Pre-Print Version

Comparative Analysis of Machine Learning Based Fraud Detection Techniques In Blockchain

Sabih Hida Tahir[1*], Muhammad Raza[2], Ageshwari Haryani[3], Almina Sherish[4]

ABSTRACT: **Fraudulent activities like phishing and pump-dump schemes clearly threaten the integrity and reliability of decentralized platforms, especially Ethereum. This paper compares the quality of fraud detection methods in Ethereum's platform. It emphasizes the potential of unsupervised and supervised learning algorithms applied to Ethereum. The aim is to have an advanced system capable of firmly protecting Ethereum by detecting fraud and putting a stop to it. This paper collected transactional data on Ethereum, smart contract interactions, and past fraud activities possibly significant to net miners. It further proposed fine-grained features targeting Ethereum transaction nuances, which are important early signs of fraud. The paper takes an integrated approach in comparing traditional supervised methods such as Random Forest, eXtreme Gradient Boosting, Decision Trees, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Naive Bayes, Logistic Regression, and Linear Discriminant Analysis (LDA), versus unsupervised learning like outlier detection or clustering algorithms such as K-Means, Gaussian Mixture Models (GMM), and BIRCH. Unlike most Ethereum fraud detection studies that rely heavily on supervised techniques, this one highlights the lack of unsupervised techniques and shifts the spotlight to a comparative analysis with three unsupervised algorithms. In addition, this paper also compares the algorithms on time efficiency. Benchmarking with traditional supervised techniques indicates that unsupervised learning is more effective in detecting new fraudulent patterns. Overall evaluation was further broken down under headings of precision, recall, F1-score and Silhouette score. It proposes a proactive fraud prevention system for Ethereum, having foreseen an event before it actually happens. The goal is to maintain security in Ethereum, as well as other decentralized networks, providing flexible defenses against this rapidly evolving form of crime.**

Keywords**— Blockchain Security, Ethereum Network, Fraud Detection, Unsupervised Learning, Comparative Analysis, Proactive Security, Anomaly Detection, Cryptocurrency**

INTRODUCTION

Blockchain technology is regarded as one of the most developing technologies in today's world and it is an innovation seen more in the present digital era; thus, the application of blockchain has not only been confined to the financial and digital currency industry but has rather expanded to all domains present in the society.[2]

Vitalik Buterin developed Ethereum in 2014, hoping its improvements would fix what was wrong with Bitcoin, such as block size and creation time. Ethereum has enhanced these aspects whereby it takes approximately 12 seconds to add a block to the chain of the network. Moreover, it also addresses the scalability issue and provides the blockchain architecture that deals with decentralized applications (DApps), smart contracts, and cryptocurrency. Since the creation of the Ethereum network, updates occurred frequently and are still being made.

As a new creation, blockchain employs consensus as a way of constructing blocks and forms a chain system that does not allow duplication and forgery due to encryptions and digital signatures. Yet there are still various risks associated with security, and this is especially true about smart contracts. The inability to construct the infrastructure securely is due to differences between programmers. For instance, on June 18, 2016, planning and malicious hackers conspired to steal around $100 million from an Ethereum-based project called The DAO and get hold of 3. 6 million ethers. There was a similar January attack when hackers exploited the weakness of the Parity Multi-Signature Library through which they stole approximately 220 million RMB, and locked over 500,000 ethers in 587 wallets. In April 2018, integer overflow issues in the contract code of the BEC American Chain were put into effect to create a project that wiped out about RMB 6 billion of tokens and almost caused the complete loss of the tokens' value. Early in 2019, the global blockchain sector is estimated to have given away more than $6 billion to security issues and a hack in December 2020 saw an equivalent of nearly $3. 8 billion.[3]

Ethereum Accounts:

Ethereum uses "accounts" to represent its state. An account's address is a 20-byte string, and state transitions are the actual transfers of money and data between accounts. Each account address consisting of four fields: account storage, which is the account data on the blockchain; bytecode, which is the code of the account; account balance, which is how much more of the digital currency the account has to spend; and nonce,

[1-3-4]SZABIST University, Karachi Campus
[2]SZABIST University, Gharo Campus
Country: Pakistan
Email: sabihhida@hotmail.com

which is a number used only once in a process known as mining. [21]
Ethereum has two types of accounts:

Externally Owned Account (EOA):
These wallets are run by an external authority via private keys and are basic, mostly for storing Ether and making transactions. A point of note is that EOAs do not have their code or data storage, and they are managed by software like a wallet application. In practice, EOAs are cryptographically signed with a private key, and the signature is verified against the known public key of the EOA.

Contract Account (Smart Contract):
They have private and public keys, and users can use the contract's executed codes to carry out internal transactions. When the contract account receives a message, its code is activated, allowing it to read and write internal storage, send other messages, and create contracts [22].

With Ethereum-based systems, fraud can take many different forms. From tricks to altering transaction data to grabbing "black-box" artifacts out of smart contracts, which will tip off the alarm and freeze everything, for social engineering Ethereum is a very amenable target. With Ethereum, transactions are publicly recorded but the identity of the person conducting them is pseudonymous. So standard fraud detection techniques simply do not work here. To tackle these problems and raise the security of Ethereum-based applications in response, this paper focuses on using machine learning techniques for blockchain fraud detection, particularly how effective unsupervised methods work.

Smart contracts can be coded in any high-level programming languages such as Serpent, Viper or Solidity where the preferred language is Solidity. Executing: They have a fee and are powered by the Ethereum Virtual Machine (EVM). All nodes in the Ethereum network must process the confirmed transactions by original consensus protocols. Consumers on Ethereum base their transactions on certain conditions which are controlled by cryptocurrency exchanges.

Ethereum transactions:
Ethereum is capable of doing internal and external transactions. Internal transactions are transmitted from one smart contract to a different smart contract without signature fields and are not documented in the underlying blockchain architecture. The functions of t' functions are sending messages containing details such as the amount of Ether. In external transactions, those initiated by EOAs, it is accomplished by signing with a private key. It is written in a shared ledger which additionally bolsters the Ethereum blockchain framework. Verifying the returned hash values, EOAs keep a tally of transaction details.

The life cycle of transactions in the Ethereum network involves several steps as shown in figure 1:

- A user gains access to an Ethereum account and performs different blockchain transactions.
- All transactions are accumulated in the Mempool when none of the participating nodes agrees to validate and add it to the chain.
- A miner is chosen to work on the transactions provided in the Mempool and is in a position to approve or deny them in any block.
- Once a miner corroborates a block, the sequential nodes within the Ethereum network are updated with the new block on the blockchain.[4]

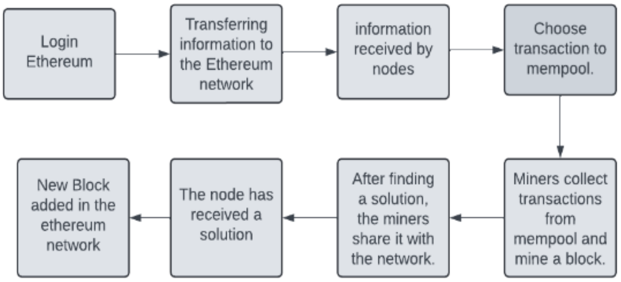
Login Ethereum
Transferring information to the Ethereum network
information received by nodes
Choose transaction to mempool.
Miners collect transactions from mempool and mine a block.
After finding a solution, the miners share it with the network.
The node has received a solution
New Block added in the ethereum network

Figure 1: The Flow of transactions in Ethereum Network [4]

This research uses a selected dataset for identifying fraudulent transactions in the Ethereum network. To achieve a better understanding of the data collected, the research methodology entails several important steps, which include data cleansing, data preprocessing, splitting, and exploratory analysis. The supervised algorithms used are Random Forest, eXtreme Gradient Boosting, Decision Trees, K-Nearest Neighbors, Support Vector Machines, Naive Bayes, Logistic Regression, and Linear Discriminant Analysis. The unsupervised learning approaches used on the other hand include the outlier detection and the clustering algorithms like the K-Means, the Gaussian Mixture Models, and the BIRCH. The findings of the study show that eXtreme Gradient Boosting achieves a perfect accuracy of 100% with a reasonable time of 0.11 seconds, while K-Means demonstrates high accuracy with a Silhouette score of 99.0 and completes its task in just 0.45 seconds. Notably, Naive Bayes emerges as the most time-efficient algorithm of 0.03 seconds. Among the unsupervised algorithms, Gaussian Mixture Models (GMM) also stand out for their time efficiency 0.008 seconds. Supervised learning requires labeled data for training and unsupervised learning can detect patterns, anomalies and other attributes within an unlabeled or unstructured data set very effectively.

In the field of Ethereum fraud detection, supervised learning systems provide insight based on labeled data, but unsupervised learning reprograms itself at each stage of its wandering phase rather than reflecting any reality. Only through recently combining these two approaches can a single, unified system for Ethereum fraud detection be created--one not only addresses known models of fraudulent activity but also can adapt in real-time to new threats and dangers as they emerge. This will further enhance the safety and reliability of the Ethereum ecosystem.

PROBLEM & SOLUTION STATEMENTS

Problem Statement

The growing number of fraudulent activities in Ethereum increases the threat of the instability of the financial market which raises worries about money laundering and other money laundering and fraudulent operations.

Growing Fraud in the Ethereum Ecosystem:

One of the leading concerns that can be pointed out on the Ethereum blockchain network has been the increased instances of fraud. Fraudulent functions include scams, Ponzi schemes, fake token sales, phishing attacks and other actions that are employed with the aim of deceiving users or investors.

Money Laundering Concerns:

Illegitimate funds mostly undergo an effort to be cleansed for use in fraudulent activities in the Ethereum community. Moreover, since Ethereum for instance provides identity theft and anonymity, money launderers are able to use the cryptocurrencies as means of concealing the source of the illicit funds earned.

Consequences on Currency Markets:

Traditional finance can be influenced by specific instances, such as fraud and money laundering, within the framework of Ethereum. Those that have been obtained through criminal activity can manipulate exchange rates, escape regulators' eyes and help accelerate international money laundering.

In order to mitigate these challenges, a multi-pronged strategy is needed, which need to comprise enhanced supervision, use of technology, as well as periodic awareness crusades and enhanced interaction with various stakeholders. Together, we can prevent threats to financial markets within the Ethereum system and promote confidence in digital currency operations by the development of appropriate measures to combat fraud and money laundering.

SOLUTION STATEMENTS

In order to solve the issues of fraud and money laundering in the Ethereum cryptocurrency ecosystem, the following solutions to the problem use machine learning techniques:

Anomaly Detection Models:

To establish the online anomaly detection models, machine learning algorithms will be trained using the historical transaction data from Ethereum blockchain. These algorithms used in the analysis can identify unusual patterns or behavior of transactions such as high number of transactions, large value transactions or strange transactions' frequency that forms fraud. Anomaly Detection Models: Develop machine learning algorithms that have been used in analyzing the activity on the Ethereum blockchain to build anomaly detection models on transaction history. These algorithms are capable of detecting other 'abnormal/a-symmetric' activities such as sudden rise in the number of transactions, sudden increase in the size of transaction, or unpredictable patterns that may correspond to a fraudulent act.

The Prediction Modeling for Risk Assessment:

This is the same as the previous task, but it requires using machine learning techniques to build prediction models that will help to assess the risk involved in the Ethereum transactions and wallet addresses. These models effectively allow cryptocurrency platforms to focus their monitoring and investigative work into certain transactions and addresses by systematically rating them according to a diverse range of factors such as the purses' traces, geographical location, and users' conduct.

Ensemble Learning and Model Fusion:

The integration of the multiple models and algorithms within the ensemble learning framework can contribute to the ability to employ multiple models with focus on the various types of frauds or money laundering techniques. Similar to model fusion strategies that are stacking and boosting, using multiple models to achieve a high level of detection and enhance the ability to counteract hostile efforts.

The members of Ethereum crypto-community must offer innovative means to prevent financial fraud and money laundering through the help of artificial intelligence and machine learning. These advanced analytical techniques enhance the ant counterfeiting of the digital asset transaction process without degrading or compromising its scalability, flexibility, and effectiveness in reacting to the dynamic threats Kraft and Dhillon, 2017).

RELATED WORK

In recent years, a significant amount of research has been conducted on fraud detection in blockchain platforms. This section reviews the key contributions from previous studies, highlighting the different methodologies and approaches taken in the domain of blockchain fraud detection.

One notable advancement in this field is the proposed a deep learning framework using graph representation learning to identify abnormal transactions. The framework consists of a structural auto-encoder and an attribute auto-encoder, which jointly learn node and attribute feature vector representations. An attention mechanism is introduced to learn the importance of the nodes and neighbors. The experiments with the multiple attributed network anomaly detection datasets reveal the performance enhancement. The problem of learning graph representations is relevant and has prospects for further development. More improvements and enhancements are required to apply this model for node classification, link prediction and clustering.[24]

This study investigates and implements six machine-learning algorithms to balance accuracy, precision, and recall. The synthetic minority over-sampling technique handles data imbalance, increasing the light gradient and boosting the machine classifier's accuracy to 98.4%. This work has the potential to enhance blockchain ecosystem security.[25]

This research paper investigates the use of ensemble machine learning models in Ethereum fraud detection using a selected dataset and a rigorous process that includes data cleaning, correlation analysis, data splitting, and exploratory data analysis. The study shows that self-optimized models, especially CATBoost and LGBM, are highly efficient in fraud detection with an accuracy of 97. 42% after oversampling, and higher F1 scores and AUC values. The K-Means SMOTE oversampling technique is identified to have the highest classification accuracy level of 97% to 98.42% with an AUC of 99. 82%.[23]

Further study analyzes security threats to the Ethereum blockchain, describes ten assault scenarios, and discusses the ways to protect the blockchain from specific attacks. The research includes a literature review, an exploration of Ethereum's history and architecture, an investigation of defense mechanisms, and an experimental assessment. The suggested protection strategies are comprehensively developed and tested using a combination of theoretical analysis and experimental evaluation.[3]

The next study discusses the problem of detecting money laundering by finding correlations in the Bitcoin blockchain. This research proposes the adoption of such forms of methodologies like unsupervised machine learning to improve the specificity, efficacy of the inquiries and the data intake machine learning approach.[11]

Another research endeavor developed a system that aims to reduce risks associated with different intrusion models and offers real-time intrusion detection capability. The current intrusion detection system proposed in this paper does not reveal consumers' information by comparing the balance book with the distributed blockchain information.[1]

In particular, to detect Ponzi schemes contracts on Ethereum a study for presenting a decentralized, secure, and privacy-preserving method is shown the approach includes a dataset of 3788 smart contracts with the management of the dataset performed with the help of Synthetic Minority Over-sampling Technique (SMOTE). Pre-processing involves the use of Long Short-Term Memory (LSTM) neural networks and the Term Frequency-Inverse Document Frequency (TF-IDF) framework for feature extractions. Evaluations regarding the distinct models are conducted, and the under-consideration model demonstrates higher recall values than the previous research, which emphasizes the effectiveness of the procedure concerning the categorization of Ethereum Ponzi schemes. The merged research papers highlighted the areas of work in intrusion detection, money laundering detection, fraud detection, and security in Blockchain networks including open problems, suggestions, questions, or future work of the related research areas. More emphasis has been placed on carrying out research and innovation studies in each of these fields. A condensed overview of key ideas is provided below: The study recommends exploring unsupervised learning techniques for fraud detection in blockchain networks alongside the prevalent focus on supervised approaches. It mentions investigating the impact of various inputs on fraud identification associated with the trust ranking of the nodes and comparing different models across different blockchain networks to resolve problems related to data privacy in conclusion while recognizing the potential of machine learning for blockchain fraud detection it emphasizes the need for continued research to address existing challenges.[9]

It is crucial to explore innovative methods that can effectively address the ever-changing nature of blockchain and the lack of labeled data. One potential area for further investigation involves exploring unsupervised machine-learning techniques capable of adapting to evolving patterns of cryptocurrency-related money laundering. Additionally, there is a need to focus on developing hybrid models that integrate supervised and unsupervised learning to accurately categorize addresses and identify suspicious activities. Moreover, examining the use of alternative data sources like network traffic and user behavior may provide valuable insights into improving anti-money laundering procedures' accuracy.[5]

Future research efforts and unresolved issues involve the requirement for additional investigations into different kinds of attacks that may evade detection by the suggested system, testing its adaptability with various blockchain cryptocurrencies, examining potential joint attacks by miners on the system, and exploring integration with conventional intrusion detection systems. [6]

Areas for further research include the examination of alliance, public chain, and cross-chain security. This encompasses the implementation of security defenses against diverse cross-chain attack scenarios and research into automatic cross-chain system attack detection. It is essential to explore further research in these areas.[7]

Utilizing a dataset from an open-source shared address collection, addressing data imbalance through oversampling with SMOTE. Feature extraction using TF-IDF and LSTM. Evaluation and comparison of models based on precision, recall, and F1-score. Assessment of feature contribution through permutation feature significance. The findings indicate the effectiveness of the proposed method in identifying Ethereum Ponzi scheme contracts.[8]

The research presents a highly exhaustive and effective approach that focuses on the identification of the malicious strings within the blockchain-based cryptocurrencies, and calling for the problem of the lack of labeled data. The proposed framework adopted different algorithms possessing an effective strategy in detecting fraud in a sheer number of records. Based on the binaries and following an AI strategy, four key aspects were identified defining the transactional profile of Ethereum entities. Data collection and feature selection are the steps in the predictive model where Ethereum blockchain data are preprocessed, web resource data collection is carried out carefully with an emphasis on determining the appropriate attributes, and addresses related to fraudulent activities are grouped. In the data pre-processing step, several resampling methods prevalent in the dataset include SMOTE, under- sampling or over – sampling in order to achieve a more balanced data set which is suitable for binary classifiers.

The ability of the framework is then assessed based on the comparison made with the actual label that is associated with the test entities. Most strikingly, the F1 score of using the proposed methodology is a notable average of 0. Or 996 for ensemble approaches have been proposed to mitigate the challenges caused by the limited availability of labeled data in blockchain applications. This research provides a practical and comprehensive approach towards screening for malicious actors within the Ethereum entities.[10]

In the realm of fraud detection for vehicle insurance, The Use of High-Quality Statistical Software and Classification of Skewed Data and it is a relief to find new approaches. Since in most cases it became a tradition to create large datasets that contain observations with non-normal distributions, the authors suggest improving the prediction process by using the meta-learning approach in which the forecast is made based on the results of a set of basic classifiers. When using the meta single classifier, methods that were employed included: Bagging, back propagation, naive Bayesian, and C4. Five of them are used, and the processes that are being used include data preparation, partition, and oversampling. When it comes to the performance of the proposed strategy the following can be seen: It is cost-friendly compared to other traditional methods; it is also better or performs better than the specific algorithm.

In summary, these publications not only highlight key challenges but also stress the importance of ongoing research to address these issues and enhance the effectiveness of security measures in blockchain networks.

PROPOSED MODEL

Objective:

To address the security issue in decentralized systems, develop an assessment of the supervised and unsupervised learning models for blockchain fraud identification. Among the suggested family of algorithms, there are Random Forest, eXtreme Gradient Boosting, K-Nearest Neighbors, Logistic Regression, Decision Tree, Gaussian Naive Bayes, Support Vector Machine, and Linear Discriminant Analysis. Unsupervised learning methods include K-Means, Gaussian Mixture Models (GMM), and Birch.

The current research will focus on enhancing the security of blockchain technologies, but more precisely on the supervised and unsupervised learning techniques. The supervised learning approach is a kind of machine learning where models are trained on historical classified data and are particularly useful in categorizing security events or threats in a blockchain. It is crucial to use the unsupervised learning method as it does not rely on the classified past data for training to discover new patterns, anomalies, and elaborate possible threats in blockchain networks. This area of research is intended to help establish good and efficient security solution models in a variety of Blockchain environments by incorporating both supervised and unsupervised learning models.

METHODOLOGY:

The step-by-step process of fraud scheme detection is outlined as follows and its graphical representation is shown in Figure 2:

The first steps include data gathering and preparation, which consist of tackling missing values, converting category variables, and creating new columns and target variables according to specific needs.

The size of the training set is 70% and that of the testing set is 30% of the data. The above pre-processing step ensures that each feature is equally important and is used to split the dataset into two sets while ensuring that the two classes are balanced. Also applied in cross-validation and hyper parameters tunings, different models are developed and optimized in machine learning.

The F1-score, accuracy, precision, and rate of recall are used in examining the effectiveness of each of the generated models.

These evaluations form the basis of any change that might be necessary which involves altering the appropriate parameters of the models and then reevaluating the models to achieve higher returns.

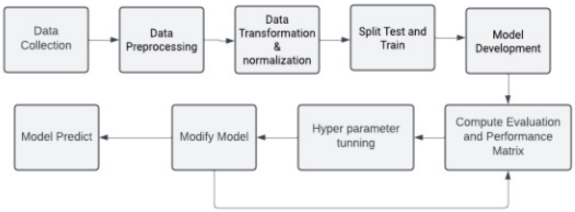

Figure 2: Proposed Model Workflow

DATASET

The dataset used in the examination of Ethereum Classic (ETC) was formed from transactions sampled from the record accessible through Google BigQuery and MySQL of the table on the Kaggle website[18]. The framework of this study lies in the 18 parameters in the dataset. The techniques and examinations used by the study are used to seek out strange patterns of transactions in Ethereum.

However, the fraudulent transactions are made by altering from the Etherscamdb3, an open-source programmable on GitHub. There are 72,500 instances and 18 fields in the dataset; here is the description of the fields: Table 1. The percentages of anomalous transactions are 20.0% and 80.0%, in that order as shown in Figure 3.

Table 1: Dataset Features and Description

	Field	Description
1	hash	Hash of the transaction
2	nonce	The number of transactions performed by the sender's account
3	transaction_index	Index of the transaction in a block
4	from_address	Source account
5	to_address	Target account
6	value	The value of transferred in Wei which is smallest Ether unit
7	gas	Amount of gas by source
8	gas_price	The price of gas in Wei that provided by the source
9	input	The data transmitted with the transaction
10	receipt_cumulative_gas_used	The amount of gas was used by this transaction when executed a block
11	receipt_gas_used	The total amount of gas was used by this given transaction alone
12	block_timestamp	Timestamp of the block was used by this transaction
13	block_number	Block number of the transaction
14	block_hash	block_hash
15	From_scam	The value 1 indicates sender address is a scam and 0 is a normal address
16	to_scam	The value 1 indicates receiver address is a scam and 0 is a normal address
17	from_category	Determine the main category (scamming or phishing) of abnormal activity that occurred from sender address, and (null) for the normal transaction
18	to_category	Determine the main category (scamming or phishing) of abnormal activity that occurred from the receiver address, and (null) for the normal transaction

EXPERIMENTAL SETUP

The dataset was loaded into the Colab and Mac mini 2 environments for analysis. It was first loaded into memory for data manipulation using Pandas. Missing values in the dataset were evaluated and eliminated. Following this, categorical variables were encoded using label encoding. Two additional features are introduced as new columns 'to_scam' and 'from scam', while a new feature was proposed as the target feature: 'is fraudulent'. In order to tackle this particular issue, a check was made and SMOTE was used with the help of scikit-learn provided preprocessing tool. The data set was then split into 70 percent training and 30 percent testing sets using the train_test_split method available on sci-kit-learn data preprocessing followed by normalization.

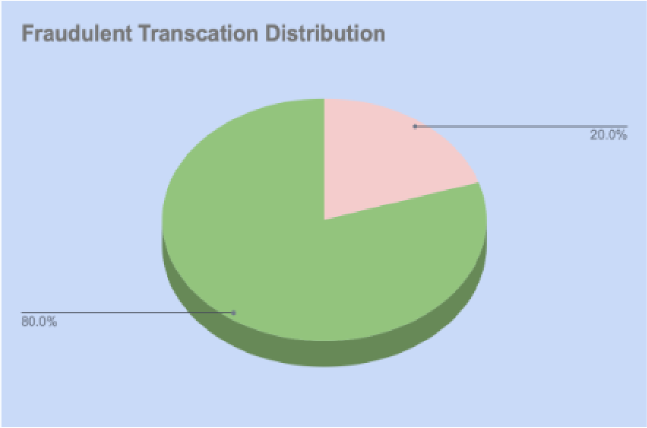

Figure 3: The distribution of transaction as a default or non default

Comparison Analysis

In the comparative analysis, investigations were conducted by applying both supervised and unsupervised learning techniques to identify patterns. Given the detailed and wide-scale experimental study, the mutual respect of these methods is implied and the fact that they are both functional and efficient in the patterns finding is being exposed.

Supervised Algorithms

The results for different comparison algorithms that are shown in Table 2, can be used to select the best algorithm for fraud detection concerning the Ethereum Blockchain, whereas its high accuracy and time efficiency are considered. This accuracy was calculated using Equation (5). Through a spot check without setting any of the hyper parameters, it was found that the Random Forest and eXtreme Gradient Boosting models had 100% accuracy in the results as shown in figure 4. On the other hand, the eXtreme Gradient Boosting model was found to take less time during the analysis as illustrated in figure 6. The second-best performing filter was the decision tree which had a 98% accuracy rate, but it had a hyper parameter tuning challenge. The third place again belonged to KNN with an accuracy of 95%. These results were comparable to achieving an 88% accuracy using support vector machines, naive Bayes, and logistic regression. The supervised algorithms could not detect some new risks and assaults because they only relied on labeled data that were not always available in all fraud cases.

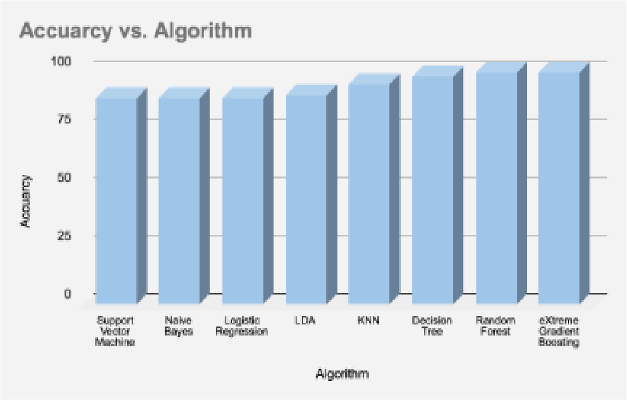

Figure 4: Accuracy comparison in supervised learning

In Figure 5, we indicate the precision, recall, and F1 scores which are calculated using the formulas provided in Equations (1), (2), and (3) respectively, of the eight models used in this study to contrast the performance of the two. The Random Forest and Extreme Gradient Boosting have almost similar and impartial values of precision, recall, and F1 score which can illustrate that the two algorithms are quite stable and provide almost balanced predicted results. KNN gives a good recall rate although it is not very accurate; this means that while KNN can predict most of the positive cases, it does so with a certain degree of error. Nonetheless, it produces quite a reasonable f1 score as the above result suggested because of relatively good performance in most aspects. Compared with the top performing models, Logistic Regression and Linear Discriminant Analysis are in between middling performance in terms of balanced performance indicators but the result or performance of Naive Bayes is also fairly favorable but with slightly lower precision which means though it classified the positive cases accurately it allows a greater number of positive and negative samples. For the last algorithm, the Support Vector Machine (SVM) with even lower values of precision and recall means it has the lowest F1 score meaning that it has a bigger problem with this dataset than the others.

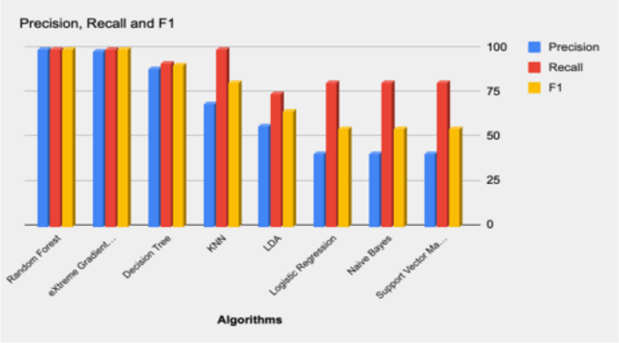

Figure 5: Results comparison in supervised learning

As depicted in figure 6, the support vector is the most time consuming at 94%, followed by random forest which is much better at only 14%. Extreme Gradient Boosting is even more efficient, coming in at just 2.2%. It has to be noted that the Naive Bayes algorithm, LDA (Linear Discriminant Analysis), and K-Nearest Neighbors (KNN) are almost as time efficient that even the time they consume is barely noticeable on the graph above. Of these three, the Naive Bayes is highly efficient compared to the others as clearly demonstrated in the below results Table 2.

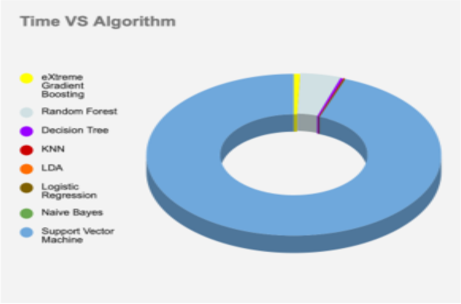

Figure 6: Time Comparison of supervised algorithm

Table 2: The evaluation of supervised machine algorithms

S.NO	Algorithm	Accuracy	Class	Precision	Recall	F1-Score	Confusion Matrix	(Time (Seconds	Hyper Parameters
1	Random Forest							14.6	None
	Train	100%	0	100%	100%	100%	[0 41695]] [[8180 0]	0.67	
			1	100%	100%	100%			
	Test	99%	0	100%	99%	99%	[231 17857]] [[3238 49]	0.29	
			1	93%	99%	96%			
2	**eXtreme Gradient Boosting**							**2.2**	None
	Train	100%	0	100%	100%	100%	[57 41687]] [[8123 8]	0.11	
			1	99%	100%	100%			
	Test	99%	0	100%	99%	100%	[130 17862]] [[3339 44]	0.05	
			1	96%	99%	97%			
3	Decision Tree							1.04	criterion=’entropy’, max_depth=19, max_features=’log2’, min_samples_leaf=13, min_samples_split=19
	Train	98%	0	99%	98%	98%	[716 41296]] [[7464 399]	0.018	
			1	89%	92%	91%			
	Test	097%	0	99%	99%	98%	[379 17654]] [[3090 252]	0.0084	
			1	89%	92%	91%			
4	KNN							0.21	n_neighbors=2
	Train	95%	0	100%	94%	97%	[2568 41695]] [[5612 0]	4.31	
			1	69%	100%	81%			
	Test	92%	0	99%	93%	96%	[1359 17638]] [[2110 268]	1.90	
			1	61%	89%	72%			
5	LDA							0.10	None
	Train	90%	0	96%	92%	94%	[3509 40116]] [[4671 1579]	0.007	
			1	57%	75%	65%			
	Test	90%	0	96%	92%	94%	[1482 17241]] [[1987 665]	0.011	
			1	57%	75%	65%			
6	Logistic Regression							0.45	random_state=42
	Train	89%	0	98%	90%	94%	[4796 40893]] [[3384 802]	0.018	
			1	41%	81%	55%			
	Test	89%	0	98%	90%	94%	[1985 17550]] [[1484 356]	0.007	
			1	43%	81%	56%			

7	Naive Bayes							0.031	alpha=0.1, fit_prior=True
	Train	89%	0	98%	90%	94%	801] [4796 40894]] [[3384	0.033	
			1	41%	81%	55%			
	Test	89%	0	98%	90%	94%	[1985 17550]] [[1484 356]	0.0117	
			1	43%	81%	56%			
8	Support Vector Machine							310.68	ernel='linear', gamma=0.1, C = 1.0
	Train	89%	0	98%	90%	94%	[4796 40896]] [[3384 799]	111.95	
			1	41%	81%	55%			
	Test	89%	0	98%	90%	94%	[1985 17550]] [[1484 356]	48.76	
			1	43%	81%	56%			

Unsupervised Algorithms:

The three non-hierarchical clustering techniques namely KMean, Birch, and GMM were discussed. As shown in Figure 7, KMean stood the best with an accuracy of 88% and a Silhouette score of 0.99 which explains the well-clustered dataset. The GMM was the second closest to achieving the best score with 87% and a silhouette score of 0.99 it was achieved after adjusting the hyperparameters as seen in Table 3. The model's accuracy for Birch averaged 80% and the silhouette score was 0. 84. The silhouette score and accuracy are calculated by using Equations (4) and (5) respectively. The implementation of unsupervised algorithms is generalizable for the identification of anomalies, it can identify new instances of fraud, and does not need to be trained with datasets labeled for the purpose. This is true if the parameters are tuned correctly, but it can be sensitive and can potentially take more time with higher false positives.

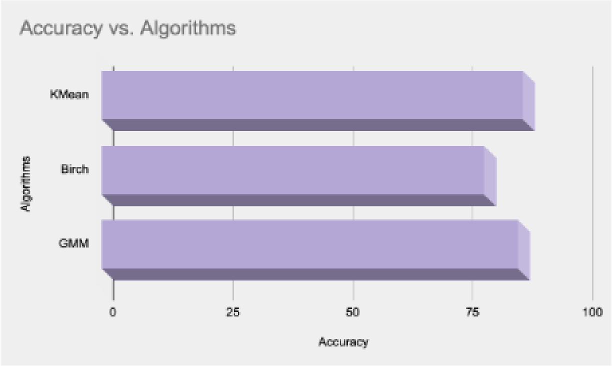

Figure 7: Accuracy comparison in unsupervised learning

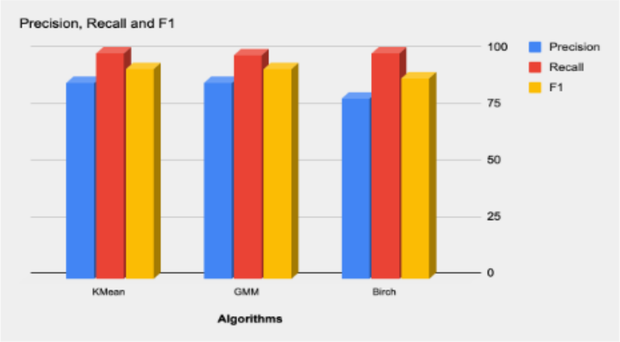

Figure 8: Results comparison in unsupervised learning

In Figure 8, The performance of three types of unsupervised learning algorithms including K-Means, GMM, and Birch are compared in terms of precision, recall, and F1 and calculated using the formulas provided in Equations (1), (2), and (3) respectively. These metrics give information on how efficient each of the algorithms is. K-Means has the highest precision, as well as a high recall, which makes for a high F1 score. This implies that besides being accurate in identifying the important data points, K-Means is also able to capture most of them. The GMM algorithm is also quite efficient but it has slightly lower precision, which means that while using this algorithm, there may be a slight decrease in accuracy. Still, at the same time, it will be possible to identify more data. Birch is a little slower than the other two: the precision and recall are good enough to yield a decent enough F1 value. This means that even though Birch could be less accurate, it is still efficient in its role of clustering.

The results prove the efficiency of all three algorithms with a variation in the proportion of precision and recall ratios in each of them. The decision as to which of them to use would depend on the nature of the task, whether one would need more accuracy, more coverage, or both.

Table 3: The evaluation of unsupervised machine algorithms

S.NO	Algorithm	Accuracy	Class	Precision	Recall	F1-Score	Confusion Matrix	Time (Sec(onds	Hyper Parameters	Silhouette Score
1	**KMean**								n_clusters=2, algorithm=’elkan’, max_iter = 1000	0.99
	Train	88%	0	87%	100%	93%	[129 39727]] [[4055 5964]	0.45		
			1	97%	40%	57%				
	Test	88%	0	87%	99%	93%	[119 17025]] [[1771 2460]	0.24		
			1	94%	42%	58%				
2	Birch								threshold=0.03, n_clusters=2	0.84
	Train	80%	0	80%	100%	89	[132 39724]] [[0 10019]	0.01		
			1	0%	0%	0%				
	Test	80%	0	80%	100	89	[51 17093]] [[0 4231]	0.01		
			1	0%	0%	0%				
3	GMM								n_components=2, random_state=42, init_params = ‘random’ , covariance_type=’spheri‘cal	0.99
	Train	87%	0	87%	99%	93%	[276 39580]] [[4056 5963]	0.008		
			1	94%	40%	57%				
	Test	88%	0	87%	99%	93%	[51 17093]] [[0 4231]	0.009		
			1	93%	42%	58%				

As Figure 9 illustrated, KMeans is the least time-efficient at 94%. Birch is even more efficient, with just 3.6%, while the GMM algorithm proves to be the most time-efficient option at only 1.8%.

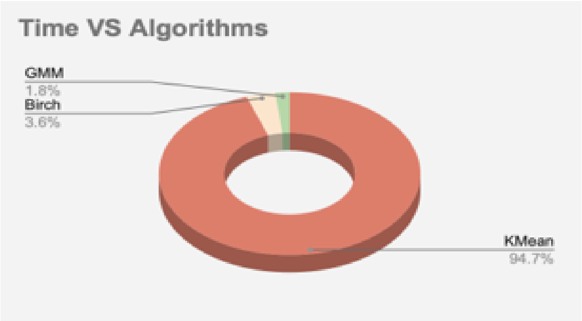

Figure 9: Time Comparison of unsupervised algorithms

PERFORMANCE EVALUATION

To analyze the performance of machine learning algorithms three common metrics are generally used that are precision, F1 and recall. Precision is the ability to correctly identify attack records among all known attacks. This equation can be used to calculate it: [13]

$$\text{Precision} \quad \frac{True\ Positives}{True\ Positives + False\ Positive}$$

Equation 1: Precision Calculation

The real fraudulent instances calculated with the help of recall, which is referred to as true positive rates. Recall is used to identify how many actual fraud cases are identified by model. This equation helps us figure out how well a model works when it's trying to predict if something is true or false. This equation can be calculated using: [13]

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Positive}$$

Equation 2: Recall Calculation

A high F1 score shows that the model is minimizing false positives and successfully recognizing real fraud situations. F1 is calculated as: [13]

$$\text{F1 Score} = \frac{2\ x\ Precision\ x\ Recall}{Precision\ +\ Recall}$$

Equation 3: F1 Calculation

The Silhouette score is an additional statistic for evaluating the quality of clusters in unsupervised learning systems. It provides an indicator of the degree of similarity between data points within a cluster as well as the degree of cluster separation. This is how the silhouette score is computed. [14]

$$\text{Silhouette score} = \frac{1}{n}\sum_{i=1}^{n} \blacksquare \left(b(i) - \frac{a\,(i)}{max\,\{a(i),b(i)\}} \right)$$

Equation 4: Silhouette Score Calculation

A further metric is Accuracy (AC), which may be computed as below and represents the accuracy percentage in the classification procedure.[13]

$$\text{Accuracy} = \quad \text{Accuracy} = \frac{TP + TN}{TP+TN+FP+FN}$$

Equation 5: Accuracy Calculation

The mentioned evaluation measures have been computed using a variety of classifiers, such as Random Forest, Decision Tree KNN, LDA, Logistic Regression, Naive Bayes, Support Vector Machine, KMeans, Birch, and GMM. The experimental findings from each classifier are displayed in Tables 2 and 3 above.

The significance of time comparison:

Comparing times is important when evaluating fraud detection algorithms. While model complexity affects estimation time, scalable techniques allow for growing datasets and computing loads. It is crucial to find a balance between batch and online processing and hyperparameter optimization. Software and hardware optimizations impact prediction time, therefore comparative research and benchmarks are necessary for an accurate assessment.

CONCLUSION

To sum up, this study has explored fraud detection in the Ethereum network in great detail, focusing on machine learning algorithms. Blockchain ecosystems are dynamic environments that require creative and proactive security solutions to maintain integrity and confidence. We have examined the benefits and drawbacks of supervised and unsupervised learning through a thorough comparison study. Hyperparameter tuning and unsupervised learning were able to achieve a top accuracy of 88%, while supervised learning was at 100%. To be sure, unsupervised learning brings with it particular advantages. In situations where data of any kind is rare or unavailable, it is very good at identifying new and subtle trends that could be fraudulent. In contrast, supervised learning is good at historical data analysis but may be unable to be identified if there is no labeled data available for real-time scenario handling. Therefore, the choice between supervised and unsupervised learning depends on the specific requirements and constraints of the fraud detection task in question.

The study puts forward several other avenues for research. For example, incorporating complex feature engineering techniques and adjusting algorithms adaptively over changing fraud trends in order to examine scalability evaluations for practical implementation the context of blockchain security, especially in Ethereum, is still evolving. In developing and building a robust, adaptive security method for this research indicates a way forward that until now was not thought about. Essentially, within an environment where blockchain technology is changing everything, this study provides a platform for further dialogue supporting the need to improve Ethereum's security posture. We aimed to contribute to this broader inaugural conversation on how decentralized systems confront fraud.

REFERENCES:

Chen, Xuhui & Ji, Jinlong & Luo, Changqing & Liao, Weixian & Li, Pan. (2018). "When Machine Learning Meets Blockchain: A Decentralized, Privacy-preserving and Secure Design." 1178-1187. 10.1109/BigData.2018.8622598.

Y. Chen, J. Sun, Y. Yang, T. Li, X. Niu, and H. Zhou, "PSSPR: a source location privacy protection scheme based on sector phantom routing in WSNs," International Journal of Intelligent Systems, vol. 37, 2021.

Duan, Li & Sun, Yangyang & Zhang, Ke-Jia & Ding, Yong. (2022). "Multiple-Layer Security Threats on the Ethereum Blockchain and Their Countermeasures. Security and Communication Networks". 2022. 10.1155/2022/5307697.

Lee, G. & Howard, D. & Ślęzak, Dominik & Hong, Y.S.. (2012). Communications in Computer and Information Science: Preface. 310. V.

Stefánsson, Hilmar Páll Grímsson, Huginn Sær Þórðarson, Jón Kristinn Oskarsdottir, Maria. (2022) "Detecting potential money laundering addresses in the Bitcoin blockchain using unsupervised machine learning", https://scholarspace.manoa.hawaii.edu/items/da95378e-b691-465c-abc6-f407efe7b546

Kim, Suah & Kim, Beomjoong & Kim, Hyoung. (2018). "Intrusion Detection and Mitigation System Using Blockchain Analysis for Bitcoin" Exchange. 40-44. 10.1145/3291064.3291075.

Li Duan, Yangyang Sun, Kejia Zhang, Yong Ding, "Multiple-Layer Security Threats on the Ethereum Blockchain and Their Countermeasures", Security and Communication Networks, vol. 2022, Article ID 5307697, 11 pages, 2022. https://doi.org/10.1155/2022/5307697

Xuezhi He, Tan Yang, and Liping Chen(2022) "Ethereum-Based Ponzi Contract Identification" https://doi.org/10.1155/2022/1554752

M. Bhowmik, T. Sai Siri Chandana and B. Rudra, "Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2021, pp. 539-541, doi: 10.1109/ICCMC51019.2021.9418470.

Detection: Classification of Skewed Data. SIGKDD Explorations. 6. 50-59. Poursafaei, Farimah & Hamad, Ghaith & Zilic, Zeljko. (2020). Detecting Malicious Ethereum Entities via Application of Machine Learning Classification. 120-127. 10.1109/BRAINS49436.2020.9223304.

Joana Lorenz, Maria Inês Silva, David Aparício, João Tiago Ascensão, Pedro Bizarro, 2021, Detecting potential money laundering addresses in the Bitcoin blockchain using unsupervised machine learning, https://doi.org/10.1145/3383455.3422549

Phua, Clifton & Alahakoon, Damminda & Lee, Vincent. (2004). "Minority Report in Fraud Detection: Classification of Skewed Data". SIGKDD Explorations. 6. 50-59.

Goutte, Cyril & Gaussier, Eric. (2005). A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. Lecture Notes in Computer Science. 3408. 345-359. 10.1007/978-3-540-31865-1_25

Rousseeuw, Peter. (1987). Rousseeuw, P.J.: Silhouettes: "A Graphical Aid to the Interpretation and Validation of Cluster Analysis". Comput. Appl. Math. 20, 53-65. Journal of Computational and Applied Mathematics. 20. 53-65. 10.1016/0377-0427(87)90125-7.

Rohit Saxena, Deepak Arora, Vishal Nagar," Classifying Transactional Addresses using Supervised Learning Approaches over Ethereum Blockchain", 2023 , https://doi.org/10.1016/j.procs.2023.01.178

Joana Lorenz, Maria Inês Silva, David Aparício, João Tiago Ascensão, ,Pedro Bizarro "Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity", https://doi.org/10.1145/3383455.3422549

Kim, Suah & Kim, Beomjoong & Kim, Hyoung. (2018). "Intrusion Detection and Mitigation System Using Blockchain Analysis for Bitcoin Exchange". 40-44. 10.1145/3291064.3291075.

https://www.kaggle.com/bigquery/crypto-ethereum-classic.

https://etherscamdb.info/.

https://github.com/MrLuit/EtherScamDB.

Huang, Qichen. (2023). Ethereum: Introduction, Expectation, and Implementation. Highlights in Science, Engineering and Technology. 41. 175-182. 10.54097/hset.v41i.6804.

https://ethereum.org/en/whitepaper/

Aziz et al. (2024). “Ethereum Fraud Detection Using Various Machine Learning Models.” International Research Journal of Engineering and Technology (IRJET), Volume 11, Issue 05, pp. 846-852. https://www.irjet.net/archives/V11/i5/IRJET-V11I5117.pdf

Ao Xiong, Chenbin Qiao, Yuanzheng Tong et al. “Block-chain Abnormal Transaction Detection Method Based on Auto-encoder and Attention Mechanism”, 30 May 2023, PREPRINT (Version 1) available at Research Square https://doi.org/10.21203/rs.3.rs-2969521/v1]

Chibuzo Obi-Okoli, Olamide Jogunola, Bamidele Adebisi, and Mohammad Hammoudeh. 2024. “Machine Learning Algorithms to Detect Illicit Accounts on Ethereum Blockchain”. In Proceedings of the 7th International Conference on Future Networks and Distributed Systems (ICFNDS ‘23). Association for Computing Machinery, New York, NY, USA, 747–752. https://doi.org/10.1145/3644713.3644838

Udit Agarwal, Vinay Rishiwal, Sudeep Tanwar, Mano Yadav, 07 December 2023, https://doi.org/10.1002/nem.2255