Advanced Fraud Detection in Blockchain Transactions: An Ensemble Learning and Explainable AI Approach

Shimal Sh. Taher
Computer Science Department, University of Duhok, Kurdistan Region, Iraq
Shimal.taher@uod.ac (corresponding author)

Siddeeq Y. Ameen
Quality Assurance Directorate, Duhok Polytechnic University, Kurdistan Region, Iraq
siddeeq.ameen@dpu.edu.krd

Jihan A. Ahmed
Computer Science Department, University of Duhok, Kurdistan Region, Iraq
drjihanrasool@uod.ac

Received: 18 November 2023 | Revised: 8 December 2023 | Accepted: 23 December 2023

 DOI: https://doi.org/10.48084/etasr.6641

ABSTRACT

In recent years, cryptocurrencies have experienced rapid growth and adoption, revolutionizing the financial sector. However, the rise of digital currencies has also led to an increase in fraudulent transactions and illegal activities. In this paper, we present a comprehensive study on the detection of fraudulent transactions in the context of cryptocurrency exchanges, with a primary focus on the Ethereum network. By employing various Machine Learning (ML) techniques and ensemble methods, including the hard voting ensemble model, which achieved a remarkable 99% accuracy, we aim to effectively identify suspicious transactions while maintaining high accuracy and precision. Additionally, we delve into the importance of eXplainable Artificial Intelligence (XAI) to enhance transparency, trust, and accountability in AI-based fraud detection systems. Our research contributes to the development of reliable and interpretable models that can significantly improve the cryptocurrency ecosystem security and integrity.

Keywords-blockchain; Ethereum; fraudulent transactions; machine learning

I. INTRODUCTION

The rapid growth of cryptocurrencies and the adoption of blockchain (BC) technology have revolutionized the financial sector, offering new opportunities for investment and secure online transactions [1]. BC, an advanced and rapidly evolving technology, offers a more secure alternative by providing a distributed, decentralized, and immutable ledger for recording transactions [2]. Despite the advantages of decentralized systems, cryptocurrencies face significant challenges in detecting and preventing fraudulent transactions [3]. These illicit activities not only harm the economy but also erode public trust in BC-based solutions. BC networks aim to detect and mitigate fraudulent transactions as quickly as possible to maintain community integrity and security [4]. The anonymous nature of BC transactions and the decentralized nature of cryptocurrencies make fraud detection a complex and challenging task [5]. The rise of digital currencies has led to an increase in financial transactions conducted online [6], with traditional currencies being converted into their digital counterparts. Although this modernization offers convenience and efficiency, it also exposes transactions to potential security breaches, such as fraud, anomalies, and privacy violations. As the volume of transactions increases, so does the risk of fraudulent activities, resulting in significant financial losses. Even within BC networks, malicious actors can exploit vulnerabilities and engage in fraudulent activities [7, 8]. Therefore, it is crucial to develop and implement robust techniques to detect and prevent fraudulent transactions in cryptocurrencies. This paper aims to explore and evaluate ML models for Ethereum Fraud Detection (EFD), focusing on ensemble learning and XAI to improve system accuracy and transparency. By addressing these challenges, we hope to contribute to the development of more secure and trustworthy BC-based financial solutions.

II. BACKGROUND

A. Blockchain and Decentralized Applications

At its core, BC is a manifestation of Distributed Ledger

Technology (DLT) [9]. This revolutionary technology facilitates consensus and validation of transactions across a computer network, thereby eliminating the need for a central or intermediary authority [10]. Each validated transaction, along with others, forms a new "block" that is added to an existing chain of transactions, giving rise to the term Blockchain [11, 12]. BC networks are typically categorized into permissioned and permissionless BCs. Permissioned BCs are exclusive networks utilized by specific individuals or entities, such as a consortium of banks conducting financial transactions [13]. On the other hand, permissionless or public BCs, like the Bitcoin network where transactions are conducted using Bitcoin as an exchange medium, are open source networks accessible to anyone [14]. A key feature of these networks is the use of smart contracts. These are self-executing contracts with the agreement terms directly written into code. Smart contracts automate and secure transactions, making them an integral part of decentralized applications (dApps). dApps are applications that run on a peer to peer computer network, leveraging BC technology principles to create secure, transparent, and resistant to censorship systems [9]. BC operates as a decentralized peer to peer network, where control over the data is not concentrated in any single node or group of nodes [15]. Instead, all nodes connected to the BC network share equal authority over the latter. Immutability is a defining characteristic of BC, safeguarding against data alteration [16]. BC maintains an append-only digital ledger, meaning that once data are added to the network, they cannot be edited or deleted. Every connected node in a BC network has a copy of the current ledger [11], making the data within the specific network accessible to all connected participants and ensuring system transparency and availability. The foundational understanding of BC technology, along with the use of smart contracts, paves the way for the development and application of dApps. These applications have a wide range of uses, from decentralized finance (DeFi) [17] where they provide financial services in a more open, accessible, and equitable manner. They are also used in voting systems [18], enhancing the voting process transparency and security. Furthermore, dApps are utilized in supply chain management, where they provide real-time, transparent tracking of goods, enhancing efficiency and accountability [19]. They are also making inroads into the gaming industry, creating decentralized gaming platforms that offer true ownership of game assets [20]. In the content management field, dApps enable creators to publish and monetize their content without the need for intermediaries [21].

B. Ethereum

Ethereum is a decentralized, open-source BC platform known for its smart contract functionality [20]. It was proposed in 2013 by programmer Vitalik Buterin and its development was crowdfunded in 2014, with the network going live on July 30, 2015 [22]. In contrast to Bitcoin, which is primarily a digital currency, Ethereum's primary goal is to serve as a platform for decentralized applications (dApps). These dApps are designed to operate without any downtime, fraud, control, or interference from a third party. Ethereum's platform specific cryptographic token, Ether, is the fuel that powers these applications [44]. Ether serves two main functions: it acts as a digital currency exchange similar to Bitcoin, and it is used within the Ethereum platform to run applications and even monetize work. A significant innovation of Ethereum is the Ethereum Virtual Machine (EVM), a Turing complete software that operates on the Ethereum network. The EVM allows any program to be run, regardless of the programming language, as long as sufficient time and memory are provided. The EVM streamlines the creation of multiple diverse BC applications on a single platform [45]. As other cryptocurrencies, Ethereum functions on a decentralized network, which makes it a compelling platform for implementing smart contracts and developing decentralized applications (dApps). However, this decentralization also opens up avenues for fraudulent activities. For instance, Ethereum is susceptible to a 51% attack, where if a miner or a group of miners possesses more than half of the network's computing capability, they can rewrite the BC, leading to potential fraud [23]

C. Explainable AI (XAI)

XAI is a research area that aims to design algorithms and methods which allow AI models to offer clear and human understandable explanations for their predictions and decisions [24]. The primary objective of Interpretable AI is to improve transparency, accountability, and trust in AI systems by helping humans comprehend the rationale behind model outcomes [25]. Within the realm of cybersecurity, Interpretable AI plays a crucial role in detecting and mitigating cyber threats [26]. AI models are utilized by cybersecurity professionals to analyze vast quantities of data and pinpoint potential threats. However, due to the opaque nature of these models, understanding their decision-making process can be challenging, leaving experts uncertain about the appropriate response to identified threats [27]. By incorporating Interpretable AI, cybersecurity specialists can gain a better understanding of the AI model's predictions, which allows them to make well informed decisions when addressing cyber threats [25].

Locally Interpretable Model Agnostic Explanations(LIME) is designed to create an intelligible model that utilizes an easily understandable representation while preserving local fidelity to the original classifier[28]. Given an instance with its original representation, $x \in Rd$, and an explanation model, $g \in G$, where G represents a set of visually expressible, interpretable models (e.g. a linear model), the explanation provided by LIME can be formulated as follows:

$$\varphi(x) = \arg\min L\,[(f, g, \omega x) + \Omega(g)] \quad (1)$$

In (1), f symbolizes the classification model, ωx denotes a similarity measure between the original and new instances (with higher values indicating greater similarity), L is the loss function that assesses the proximity of the predictions between the explanation and original models, and $\Omega(g)$ quantifies the complexity of model g. LIME strives to develop a model that is both locally focused and interpretable. To accomplish this, LIME minimizes the function $L[(f, g, \omega x) + \Omega(g)]$, where f is the original model, g is the locally derived interpretation model, and ωx is a weight vector for instance x. The regularization term $\Omega(g)$ assists in preventing overfitting of the interpretation model. Upon minimizing the objective function, LIME generates an explanation for a specific instance using the locally derived interpretation model $\phi(x)$. The interpretation

model $\phi(x)$ is intended to be straightforward and transparent, which makes it more accessible for humans to grasp the reasoning behind a particular prediction. By employing a locally focused and interpretable model, LIME offers insights into the decision making processes of complex models.

D. Ensemble Techniques

Ensemble techniques combine the predictions of multiple base models to improve the performance and accuracy of the final predictions [29]. In this study, we employed two widely used ensemble methods: Hard Voting and Soft Voting. Both methods utilize the outputs of multiple classifiers to produce a more robust and accurate final decision.

- Majority Voting: Majority Voting, also known as Hard Voting, is a straightforward ensemble method that considers the class labels predicted by individual base models. For each input, the class label that receives the majority of votes from the base models is chosen as the final prediction [30]. This approach is particularly effective when the base models have complementary strengths and weaknesses, as it enables the ensemble to collectively capitalize on their individual advantages.
- Probabilistic Voting: Probabilistic Voting, or Soft Voting, is a more sophisticated ensemble technique that takes into account the predicted class probabilities generated by the base models [31]. Instead of simply counting the votes for each class label, Soft Voting computes the average of the predicted probabilities for each class and selects the class with the highest average probability as the final prediction [32]. This method can yield better performance than Hard Voting, especially when the base models provide well calibrated probability estimates.

III. ETHEREUM FRAUD DETECTION USING MACHINE LEARNING

BC technologies are being implemented across various public and private sectors due to their ability to secure and monitor auditing systems effectively [14]. These technologies facilitate the evaluation of data repositories, allowing auditors to submit queries in a secure and user-friendly manner without disclosing their identities to unauthorized parties [33]. In [34], consensus algorithms are utilized to confirm the validity of conducted transactions; however, they fall short in accurately identifying transactions. Consequently, solely depending on BC for fraud detection is not a comprehensive solution. To tackle this issue, alternative strategies, such as the application of ML algorithms, are explored to eliminate existing system vulnerabilities. Multiple supervised ML techniques are deployed to detect fraudulent transactions, and an extensive comparison of these approaches is conducted. KaRuNa [35] is a decentralized framework that leverages BC technology fr sentiment analysis to detect fraudulent cryptocurrency schemes. Operating on a public BC, KaRuNa is composed of three trust modeling phases involving stakeholders. The initial phase includes transaction execution on the BC, which fosters trust, auditability, and transparency among participants. The subsequent phase proposes a sentiment analysis of cryptocurrencies, employing a unique hashing algorithm for addresses to produce classification scores. This phase takes into account various factors such as social trends, cryptocurrency price volatility, calculated standard deviation, and peak and trough values. These factors are integrated into a novel Long-Short Term Memory (LSTM) classifier to generate recommendations. Impressively, the LSTM classifier demonstrates an accuracy of 98.99% in evaluating investment risks based on the generated classification scores.

By analyzing 2,179 accounts flagged for illicit activity and 2,502 normal accounts within the Ethereum community, authors in [3] aim to detect malicious accounts based on their transaction history using the XGBoost classifier. Through 10-fold cross-validation, XGBoost achieved an average accuracy of 0.963 (± 0.006) and an average AUC of 0.994 (± 0.0007). The most influential features for the final model were identified as "Time diff between first and last (μins)," "Total Ether balance," and "Min value received." Authors in [36] explore the application of various supervised ML approaches to identify and differentiate between fraudulent and legitimate transactions. An in depth comparative analysis of several supervised ML algorithms is conducted for this purpose. It was found that the most effective results (accuracy: 97%) were achieved using Ada Boost, Support Vector Machine (SVM), and Random Forest (RF) classifiers, outperforming the other seven algorithms examined. In [37], the researchers focused on extracting more comprehensive features from Bitcoin transaction data to improve the detection of fraudulent activities. To address the issue of imbalanced data, measures were taken to equalize the dataset. Several supervised methods, including KNN, SVM, RF, AdaBoost, and MLP, were employed. Additionally, three unsupervised methods, namely the One Class SVM (OCSVM), Local Outlier Factor (LOF), and Mahalanobis Distance Based (MDB) approach, were utilized for detection purposes. The best performing algorithm among these approaches was the RF, exhibiting impressive recall, precision, and F1 scores of 95.9%. These findings demonstrate the efficacy of the RF algorithm in accurately identifying fraudulent activities in Bitcoin transactions. In a related study [33], supervised methods including RF, SVM, and XGBoost were employed to detect fraudulent accounts in the Ethereum BC. The study successfully achieved high recall and precision values, which enabled the design of an effective antifraud rule for digital wallets or currency exchanges. This study introduces a novel detection mechanism called Ethereum Phishing Scam Detection (Eth-PSD) aimed at identifying phishing scam related transactions using an ML based approach [38]. Eth-PSD addresses various limitations present in existing works, including imbalanced datasets, complex feature engineering, and lower detection accuracy. Additionally, an investigation was conducted into constructing a new, updated, and balanced dataset specifically designed for effectively evaluating Eth-PSD performance. The experimental results demonstrate that Eth-PSD exhibits high efficiency in detecting phishing scams on the Ethereum platform, achieving a remarkable detection accuracy of 98.11% while maintaining a very low False Positive Rate of 0.01. By mitigating the shortcomings of previous approaches, Eth-PSD presents a promising solution for combating phishing scams in the Ethereum ecosystem. Authors in [40] present an innovative approach that leverages the integration of ML techniques and

BC technology to combat fraud in the healthcare sector, specifically in claims processing. The proposed methodology revolves around the utilization of a decision tree classification algorithm to accurately categorize the original claims dataset. Subsequently, the acquired knowledge is translated into a smart contract deployed on the Ethereum BC, facilitating the detection and prevention of healthcare fraud. Through meticulous comparative experiments, the results underscore the remarkable performance of the proposed system, achieving an impressive classification accuracy of 97.96% and an exceptional sensitivity of 98.09%. These outcomes highlight the profound impact of the BC smart contract in fortifying fraud detection capabilities, leading to an unparalleled accuracy rate of 97.96%. Authors in [40] introduce ContractWard, an original approach aimed at identifying vulnerabilities in smart contracts through the utilization of machine learning techniques. By leveraging bigram features extracted from simplified operation codes of smart contracts, ContractWard constructs robust detection models using a combination of five distinct ML algorithms and two sampling algorithms. Notably, the evaluation conducted on a comprehensive dataset of real-world smart contracts deployed on Ethereum showcases ContractWard's exceptional performance, achieving remarkably high predictive scores of over 96% for both Micro-F1 and Macro-F1 metrics. Moreover, ContractWard demonstrates impressive efficiency, with an average detection time of 4 s per smart contract when trained with XGBoost whereas it was balanced with the SMOTE Tomek. These findings unequivocally underscore the efficacy and proficiency of ContractWard in effectively identifying vulnerabilities and fortifying the security of smart contracts.

IV. PROPOSED SCHEME

In our methodology (Figure 1), we will begin by collecting the EFD dataset. This dataset will undergo preprocessing to prepare the data for analysis. Following that, we will use three different algorithms for making predictions: Random Forest (RF), AdaBoost, and Decision Tree (DT). Each of these algorithms will produce its own prediction (Pred1, Pred2, and Pred3, respectively). Once the predictions are made, we will apply Hard Voting and Soft Voting, leading to two sets of predictions (Pred_H and Pred_S). The predictions will then be evaluated to assess the performance of our ensemble method. As part of the evaluation, we will also employ techniques from XAI to interpret the results and provide insights into how the decisions were made by the ensemble.

A. Dataset

The dataset [41] we used in our study provides a comprehensive view of Ethereum transactions, with a specific focus on identifying fraudulent activities. It encompasses various attributes ranging from basic account information to more detailed transactional data. Each record in the dataset represents an Ethereum account and includes the unique account address and a flag indicating whether transactions from the account are fraudulent. Transaction timings are dissected into average minutes between transactions, both sent and received, as well as the time difference between the first and last transaction, illustrating account activity over time.

The dataset quantifies the number of transactions sent and received, including the creation of contract transactions, which are crucial for understanding account behavior. It also delves into the uniqueness of interaction by accounting for the total unique addresses involved in both incoming and outgoing transactions. The transactional values are broken down into minimum, maximum, and average Ether values sent and received, providing a financial profile of each account's activity. This extends to transactions involving contracts, with separate metrics for Ether sent to contracts, thus highlighting the role of smart contracts in the transactional ecosystem of an account.

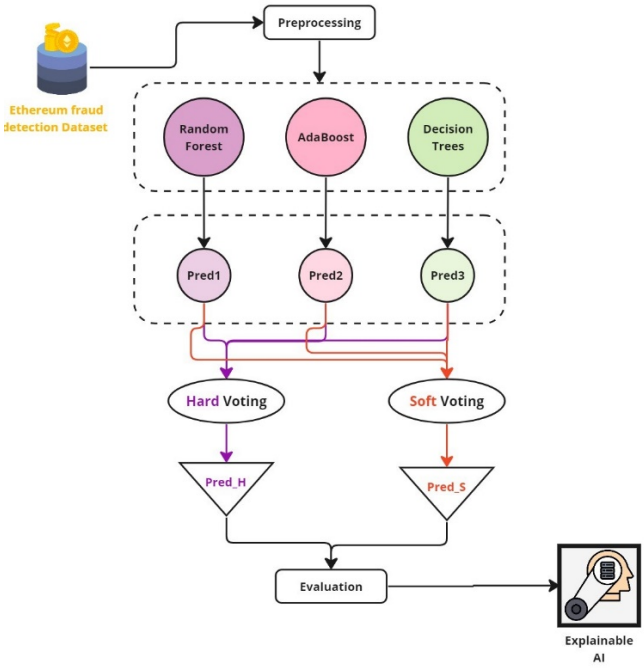

Fig. 1. The proposed methodology.

Moreover, the dataset provides a holistic view of an account's Ether flow, summarizing total transactions, total Ethers sent and received, and the overall balance of Ether post-transactions. It paints a detailed picture of an account's engagement with ERC20 token transactions, including the total number of these transactions, the Ether value of ERC20 tokens sent and received, and interactions with contract addresses specifically for ERC20 tokens. The dataset does not just stop at quantity but also explores the diversity of tokens associated with the accounts, listing the number of unique ERC20 tokens sent and received, as well as the most frequently sent and received token types. This rich dataset serves as the backbone of our analysis, providing us with the necessary depth and breadth of data to accurately model and detect fraudulent transactions in the Ethereum network.

B. Data Pre-processing

Data preprocessing stands as a pivotal phase in the deployment of ML methodologies, aiming to enhance model accuracy and accelerate the learning process. This stage is critical for the elimination of non-contributing attributes, the

resolution of missing values, and the rectification of imbalances within the dataset, which are essential to optimize the predictive prowess of the algorithms employed. In the realm of data cleansing, we judiciously excised three variables from our dataset deemed non-essential for the task of EFD. These excluded features—specifically the "Address," "ERC20 most sent token type," and "ERC20 most received token type" were identified as extraneous in the context of our analytical objectives, thereby streamlining the feature space to those of greater pertinence[43]. Addressing the prevalence of incomplete records, we invoked the capabilities of the datasets library, a Python toolkit tailored for facile data operations and exploration. We engaged the "fill missing num" functionality to impute voids within numerical attributes, substituting absent values with the feature's mean. Concurrently, the "fill missing cats" mechanism was employed to rectify gaps in categorical features, defaulting missing entries to the mode of the respective attribute.

Given the inherent challenge posed by skewed class distributions, our dataset's imbalance was manifest, with a scant proportion of fraudulent transactions (2,179 out of 9,841). To mitigate the bias towards the predominant class and enhance the model sensitivity to the minority class, we implemented two resampling strategies: Random Under Sampling (RUS) and Synthetic Minority Over-sampling Technique (SMOTE). These techniques are instrumental in recalibrating the dataset to a more balanced composition, thereby facilitating a more equitable and nuanced learning process

C. Model Implementation

In our ensemble framework, we methodically implemented three machine learning models: RF, DT, and AdaBoost, each chosen for its distinct characteristics and ability to uncover patterns of fraudulent activity in the Ethereum dataset.

The RF model, known for its proficiency in managing datasets with numerous features, was applied by constructing a series of DTs, each on different data and feature subsets. This ensemble approach reduces overfitting risks inherent in single DTs by averaging out biases and errors, which results in improved prediction accuracy. We tailored the RF hyperparameters, such as the number of trees and the maximum depth of the trees, to suit the complexity of our data.

In the case of the DT model, we capitalized on its straightforward structure to build a model that splits the data based on certain feature thresholds. These splits were carefully chosen to maximize the differentiation between fraudulent and non-fraudulent transactions. The simplicity of a single DT allows for easy interpretation of decision paths but requires careful monitoring to prevent overfitting to the training data.

The AdaBoost algorithm was chosen for its adaptive qualities, boosting the classification capabilities of weak learners, typically DTs in our case. AdaBoost focuses on instances that have been challenging for previous models, iteratively adjusting the weights of these instances. By doing so, subsequent models prioritize these difficult cases in the training process. The final model is then a composite of these individual learners, weighted by their accuracy, to form a robust classifier. For AdaBoost, we selected an appropriate number of DTs to serve as weak learners, ensuring they collectively form a strong predictive model without overcomplicating the learning process.

For each model, we meticulously adjusted their respective hyperparameters, such as the number of estimators for RF and AdaBoost and the depth for DT, according to the specific demands of our dataset. This fine-tuning was essential to balance the bias-variance tradeoff, aiming to maximize the predictive performance while maintaining the models' abilities to generalize new data. The individual strengths of RF, DT, and AdaBoost, when integrated within our ensemble strategy, resulted in a comprehensive and robust fraud detection mechanism.

D. Ensemble Learning

Ensemble learning in the context of our fraud detection system is the strategic combination of multiple machine learning models to improve the overall predictive performance. This approach leverages the strengths of individual models to create a collective decision-making entity that is more accurate and reliable than any single model alone. The ensemble technique is particularly effective in complex problems like fraud detection, where the nuances of fraudulent behavior can be difficult for a single model to capture.

Our ensemble consists of three well-established algorithms, RF, DT, and AdaBoost. The RF model contributes its ensemble of DTs that individually assess different aspects of the data, thus offering a comprehensive view through majority voting. The strength of this model lies in its ability to mitigate the overfitting problem commonly seen in individual DTs by averaging multiple decision paths. The DT model brings clarity and interpretability to the ensemble. It offers a depth of analysis at each decision node, which is valuable for understanding the specific conditions that lead to a classification of fraud. The tree's structure provides clear rules and criteria for decision-making, which can be crucial for stakeholders requiring insight into the model's reasoning. AdaBoost complements the ensemble by focusing on instances that are difficult to classify. It operates by sequentially applying weak learners and adjusting their focus on misclassified instances from the previous models. The learning algorithm adapts by giving more weight to the misjudged instances, ensuring that subsequent learners give them more attention. The resulting combination of these learners forms a potent model that can handle varied and complex fraud patterns.

In our ensemble, each model votes on the outcome of a transaction being fraudulent or not, with the final decision achieved through aggregation methods, like hard voting or soft voting. Hard voting considers the most common outcome predicted by the models, while soft voting takes into account the confidence level of each prediction. This harmonization of diverse models and voting mechanisms ensures that the ensemble captures an expansive spectrum of data patterns and anomalies, leading to a robust and nuanced fraud detection system. By integrating these models' predictions, we aim to harness their collective intelligence, thereby enhancing the

accuracy and reliability of detecting fraud within the Ethereum network.

V. MODEL TRAINING AND EVALUATION

The dataset was divided into two portions: one for training and evaluation, accounting for 80% of the data, and the other for testing with unseen data, comprising the remaining 20%. For our experiments, we employed Google Colab, a collaborative Jupyter notebook platform, to conduct our ML research. We utilized Keras (version 2.12.0) and TensorFlow (version 2.12.0), as a backend engine. Our experimental setup in the Google Colab environment provided us with 13.62 GB of available RAM, offering a powerful and accessible resource for implementing and testing our models. The primary objective of our study was to design and evaluate ML models using the EFD dataset to categorize Ethereum transactions into two groups: legitimate and fraudulent. The "Flag" feature served as the target variable for training and evaluation. We assessed the performance of various ML methods by measuring metrics such as accuracy, training time, recall, precision, and F-score.

The performance results for the three individual ML classifiers and two ensemble voting strategies used in EFD can be seen in Table I. The RF classifier showcases the highest accuracy of 98.7%, with a precision, recall, and F1 score of 0.99 for both fraudulent and legitimate labels. It, however, requires a considerable training time of 3.35 s. The DT classifier exhibits an accuracy of 97.3%, with corresponding precision, recall, and F1 scores of 0.97. Despite its slightly lower accuracy, it has the benefit of a relatively shorter training time of just 0.2 s. The AdaBoost classifier aligns closely with the DT classifier, with accuracy of 97.6% and similar precision, recall, and F1 scores, but at a longer training time of 2.32 s. When we consider the ensemble methods, both soft and hard voting mechanisms show promising results. The Soft Voting method achieves an accuracy of 98%, albeit with the longest training time of 4.59 s. The Hard Voting method, in contrast, achieves the highest accuracy of 99%, surpassing all individual classifiers and the Soft Voting method, while requiring a moderate training time of 3.85 s.

TABLE I. ETHEREUM FRAUD DETECTION RESULTS

Classifier	ACC	Training Time (s)	Label	PREC	REC	F-S
RF	98.7%	3.35	Fraudulent	0.99	0.99	0.99
			Legitimate	0.99	0.99	0.99
DT	97.3%	0.2	Fraudulent	0.97	0.97	0.97
			Legitimate	0.97	0.97	0.97
AdaBoost	97.6%	2.32	Fraudulent	0.98	0.97	0.98

Based on these outcomes, the Ensemble Hard Voting classifier emerges as the most effective model for EFD due to its highest accuracy of 99%. Although its training time is not the shortest, the superior accuracy compensates for the extra computational resources, given the critical nature of fraud detection in BC transactions.

The results underscore the importance of ensemble methods in enhancing the robustness and accuracy of fraud detection models, thus contributing significantly to the security aspect of BC technologies.

The comparative evaluation of our proposed Ensemble Hard Voting classifier against other methods presented in the existing literature further underscores its superior performance (Table II). With an ACC of 99%, the proposed model surpasses the accuracies reported in [3, 36, 38, 39, 42]. Not only does this comparison illustrate the efficiency of the proposed model, but also demonstrates its contribution to the ongoing research in EFD. The incorporation of ensemble methods and specifically the Hard Voting classifier enhances the robustness and accuracy of the detection system, thereby strengthening the security framework of BC transactions. These findings emphasize the growing potential of ML-driven solutions in managing the intricate cybersecurity challenges within the constantly evolving BC sphere.

TABLE II. COMPARATIVE ANALYSIS OF EFD ACCURACY: PROPOSED VS. EXISTING METHODS

Reference	ACC
Proposed	99%
[3]	96.3%
[36]	98%
[42]	80.2%
[38]	98.11%
[39]	97.96%

The significance of explainability in AI cannot be overstated, particularly when it pertains to high-stake domains, such as fraud detection. A fundamental step towards achieving transparency in AI models is the examination of feature importance, which sheds light on the variables that most significantly influence the model's predictions.

In our analysis (Figure 2), the RF classifier has highlighted the differential impact of various features. The prominence of "Time diff between first and last (mins)" at the apex of importance underscores its pivotal role in identifying potential fraud. Such insights are invaluable, as not only do they inform the model's predictive behavior, but also provide stakeholders with an understanding of the underlying factors that the AI considers indicative of fraudulent activities. The feature importance graph also accentuates the relevance of ERC20 token transactions, with features such as "ERC20 min val rec," "ERC20 max val rec," and "ERC20 avg val rec" ranking highly. The significance attributed to these features reveals the model's sensitivity to the nuances of smart contract interactions on the Ethereum platform. By delineating the hierarchy of feature importance, XAI demystifies the model's decision-making process, enabling a more informed and transparent evaluation of the model's outputs. Not only does this approach ensure that the most impactful features are incorporated into the model, enhancing its accuracy, but also fosters trust and credibility in the AI system by making its inner workings more accessible to users and practitioners.

The LIME technique stands as a pivotal tool in the realm of explainable AI, providing clarity on how predictive models make their decisions. It achieves this by generating local surrogate models that approximate the predictions of the complex model in a comprehensible way. The bar chart

produced by LIME offers an intuitive visualization of the importance of individual features for a particular prediction. In the context of our fraud detection model, LIME elucidates the contributions of specific features to the classification of a transaction (Figure 3). For instance, the "Unique received from addresses" feature has a significant positive impact on classifying a transaction as class 1, which could represent a fraudulent transaction. Similarly, "ERC20 min val sent" and "ERC20 total Ether sent contract" are also influential in the model's prediction, as depicted by their weight in the chart. The negative weight of "Time diff between first and last (mins)" suggests that shorter time differences may decrease the likelihood of a transaction being classified as class 1.

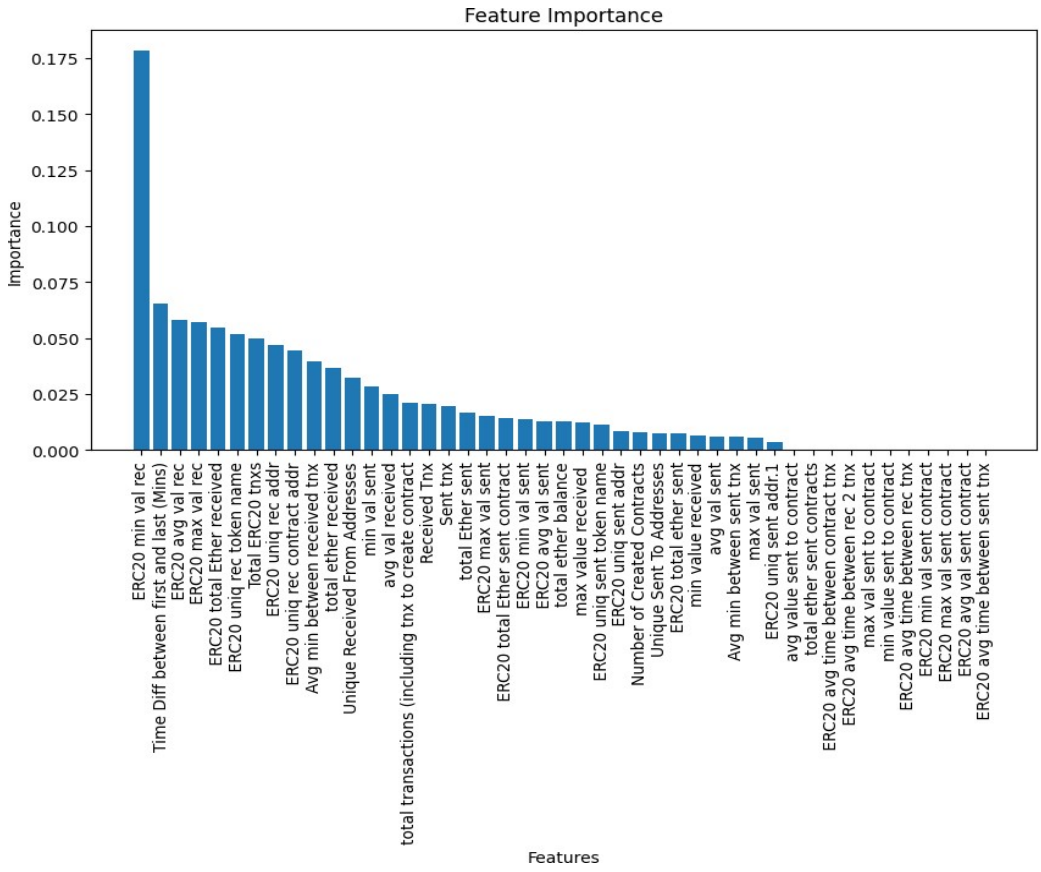

Fig. 2. Feature importance rankings utilizing the RF classifier.

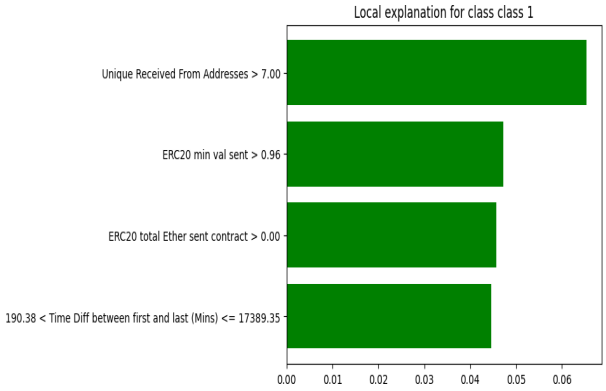

Fig. 3. Feature importance rankings utilizing the LIME technique.

LIME's granular explanations offer valuable insights, particularly in scenarios where model trustworthiness and understanding are critical. By dissecting the model's predictions into interpretable contributions from each feature, stakeholders can gain a nuanced understanding of the model's behavior, which is instrumental for validation and trust in AI applications.

VI. CONCLUSION AND FUTURE WORK

This study has underscored the power of ML techniques, notably the Hard Voting ensemble model, in detecting fraudulent transactions within the complex world of cryptocurrencies. The research has illuminated the capability of these methodologies in identifying questionable activities, while simultaneously underscoring the value of Explainable AI (XAI) in ensuring transparency, trust, and accountability within AI- empowered fraud detection systems. One significant direction is the proposed model application in real-time scenarios. With its exceptional performance, implementing the Hard Voting ensemble model to oversee live transactions within the BC could prove to be profoundly beneficial. This real-time deployment could potentially deliver immediate and accurate fraud detection, thereby substantially improving the

security and dependability of BC ecosystems. Not only does the research showcase the theoretical potential of ML solutions, but also underscores the imperative need to translate these theories into practice to enhance the burgeoning domain of digital currencies.

REFERENCES

[1] F. Allen, X. Gu, and J. Jagtiani, "Fintech, Cryptocurrencies, and CBDC: Financial Structural Transformation in China," *Journal of International Money and Finance*, vol. 124, Jun. 2022, Art. no. 102625, https://doi.org/10.1016/j.jimonfin.2022.102625.

[2] A. Raja Santhi and P. Muthuswamy, "Influence of Blockchain Technology in Manufacturing Supply Chain and Logistics," *Logistics*, vol. 6, no. 1, Mar. 2022, Art. no. 15, https://doi.org/10.3390/logistics6010015.

[3] S. Farrugia, J. Ellul, and G. Azzopardi, "Detection of illicit accounts over the Ethereum blockchain," *Expert Systems with Applications*, vol. 150, Jul. 2020, Art. no. 113318, https://doi.org/10.1016/j.eswa.2020.113318.

[4] P. Bains, *Blockchain Consensus Mechanisms: A Primer for Supervisors*. Washington, DC, USA: International Monetary Fund, 2022.

[5] M. Ul Hassan, M. H. Rehmani, and J. Chen, "Anomaly Detection in Blockchain Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 289–318, 2023, https://doi.org/10.1109/COMST.2022.3205643.

[6] Y. Huang and M. Mayer, "Digital currencies, monetary sovereignty, and U.S.–China power competition," *Policy & Internet*, vol. 14, no. 2, pp. 324–347, 2022, https://doi.org/10.1002/poi3.302.

[7] T. Volety, S. Saini, T. McGhin, C. Z. Liu, and K.-K. R. Choo, "Cracking Bitcoin wallets: I want what you have in the wallets," *Future Generation Computer Systems*, vol. 91, pp. 136–143, Feb. 2019, https://doi.org/10.1016/j.future.2018.08.029.

[8] J. Osterrieder, S. Chan, J. Chu, and Y. Zhang, "A Primer on Anomaly and Fraud Detection in Blockchain Networks." SSRN, Rochester, NY, USA, Jan. 04, 2023, https://doi.org/10.2139/ssrn.4317520.

[9] T. Hewa, M. Ylianttila, and M. Liyanage, "Survey on blockchain based smart contracts: Applications, opportunities and challenges," *Journal of Network and Computer Applications*, vol. 177, Mar. 2021, Art. no. 102857, https://doi.org/10.1016/j.jnca.2020.102857.

[10] P. Sharma, R. Jindal, and M. D. Borah, "Blockchain-based decentralized architecture for cloud storage system," *Journal of Information Security and Applications*, vol. 62, Nov. 2021, Art. no. 102970, https://doi.org/10.1016/j.jisa.2021.102970.

[11] M. Niranjanamurthy, B. N. Nithya, and S. Jagannatha, "Analysis of Blockchain technology: pros, cons and SWOT," *Cluster Computing*, vol. 22, no. 6, pp. 14743–14757, Nov. 2019, https://doi.org/10.1007/s10586-018-2387-5.

[12] G.-T. Nguyen and K. Kim, "A Survey about Consensus Algorithms Used in Blockchain," *Journal of Information Processing Systems*, vol. 14, no. 1, pp. 101–128, 2018, https://doi.org/10.3745/JIPS.01.0024.

[13] J. Bernal Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. Torres Moreno, and A. Skarmeta, "Privacy-Preserving Solutions for Blockchain: Review and Challenges," *IEEE Access*, vol. 7, pp. 164908–164940, 2019, https://doi.org/10.1109/ACCESS.2019.2950872.

[14] M. E. Khatib, A. A. Mulla, and W. A. Ketbi, "The Role of Blockchain in E-Governance and Decision-Making in Project and Program Management," *Advances in Internet of Things*, vol. 12, no. 3, pp. 88–109, Jul. 2022, https://doi.org/10.4236/ait.2022.123006.

[15] A. B. Haque, A. K. M. N. Islam, S. Hyrynsalmi, B. Naqvi, and K. Smolander, "GDPR Compliant Blockchains–A Systematic Literature Review," *IEEE Access*, vol. 9, pp. 50593–50606, 2021, https://doi.org/10.1109/ACCESS.2021.3069877.

[16] N. O. Nawari and S. Ravindran, "Blockchain and the built environment: Potentials and limitations," *Journal of Building Engineering*, vol. 25, Sep. 2019, Art. no. 100832, https://doi.org/10.1016/j.jobe.2019.100832.

[17] Y. Chen and C. Bellavitis, "Blockchain disruption and decentralized finance: The rise of decentralized business models," *Journal of Business Venturing Insights*, vol. 13, Jun. 2020, Art. no. e00151 https://doi.org/10.1016/j.jbvi.2019.e00151.

[18] C. Denis Gonzalez, D. Frias Mena, A. Masso Munoz, O. Rojas, and G. Sosa-Gomez, "Electronic Voting System Using an Enterprise Blockchain," *Applied Sciences*, vol. 12, no. 2, Jan. 2022, Art. no. 531, https://doi.org/10.3390/app12020531.

[19] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, "A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain," *IEEE Access*, vol. 5, pp. 17465–17477, 2017, https://doi.org/10.1109/ACCESS.2017.2720760.

[20] C. Karapapas, G. Syros, I. Pittaras, and G. C. Polyzos, "Decentralized NFT-based Evolvable Games," in *4th Conference on Blockchain Research & Applications for Innovative Networks and Services*, Paris, France, Sep. 2022, pp. 67–74, https://doi.org/10.1109/BRAINS55737.2022.9909178.

[21] N. Sasikala, B. M. Sundaram, S. Biswas, A. Sai Nikhil, and V. S. Rohith, "Survey of latest technologies on Decentralized applications using Blockchain," in *Second International Conference on Artificial Intelligence and Smart Energy*, Coimbatore, India, Feb. 2022, pp. 1432–1436, https://doi.org/10.1109/ICAIS53314.2022.9742768.

[22] J. P. Cruz, Y. Kaji, and N. Yanai, "RBAC-SC: Role-Based Access Control Using Smart Contract," *IEEE Access*, vol. 6, pp. 12240–12251, 2018, https://doi.org/10.1109/ACCESS.2018.2812844.

[23] F. A. Aponte-Novoa, A. L. S. Orozco, R. Villanueva-Polanco, and P. Wightman, "The 51% Attack on Blockchains: A Mining Behavior Study," *IEEE Access*, vol. 9, pp. 140549–140564, 2021, https://doi.org/10.1109/ACCESS.2021.3119291.

[24] G. Alicioglu and B. Sun, "A survey of visual analytics for Explainable Artificial Intelligence methods," *Computers & Graphics*, vol. 102, pp. 502–520, Feb. 2022, https://doi.org/10.1016/j.cag.2021.09.002.

[25] A. Barredo Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, Jun. 2020, https://doi.org/10.1016/j.inffus.2019.12.012.

[26] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable Artificial Intelligence (XAI) to Enhance Trust Management in Intrusion Detection Systems Using Decision Tree Model," *Complexity*, vol. 2021, Jan. 2021, Art. no. e6634811, https://doi.org/10.1155/2021/6634811.

[27] T. Zahavy, N. Ben-Zrihem, and S. Mannor, "Graying the black box: Understanding DQNs," in *33 rd International Conference on Machine Learning*, New York, NY, USA, Jun. 2016, pp. 1899–1908.

[28] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?': Explaining the Predictions of Any Classifier," in *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 1135–1144, https://doi.org/10.1145/2939672.2939778.

[29] O. Sagi and L. Rokach, "Ensemble learning: A survey," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 4, 2018, Art. no. e1249, https://doi.org/10.1002/widm.1249.

[30] A. Khatri, S. Agrawal, and J. M. Chatterjee, "Wheat Seed Classification: Utilizing Ensemble Machine Learning Approach," *Scientific Programming*, vol. 2022, Feb. 2022, Art. no. e2626868, https://doi.org/10.1155/2022/2626868.

[31] M. U. Salur and I. Aydın, "A soft voting ensemble learning-based approach for multimodal sentiment analysis," *Neural Computing and Applications*, vol. 34, no. 21, pp. 18391–18406, Nov. 2022, https://doi.org/10.1007/s00521-022-07451-7.

[32] G. P. de Oliveira, A. Fonseca, and P. C. Rodrigues, "Diabetes diagnosis based on hard and soft voting classifiers combining statistical learning models," *Brazilian Journal of Biometrics*, vol. 40, no. 4, pp. 415–427, Dec. 2022, https://doi.org/10.28951/bjb.v40i4.605.

[33] A. Khanna *et al.*, "Blockchain: Future of e-Governance in Smart Cities," *Sustainability*, vol. 13, no. 21, Jan. 2021, Art. no. 11840, https://doi.org/10.3390/su132111840.

[34] M. Ostapowicz and K. Zbikowski, "Detecting Fraudulent Accounts on Blockchain: A Supervised Approach," in *International Conference on Web Information Systems Engineering*, Hong Kong, China, Jan. 2020, pp. 18–31, https://doi.org/10.1007/978-3-030-34223-4_2.

[35] P. N. Sureshbhai, P. Bhattacharya, and S. Tanwar, "KaRuNa: A Blockchain-Based Sentiment Analysis Framework for Fraud Cryptocurrency Schemes," in *IEEE International Conference on Communications Workshops*, Dublin, Ireland, Jun. 2020, pp. 1–6, https://doi.org/10.1109/ICCWorkshops49005.2020.9145151.

[36] M. Bhowmik, T. Sai Siri Chandana, and B. Rudra, "Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain," in *5th International Conference on Computing Methodologies and Communication*, Erode, India, Apr. 2021, pp. 539–541, https://doi.org/10.1109/ICCMC51019.2021.9418470.

[37] B. Chen, F. Wei, and C. Gu, "Bitcoin Theft Detection Based on Supervised Machine Learning Algorithms," *Security and Communication Networks*, vol. 2021, Feb. 2021, Art. no. e6643763, https://doi.org/10.1155/2021/6643763.

[38] A. H. H. Kabla, M. Anbar, S. Manickam, and S. Karupayah, "Eth-PSD: A Machine Learning-Based Phishing Scam Detection Approach in Ethereum," *IEEE Access*, vol. 10, pp. 118043–118057, 2022, https://doi.org/10.1109/ACCESS.2022.3220780.

[39] A. A. Amponsah, A. F. Adekoya, and B. A. Weyori, "A novel fraud detection and prevention method for healthcare claim processing using machine learning and blockchain technology," *Decision Analytics Journal*, vol. 4, Sep. 2022, Art. no. 100122, https://doi.org/10.1016/j.dajour.2022.100122.

[40] W. Wang, J. Song, G. Xu, Y. Li, H. Wang, and C. Su, "ContractWard: Automated Vulnerability Detection Models for Ethereum Smart Contracts," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1133–1144, Apr. 2021, https://doi.org/10.1109/TNSE.2020.2968505.

[41] V. Aliyev, "Ethereum Fraud Detection Dataset." kaggle, 2020, [Online]. Available: https://www.kaggle.com/datasets/vagifa/ethereum-frauddetection-dataset.

[42] Y. Elmougy and O. Manzi, "Anomaly Detection on Bitcoin, Ethereum Networks Using GPU-accelerated Machine Learning Methods," in *31st International Conference on Computer Theory and Applications*, Alexandria, Egypt, Dec. 2021, pp. 166–171, https://doi.org/10.1109/ICCTA54562.2021.9916625.

[43] O. M. Ahmed, L. M. Haji, A. M. Ahmed, and N. M. Salih, "Bitcoin Price Prediction using the Hybrid Convolutional Recurrent Model Architecture," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11735–11738, Oct. 2023, https://doi.org/10.48084/etasr.6223.

[44] N. K. Al-Shammari, T. H. Syed, and M. B. Syed, "An Edge – IoT Framework and Prototype based on Blockchain for Smart Healthcare Applications," *Engineering, Technology & Applied Science Research*, vol. 11, no. 4, pp. 7326–7331, Aug. 2021, https://doi.org/10.48084/etasr.4245.

[45] K. Rajeshkumar, C. Ananth, and N. Mohananthini, "Blockchain-Assisted Homomorphic Encryption Approach for Skin Lesion Diagnosis using Optimal Deep Learning Model," *Engineering, Technology & Applied Science Research*, vol. 13, no. 3, pp. 10978–10983, Jun. 2023, https://doi.org/10.48084/etasr.5594.