1_7
30. Wu X, Bertino E (2007) An Analysis Study on Zone-Based Anonymous Communication in Mobile Ad Hoc Networks. IEEE Trans. on Dependable and Secure Computing 4(4): 252– 264.

https://doi.org/10.1109/TDSC.2007.70213
31. Froomkin, AM (1995) Anonymity and its enmities. Journal of Online Law, Online available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2715621. Last accessed on May 15, 2023).
32. Froomkin AM (1999) Legal issues in anonymity and pseudonymity. The Information Society 15(2):113-127. https://doi.org/10.1080/019722499128574
33. Hastie T, Tibshirani R, Friedman JH, Friedman JH(2009) The elements of statistical learning: data mining, inference, and prediction. New York: Springer; https://doi.org/10.1007/978-0-387-84858-7.
34. Sun Yin HH, Langenheldt K, Harlev M, Mukkamala RR, Vatrapu R(2019). Regulating cryptocurrencies: a supervised machine learning approach to de-anonymizing the bitcoin blockchain. Journal of Management Information Systems.;36(1):37-73. https://doi.org/10.1080/07421222.2018.1550550
35. Herrera-Joancomartí, J. (2015). Research and challenges on bitcoin anonymity. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8872, 3-16. https://doi.org/10.1007/978-3-319-17016-9_1 https://doi.org/10.1007/978-3-662-48051-9_1
36. Blockchair Database, Online available at: https://gz.blockchair.com/bitcoin/ transactions/, Last accessed on 29 March, 2023.
37. Wallet labelling, Online available at: https://www.walletexplorer.com/ Last accessed on 30 March, 2023.
38. Jain YK, Bhandare SK.(2011), Min max normalization based data perturbation method for privacy protection. International Journal of Computer & Communication Technology.2(8):45-50. https://doi.org/10.47893/ijcct.2013.1201
39. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research 1(16): 321–357. https://doi.org/10.1613/jair.953
40. Batista GE, Prati RC, Monard MC (2004) A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD explorations newsletter 6(1):20-29. https://doi.org/10.1145/1007730.1007735
41. Saxena R, Arora D, Nagar V. (2023) Classifying blockchain cybercriminal transactions using hyperparameter tuned supervised machine learning models, Int. J. Computational Science and Engineering, 26(6), 615–626, https://doi.org/ 10.1504/IJCSE.2022.10056854
42. Schratz P, Muenchow J, Iturritxa E, Richter J, Brenning A (2019) Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. Ecological Modelling 406:109-120. https://doi.org/10.1016/j.ecolmodel.2019.06.002
43. Probst P, Wright MN, Boulesteix AL (2019) Hyperparameters and tuning strategies for random forest. Wiley Interdisciplinary Reviews: data mining and knowledge discovery 9(3):1-15 https://doi.org/10.1002/widm.1301
44. Zhang L, Zhan C (2017) Machine learning in rock facies classification: An application of XGBoost. In International Geophysical Conference on Society of Exploration Geophysicists and Chinese Petroleum Society, 1371-1374 https://doi.org/10.1190/IGC2017-351
45. Ifraz GM, Rashid MH, Tazin T, Bourouis S, Khan MM (2021) Comparative Analysis for Prediction of Kidney Disease Using Intelligent Machine Learning Methods. Computational and Mathematical Methods in Medicine, Hindawi 2021(6141470): 1-10 https://doi.org/10.1155/2021/6141470
46. Putatunda S, Rama K (2018) A comparative analysis of hyperopt as against other approaches for hyper-parameter optimization of XGBoost. In Proceedings of the 2018 International Conference on signal processing and machine learning, 6-10 https://doi.org/10.1145/3297067.3297080
47. Syarif I, Prugel-Bennett A, Wills G (2016) SVM parameter optimization using grid search and genetic algorithm to improve classification performance. (TELKOMNIKA) Telecommunication Computing Electronics and Control 14(4): 1502-1509 https://doi.org/10.12928/TELKOMNIKA.v14i4.3956
48. Ataei M, Osanloo M (2004) Using a Combination of Genetic Algorithm and the Grid Search Method to Determine Optimum Cutoff Grades of Multiple Metal Deposits, International Journal

of Surface Mining, Reclamation and Environment, 18(1): 60-78 https://doi.org/10.1076/ijsm.18.1.60.23543

49. Xiao T, Ren D, Lei S, Zhang J, Liu X (2014) Based on grid-search and PSO parameter optimization for Support Vector Machine. In Proceeding of the 11th World Congress on Intelligent Control and Automation, 1529-1533, https://doi.org/10.1109/WCICA.2014.7052946