big data and cognitive computing

MDPI

Article

AHEAD: A Novel Technique Combining Anti-Adversarial Hierarchical Ensemble Learning with Multi-Layer Multi-Anomaly Detection for Blockchain Systems

Muhammad Kamran [1,2], Muhammad Maaz Rehan [1,*], Wasif Nisar [1] and Muhammad Waqas Rehan [3]

[1] Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah 47040, Pakistan; kamran.uow@gmail.com (M.K.); wasif@ciitwah.edu.pk (W.N.)
[2] Department of Computer Science, Air University Islamabad, Aerospace and Aviation Campus Kamra, Attock 43600, Pakistan
[3] Institute for Software Engineering and Programming Languages (ISP), University of Lübeck, 23562 Lübeck, Germany; waqas.rehan@isp.uni-luebeck.de
* Correspondence: maazrehan@gmail.com or maazrehan@ciitwah.edu.pk

Abstract: Blockchain technology has impacted various sectors and is transforming them through its decentralized, immutable, transparent, smart contracts (automatically executing digital agreements) and traceable attributes. Due to the adoption of blockchain technology in versatile applications, millions of transactions take place globally. These transactions are no exception to adversarial attacks which include data tampering, double spending, data corruption, Sybil attacks, eclipse attacks, DDoS attacks, P2P network partitioning, delay attacks, selfish mining, bribery, fake transactions, fake wallets or phishing, false advertising, malicious smart contracts, and initial coin offering scams. These adversarial attacks result in operational, financial, and reputational losses. Although numerous studies have proposed different blockchain anomaly detection mechanisms, challenges persist. These include detecting anomalies in just a single layer instead of multiple layers, targeting a single anomaly instead of multiple, not encountering adversarial machine learning attacks (for example, poisoning, evasion, and model extraction attacks), and inadequate handling of complex transactional data. The proposed AHEAD model solves the above problems by providing the following: (i) data aggregation transformation to detect transactional and user anomalies at the data and network layers of the blockchain, respectively, (ii) a Three-Layer Hierarchical Ensemble Learning Model (HELM) incorporating stratified random sampling to add resilience against adversarial attacks, and (iii) an advanced preprocessing technique with hybrid feature selection to handle complex transactional data. The performance analysis of the proposed AHEAD model shows that it achieves higher anti-adversarial resistance and detects multiple anomalies at the data and network layers. A comparison of the proposed AHEAD model with other state-of-the-art models shows that it achieves 98.85% accuracy against anomaly detection on data and network layers targeting transaction and user anomalies, along with 95.97% accuracy against adversarial machine learning attacks, which surpassed other models.

Keywords: blockchain; anomaly detection; adversarial machine learning; cybersecurity

Citation: Kamran, M.; Rehan, M.M.; Nisar, W.; Rehan, M.W. AHEAD: A Novel Technique Combining Anti-Adversarial Hierarchical Ensemble Learning with Multi-Layer Multi-Anomaly Detection for Blockchain Systems. *Big Data Cogn. Comput.* **2024**, *8*, 103. https://doi.org/10.3390/bdcc8090103

Academic Editor: Michele Melchiori

Received: 28 May 2024
Revised: 20 August 2024
Accepted: 28 August 2024
Published: 2 September 2024

1. Introduction

Blockchain technology's decentralized, secure, and transparent nature has positively impacted various industries. The distributed ledger system of blockchain enables trustless record keeping by verifying and timestamping each transaction through a network of nodes. This decentralized structure allows for secure, transparent, and immutable transactions without a central authority, facilitating trust among parties in a transaction [1]. Blockchain technology not only laid the foundation of cryptocurrencies like Bitcoin and Ethereum but has also transformed several industries such as the Internet of things, supply chain

management, smart healthcare, education, fintech, energy sector, smart cities, e-governance, voting systems, and many more by changing the way data and assets are managed and exchanged [2]. Recent advancements in blockchain technology, such as web 3.0, non-fungible tokens, central bank digital currencies, blockchain-as-a-service, the metaverse, decentralized finance, green blockchain, and Ricardian Contracts, are impacting the evolution of technology globally [3]. Since then, research in blockchain technology has been growing.

Blockchain technology works on a decentralized network architecture, ensuring secure transactions through a distributed ledger spread across nodes. These nodes validate transactions using consensus algorithms, such as proof-of-work or proof-of-authority, depending on the blockchain type (public or private). Cryptography secures these transactions, with smart contracts automating processes according to predefined codes. The application layer provides a platform for applications based on blockchain, such as cryptocurrencies, and supply chains. The overall blockchain working process consists of these steps: broadcasting transactions, verifying them through consensus, grouping them into blocks secured by cryptographic hashes, and updating the distributed ledger with uniformity and integrity across the network [4].

The blockchain technology follows a structured approach that divides the blockchain into multiple layers, each with a distinct role. The data layer organizes the transaction blocks for transparency and security. The network layer creates consensus among participating nodes to maintain their integrity. The incentive layer executes the rewards policies. The contract layer ensures the deployment of smart contracts to facilitate decentralized agreements. The application layer provides a user-friendly interface for decentralized applications (dApps) for end-users of blockchain services. The execution layer executes smart contracts, ensuring trustless and decentralized operations. This layered approach helps create a robust, secure, transparent, and efficient blockchain architecture having a wide range of applications ranging from secure financial transactions to decentralized governance systems. A high-level overview of these layers is presented in Figure 1.

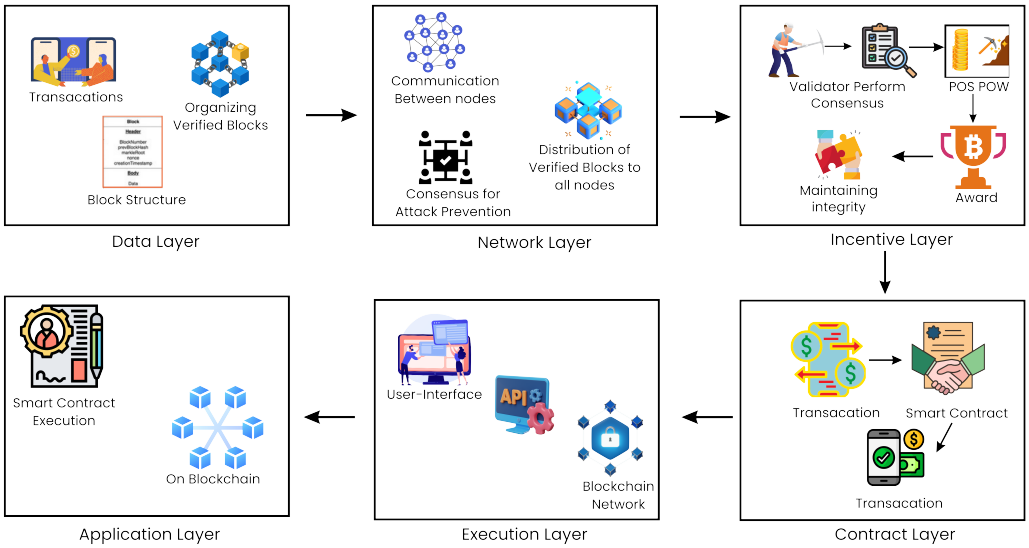

Figure 1. Blockchain layers and their functionalities.

An anomaly is an irregularity or deviation from the norm or expected pattern. It represents something that stands out as unusual or atypical compared to what is considered standard or normal [5]. Anomaly detection is playing a key role in cybersecurity by detecting intrusions and fraud, monitoring network performance, ensuring data quality, and predicting equipment failures. It is also used in healthcare to detect critical health events, in spam detection, and in surveillance to identify unusual activities [6]. This makes anomaly detection crucial for maintaining security, performance, and reliability across various applications.

Due to blockchain's decentralized nature and wide range of applications in various domains, blockchain technology faces some challenges due to some inherent vulnerabilities/anomalies like double-spending, where a single cryptocurrency is utilized multiple times, phishing attacks that aim to steal a user's private keys, and Ponzi schemes where returns are paid to earlier investors from the capital of newer ones, money laundering, scam operations, and so on [7]. Abnormal user behaviour poses a critical threat to the integrity and security of blockchain systems. These anomalous activities can lead to a huge loss for an organization if not handled properly, like financial losses due to fraud, loss of user confidence, exploitation of the network for illicit activities, and so forth. Thus, it is important to detect and mitigate these anomalies to ensure trust, integrity, and security in blockchain networks.

Adversarial attacks manipulate machine learning models, including those used for anomaly detection in blockchain. These attacks exploit vulnerabilities, leading to incorrect results [8]. Common types include the following: (i) poisoning attacks (contaminating training data), (ii) evasion attacks (manipulating input data), (iii) extraction attacks (stealing sensitive information), (iv) input manipulation attacks (modifying transactions to evade detection), (v) model inversion attacks (reverse-engineering model parameters), and (vi) impersonation attacks (mimicking legitimate users or transactions).

Adversarial attacks on blockchain anomaly detection models lead to financial losses, reputational damage, and compromised security. Developing robust and resilient models is crucial to preventing financial fraud, protecting user data, and maintaining transaction integrity. Anomaly detection models are critical in identifying suspicious transactions, such as money laundering or fraud. However, the entire blockchain ecosystem is at risk if these models are vulnerable to adversarial attacks. Therefore, developing adversarial-resistant models is essential to ensuring the security and trustworthiness of blockchain transactions.

The existing studies on anomaly detection techniques in blockchain focus on single aspects, such as a single type of anomaly (for example, transactions or users) and a single layer (for example, the data or network layer). This results in a narrow scope of detection, inefficient utilization of computational resources, and limited anomaly detection. The existing methods can only detect anomalous transactions; however, they fail to identify the anomalous user behind the transaction. Moreover, traditional algorithms for anomaly detection face challenges when working with the complex and imbalanced nature of blockchain datasets, which leads to less optimal performance. Another concern is the vulnerability of these ML models when they face adversarial attacks because adversaries can exploit the models to manipulate the network. Lastly, when detecting anomalies, the existing models may be biased because they depend only on one ML technique. This research aims to develop a Hierarchical Ensemble Learning Model (HELM) for an optimized, secure, and multi-layered ML-based anomaly detection mechanism tailored for blockchain networks to address the aforementioned challenges. The proposed AHEAD makes the following key contributions:

- *Anti-adversarial resilience of the proposed ML model* from attacks like data poisoning, evasion, extraction, and input manipulation based on the novel Three-Layer Hierarchical Ensemble Learning Model (HELM) employing stratified random sampling.
- *Multi-layered, multi-anomaly detection* at the data and network layers of blockchain, targeting both anomalous transactions and users.

This manuscript is organized as follows: The second section provides a comprehensive review of the existing literature. Section 3 introduces our proposed methodology, detailing the innovative approaches and techniques. Section 4 presents the evaluation of the proposed AHEAD model. Further, Section 5 compares AHEAD with existing ML mechanisms. Finally, in Section 6, we conclude the paper by summarizing the key insights derived from our study.

2. Literature Review

As discussed earlier, the decentralized nature and other properties of blockchain networks lead to their vast deployment in various domains from business to governance. The blockchain, however, faces similar challenges and security threats as other similar widely welcomed technologies could. This includes various attacks like fraudulent transactions, cyber-attacks, traffic bottlenecks, and so on. The research community has proposed a variety of solutions to address these challenges. These include strategies like machine learning approaches, graph-based approaches, real-time anomaly detection, network traffic analysis, and data mining approaches. This section discusses the limitations and efficiency of existing transaction-based and user-based anomalies.

2.1. Anomalous Transaction-Based Anomaly Detection

In blockchain networks, the integrity and security of transactions play an important role in maintaining trust and functionality. Transactional anomalies in blockchain refer to irregular or suspicious activities that deviate from expected patterns. This indicates potential fraud, theft, or other malicious intents. It is important to identify these anomalies to prevent financial losses and protect against cyber threats. This subsection explores various methodologies aimed at identifying and mitigating such anomalous transactions in blockchain networks.

2.1.1. Machine Learning Approaches

Several studies have applied machine learning techniques to improve anomaly detection within blockchain networks. For instance, the Isolation Forest (IF) algorithm has been employed to develop a method for automatically signing blockchain transactions [9]. This approach utilizes historical transaction data to automatically authenticate transactions. Only those transactions identified as anomalous require manual verification by a user. This method enhances the efficiency of the transaction process while ensuring security against fraudulent activities. However, personalized anomaly detection's effectiveness may be limited by the variability in transaction patterns across different users, requiring continual adaptation and refinement of the detection model to cater to individual user behaviours.

Efficient fraud detection [10] within Bitcoin transactions is proposed using XGBoost and random forest algorithms. It is validated by precision and AUC metrics; this approach demonstrates notable accuracy in fraud detection. However, its focus on a single anomaly and blockchain layer results in incomplete detection and inefficient computational use. Additionally, security analysis against adversarial attacks is performed using the Oyente tool. However, the model's robustness against actual adversarial inputs has not been tested. The model cannot fully address such sophisticated threats.

ADOBSVM [11], an anomaly detection model for Bitcoin transactions using a support vector machine (SVM), extracts comprehensive features from transaction data for refining labels to eliminate noise and employs an SVM to identify illicit activities. This model claims to enhance security and also optimize power consumption and execution time, but it lacks empirical evidence to prove these claims. Moreover, an SVM relies heavily on data points for classification purposes. This research uses too few features during training, raising the risk associated with false positives.

Theft detection [12] is employed in Bitcoin using unsupervised machine learning algorithms like K-nearest neighbour, support vector machine, random forest, Ada Boost, and multi-layer perception to predict Bitcoin transaction patterns, with random forest outperforming others by achieving recall, precision, and F1 values of 95.9%. However, focusing on specific aspects of anomalies may lead to inefficient computational use and incomplete detection. Additionally, the approach might not adequately address the threat of adversarial attacks, which could exploit vulnerabilities in the detection system.

2.1.2. Graph-Based Approaches

Graph-based approaches are notably effective in enhancing the detection of anomalies within blockchain networks. A method for detecting abnormal transactions, leveraging node and neighbourhood features through a random walk method [13], is proposed for enhanced anomaly detection. It combines these features to mine information from the network's structure and employs an unsupervised algorithm to identify and rank abnormal transactions. However, the research is limited to basic network metrics like degree centrality, which may affect the accuracy of feature extraction. Additionally, the reliance on unsupervised algorithms and the complexity of the fusion process may limit the model's precision.

The multi-layer temporal transaction [14] anomaly detection model employs a graph neural network approach for anomaly detection in Ethereum networks, combining multi-layer temporal analysis and graph convolutional networks for graph classification. However, the lack of detailed evaluation of large-scale, complex networks may challenge the model's scalability and generalizability.

A scalable anomaly detection method [15] was proposed for blockchain transactions, utilizing a sub-graph approach and GPU acceleration to identify illegal transactions. By focusing on partial blockchain data and employing parallel processing, this method reduces detection time. In evaluations with real Bitcoin data, it achieved an 11.1× speed improvement over previous GPU-based methods. This advancement enables a quicker response to fraudulent transactions, although its effectiveness is directly tied to the chosen sub-graph size and the algorithm's adaptability to transaction complexity.

2.1.3. Real-Time Anomaly Detection

Real-time anomaly detection in blockchain is crucial for identifying and mitigating threats in a timely manner. BAD [16], a blockchain anomaly detection framework, protects blockchain networks from unforeseen attacks by identifying and mitigating anomalous activities. This method leverages blockchain metadata to detect anomalies. Despite its novel approach to enhancing blockchain security, the framework's reliance on unsupervised detection algorithms may limit its ability to adapt to and identify new, evolving attack patterns. Continuous model evolution and integration of more dynamic detection mechanisms are required in it.

2.1.4. Network Traffic Analysis

A network traffic analysis approach was proposed for detecting anomalies in blockchain networks. An anomaly detection framework was developed for blockchain networks based on traffic monitoring [17], using a semi-supervised learning model with an AutoEncoder (AE) for profiling normal network behaviour. This approach focuses on network traffic statistics rather than blockchain ledger data and enables the detection of previously unseen attack patterns. The system demonstrated effective online detection of malicious activities, reducing time complexity for both training and testing phases by up to 66.8% and 85.7%, respectively. However, its reliance on traffic data may overlook ledger-based anomalies, and the semi-supervised model might not adapt quickly to novel, complex attack vectors, highlighting areas for future improvement.

2.1.5. Data Mining Approaches

Data mining approaches in blockchain anomaly detection employ algorithms for effective analysis. The label scarcity method [18] was presented for detecting money laundering in the Bitcoin blockchain. The study highlights the cons of unsupervised anomaly detection for identifying illicit transactions. The authors proposed an active learning solution that achieved comparable results to fully supervised baselines with only 5% of the labels, leveraging a limited number of expert-annotated labels in real-world scenarios. However, the approach may struggle to adapt rapidly to evolving laundering techniques without continuous input from costly manual labelling, a drawback to this approach.

2.2. Anomalous User-Based Anomaly Detection

In blockchain technology, ensuring the authenticity and legitimacy of user activity is important for maintaining the network's integrity and trustworthiness. Anomalous user-based activities, including illicit account operations, fraudulent transactions, and suspicious behavioural patterns, pose challenges to blockchain network security. Detecting such anomalies is critical for preempting potential threats and mitigating risks that can compromise the blockchain ecosystem. This subsection explores the research in anomaly detection focused on user behaviour.

2.2.1. Machine Learning Approaches

A novel approach was developed to improve Bitcoin ownership identification [19] by analyzing transaction patterns to cluster addresses that share the same ownership. This method analyzed over 46 million Bitcoin addresses and used transaction patterns such as relay, sweep, and distributing transactions, combined with traditional heuristics, to identify clusters of addresses. This method of combining features enhances anomaly detection in Bitcoin. However, focusing on predefined patterns may miss other blockchain transactions and not consider evolving adversarial attacks, compromising blockchain security.

In another study, the XGBoost classifier is applied to detect illicit accounts in the Ethereum blockchain, focusing on transaction history [20]. This method, leveraging 2179 accounts flagged by the Ethereum community and 2502 normal accounts, achieved an average accuracy of 0.963 with an AUC of 0.994, advancing the detection of illicit activities on the Ethereum network. Despite its effectiveness, the model's reliance on transaction history could limit adaptability to evolving illicit patterns not represented in the dataset, emphasizing the need for continuous data updates and model refinement to maintain its detection capabilities.

Another Fraud Detection Framework [21] experimented on various machine learning techniques, including K-nearest neighbour, decision tree (DT), and random forest, to identify unauthorized accounts within the Ethereum blockchain. A limitation of this study is its reliance on a relatively small dataset from Kaggle.com, which might not provide the comprehensive representation required for accurate detection of the Ethereum blockchain.

2.2.2. Graph-Based Approaches

A temporal graph properties-based approach [22] was presented for detecting malicious accounts in permission-less blockchains. This approach utilizes a directed graph model. The study leverages machine learning models to classify accounts as malicious or benign based on newly introduced temporal features such as burst and attractiveness, in addition to traditional graph metrics. However, the study's reliance on temporal features may not fully capture blockchain fraud's dynamic and sophisticated nature.

Another study presents an anomaly detection method for public blockchain networks using an evolved graph attention [23] network and a directed dynamic attribute graph, enhancing transaction attribute granularity and updating node learning weights based on temporal changes. The authors use the Graph-SMOTE method to handle imbalanced data. The proposed method may encounter scalability challenges due to the computational complexity inherent in handling larger and complicated blockchain networks.

2.2.3. Data Mining Approaches

Various data mining approaches have been proposed in the research community for anomaly detection in blockchain. A scalable anomaly detection method is proposed in [24], using data sketches for a compact block for the identification of suspicious activities without analyzing the entire blockchain. The proposed technique utilizes machine learning approaches and frequency estimation with sketches like HyperLogLog, which can identify complex patterns; however, it can suffer from the need for extensive training data and potential accuracy issues due to their probabilistic nature, leading to a trade-off between efficiency and precision in detecting blockchain anomalies.

Similarly, the authors in [25] proposed a sketch-based framework for detecting anomalies in blockchain networks, using Ethereum as a test case. It identifies suspicious accounts while reducing memory and time complexity (90–96% and 86%) as compared to conventional methods. This approach may face challenges in identifying complex, low-frequency anomalies and adapting to new threats due to its reliance on summarized data and fixed parameters.

A collective anomaly detection approach [26] was proposed to find fraud in Bitcoin. The authors focused on user behaviour across multiple wallets, using the trimmed K-means algorithm. According to the article, the proposed method identified 14 fraudulent users and 26 associated addresses. This work utilized high computational and operational power to extract features and execute algorithms. Also, this approach targets a single blockchain layer, leading to incomplete detection and inefficient computational usage. The approach might be susceptible to adversarial attacks, exposing vulnerabilities in the detection system and potentially compromising blockchain security.

A summary of related work on anomaly detection is presented in Table 1. In conclusion, detecting anomalies in blockchain networks requires advanced, multi-dimensional approaches due to the complexities of fraud, unauthorized transactions, and cyber threats. This review illuminates the necessity for innovative solutions that transcend traditional methods. This research aims to create a secure, efficient, and comprehensive framework for detecting anomalies.

Table 1. Summary of related work on anomaly detection in blockchain networks.

Existing Work	Year	Target Anomaly	Target Layer	Multi-Layer	Multi-Anomaly	ML Model	Feature Selection Method	Dataset	Approach
Evolved graph attention [13]	2024	Transaction (Txn)	Data	×	×	×	Domain-Driven	Bitcoin	Graph theory
Multi-layer temporal transaction [14]	2023	Txn	Data	×	×	×	Domain-Driven	Ethereum	Graph neural network
Efficient fraud detection [10]	2022	Txn	Data	×	×	✓	Domain-Driven	Bitcoin	XGBoost, RF
ADOBSVM [11]	2022	Txn	Data	×	×	✓	Domain- Driven	Bitcoin	SVM
Bitcoin theft detection [12]	2021	Txn	Data	×	×	✓	Domain-Driven	Bitcoin	KNN, SVM, RF, Ada Boost, MLP
Scalable anomaly detection [15]	2021	Txn	Data	×	×	×	Domain-Driven	Bitcoin	Graph theory
Monitoring traffic [17]	2021	Txn	Data	×	×	✓	Statistical	Bitcoin	Semi-supervised learning model
BAD [16]	2020	Txn	Data	×	×	×	Domain-Driven	Bitcoin	Leveraging blockchain metadata
Random walk [13]	2020	Txn	Data	×	×	✓	Domain-Driven	Bitcoin	K-means clustering
Label scarcity [18]	2020	Txn	Data	×	×	×	Machine learning	Bitcoin	Active learning
Blockchain Transaction Signing [9]	2019	Txn	Data	×	×	✓	Domain-Driven	Ethereum	IF
Sketch-based framework [25]	2023	User	Network (N/W)	×	×	×	Domain-Driven	Ethereum	Sketching technique
Collective anomaly detection [26]	2022	User	N/W	×	×	✓	Domain-Driven	Bitcoin	Trimmed K-means clustering
Scalable anomaly detection [24]	2021	User	N/W	×	×	×	Statistical	Ethereum	Sketching
Temporal graph properties [22]	2021	User	N/W	×	×	✓	Domain-Driven	Ethereum	K-means
Fraud Detection Framework [21]	2021	User	N/W	×	×	✓	Statistical	Ethereum	DT, RF, K-nearest neighbours
Transactional History Approach [20]	2020	User	N/W	×	×	✓	Machine learning	Ethereum	XGBoost, ETC
Ownership identification [19]	2018	User	N/W	×	×	✓	Domain-Driven	Bitcoin	Clustering
Proposed AHEAD	2024	Transaction and User	Data and N/W	✓	✓	✓	Hybrid (Statistical + ML + Data Science + Domain-Driven)	Ethereum	Ensemble (ETC, GBC, ABC, RFC, KNC, eXGB, LGBM, BC, DTC, CB)

2.3. Problem Statement

Based on the survey in the last section, the problem can be stated as follows. The majority of the research work on detecting and mitigating adversarial attacks is either focused on a single aspect of an anomaly (such as only anomalous transactions or anomalous users) or a single layer of the blockchain (such as the data or the network layer). This leads to inefficient computational usage, incomplete detection, mitigation, and remedial steps, thereby limiting the scope of anomaly detection [9,11]. Moreover, blockchain data possess inherent complexities such as high dimensionality, huge volumes of data, and imbalanced classes, which makes data preprocessing challenging and negatively affects the accuracy of anomaly detection [27]. Lastly, to the best of our survey, the current anomaly detection ML models are vulnerable to adversarial attacks which can exploit their weaknesses and compromise the overall integrity of the blockchain network [8,10,28].

3. Proposed Methodology

This section discusses the methodology of our proposed work for anomaly detection in the blockchain on multiple layers. The first subsection discusses preprocessing steps. The selection of optimal algorithms for enhanced anomaly detection is discussed in subsection two, followed by the three-layered ensemble model approach in subsection three. Figure 2 describes the phases of our proposed work for anomaly detection.

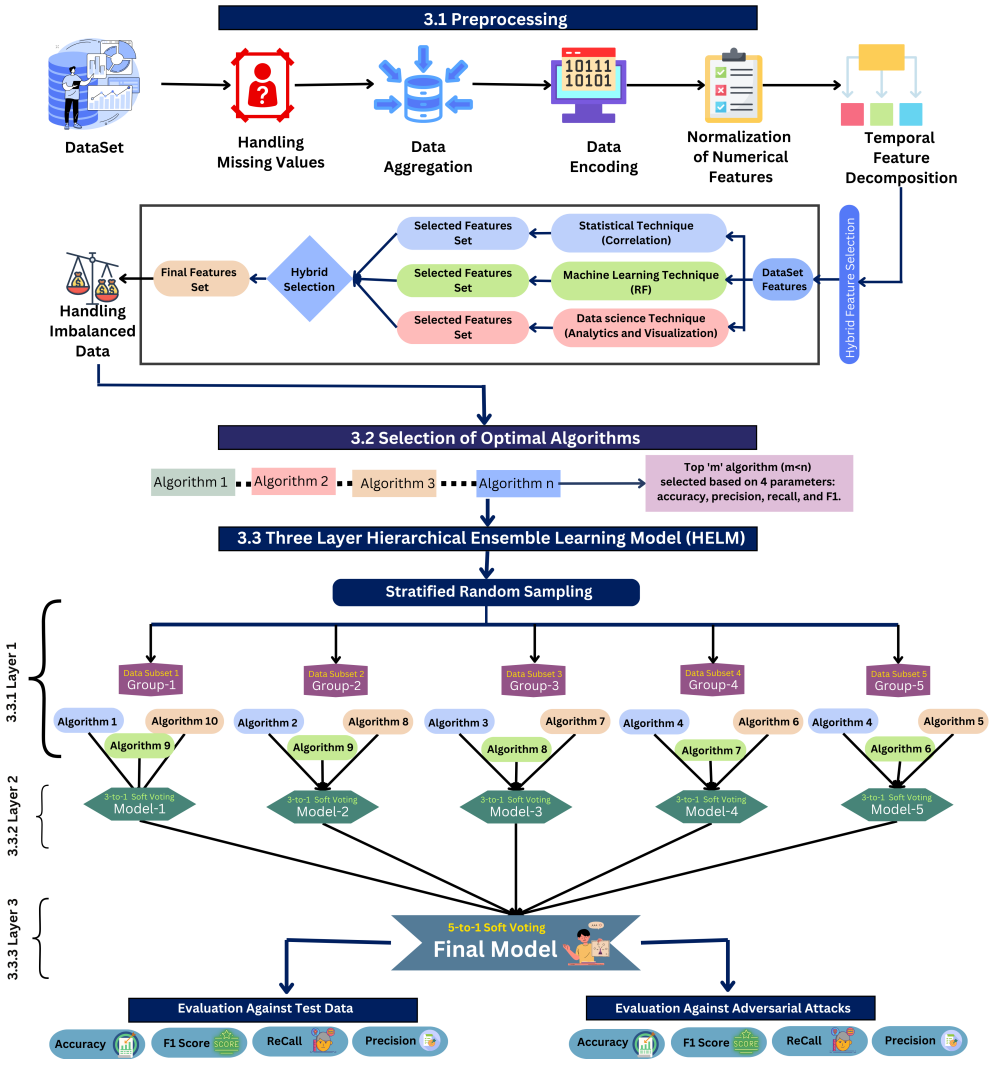

Figure 2. Proposed AHEAD: multi-layer multi-anomaly detection with adversarial ML model resilience.

3.1. Preprocessing of Transactional Dataset

Data preprocessing in blockchain anomaly detection is essential for transforming complex and raw data into a clear, usable format. In this stage, we perform cleaning, handling missing values, data balancing, scaling, encoding, feature engineering, and normalizing the data. In this process, we enhance data quality, which is crucial for accurate and efficient anomaly detection, strengthening blockchain network security. This subsection will present the details of the specific steps of the preprocessing.

3.1.1. Dataset

Our experiments utilize the Ethereum network transaction dataset [29]. This dataset was chosen after extensive reviews of several pertinent datasets, with the final selection strongly aligning with the objectives of our research. The Ethereum dataset contains transactions, where each transaction consists of two components: first, transaction-data, which are required to detect data layer anomalies, and transaction-users, which are required to detect network layer anomalies. Features related to transaction-data include 'Hash', 'Transaction_index', 'Value', 'Input', 'Receipt_cumulative_gas_used', 'Receipt_gas_used', 'Block_timestamp', 'Block_number', and 'Block_hash'. Similarly, the features related to transaction-users include 'From_address', 'To_address', 'Nonce', 'Gas', 'Gas_price', 'From_scam', 'To_scam', 'From_category', and 'To_category'. The dataset comprises 71,250 transactions, each described by 18 distinct features. Among these, 57,000 transactions are categorized as normal, while 14,250 are identified as anomalous. A detailed description of the dataset's features is provided in Table 2.

Table 2. Feature description in blockchain transactions.

Feature Name	Description
Hash	Hash of the transaction.
Nonce	Number of transactions made by the sender.
Transaction_index	Index of the transaction.
From_address	Transaction sender address.
To_address	Transaction receiver address.
Value	Value of the transaction in Wei.
Gas	Gas used in the transaction.
Gas_price	Price of gas provided by the sender.
Input	Data sent with the transaction.
Receipt_cumulative_gas_used	Accumulative gas used in block until current transaction.
Receipt_gas_used	Actual gas used in the transaction.
Block_timestamp	Timestamp of the block containing the transaction.
Block_number	Number of the block containing the transaction.
Block_hash	Hash of the block containing the transaction.
From_scam	Indicator if the sender is anomalous.
To_scam	Indicator if the receiver is anomalous.
From_category	Category of sender anomaly, if available.
To_category	Category of receiver anomaly, if available.

3.1.2. Handling Missing Values

The dataset has several missing values in the 'from_category' and 'to_category' columns. These columns contain 68,622 and 59,601 null values, respectively, out of 71,250 records. We dropped these two columns to enhance our dataset's quality and reliability and minimize the noise and potential data skewness. We performed this step to maintain data integrity and improve the accuracy of our anomaly detection model.

3.1.3. Data Aggregation

Further, the aggregation of data was performed, an important preprocessing step in the development of our anomaly detection model for blockchain. This process is pivotal in identifying anomalies at data and network layers, one of the objectives of our research.

In data aggregation, we introduced a new target attribute variable 'class'. It was derived from the 'to_scam' and 'from_scam' columns. We updated the 'to_scam' column through strategic data transformations, replacing all 1s with 2s, and engineered the 'class' attribute using condition-based classifications. This attribute classifies the transactions into three classes to distinguish between normal and anomalous transactions and their users. The classification of transactions and users are presented in Table 3.

We introduce a new abstract feature set by labelling these classes and removing the original 'from_scam' and 'to_scam' columns. This refined data aggregation and feature engineering approach advances our model's capacity to detect and analyze anomalies in the blockchain.

Table 3. Classification of anomalous transactions and users.

Class	Description
0	Normal transaction and normal user (Normal Tx)
1	Anomalous transactions with anomalous user at the sender side (Anom Tx Sndr)
2	Anomalous transactions with anomalous user at the receiver side (Anom Tx Rcvr)

3.1.4. Data Encoding

In the preprocessing stage, an essential step involves transforming categorical variables into a format that machine learning algorithms can efficiently process. Categorical data in their original form can introduce complexities and ambiguities, hindering the model's performance and interpretability. We used the Label-Encoder class from the scikit-learn preprocessing module for Encoding. The mathematical algorithm used by Label-Encoder is a mapping function: it assigns a unique integer (starting from 0) to each distinct category in a column, based on the alphabetical order of the categories, using Equation (1):

$$f(x) = \text{index of } x \text{ in the sorted list of unique categories.} \tag{1}$$

We identify three categorical columns, i.e., 'block_hash', 'from_address', and 'to_address'. Further, we encoded them into numerical values. In this transformation, we appended the suffix '_encoded' to new columns and dropped the categorical columns. This step is beneficial in blockchain contexts, where data categories like addresses and hashes do not possess a natural order. We enhanced the precision and interpretability of our anomaly detection model through this encoding process.

3.1.5. Normalization of Numerical Features

When working with anomaly detection in blockchain systems, it is crucial to normalize the numerical features to boost the effectiveness of the models. We used the Standard-Scaler module from the scikit-learn library to accomplish this. The purpose of scaling is to modify the distribution of each variable to have a mean value (μ) of 0 and a standard deviation (σ) of 1. The mathematical formula used in Standard-Scaler for scaling a feature is described in Equation (2):

$$X_{\text{scaled}} = \frac{X - \mu}{\sigma} \tag{2}$$

3.1.6. Temporal Feature Decomposition

After the normalization step, temporal feature decomposition was performed on time-based data. This section explores how we transformed the 'block_timestamp' data from blockchain datasets into several detailed time-based features. The 'block_timestamp' is recorded when transactions or blocks are logged in the system. These raw data are informative but often need to be broken down into more descriptive parts for a thorough analysis. We extracted the year, month, day, hour, minute, second, and day of the week as separate features.

This step was performed to analyze in depth the temporal patterns within the blockchain data. For instance, analyzing transactions with respect to the time of day or day of the week can reveal repeated patterns or anomalies that might not be possible to find from the raw timestamp alone. Such insights are valuable in detecting fraudulent activities, irregularities, or inefficiencies within blockchain networks.

3.1.7. Feature Selection

In anomaly detection within blockchain technology, feature selection plays an important role in enhancing model accuracy and computational efficiency. This study introduces a novel, multi-faceted approach to feature selection, integrating statistical methods, machine learning algorithms, and data science techniques. This hybrid strategy ensures the selection of the most impactful features for subsequent study.

Initially, we employed a statistical technique, SelectKBest, one of the most used techniques, paired with the ANOVA F-test to find important features. This method emphasizes features with the strongest relationships with the output variable. The top ten features identified through this technique include 'nonce', 'receipt_cumulative_gas_used', 'receipt_gas_used', 'block_number', 'to_address_encoded', 'year', 'month', 'day', 'hour', and 'day_of_week'.

Subsequently, we used a machine learning-based approach, utilizing a Random Forest (RF) Classifier for feature selection. This ensemble learning method offers insights into feature importance, derived from the aggregated decision trees. The top ten features identified by this approach are 'block_number', 'receipt_gas_used', 'month', 'to_address_encoded', 'value', 'gas', 'nonce', 'gas_price', 'year', and 'day'.

Further enhancement of the feature selection process involves data science techniques, where Domain-Driven insights and analytical methods are applied. For instance, a boxenplot visualization in Figure 3 shows the distribution of the 'value' feature across three distinct classes on a logarithmic scale. This visualization reveals that normal transactions mostly involve lower transaction values, whereas anomalous transactions involve higher transaction values. Such distinct patterns suggest the 'value' feature's influence on the model's performance.

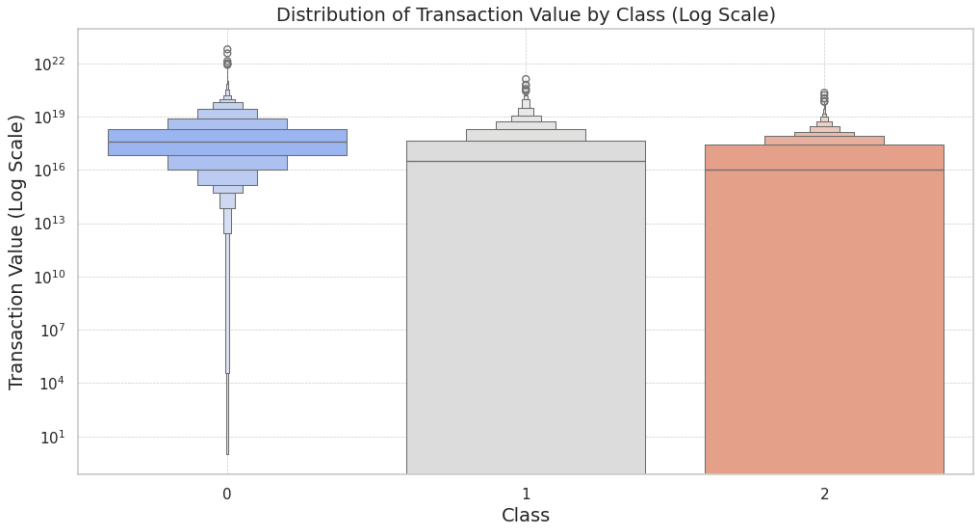

Figure 3. A representation of the 'value' attribute differentiated by class.

Moreover, in Figure 4, a scatter plot with 'gas' and 'gas_price' on logarithmic scales provides further insights. Normal transactions are characterized mainly by lower gas usage and gas prices, while anomalous transactions display high levels of gas usage and gas prices, indicative of cost-intensive transactions. These observations suggest the inclusion of 'gas' and 'gas_price' as important features. Our further analyses highlight the significance

of the 'to_address_encoded' feature, which shows clear transaction patterns across different classes, emphasizing its crucial role in our feature set.

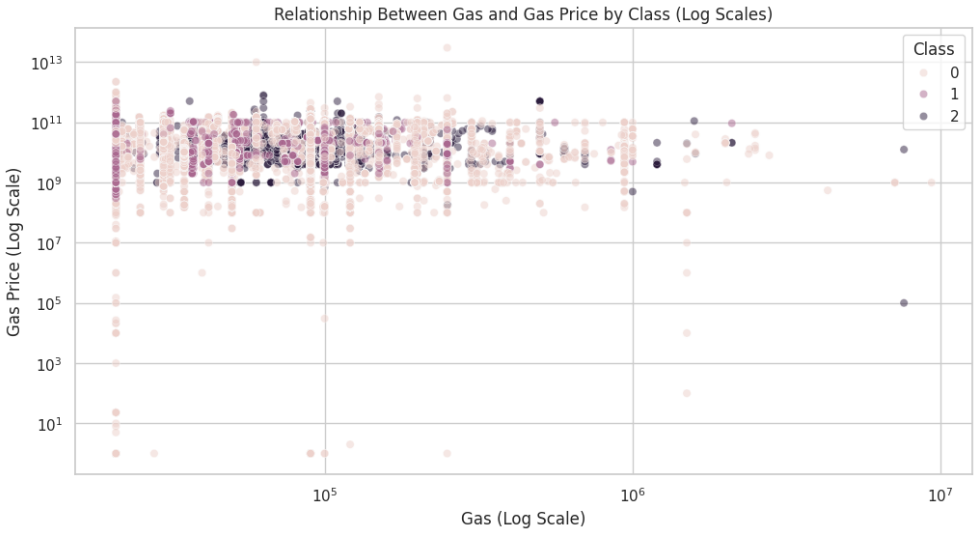

Figure 4. Class-wise relationship between 'gas' and 'gas_price'.

However, we also identified features that showed a limited level of importance for anomaly detection. For instance, while 'block_number' and 'transaction_index' provide some structural information, they lack the depth in relational and behavioural insights that is important for detecting anomalies effectively. Similarly, 'receipt_cumulative_gas_used', which aggregates the gas used across all transactions in a block, proved to be less important.

After detailed experimentation and analysis, we selected a more targeted feature set for further research. This set includes 'month', 'receipt_gas_used', 'to_address_encoded', 'value', 'nonce', 'gas', and 'gas_price'. Our hybrid feature selection approach combines statistical analysis, machine learning techniques, and data science insights using domain expertise and extensive experimentation.

3.1.8. Handling Imbalanced Data

In blockchain anomaly detection, one common challenge is imbalanced classes in transaction data that can affect the accuracy of predictive models. Our study addresses this issue with a dataset of 71,250 blockchain transactions. Initially, this dataset showed a significant imbalance: it consisted of 57,000 normal transactions (label 0), 2628 transactions labelled as anomalous due to an anomalous sender (label 1), and 11,622 transactions labelled as anomalous due to an anomalous receiver (label 2). The unbalanced distribution of classes leads to biased predictions favouring the majority class, reduces the model's ability to generalize well to unseen data, and necessitates complex evaluation metrics to assess and improve model performance.

We used the Adaptive Synthetic Sampling (ADASYN) method, a robust oversampling technique designed to generate synthetic samples for the minority class, to resolve the issue of an imbalanced dataset. ADASYN adaptively adjusts the weights of different minority class examples based on their level of difficulty in learning. It improves model performance on minority samples without compromising the overall accuracy. After applying the ADASYN technique, the dataset had 56,860 for each class, i.e., normal transactions (label 0), anomalous transactions with the sender as anomalous (label 1), and anomalous transactions with the receiver as anomalous (label 2).

3.2. Selection of Optimal Algorithms for Enhanced Anomaly Detection

We performed experimental analysis on various machine learning algorithms to identify an ensemble of algorithms that exhibits superior performance for an efficient,

reliable, and secure anomaly detection system. We selected the top-performing algorithms based on a comprehensive evaluation of key performance indicators. These indicators included accuracy, precision, recall, and F1 score, each providing an understanding of the model's predictive capabilities and their adeptness in managing the complex dynamics of anomaly detection in blockchain networks. We selected the algorithm that achieved a metric score of over 75% in any of the evaluated indicators. The selected classifiers are Extra Trees Classifier (ETC), Random Forest Classifier (RFC), eXtreme Gradient Boosting (eXGB), Bagging Classifier (BC), Categorical Boosting (CB), Decision Tree Classifier (DTC), Light Gradient Boosting Machine (LGBM), K-Neighbours Classifier (KNC), Gradient Boosting Classifier (GBC), and Ada Boost Classifier (ABC).

The ensemble technique incorporates these top-tier algorithms and leverages the strengths and mitigates the limitations of individual models. This strategy ensures a comprehensive and nuanced anomaly detection mechanism essential for maintaining the efficiency and security of blockchain networks.

3.3. Three-layer Hierarchical Ensemble Learning Model (HELM)

In addressing the complex challenges of anomaly detection in blockchain technology, this study has presented an advanced three-layer model employing ensemble techniques. Ensemble models are more robust to adversarial attacks compared to single models, and ensemble learning can improve the model's ability to learn complex patterns and reduce overfitting, leading to higher accuracy. Hence, this approach aligns with the research objectives: to develop an anti-adversarial, accurate, and multi-layered anomaly detection framework for blockchain networks.

3.3.1. Layer 1: Anti-Adversarial Stratified Random Sampling

In the first layer, distinct model groups were created. We performed experiments in different settings by changing the number of groups and the combination of different optimal selected algorithms within each group. After extensive experiments, based on the diversity of algorithms and evaluation results, five groups, each with the combination of three algorithms, were constituted to ensure a robust and comprehensive anomaly detection capability. From the preprocessed dataset, each group was given a unique subset of the dataset using stratified sampling. This approach increases the complexity of the model and makes it more resilient to adversarial attacks such as data poison, evasion, extraction, and input manipulation.

The composition of each group along with their algorithms are presented in Table 4.

Table 4. Summary of Layer 1 groups.

Group No.	Group Models
1	ETC, GBC, ABC
2	RFC, KNC, GBC
3	eXGB, LGBM, KNC
4	BC, DTC, LGBM
5	BC, CB, DTC

3.3.2. Layer 2: Optimal Grouping of Classifiers Based on Soft Voting

In the second layer, we leveraged soft voting to combine predictions from algorithms in each group trained on different data subsets. Soft voting aggregates the class probability distributions from each model, leading to a more robust ensemble classifier. This approach improves the overall performance, stability, and accuracy of predictions.

3.3.3. Layer 3: Final Ensemble Model Based on Soft Voting

We employed group-level ensembles from Layer 2 and developed Layer 3 of the final anomaly detection model. We used a soft voting mechanism to merge the predictions from each group, which were trained on different subsets of data. The final ensemble

benefits from the strengths of predictions from each group, enhancing the model's ability to generalize across varied scenarios and improving its overall robustness.

4. Evaluation of the Proposed AHEAD Model

In this section, we will first explain the evaluation metrics. Then, we will evaluate two aspects of AHEAD, the anti-adversarial resilience through the HELM (Three-Layer Ensemble Learning Model) and multi-layer multi-anomaly detection capability. In the next section, we will compare the performance of AHEAD with other machine learning mechanisms.

4.1. Evaluation Metrics

For blockchain anomaly detection evaluation, we chose to employ weighted metrics for precision, recall, and the F1 score along with accuracy metrics. The details of the metrics are as follows:

- **Accuracy:** This provides a measure of the overall effectiveness of the model using the formula written in Equation (3).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}} \tag{3}$$

- **Precision:** This is calculated by weighting the precision of each class according to its representation in the data. It is calculated as written in Equation (4).

$$\text{Weighted Precision} = \frac{\sum_{i=1}^{n}(\text{Precision of class } i \times \text{instances in class } i)}{\text{Total instances}} \tag{4}$$

- **Recall:** Similarly, weighted recall ensures the model's sensitivity. It is calculated as written in Equation (5).

$$\text{Weighted Recall} = \frac{\sum_{i=1}^{n}(\text{Recall of class } i \times \text{instances in class } i)}{\text{Total instances}} \tag{5}$$

- **F1 score:** The weighted F1 score corresponds to precision and recall, providing a metric reflecting the model's balanced performance across both dimensions. It is calculated as written in Equation (6).

$$\text{Weighted F1 Score} = 2 \times \frac{\sum_{i=1}^{n}(\text{Precision of class } i \times \text{Recall of class } i \times \text{instances in class } i)}{\sum_{i=1}^{n}((\text{Precision of class } i + \text{Recall of class } i) \times \text{instances in class } i)} \tag{6}$$

We use a confusion matrix to complement these quantitative metrics to visualize the model's performance, presenting the true positives, true negatives, false positives, and false negatives in a matrix format. The confusion matrix for our model is depicted for each class: normal transaction, anomalous transaction and sender, and anomalous transaction and receiver.

4.2. Anti-Adversarial Resilience of AHEAD's HELM

Securing anomaly detection models from adversarial attacks is crucial as adversarial attacks can cause incorrect predictions, manipulate input data, erode trust, and lead to financial losses. The security of the model was tested through a perturbation analysis, where the input features of the test dataset were intentionally altered within a specified range to simulate an adversarial attack scenario. The core objective of this method is to assess the prediction ability of a model under potentially compromised conditions.

Table 5 outlines the final model's performance metrics, offering a detailed view of its accuracy, precision, recall, and F1 score. The experimental results demonstrate the resilience of the AHEAD model to adversarial attacks. In a situation of perturbed inputs, this model has the ability to maintain high accuracy, precision, recall, and F1 score. This ability of the model enhanced the robustness and reliability of anomaly detection systems in blockchain networks. Figure 5 presents the performance of AHEAD against adversarial

attacks for all classes. The figure illustrates the consistent and high performance of the HELM against all metrics across different classes under adversarial conditions. This shows that the HELM exhibits high accuracy and precision, which are crucial for minimizing false positives and false negatives. Figure 6 further supports these findings for the proposed HELM of AHEAD's model. The confusion matrix shows that the model achieves high true positive rates and low false positive rates across all classes, indicating its effectiveness in distinguishing between normal and anomalous transactions under adversarial conditions, demonstrating the optimal performance of the HELM.

Table 5. Performance evaluation for HELM (anti-adversarial resilience of AHEAD).

Evaluation Metrics	Final Results
Accuracy	95.97%
Precision	96.03%
Recall	95.97%
F1 score	95.98%

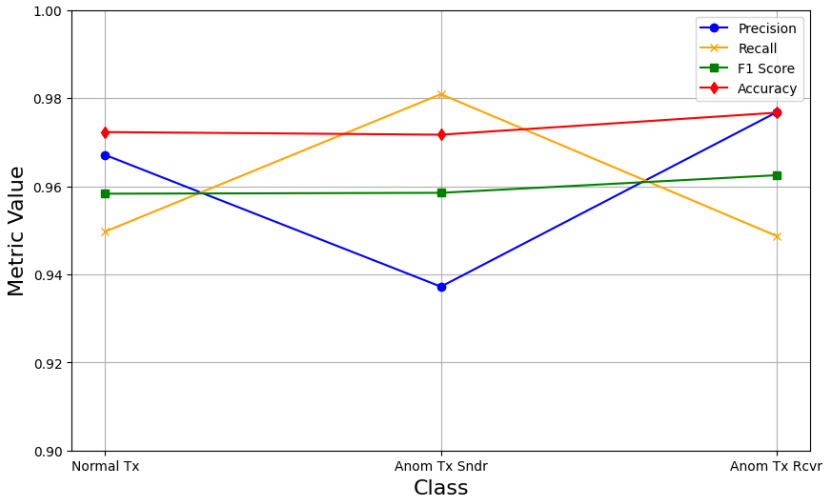

Figure 5. Performance of HELM against adversarial attacks for all classes.

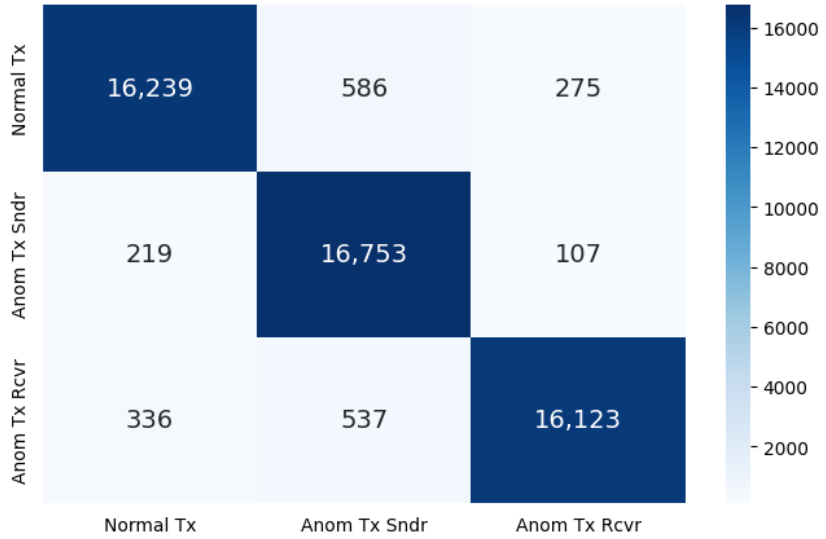

Figure 6. Confusion metrics showing performance of HELM against adversarial attacks.

4.3. Multi-Layer Multi-Anomaly Detection Capability of AHEAD

In this section, our research describes the implementation and outcomes of a novel AHEAD method for multi-layer multi-anomaly detection within blockchain technology. This methodology is designed to address the challenges of identifying anomalies in two layers of blockchain networks, i.e., data and network layers targeting transaction and user anomalies. We performed advanced preprocessing techniques, e.g., data aggregation, temporal feature selection, hybrid feature selection, and handling imbalanced data, to enhance model accuracy and computational efficiency. Further, we selected optimal algorithms that exhibit superior performance metrics, ensuring robustness and reliability in anomaly detection. The AHEAD adopts the train–test split method with 30% of the data used for testing while the remaining 70% is used for training. This approach ensures that our model is well trained and tested with a substantial amount of unseen data.

The implementation of the AHEAD model concluded with the refinement of anomaly detection. Table 6 outlines the final model's performance metrics with a detailed view of its accuracy, precision, recall, and F1 score, which showcased an enhanced overall accuracy of 98.85%. This increment highlights the effectiveness of our hierarchical voting system in improving decision making accuracy. Figure 7 displays the confusion metrics of the proposed AHEAD model achieving high true positive rates and low false positive rates, indicating its effectiveness in distinguishing between normal and anomalous transactions. Figure 8 presents the performance of AHEAD for multiple layers and multiple anomalies for all classes. The figure illustrates the consistent and high performance of the AHEAD against all metrics across different classes and exhibits high accuracy and precision, which are crucial for minimizing false positives and false negatives, thus underscoring the advancements achieved through our innovative three-layered ensemble model in the realm of blockchain anomaly detection.

Table 6. Performance evaluation for multi-layer anomaly detection model.

Evaluation Metrics	Final Results
Accuracy	98.8510%
Precision	98.8524%
Recall	98.8510%
F1 score	98.8510%

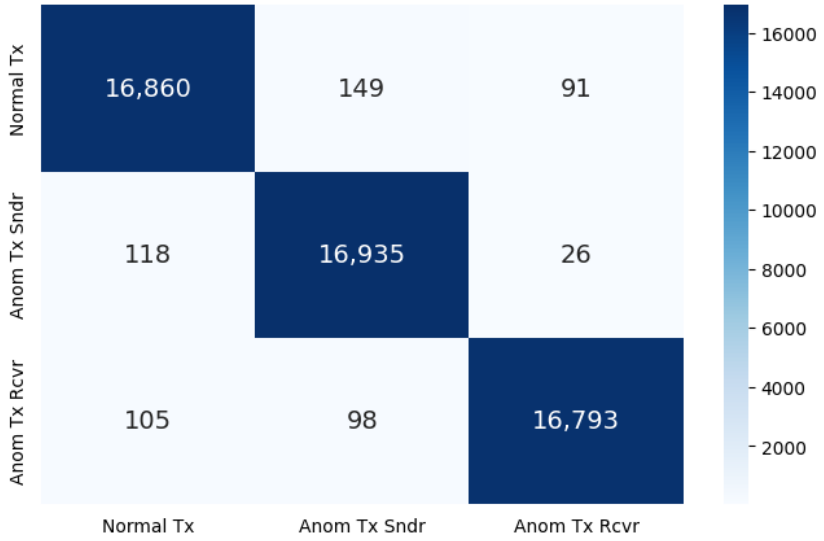

Figure 7. Confusion metric for AHEAD's multi-layer multi-anomaly detection capability.

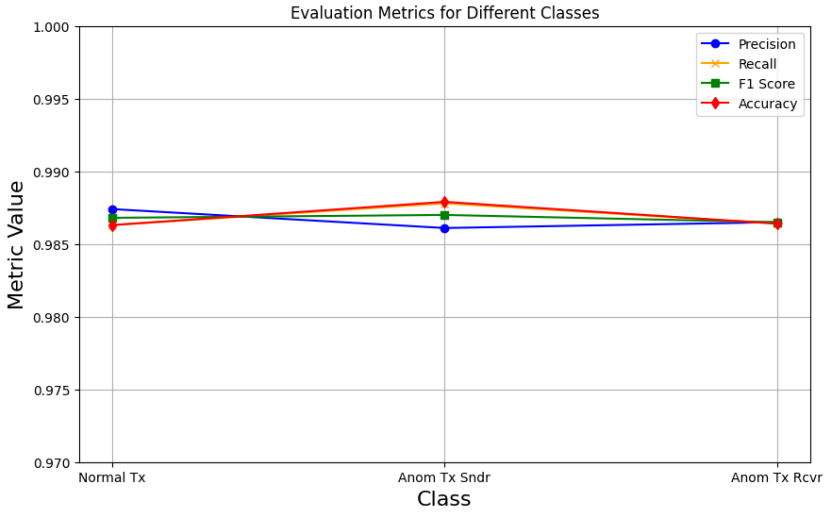

Figure 8. Performance of AHEAD's multi-layer multi-anomaly detection capability for all classes.

5. Comparison of AHEAD with Existing ML Approaches

In this section, firstly, we will compare the ML model of AHEAD, the Three-Layer Hierarchical Ensemble Learning Model (HELM), with the existing ML models, and then we will compare AHEAD with the prevalent ML approaches, to highlight their multi-layer multi-anomaly detection capability.

5.1. Comparing Adversarial Resilience of AHEAD's HELM with Existing ML Models

This section evaluates how an ML model reacts when exposed to adversarial attacks (like data poisoning, evasion, extraction, and input manipulation). The comparison of AHEAD's HELM with existing ML models was conducted under identical preprocessing conditions to ensure a fair evaluation. For this, we followed the preprocessing steps explained in Section 3.1. Additionally, we intentionally altered the preprocessed dataset within a specified range to simulate an adversarial attack scenario for testing purposes. The performance results of the HELM surpassed those of other models. The HELM achieved 95.97% accuracy, 96.03% precision, 95.97% recall, and a 95.98% F1 score. The superior performance of the HELM is due to its three-layer hierarchical ensemble approach, which leverages the following: (i) An optimal grouping of classifiers based on soft voting. Soft voting aggregates the probability distributions of each classifier's predictions, allowing for a more nuanced decision making process. This method enhances the model's ability to correctly classify inputs even when some classifiers are fooled by adversarial data. (ii) Anti-adversarial stratified random sampling by which we provided a separate dataset to each group. By stratifying the data, the HELM minimizes the impact of data poisoning attacks. (iii) The final layer of HELM combines outputs from grouped classifiers, enhancing accuracy and robustness by leveraging their strengths and mitigating weaknesses. This multi-layered approach ensures that even if initial defenses are bypassed, subsequent layers can detect and counteract attacks, resulting in superior overall performance. The closest competitor is the Categorical Boosting model, which attained 90.52% accuracy, 90.66% precision, 90.52% recall, and 90.49% F1 score. The Extra Tree Classifier exhibited the next best performance with 89.95% accuracy, 90.31% precision, 89.95% recall, and an 89.95% F1 score.

Out of the 11 total competitor models, including the proposed HELM, the worst performance reported was of the Ada Boost Classifier. It achieved 70.34% accuracy, 70.75% precision, 70.30% recall, and 69.53% F1 score. The performance comparison of the HELM with existing ML models against adversarial attacks is visualized in a bee swarm in Figure 9. This illustrates the distribution and density of performance scores for different metrics: accuracy, precision, recall, and F1 score. Each coloured dot represents a model's score for a visual comparison. The proposed HELM, depicted by purple dots, consistently shows

superior performance across all metrics. This visualization effectively demonstrates the proposed model's robustness and reliability against adversarial attacks. The performance results of ML models against adversarial attacks are presented in Figure 10.

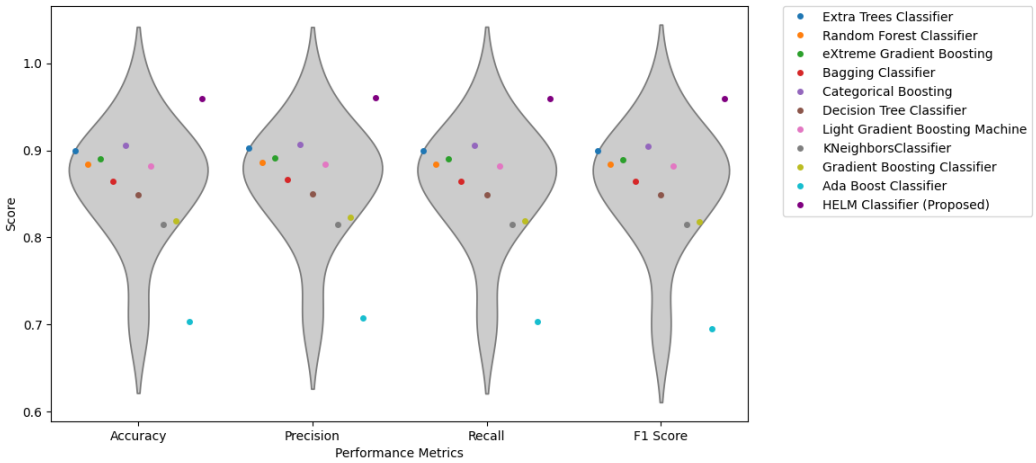

Figure 9. Performance comparison of HELM with other ML models against adversarial attacks.

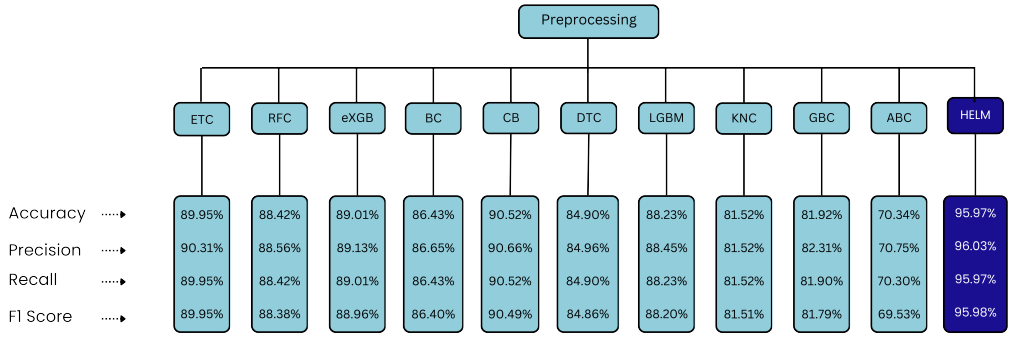

Figure 10. Performance comparison of HELM with other ML models against adversarial attacks.

5.2. Comparing Multi-Layer Multi-Anomaly Capability of AHEAD with Other ML Models

In this section, we compare the performance of the AHEAD model for multi-layer multi-anomaly detection with state-of-the-art existing anomaly detection techniques. The data aggregation of preprocessing in the proposed AHEAD model (explained in Section 3.1) gives it the capability to detect multiple anomalies at multiple layers. However, this feature does not exist in state-of-the-art anomaly detection techniques. For a performance comparison of AHEAD and prevalent techniques, an identical preprocessing feature was added to ensure a fair evaluation. AHEAD surpasses other techniques by achieving the highest accuracy of 98.85%. We achieved this accuracy by utilizing advanced preprocessing steps, e.g., we used a hybrid feature selection approach to ensure that only the most impactful features were selected, enhancing model accuracy and computational efficiency. Further, AHEAD employs a diverse set of algorithms that exhibit superior performance metrics, ensuring robustness and reliability in anomaly detection. Finally, the Three-Layer Hierarchical Ensemble Learning Model (HELM) increases model resilience to adversarial attacks and improves its ability to learn complex patterns, reducing overfitting and enhancing accuracy. The closest accuracy is achieved by a Fraud Detection Framework (FDF) [21], a value of 98.67%, followed by the Transactional History Approach (THA) [20], which achieves 97.79% accuracy. Blockchain Transaction Signing (TS) [9] achieves only 51.08% accuracy. The performance comparison of AHEAD with other anomaly detection techniques is visualized in a bee swarm in Figure 11. The proposed AHEAD model, depicted by purple dots, consistently shows superior performance across all metrics. Figure 12 presents

the detailed performance metrics of all techniques, showcasing their respective accuracy, precision, recall, and F1 scores.

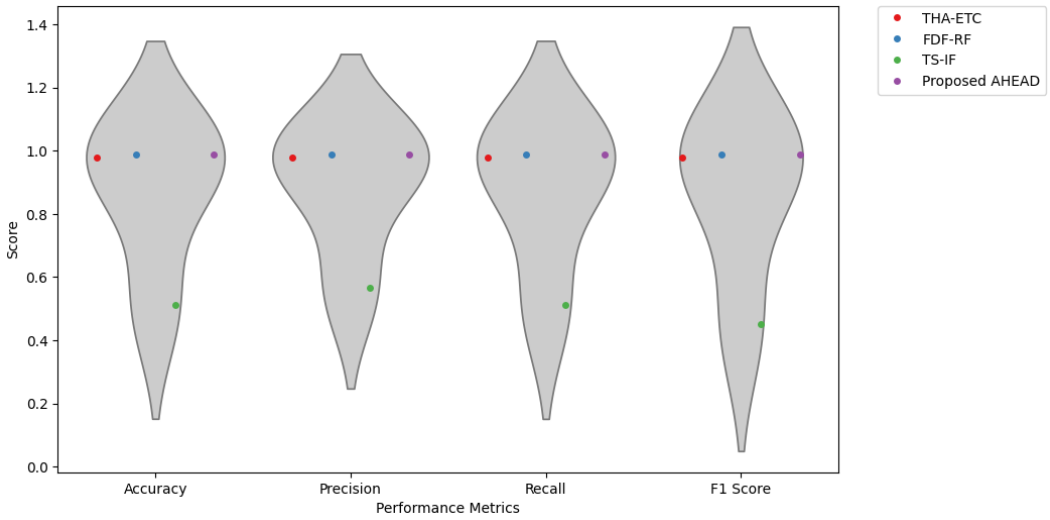

Figure 11. Performance comparison of AHEAD with other anomaly detection techniques.

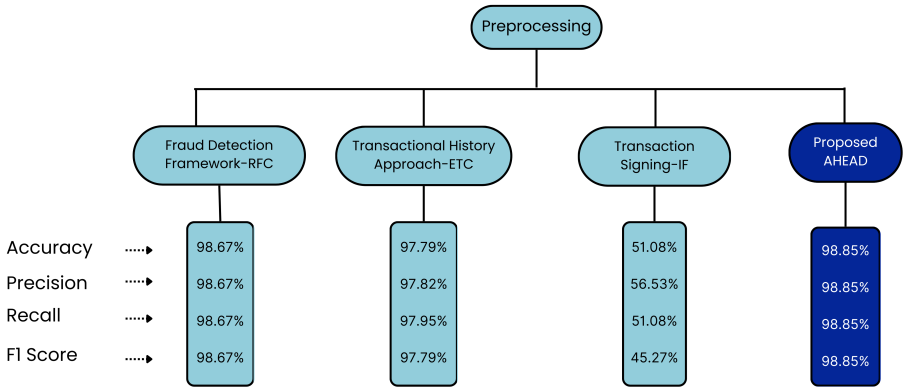

Figure 12. AHEAD performance for multi-layer multi-anomaly detection with state-of-the-art models.

Our work is a pioneer in demonstrating the resilience of the proposed AHEAD against adversarial attacks and effectively mitigates individual model biases. This dual-layer, dual-target approach enhanced by a novel feature selection strategy sets a new benchmark in blockchain anomaly detection.

6. Conclusions and Future Work

This research presents a pioneering Novel Anti-Adversarial Hierarchical Ensemble Model for Multi-Layered Anomalies Detection in Blockchain. We employ advanced preprocessing techniques and introduce a hybrid feature selection mechanism by integrating statistical analysis, machine learning algorithms, and data science methodologies for effective feature selection, ensuring precise and efficient anomaly detection. The proposed AHEAD model detects anomalies in both the data layer and network layer, simultaneously targeting anomalous transactions and anomalous users, significantly enhancing the security and reliability of blockchain networks. This research contributes a multi-layered anomaly detection framework that employs ensemble methods and exhibits resilience to adversarial threats, demonstrating the model's robustness. The empirical findings from our study outperform existing models, underscoring the proposed approach's effectiveness by achieving 98.85% accuracy against anomaly detection on data and network layers targeting transaction and user anomalies, along with 95.97% accuracy against adversarial machine learning attacks. This research highlights the importance of continuously improving anomaly de-

tection techniques to face evolving security threats in blockchain technology. Our future work will focus on integration with sophisticated algorithms and detailed case studies to allow the model to achieve a high level of adaptability, scalability, real-time processing, and long-term stability within any blockchain environment.

Author Contributions: Conceptualization, M.K.; methodology, M.K. and M.M.R.; software, M.K.; validation, M.K., M.M.R., M.W.R. and W.N.; formal analysis, M.K. and M.M.R.; investigation, M.K. and M.M.R.; resources, M.K., M.M.R., M.W.R. and W.N.; data curation, M.K.; writing—original draft preparation, M.K. and M.M.R.; writing—review and editing, M.M.R., M.W.R. and W.N.; visualization, M.K. and M.M.R.; supervision, M.M.R. (main), W.N. (Co) and M.W.R. (field). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare that there are no conflicts of interest.

References

1. Guo, H.; Yu, X. A survey on blockchain technology and its security. *Blockchain Res. Appl.* **2022**, *3*, 100067. [CrossRef]
2. Monrat, A.A.; Schelén, O.; Andersson, K. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access* **2019**, *7*, 117134–117151. [CrossRef]
3. Nehra, M. Latest Trends in Blockchain Technology. 2022. Available online: https://www.decipherzone.com/blog-detail/blockchain-trends (accessed on 10 April 2023).
4. IBM. What Is Blockchain? 2023. Available online: https://www.ibm.com/topics/blockchain (accessed on 10 April 2023).
5. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv. (CSUR)* **2009**, *41*, 1–58. [CrossRef]
6. Nassif, A.B.; Talib, M.A.; Nasir, Q.; Dakalbab, F.M. Machine learning for anomaly detection: A systematic review. *IEEE Access* **2021**, *9*, 78658–78700. [CrossRef]
7. Sanjay Rai, G.; Goyal, S.B.; Chatterjee, P. Anomaly Detection in Blockchain Using Machine Learning. In *Proceedings of the Computational Intelligence for Engineering and Management Applications*; Chatterjee, P., Pamucar, D., Yazdani, M., Panchal, D., Eds.; Springer: Singapore, 2023; pp. 487–499.
8. Bhagoji, A.N.; Shirani, P. Adversarial Attacks on Anomaly Detection. In *Encyclopedia of Machine Learning and Data Science*; Phung, D., Webb, G.I., Sammut, C., Eds.; Springer: New York, NY, USA, 2020; pp. 1–4. [CrossRef]
9. Podgorelec, B.; Turkanović, M.; Karakatič, S. A machine learning-based method for automated blockchain transaction signing including personalized anomaly detection. *Sensors* **2019**, *20*, 147. [CrossRef] [PubMed]
10. Ashfaq, T.; Khalid, R.; Yahaya, A.S.; Aslam, S.; Azar, A.T.; Alsafari, S.; Hameed, I.A. A machine learning and blockchain based efficient fraud detection mechanism. *Sensors* **2022**, *22*, 7162. [CrossRef] [PubMed]
11. Rwibasira, M.; Suchithra, R. ADOBSVM: Anomaly detection on blockchain using support vector machine. *Meas. Sens.* **2022**, *24*, 100503. [CrossRef]
12. Chen, B.; Wei, F.; Gu, C. Bitcoin theft detection based on supervised machine learning algorithms. *Secur. Commun. Netw.* **2021**, *2021*, 643763. [CrossRef]
13. Liao, Q.; Gu, Y.; Liao, J.; Li, W. Abnormal transaction detection of Bitcoin network based on feature fusion. In Proceedings of the 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 11–13 December 2020; Volume 9, pp. 542–549.
14. Han, B.; Wei, Y.; Wang, Q.; Collibus, F.M.D.; Tessone, C.J. MT^{2}AD: Multi-layer temporal transaction anomaly detection in ethereum networks with GNN. *Complex Intell. Syst.* **2024**, *10*, 613–626. [CrossRef]
15. Morishima, S. Scalable anomaly detection in blockchain using graphics processing unit. *Comput. Electr. Eng.* **2021**, *92*, 107087. [CrossRef]
16. Signorini, M.; Pontecorvi, M.; Kanoun, W.; Di Pietro, R. BAD: A blockchain anomaly detection solution. *IEEE Access* **2020**, *8*, 173481–173490. [CrossRef]
17. Kim, J.; Nakashima, M.; Fan, W.; Wuthier, S.; Zhou, X.; Kim, I.; Chang, S.Y. Anomaly detection based on traffic monitoring for secure blockchain networking. In Proceedings of the 2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Sydney, Australia, 3–6 May 2021; pp. 1–9.
18. Lorenz, J.; Silva, M.I.; Aparício, D.; Ascensão, J.T.; Bizarro, P. Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity. In Proceedings of the First ACM International Conference on AI in Finance, New York, NY, USA, 15–16 October 2020; pp. 1–8.

19. Chang, T.H.; Svetinovic, D. Improving bitcoin ownership identification using transaction patterns analysis. *IEEE Trans. Syst. Man, Cybern. Syst.* **2018**, *50*, 9–20. [CrossRef]
20. Farrugia, S.; Ellul, J.; Azzopardi, G. Detection of illicit accounts over the Ethereum blockchain. *Expert Syst. Appl.* **2020**, *150*, 113318. [CrossRef]
21. Ibrahim, R.F.; Elian, A.M.; Ababneh, M. Illicit account detection in the ethereum blockchain using machine learning. In Proceedings of the 2021 international conference on information technology (ICIT), Amman, Jordan, 14–15 July 2021; pp. 488–493.
22. Agarwal, R.; Barve, S.; Shukla, S.K. Detecting malicious accounts in permissionless blockchains using temporal graph properties. *Appl. Netw. Sci.* **2021**, *6*, 9. [CrossRef]
23. Liu, C.; Xu, Y.; Sun, Z. Directed dynamic attribute graph anomaly detection based on evolved graph attention for blockchain. *Knowl. Inf. Syst.* **2024**, *66*, 989–1010. [CrossRef]
24. Voronov, T.; Raz, D.; Rottenstreich, O. Scalable Blockchain Anomaly Detection with Sketches. In Proceedings of the 2021 IEEE International Conference on Blockchain (Blockchain), Melbourne, Australia, 6–8 December 2021; pp. 1–10.
25. Voronov, T.; Raz, D.; Rottenstreich, O. A Framework for Anomaly Detection in Blockchain Networks With Sketches. *IEEE/ACM Trans. Netw.* **2023**, *32*, 686–698. [CrossRef]
26. Shayegan, M.J.; Sabor, H.R.; Uddin, M.; Chen, C.L. A collective anomaly detection technique to detect crypto wallet frauds on bitcoin network. *Symmetry* **2022**, *14*, 328. [CrossRef]
27. Hassan, M.U.; Rehmani, M.H.; Chen, J. Anomaly detection in blockchain networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2022**, *25*, 289–318. [CrossRef]
28. Rosenberg, I.; Shabtai, A.; Elovici, Y.; Rokach, L. Adversarial machine learning attacks and defense methods in the cyber security domain. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–36. [CrossRef]
29. Al-E'mari, S.; Anbar, M.; Sanjalawe, Y.; Manickam, S. A labeled transactions-based dataset on the ethereum network. In Proceedings of the International Conference on Advances in Cyber Security, Penang, Malaysia, 8–9 December 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 61–79.