Preprints.org

Review

Not peer-reviewed version

DeepChainIoT: Exploring the Mutual Enhancement of Blockchain and Deep Neural Networks (DNN) in the Internet of Things (IoT)

Sabina Sapkota , Yining Hu * , Asif Gill , Farookh Khadeer Hussain

Posted Date: 8 July 2025

doi: 10.20944/preprints202507.0645.v1

Keywords: blockchain; deep neural networks (DNNs); internet of things (IoTs); security

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

DeepChainIoT: Exploring the Mutual Enhancement of Blockchain and Deep Neural Networks (DNN) in the Internet of Things (IoT)

Sabina Sapkota [1], Yining Hu [2,*], Asif Gill [2] and Farookh Hussain [2]

[1] Onion Innovation Pvt. Ltd, Dharmachowk, Bharatpur, 44200, Bagmati, Nepal; elecengsab@gmail.com
[2] School of Computer Science, University of Technology Sydney, 15 Broadway, Ultimo, 2007, NSW, Australia
* Correspondence: Yining.Hu@uts.edu.au

Abstract

The Internet of Things (IoT) is widely used across domains such as smart homes, healthcare, and grids. As billions of devices become connected, strong privacy and security measures are essential to protect sensitive information and prevent cyber-attacks. However, IoT devices often have limited computing power and storage, making it difficult to implement robust security and manage large volumes of data. Existing studies have explored integrating blockchain and Deep Neural Networks (DNNs) to address security, storage, and data dissemination in IoT networks, but they often fail to fully leverage the mutual enhancement between them. We propose DeepChainIoT, a blockchain–DNN integrated framework designed to address centralization, latency, throughput, storage, and privacy challenges in generic IoT networks. It integrates smart contracts with a Long Short-Term Memory (LSTM) autoencoder for anomaly detection and secure transaction encoding, along with an optimized Practical Byzantine Fault Tolerance (PBFT) consensus mechanism featuring transaction prioritization and node rating. On a public pump sensor dataset, our LSTM autoencoder achieved 99.6% accuracy, 100% recall, 97.95% precision, and a 98.97% F1-score, demonstrating balanced performance, along with a 23.9× compression ratio. Overall, DeepChainIoT enhances IoT security, reduces latency, improves throughput, and optimizes storage, while opening new directions for research in trustworthy computing.

Keywords: blockchain; deep neural networks (DNNs); internet of things (IoTs); security

1. Introduction

The Internet of Things (IoTs) are connected devices equipped with sensors and software that continuously collect and exchange a large amount of data. With the use of IoTs expanding across various domains such as smart homes, smart healthcare, agriculture, it is estimated that up to 500 billion connected devices will be deployed by 2030 [1]. However, IoT networks are inherently vulnerable due to their heterogeneity and the lack of strong security mechanisms in low-power devices [2,3], which results in many IoT networks relying on centralized architectures that can be easily exploited by attackers [4]. These make IoTs susceptible to numerous cyber-attacks, including data interception, Denial of Service (DoS), and Distributed Denial of Service (DDoS) [5,6]. For example, the Mirai botnet attack on DNS provider Dyn was reported to have affected more than 600,000 low-power IoT devices, including IP cameras, printers, and routers [7], which as a result disrupted major websites such as Amazon, Airbnb, and Netflix [5,8]. Flaws in the device authentication mechanisms in home routers have also led to complete network compromises [9], and the centralized architecture of IoTs have caused a Single Points of Failures (SPOF) that affected all connected devices [10]. As new variants of attacks continue to emerge [6], developing robust and reliable security measures to protect IoT networks has become increasingly crucial.

Blockchain has emerged as a trustworthy and secure mechanism for storing and exchanging data. Despite being originally designed for cryptocurrencies [11], blockchains are increasingly used

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

for identity verification, creating decentralization, and providing tamper-proof records in the Internet of Vehicles (IoVs), Industrial Internet of Things (IIoTs), Internet of Medical Things (IoMT), etc. A blockchain network effectively follows the CIA security model by ensuring confidentiality through public-key cryptography, maintaining integrity through hashing, and guaranteeing availability through decentralization [12]. However, as all nodes independently verify transactions, when the transaction volume increases or the network expands, the latency also increases, leading to different nodes having different versions of the transaction history, which is referred to as forking. The immutable and append-only ledger requires significant space for storing transactions, and storing transactions in their original format can raise security concerns.

Deep Neural Networks (DNNs) are significantly advancing multiple industries by incorporating Artificial Intelligence (AI) for decision-making [13]. DNNs have been shown to enhance the anomaly detection, classification, and prediction functionalities in wearables, self-driving cars, smart homes, smart offices, and virtual assistants [13]. Nonetheless, today's DNN systems heavily rely on centralized servers, making them susceptible to SPOF. There is also no mechanism in place to ensure the integrity of DNN models and their training data.

Extensive research has been conducted on integrating blockchain and DNNs to enhance IoT networks by establishing trust, identifying anomalies, and ensuring secure data storage and transmission [14,15]. Despite significant advancements, existing studies have not fully addressed the challenges in generic IoT networks primarily due to the lack of an effective blockchain-DNN integration framework that optimizes network latency, throughput, storage, computational efficiency, and data security [16,17]. Sapkota et al. [18] proposed an optimized blockchain-DNN framework leveraging an LSTM autoencoder to detect anomalous transactions and reduce storage overhead in IoT networks. Nonetheless, this solution has overlooked the blockchain consensus process, leaving the network vulnerable to malicious nodes while limiting the improvements on latency and throughput.Goh et al. [19] and Papaioannou et al. [20] proposed to optimize blockchain consensus with DNN, ensuring only honest nodes participate in the consensus process. However, neither study addressed the handling of anomalous transactions and nodes. How to *fully leverage the mutual enhancement of blockchain and DNN within an integration framework to mitigate anomalies, reduce storage overhead, and enable efficient and secure data dissemination in generic IoT networks* remains an open question.

To address this remaining gap, we propose DeepChainIoT, a comprehensive blockchain-DNN integration framework that detects anomalies, enhances the consensus with transaction prioritization and node rating, enables efficient transaction storage and dissemination through encoding, and decentralizes DNNs. Overall, this paper makes the following major contributions:

- We analyze blockchain and DNN as independent technologies in depth, and find that blockchain is suitable for IoT transaction processing due to its decentralization, but it suffers from issues such as high latency, low throughput, limited storage space, privacy, and is susceptible to network and smart contract attacks. DNNs can be used to detect anomalies in transactions, devices and users, but are often deployed in centralized systems without mechanisms to guarantee the integrity of the models and their training data.
- We conduct an extensive literature survey on existing frameworks that integrate blockchain and DNNs in IoT networks, and categorize these frameworks into three main types: *anomaly detection*, *secure data storage*, and *secure distributed networks*. We also identify their limitations, including a lack of integration between blockchain and DNN, using complex architectures with separate components which as a result increases the network latency and storage overhead, and overlooking the importance of the consensus algorithms.
- We propose a novel framework–DeepChainIoT–that fully leverages the mutual enhancement of blockchain and DNN to address security, storage and data dissemination issues in generic IoT networks. We show that DeepChainIoT offers efficient and secure transaction processing, storage and dissemination through a blockchain network that uses Long Short-Term Memory (LSTM) autoencoders to analyze and detect anomalies in transactions, nodes, network traffic and

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

smart contracts, and uses an optimized Practical Byzantine Fault Tolerance (PBFT) consensus with node rating to prioritize critical transactions and prevent malicious nodes. At the same time, DeepChainIoT uses a blockchain network to decentralize DNN models and preserves the integrity of the models and training data.

- We evaluated the LSTM autoencoder with an anomaly detection task on a pump sensor dataset collected from a smart water system. The LSTM autoencoder achieved an accuracy of 99.6%, a recall of 100%, a precision of 97.95%, and an F1 score of 98.97%. It also achieved a data compression ratio of 23.9, showing its ability to compress data and hence reduce storage and bandwidth requirements for blockchain nodes. Malicious sensors were categorized based on the number of anomalous transactions they sent within a defined transaction window, and transactions from non-malicious sensors were prioritized. Additionally, we evaluated the effectiveness of the LSTM autoencoder model in real-time anomaly detection. This marks significant improvements in anomaly detection compared to the results presented by Sapkota et al. [18], which only optimized the compression ratio.

The rest of the paper is organized as follows. Section 2 introduces blockchain and DNN technologies and discusses their limitations separately. Section 3 reviews existing studies on blockchain-DNN integration for IoT networks, categorizes and compares the frameworks proposed in these studies, discusses their limitations, and summarizes the overlooked potentials in integrating blockchain and DNN, which leads to the introduction and evaluation of the proposed DeepChainIoT framework in Section 4 and Section 5. Finally, Section 6 concludes the paper and identifies avenues for future research.

2. Background

This section provides background on blockchain and DNN as individual technologies, explores their potential applications in IoT networks, and discusses their limitations.

2.1. Blockchain

Blockchain offers a decentralized, Peer to Peer (P2P), immutable ledger that stores transactions in blocks [21]. Many cryptocurrencies like Bitcoin, Ethereum [22], Litecoin [23] and Dogecoin [24] were developed based on Satoshi Nakamoto' paper on Bitcoin [11]. Today, blockchain is not only used in finance, but also used in fields including IoTs, identity verification, healthcare, etc.

2.1.1. Working of a Generic Blockchain

Public Key Cryptography

Upon joining, participants first register to obtain valid identities based on public-key cryptography. The private key remains confidential to its owner and is used to generate digital signatures when sending transactions, while the public key is distributed to other participants and serves as a means of identification within the network.

Transaction Broadcasting

When a transaction is sent, it is broadcast to the networks. Ethereum uses the Ethereum Wire Protocol (Eth protocol), a variant of the gossip protocol, to propagate transactions and blocks [25,26]. Similarly, Bitcoin uses a P2P broadcasting mechanism [11], and Hyperledger Fabric also uses a gossip-based data dissemination protocol [27].

Smart Contract

A smart contract is a digitized and computerized piece of code stored on the shared ledger and is executed by all nodes [28]. Smart contracts can encode predefined transaction conditions, such as escrow and bidding, similar to contractual agreements. Notable examples include Ethereum's smart contracts [22] and Hyperledger Fabric's chaincode [29].

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Block

A block contains the hash of its transactions, a timestamp, and the header of the previous block. This structure allows blocks to be chronologically chained together, forming a blockchain, or the shared ledger. The genesis block is the first block mined on the blockchain. The Secure Hash Algorithm (SHA) is used to ensure immutability in many blockchain networks [30].

Decentralization and Consensus

Decentralization is achieved by allowing validating nodes to collectively validate transactions. In a cryptocurrency-oriented blockchain, the validating nodes validate the cryptographic identities of transaction senders and confirm their rights to spend. In a non-cryptocurrency blockchain such as Hyperledger Fabric [29], validators also verify if the senders' identities and access permissions are valid. After the identity validation, the validators then validate transaction details such as the amount to spend and verify the transaction against predefined endorsement policies. Next, the validators order and record transactions into a shared ledger following the consensus algorithms. Bitcoin uses Proof of Work (PoW) [11], Ethereum uses Proof of Stake (PoS) [31], and Fabric uses the Practical Byzantine Fault Tolerance (PBFT) [32] to achieve consensus in their respective networks. PoW ensures the security and integrity of the shared ledger by making it prohibitively expensive and time-consuming to confirm transactions and alter the transaction history. This often leads to concerns about its environmental impact and scalability as the network grows. The PoS consensus mechanism removes the need for using vast computational resources by selecting validators based on their stakes. This improves the transaction processing time, scalability, and energy efficiency compared to PoW. PBFT divides nodes into primary and backup nodes and allows a certain number of faulty nodes in the network. Upon initiation, a transaction is forwarded to and broadcast by a primary node selected in a round-robin manner. If the primary node fails or behaves maliciously, the backup nodes initiate a 'view change' process to elect a new primary node and continue the process. The transaction sender, or client, awaits at least $(f+1)$ confirmations, where f is the number of faulty nodes, from distinct nodes before adding the transaction to the shared ledger. While PBFT offers strong security guarantees and low latency, its communication overhead increases with the number of nodes.

2.1.2. Types of Blockchain

Based on permissions of joining, blockchains can be classified into permissionless blockchains, permissioned blockchains, and hybrid blockchains [33].

Permissionless blockchains such as Bitcoin and Ethereum allow anyone to join by executing a local instance of the associated P2P protocol. Global-scale permissionless blockchains often struggle to handle high volumes of transactions, and suffer from significant storage overhead and compromise user privacy as data is often duplicated across all nodes in their original format. Forking also frequently occurs in a large permissionless blockchain due to latencies, making it difficult to achieve complete consensus (cf. Section 2.3).

Permissioned blockchains, such as Ripple [34] and Hyperledger Fabric [29], limit participation in some or all roles to specific user groups and utilize PBFT consensus for fast transaction processing. These blockchains better protect user privacy than permissionless blockchains, making them more suitable for business collaborations. Permissioned blockchains also experience a reduced level of decentralization, as they often rely on a predetermined set of validators and a centralized identity certification body.

Hybrid blockchains, such as Dragonchain and Komodo [35], use a fixed set of validators while allowing regular users to join without being formally identified. These blockchains offer protection against network attacks (cf. Section 2.3), safeguard privacy, facilitate communication with external parties, ensure fast transaction processing, and exhibit improved scalability. However, upgrading the system can pose challenges due to the need for coordination between the public and private components of the network. Additionally, there often lacks a reward mechanism for users to actively

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

engage, which can lead to lower participation and reduced incentive for maintaining network security and validating transactions [35].

2.2. Deep Neural Networks (DNNs)

Neural networks consist of neurons, each with inputs, outputs, and constant or activation functions. The activation functions are applied to the input data to produce the desired output. Complex neural networks, composed of an input layer, multiple hidden layers, and an output layer, are known as Deep Neural Networks (DNNs) [36].

2.2.1. Working of DNNs

During training, input data passes through multiple layers of a DNNs. The DNN applies weights, which are numerical constants that transform the input data into an abstract representation through a series of linear and non-linear transformations [37,38] to the connections between the neurons. The output of each layer is computed by applying an activation function to the weighted sum of its inputs [38]. The weights are initially assigned randomly, and are adjusted to minimize the difference between the predicted output and the actual output during training. This adjustment is performed using back-propagation, an algorithm that calculates the gradient of the loss function and updates the weights [39,40]. This iterative process continues until the DNN converges to a set of weights that minimize the error or loss function, effectively allowing the network to recognize patterns in the training data [40]. Once the training is complete, the DNN can process new input data using the learned weights and transformations. If the incoming data matches the learned patterns, the DNN can accurately classify it or make predictions. However, if the incoming data deviates significantly from the training patterns, the DNNs will fail to produce an output and classify the data as anomalous.

Below, we illustrate the working of DNN with the sigmoid function as the activation function. Here, we have several inputs to the sigmoid-activated node, giving an output denoted as $a_0(1)$. Each line connecting the input to the node has a weight w, and the node has a bias b to appropriately adjust itself, making it more linear as in equation 1:

$$a_0(1) = \sigma(w_{00}a_0(0) + w_{01}a_1(0) + w_{02}a_2(0) + H(0) + \cdots + b_0) \tag{1}$$

Where σ represents the sigmoid function, which converts the weighted total of the inputs into a number between 0 and 1. With k inputs and n nodes, we obtain Equation 2:

$$a_k(1) = \sigma\left(\sum_{i=0}^{n} w_{ki}a_i(0) + b_i + H(0)\right) \tag{2}$$

The output of Equation 2 will serve as the input for the next layer, and that layer's output will become the input for the subsequent layer, and so on. The activation functions are chosen according to the network's needs, and the weights and biases keep adjusting until a stable output is generated. These parameters are adjusted with the help of hyper-parameters: optimizer, metrics, loss, and epochs [41].

2.2.2. Learning Algorithms

There are numerous learning algorithms to train DNNs for classification and prediction [42], which can be categorized into *supervised learning*, *unsupervised learning*, and *reinforcement learning*. In *supervised learning*, DNN models learn to map input features to corresponding outputs based on labeled input-output pairs. Convolutional Neural Networks (CNNs) [43] and Recurrent Neural Network (RNNs) [44] are the most renowned supervised learning algorithms. In *unsupervised learning*, models learn and extract information from data without the explicit labels or human guidance. The learning algorithm explores the data on its own to uncover underlying patterns, structures and relationships. Unlike supervised learning, unsupervised learning focuses on discovering hidden insights and making sense of the data without predefined labels. This allows models to identify clusters or groups of

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

similar data points, and uncover valuable knowledge. Unsupervised learning opens new possibilities for exploring and extracting meaningful information from unstructured or unlabeled datasets. Self-organizing maps [45] and autoencoder [46] are examples of unsupervised learning. *Reinforcement Learning (RL)* can identify optimal behavior in given circumstances through a sequence of choices to maximize rewards. This makes RL an independent, self-educating system that acquires knowledge through experimentation. Model-free and model-based RL [47] are widely used.

2.3. Using Blockchain and DNN in IoTs

IoTs have revolutionized our interaction with the environment by enabling the connection of devices and sensors to the Internet [48]. However, managing the large volume of diverse data generated by these devices presents security, privacy, and adaptability challenges [2,3]. IoT devices are also limited in their computing and storage capabilities and are therefore susceptible to cyber-attacks [2]. More recently, the integration of blockchain and DNN has been explored to establish trust and identify anomalies in IoT networks [49,50]. The tamper resistance, immutability, and distributed consensus of blockchains may help address the security challenges faced by IoTs. The advancement in deep learning algorithms can further enable IoT networks with tasks such as anomaly detection and prediction.

Nonetheless, blockchain and DNNs each have their own limitations that hinders their advantages when used directly in IoT networks. We summarise the limitations of blockchain technology in Table 1 and the limitations of DNNs in Table 2.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Table 1. Limitations of Blockchain.

Limitation	Description	Impact
Forking [51–53]	Multiple versions, or *branches* of the shared ledger exist, caused by block propagation delays or intentional malicious behaviors. Forking is more prominent and harder to resolve in larger networks.	Forking consumes the network's computing resources, slows down transaction processing, and introduces security vulnerabilities.
Latency [27,54–56]	The time required to verify transactions and store them on the shared ledger, affected by transaction validation, consensus, and block propagation.	Increased transaction volume results in higher latency, which slows down transaction processing compared to centralized systems and increases the likelihood of forking.
Throughput, measured in transaction per second (tps)[57–59]	The public Bitcoin network has a throughput of 7 tps, and the public Ethereum network has a throughput of 16.5 tps, while Hyperledger Fabric has a throughput of several thousands tps.	Blockchains, especially public blockchains have lower throughput than centralized systems and struggle to handle a large number of transactions simultaneously as validation is required from all nodes.
Network Attacks [60–62]	In PoW blockchains, an attacker controlling 51% or more network computing power can rewrite the transaction history. In PBFT blockchains, if more than one-third of the nodes are malicious, the network cannot reliably process transactions. Network traffic manipulation, such as Sybil attacks, affect all consensus mechanisms.	These attacks can lead to significant financial losses, manipulation of the shared ledger, disruption of consensus, and compromise of the network.
Privacy [63,64]	Decentralization leads to challenges in protecting privacy, especially when data is propagated in their original formats.	Sensitive information shared in the network may be exposed, leading to potential exploitation by other organizations or malicious actors.
Smart Contract Vulnerabilities [65,66]	Flash loans, arithmetic bugs, re-entrancy, and DOS exploit vulnerabilities in smart contract code. There is lack of advanced development languages and effective techniques for detecting and fixing bugs.	These vulnerabilities have led to billions of dollars in losses. Correcting them is costly and time-consuming due to blockchain immutability.
Message Sequences in Blockchain Consensus [11,32,67,68]	Message sequences are ordered communications such as transaction proposals, votes, and confirmations exchanged between validating nodes to reach agreements. In the absence of a prioritization mechanism, messages are processed based on transaction fees or on a first-come, first-served basis.	The lack of prioritization leads to inefficiencies, as non-critical messages, such as status updates or notifications, consume resources and delay the processing of more critical messages like transaction proposals.
Storage Space [56,69,70]	All nodes in a blockchain network store a full copy of the ledger, which requires significant storage capacity. E.g., A Bitcoin node needs about 200 GB of space, with daily uploads of 5 GB and downloads of 500 MB.	As the blockchain grows, the demand for storage increases, potentially leading to network congestion and higher energy consumptions.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Table 2. Limitation in DNNs.

Limitation	Description	Impact
Centralization [71]	DNN models are typically deployed on centralized servers, making them vulnerable to SPOF.	A failure in the central server can disrupt the entire system, impacting transaction processing and system reliability and availability.
DNN Model Integrity [72]	DNN models are susceptible to parameter-oriented attacks, such as the Bit-flip attacks, where an attacker alters a small number of parameter bits. Model theft attacks can also destroy the model integrity.	Compromised model integrity leads to unreliable results and a loss of trust in the system.
Training Data Integrity [13,72,73]	Maintaining training data integrity is crucial for DNN models.	Unauthorized injections, such as backdoor attacks, can significantly compromise model behavior and lead to security breaches and privacy violations.

3. The Blockchain-DNN Integration for IoTs

Multiple research studies have shown the integration of blockchain and DNNs allows them to effectively complement each other's limitations and provide a robust solution to solve the security, privacy and storage issues in generic IoT networks. To ensure a comprehensive review, we select the frameworks according to the following criteria:

- Relevance: We consider blockchain-DNN integrated frameworks used to solve security issues regardless of their application domains.
- Publication quality and impact: We select peer-reviewed articles from high-impact conferences and journals published within the last five years.
- Diversity: We include a variety of studies to ensure a broad perspective.

Based on these criteria, we selected twenty frameworks for review and classify them into three categories according to their objectives: *anomaly detection*, *secure data storage*, and *secure data distribution*. Among these twenty frameworks, eight focus on anomaly detection, six on secure data storage, and six on secure distributed networks.

Below, we discuss each framework in detail, along with their limitations, and summarize the overlooked potential in integrating blockchain and DNN in IoT networks.

3.1. Review of Blockchain-DNN Integration Frameworks

3.1.1. Anomaly Detection

Wang et al. proposed to combine blockchain and DNN to prevent cyber-attacks in IoMTs [74] that are highly reliant on centralized third-party services. A blockchain network was used to provide decentralization, data immutability, and fault tolerance. A Hierarchical Features Multi-Model Sequence Anomaly Detection algorithm (HFMMAD) combining low-level features extracted from various traffic sub-spaces and high-level contextual information in structured traffic data captured by a Bidirectional Long Short-Term Memory (BiLSTM) [75] was implemented to detect abnormal network activities that deviate from the established patterns of normal behaviors. Experiments demonstrated that HFMMAD significantly improved abnormal traffic detection in the IoMT-Blockchain environment compared to other traditional deep learning methods. However, the paper did not show how the anomaly detection model was integrated with the blockchain.

Shobana et al. implemented an optimized DNN with a blockchain [76] in an IoT network to enhance its security, trustworthiness, reliability, and confidentiality. The framework has two main stages: authentication and data processing. Initially, IoT data is directed to the blockchain for authentication using the Enhanced Proof of Work (EPoW) consensus mechanism to ensure end-to-end security and data integrity. The authenticated data is processed using the Modified Independent Component Algorithm (MICA) [77] for feature mapping, selection, normalization, and transformation. The pro-

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

cessed data is then fed into a Group Theory (GT) [78] based Binary Spring Search (BSS) [79] algorithm with a Hybrid DNN [80] model (GTBSS-HDNN) that uses trained data patterns to determine whether incoming packets are anomalous, sending alerts for anomalies while storing non-anomalous data in the cloud. The performance of the proposed framework was evaluated on the Telecommunications Network Internet of Things (ToN-IoT) [81] and the Botnet Internet of Things (BoT-IoT) [82] datasets. ToN-IoT includes common IoT/IIoT observations like backdoor, DDoS, DoS, injection, MITM, normal, password, ransomware, scanning, and XSS, with 1,498,334 attack instances and 79,053 normal instances. BoT-IoT also contain a diverse range of IoT network traffic and attack patterns. The framework achieved 95.3% accuracy, 96.54% precision, 95.23% recall, and 95.67% F1-score on the ToN-IoT dataset, and 96.23% accuracy, 95.94% precision, 97.03% recall, and 96.70% F1-score on the BoT-IoT dataset. These results demonstrate the model's effectiveness in real-time IoT attack prediction. However, the multi-step data processing, which involves transformation using MICA, authentication on-chain and anomaly detection significantly increased latency and reduced transaction throughput. Storing data in the cloud without any authentication mechanisms also poses risks of data tampering and unauthorized access. Simulation showed that the MDNN classifier achieved 96% accuracy, 0.04% error, and 95% precision, outperforming the other deep learning techniques. However, the transactions are stored in their original formats, which increases the storage overhead and compromises privacy.

Pian et al. [83] tackled the issue of detecting patients' stress levels in healthcare systems. They used a deep learning model that combines Attention Based BiLSTM (ABiLSTM) [84] and CNN [43] to classify medical time-series data into stress and non-stress categories, and blockchain smart contracts as intermediaries to fetch data from patients and supply it to the deep learning models. Sensitive data were encrypted using Elliptic Curve Cryptography (ECC) [85] and stored in IPFS. Evaluation of the models using the Wearable Stress and Affect Detection (WESAD) dataset [86] showed over 99% accuracy. Despite the connection between smart contracts and DNN, their implementations on separate systems can lead to an increased latency in propagating transactions and cause security and integrity vulnerabilities in the DNN model.

Jagdish et al. [87] combined blockchain and DNN for cyber-physical systems to enable smart, secure, and intelligent transmission with real-time intrusion detection across heterogeneous device connections. The DNN model employs four hidden layers to classify normal and attack data. After the classification, normal data is processed on the Ethereum blockchain, and later stored in the cloud. The model's authentication was tested on two real-time datasets, NSL-KDD15 and CIDDS-001 [88], achieving a maximum accuracy of 99.8%, precision of 99.5%, and an F1-score of 99% on the NSL-KDD15 dataset, and a maximum accuracy of 99.8%, precision of 99.9%, and an F1-score of 99% on the CIDDS-001 dataset. Despite its effectiveness, the DNN and blockchain are not optimized together and the data stored in the cloud is not tamper-free.

Konstantinos et al. [89] proposed to directly integrate AI with blockchain as an active structural element in an innovative blockchain security architecture for IIoT devices by implementing a bilateral traffic control agreement on smart contracts that consists of a trained deep autoencoder neural network to detect malicious web traffic and malfunctions, and security operations such as sending an alert to the security operation center when abnormalities are detected. This framework cannot handle large datasets of terabytes or support complex smart contracts due to memory limitation.

More recently, Intrusion Detection Systems (IDS) are being used to monitored computer networks by analyzing and identifying user behavior and predicting attacks with automatic responses. Saravanan et al. [90] proposed a Blockchain Based African Buffalo (BbAB) scheme with a RNN model for continuous monitoring and intrusion detection. Incoming data is encrypted using Identity Based Encryption (IBE) [91], validated by a blockchain network, and then sent to the cloud for intrusion detection. This approach achieved a remarkable accuracy of 99.87%, precision of 99.66%, and a recall of 99.92% on normal and malware user datasets, with the RNN model capable of locating suspicious activities and issuing alerts. However, transferring encrypted data from blockchain to the cloud can

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

cause additional latency, and ensuring the integrity and security of the DNN model in the cloud is more challenging than storing it on-chain.

Ashfaq et al. [92] proposed a blockchain based smart contract with XGBoost and Random Forest [93] models to enhance the security of digital transactions. The models classify transactions as malicious or legitimate based on a Bitcoin transaction dataset. The model achieved a precision and Area Under the Curve (AUC) of 0.9 and 0.92 respectively. Ashfaq et al. also evaluated the proposed framework by simulating Sybil attacks and double-spending attacks and performed a security analysis of the proposed smart contract and proved the effectiveness of the framework in detecting frauds and anomalies in the Bitcoin network. However, the paper did not detail how the anomaly detection model operated within the smart contract.

3.1.2. Secure Data Storage

To securely store employee attendance data as tamper-proof records, Dixit et al. [94] proposed a two-phase framework consisting of a machine learning phase for face detection and a blockchain phase for secure storage. The system used a cascade classifier and local binary pattern histogram [95] for face detection and recognition. When an employee stands in front of the camera, a trained model detects and recognizes their face. Upon identification, their attendance is stored on-chain network using the Message Queuing Telemetry Transport (MQTT) protocol [96]. Data within the blockchain is encrypted using the Advanced Encryption Standard (AES) algorithm [97], with access restricted to authorized nodes. The proposed classifier achieved 88% accuracy, demonstrating a reasonable reliability. However, there is a lack of effective integration between the neural network and blockchain. The encryption also increases data size and may lead to a higher storage overhead.

Albakri et al. developed a framework to securely store and distribute medical data using a Blockchain Based Smart Healthcare System With an Optimal Deep Learning Model (BSHS-EODL) [98]. IoMT devices first collect images and store them on-chain. Then, image encryption is applied for key selection, chaotic sequence pre-processing, diffusion, block scrambling, confusion, and expansion [99]. Finally, a Voting Extreme Learning Machine (VELM) classifier [100] with a single-hidden-layer Artificial Neural Network (ANN) is used for disease diagnosis. Simulation on medical datasets show BSHS-EODL achieves a maximum accuracy of 98.51%, outperforming existing methods such as the Deep Belief Network (DBN), You Only Look Once - Generalized Convolutional (YOLO-GC), Residual Network (ResNet), Visual Geometry Group-19-layer network (VGG-19), and Convolutional Deep Neural Network (CDNN) [101,102]. While the framework effectively integrates neural networks, blockchain and encryption, it also suffers an increased storage overhead due to encryption.

Hannah et al. [14] proposed to use a blockchain optimized with DNNs to efficiently manage cognitive data collected by IoTs. Initially, data gathered by the IoT devices is securely transmitted to the DNN based cloud through a blockchain network. Once stored on the cloud servers, the data undergoes pre-processing, feature extraction, and classification to identify disease types. An autoencoder network with multiple hidden layers is employed to learn characteristics from the data and identify diseases. The reliable information verified by DNNs is then distributed as transactions to the participants using a decentralized blockchain network. This framework promises fast and efficient delivery of healthcare data within a healthcare management system and offers a tamper-proof medium to continuously track data produced by IoT devices. Nonetheless, accessing data from the cloud through blockchain can increase latency.

Shailendra et al. [103] proposed to reduce cloud processing latencies by utilizing fog and edge layers in an intelligent 5G IoT framework. This framework uses Deep Learning (DL) for data analysis and blockchain for data security across cloud, fog, edge, and user layers. The fog layer uses Software Defined Network (SDN) controllers [104] to distribute tasks among fog nodes to enhance processing efficiency and data availability. The edge layer uses a blockchain to decentralize the processing across multiple nodes while only sharing essential data such as DL model parameters with the fog node. The framework was evaluated for latency, accuracy, and security in an object detection task. Results showed the decentralized edge layer satisfied more detection requests than the centralized fog and cloud layers.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

The fog layer also satisfied more requests than the cloud layer due to task distribution. The cloud layer handled more data traffic. Despite the reduced latencies and improved security, implementing this system across cloud, fog, edge, and user layers is resource-intensive and challenging, with high computational resource requirements at the edge and fog nodes potentially leading to bottlenecks and increased energy consumption.

Cai et al. [105] proposed GTxChain, a secure smart blockchain IoT architecture leveraging Graph Neural Networks (GNN) [106] for secure data processing and storage. GTxChain uses a distributed intelligent prophecy machine to obtain off-chain data and construct the on-chain transaction data structure through a Directed Acyclic Graph (DAG) [107]. This architecture enhances user privacy and reduces data processing time, achieving a 10.51% improvement in transaction processing time compared to the DAG Block [108] and STBitcoin [109]. However, GTxChain did not leverage edge or cloud resources, which increases the computational complexity and energy consumption for the GNN and blockchain integration, making it not feasible for many IoT devices.

Ruchi et al. [110] proposed a blockchain and fog computing model with Hybrid Encryption Algorithms (HEAs) for secure data access over distributed data storage and authentication in IoT devices. Integrating fog computing with blockchain extends cloud services to network edges, effectively addressing authentication, identification, and verification challenges in IoT devices while enabling frequent and scalable decentralized data transmission. HEAs enhance data security by combining Elliptic Curve Diffie-Hellman (ECDH) [111] and Secure Hash Algorithm 512 (SHA-512), and using the deep adaptive power probabilistic clustering algorithm [112] for optimal cluster head selection. This fusion of ECDH and SHA-512 provides secure key exchange and hashing, protecting against unauthorized access and data manipulation and achieved a score of 95% in ensuring secure and reliable data access and authentication in distributed environments. However, the computational overhead of encryption, particularly in hybrid algorithms, negatively affect the system performance.

3.1.3. Secure Distributed Network

Kumar et al. [50] propose a two-level Blockchain Enabled Deep Learning approach for Secure Data Transmission (BDSDT) in an IoT-enabled HS. The first level contains a scalable blockchain architecture using Zero-Knowledge Proofs (ZKPs) [113]. IoT devices join the blockchain network via a smart contract based ePoW consensus. The IPFS [114] was used to manage large volumes of IoT data off-chain to reduce storage costs. In the second level, authenticated data is used in a deep learning architecture for intrusion detection. This involves feature mapping, selection and normalization, with data encoded using a Deep Sparse Autoencoder (DSAE) [115]. The reduced-dimensional features extracted by the DSAE are then fed into a BiLSTM network, which employs two hidden layers: a forward LSTM layer capturing past information and a backward LSTM layer gathering future information. BDSDT was evaluated on the ToN-IoT and CICIDS-2017 [116,117] datasets. CICIDS-2017 contains various updated attack observations, including ransomware, SSH-Patator, FTP-Patator, DoS-Hulk, DoS-Slowhttptest, DoS-Goldeneye, injection, and MITM, with 390,222 attack instances and 2,035,505 benign observations. BDSDT outperformed state-of-the-art methods in both non-blockchain and blockchain settings, achieving nearly 99% accuracy with both datasets and validating data records, standardizing healthcare data transmission, and safeguarding against attacks. However, separating blockchain and DNN model leaves the DNN model vulnerable.

Dawid et al. [118] proposed a framework to securely share medical data among geographically diverse practitioners. This approach combines Federated Learning (FL) [119] with a public blockchain to protect the integrity of deep learning algorithms. The architecture consists of a global model provided by a medical center, local models trained on individual IoMT devices, and a blockchain. The iterative learning process involves sharing the global model with clients, who train local models and store the trained models on-blockchain as transactions. The central server aggregates these updates to refine the global model, which is then shared for further training. The blockchain ensures updates for FL come from trusted devices. The impact of FL using CNNs was evaluated on a dataset of 110 tuberculosis chest X-ray images [120], with the best result achieved using the inception model, which

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

showed a 1.7% increase in accuracy when trained with local models, demonstrating that FL with private data can enhance classifier accuracy and security. The validated data is stored and distributed in its original form, which can result in significant storage and bandwidth demands in a public blockchain network.

Randhir et al. [121] proposed PBDL, an efficiently optimized framework integrating a permissioned blockchain and smart contracts with DL techniques, to enhance real-time patient monitoring through data sharing among intelligent wearable devices and sensors over an insecure public network. Initially, the blockchain registers, verifies (using ZKPs), and validates communicating entities through a smart contract based PoW consensus mechanism. Then, the authenticated data undergoes DL processing with Stacked Sparse Variational Autoencoder (SSVAE) [122] and Self Attention Based Bidirectional Long Short Term Memory (SA-BiLSTM) [123] for format transformation and enhanced attack detection. A distributed IPFS storage maintains the complete transactions, with hashes stored on-chain for scalability and real-time data access, supplemented by cloud storage for long-term needs. Security analysis and experiments using BoT-IoT and ToN-IoT datasets achieved 94.34% accuracy and 8.89% loss, and 88.38% accuracy and 8.92% loss respectively. The framework offers a secure and efficient means of healthcare data transmission. However, as a blockchain was used to authenticate data before malicious transactions were detected with the neural network, malicious transactions are also validated by the blockchain.

Chandan et al. [124] proposed the Deep Learning and Blockchain Enabled Secure Data Sharing (DBSDS) for secure information exchange between power providers and consumers in smart grids. In DBSDS, service providers form a private blockchain network that generates and validates blocks associated with individual smart meters. DBSDS combines a Variational Autoencoder (VAE) [125] and ABiLSTM to an effective IDS named RENS. Normal transactions identified by RENS are used in a blockchain based access control mechanism to ensure secure and immutable data exchange. This decentralization ensures mutual authentication, decentralized computing, and tamper-resistant intrusion detection, protecting against replay, impersonation, Man-In-The-Middle (MITM), and physical capture attacks. Transactions validated by the neural network are stored in an IPFS storage, with the corresponding hashes stored on-chain. RENS IDS achieved 99.99% accuracy, 98.99% detection rate, 99.99% precision, 99.91% F1 score on the ToN-IoT dataset, and 99.99% accuracy, 98.97% detection rate, 90.65% precision, and 94.63% F1 score on the BoT-IoT dataset. Nonetheless, the intrusion detection model and blockchain were not optimized together.

Rathod et al. [126] proposed an AI and blockchain-driven architecture for secure and privacy-preserving data dissemination in IoT-enabled critical infrastructure. This architecture consists of four layers: *data collection*, *intelligence*, *blockchain*, and *application*. In the data collection layer, both raw and tampered data is gathered from various IoT devices in water treatment and thermal plants to detect data poisoning attacks. The data is then processed in the intelligence layer using dimensionality reduction techniques like Principal Component Analysis (PCA) and Explainable AI (XAI) [127], and passed to classifiers such as random forest, decision tree, Support Vector Machine (SVM), perceptron, and Gaussian Naive Bayes (GaussianNB) [93] for anomaly detection. The non-malicious data was sent to the blockchain, where smart contracts were used to support advanced user agreements, and IPFS was used for distributed file storage and version tracking. A file based web console dashboard was used to select IPFS content identifiers and generate hashes using the SHA-256 algorithm. In the application layer, authorized personnel can then access critical infrastructure data. This architecture was evaluated with metrics such as accuracy, precision, recall, F1 score, and ROC curve, showing the random forest classifier outperforming others with 98.46% accuracy. However, there is a lack of integration between the anomaly detection and blockchain storage phases.

Varun et al. [128] proposed a secure platform to enhance digital governance trustworthiness and data exchange using blockchain and deep learning. Initially, a blockchain with a bonobo optimization algorithm [129] authenticates data from smart cities. A lightweight Feistel structure is integrated to preserve privacy and secure data exchange. A Deep Reinforcement Learning (DRL) model [130]

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

detects and prevents intrusions such as fraud and corruption, thereby improving transparency and accountability. The framework outperformed similar proposals on BoT-IoT and ToN-IoT datasets and was proved effective in real-time waste management and electronic voting systems. Operating on both fog and cloud, the framework also mitigates the drawbacks of standalone architectures and collaborative networks. Nonetheless, as the neural network detects malicious activities only after blockchain validation, the blockchain validators will potentially validate malicious transactions.

Table 3 summarizes the above frameworks in terms of their ability in anomaly detection, storage overhead reduction, and more.

Table 3. Comparison of Blockchain-DNN Integration Frameworks.

Framework	Detect Anomalous Transaction/Behavior	Reduce Storage Overhead	Optimize Transaction Latency	Enhanced Data Privacy	Optimized Consensus	Data Encryption Used	Cloud/IPFS Storage Dependency	Blockchain-DNN Optimization
IoMTs [74]	✓	×	✓	×	N/A	×	×	✓
[76]	✓	×	×	×	✓	×	Cloud	✓
[49]	✓	×	✓	×	N/A	×	×	✓
[83]	×	✓	×	✓	N/A	ECC	IPFS	×
[87]	✓	×	×	×	✓	×	Cloud	×
[89]	✓	×	×	×	N/A	×	×	✓
BbAB [90]	✓	×	×	✓	N/A	IBE	Cloud	×
[92]	✓	×	✓	×	✓	×	×	✓
[94]	×	×	×	✓	N/A	AES	×	×
BSHS-EODL [98]	×	×	×	✓	N/A	Image Encryption	×	✓
[14]	×	✓	×	×	N/A	×	Cloud	×
[103]	×	✓	×	×	N/A	×	Cloud	×
GTxChain [105]	×	×	✓	×	N/A	×	×	✓
[110]	×	×	✓	✓	N/A	ECDH and SHA-512	Cloud	✓
BDSDT [50]	✓	✓	✓	✓	✓	ZKPS	IPFS	×
[118]	×	×	×	×	N/A	×	×	✓
PBDL [121]	✓	✓	✓	✓	N/A	×	IPFS	×
DBSDS [124]	✓	✓	✓	✓	N/A	×	IPFS	×
[126]	✓	✓	✓	✓	N/A	×	IPFS	×
[128]	✓	✓	✓	×	N/A	lightweight Feistel Structure	Cloud	×

3.2. Remaining Challenges

We highlight areas where further research and development are needed to enhance the effectiveness, reliability, and usability of blockchain-DNN integration frameworks by exploring the remaining challenges identified in existing studies.

Consensus Mechanism and Energy Consumption

Many existing studies tend to overlook the importance of the consensus mechanism. The most common consensus algorithms are PoW, PoS, and PBFT. PoW consumes excessive energy and often results in slow transaction processing. PoS validators are selected based on stakes, centralization is inevitable when minorities control majority of the stake. PBFT is fast but allows up to one-third of the nodes to behave maliciously. The efficiency of these consensus algorithms is crucial for minimizing network delays, as synchronization in a decentralized network is often challenging and time-consuming. The energy consumption of these mechanisms is influenced more by the network size than by transaction volume. Therefore, optimizing and accelerating the consensus is vital for improving the overall performance and sustainability.

Integration of Neural Networks and Blockchain

The neural network algorithms and blockchain platforms used in these frameworks are often used separately and not optimized together. This lack of integration prevents neural networks from utilizing the tamper-proof and immutable properties of blockchain, leaving training data and models vulnerable.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Data Storage and Confidentiality

Most existing studies proposed to store transactions in their original forms on-chain, which requires substantial storage spaces. This also allows any node to easily view transaction information, which may compromise privacy. Some frameworks proposed to use cloud for data storage, leaving authentication on-chain. However, this results in increased latency especially when data is frequently transmitted between the blockchain and the cloud.

Therefore, there is a need for a novel framework where blockchain and neural networks are optimized together as a single platform. By utilizing blockchain's immutability, we can prevent the neural network models and their training data from being tampered. With neural networks, we can detect anomalies in transactions and discard them before further processing in the blockchain to reduce the transaction volume. With transaction compression before sending them to the blockchain, we can reduce the storage space required on-chain without losing important information. The DNN model should also be simple, yet effective, fast, and reliable.

3.3. The Overlooked Potentials

Considering the limitations of DNN and blockchain as standalone technologies, as well as the remaining gaps in existing integrated blockchain-DNN frameworks, we identify the overlooked potentials in integrating blockchain and DNNs within IoT networks, and summarize how blockchain can improve DNN in Table 4, and how DNN can enhance blockchain in Table 5 and Table 6.

Table 4. Blockchain for DNN.

Enhancement	Blockchain	Description
Overcoming Centralization	Decentralized Model Updates, Fault Tolerance, Distributed Computation	Blockchain enables decentralized model updates, allowing all nodes to access updated models simultaneously through smart contracts. It ensures fault tolerance by enabling other nodes to continue tasks despite node churns, and it distributes computational tasks across the network [131,132], leading to faster response times and higher accuracy [133]. This overcomes the SPOF of centralized DNN servers, making them suitable in IoT applications.
Preserving Integrity of DNN Models and Their Training Data	Immutability of blockchain, Consensus mechanisms, Smart contracts, Cryptography	Blockchain's immutable ledger ensures the integrity of DNN models and their training data by storing transparent, auditable records that prevent unauthorized tampering. Consensus mechanisms validate updates, safeguarding against malicious actions such as DNN backdoor attacks [73], while smart contracts enforce strict access control, allowing only authorized changes. To protect sensitive training data, cryptographic techniques like zero-knowledge proofs (ZKP), secure multi-party computation, and homomorphic encryption [134] enable secure data verification and computation without revealing private information [135].

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Table 5. DNN for Blockchain-1.

Enhancement	DNN Model	Description
Reducing Forking	Autoencoder	Detects and discards double-spend transactions, reducing the occurrence of conflicting transactions and preventing forks [136].
	RNN/LSTM	Dynamically adjusts consensus rules based on real-time analysis of transaction volumes and network activities, such as modifying the difficulty level in a PoW system, adjusting stake requirements in a PoS system, or changing parameters, e.g., the number of required confirmations or the selection criteria for primary and backup nodes in PBFT. This ensures consensus rules remain efficient and responsive to the current network state, minimizing the likelihood of forks [137].
	CNN	Analyzes block data to optimize hashing algorithms, reducing the likelihood of nodes mining on different blocks and preventing divergence from the main chain [138].
	GNN	Predicts network failures and optimizes block propagation paths to reduce forks [106].
	GNN/DRL, Multi-Agent Cooperation Models	Enhances blockchain scalability by efficiently managing computational resources to handle forking in large networks [139,140], learning optimal resource allocation strategies to prevent system resource waste, mitigating weak computing power issues, and avoiding slowdowns in network performance [130].
Reducing Latency and Improving Throughput	DNN	Identifies and discards anomalous transactions and nodes by analyzing typical network behavior, including transaction frequency, size, and timing. These DNN models can be integrated into smart contracts, allowing detecting and blocking malicious nodes [141]. By eliminating anomalous transactions and nodes, the honest part of the network has fewer transactions to process, which leads to reduced latency and improved throughput.
	Autoencoder	Autoencoders compress transactions without losing important information. By applying them to legitimate transactions before sending them to the blockchain, the transaction size is reduced, making transaction processing and storage more efficient.
	DNNs	DNNs can be trained to mitigate network attacks [142] by learning characteristics such as data prioritization, packet routing, and financial or smart contract-related traffic patterns. The model can also detect peak traffic hours and activity fluctuations [143]. When unusual traffic or behaviors are detected, automatic alerts can be initiated to all participating nodes [144].
Mitigating Smart Contract Vulnerabilities	DNN	DNNs trained with known vulnerabilities and malicious code patterns can analyze smart contract code for issues such as reentrancy attacks, integer overflow, and unauthorized access [145] by flagging suspicious lines [146] and suggesting or applying patches or modifications to mitigate exploitation risks [147,148].

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Table 6. DNN for Blockchain-2.

Enhancement	DNN Model	Description
Optimizing Message Sequences in Blockchain Consensus	DNNs, Encoder-Decoder RNN	An integrated neural network can prioritize crucial messages, e.g., transaction validity confirmations for immediate processing. This streamlines the consensus process, reduces latency and improves transaction processing efficiency. Prioritization techniques have been applied in smart city protocols to handle critical transactions like emergency vehicle notifications [67] and in distributed task processing for Unmanned Aerial Vehicles (UAVs) [149].
Reducing Storage Space	Autoencoders	Autoencoders compress transactions to significantly reduce the required storage space. As the shared ledger grows, efficiently compressing non-anomalous data reduces storage needed and optimizes blockchain resource utilization for IoT applications.
Preserving Privacy	Autoencoders	Autoencoders can be utilized to encode transactions before storing them on-chain. This ensures data privacy by allowing only the corresponding decoding algorithm to decode the transactions. Transactions that fail to be decoded are flagged as anomalous and discarded.
Node Rating	ANN Model, Reputation System	Node rating evaluates and ranks nodes based on historical activities such as transaction frequency, type, and timing. This helps identify and exclude anomalous nodes from the consensus process in real-time, reflecting current network conditions and ensuring network integrity and security [150].

Overall, we have shown that DNN can enhance the performance of the the blockchain network by reducing forking, latency, and storage overhead, improving transaction throughput, optimizing message sequences to improve consensus efficiency, mitigating smart contract vulnerabilities, preserving privacy, and enabling better network management through node rating. Blockchain eliminates centralization and preserves the integrity of the DNN models and its training data. Figure 1 summarizes and visualizes these mutual enhancements. By fully leveraging these overlooked potentials, the integration of DNN and blockchain can be used to create more secure, efficient, and resilient IoT applications.

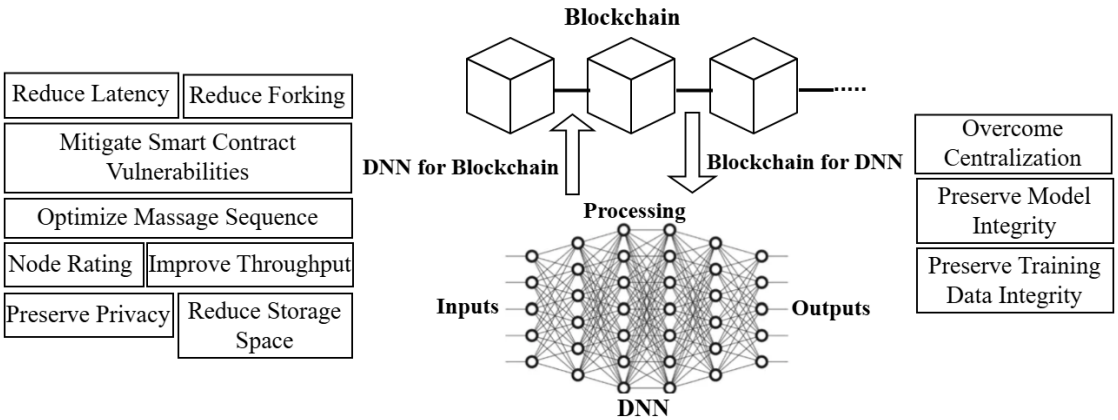

Figure 1. Potential of Blockchain-DNN Integration.

4. DeepChainIoT: Optimized Blockchain-DNN Framework for Secure IoT

We now present a novel, optimized blockchain-DNN integration framework for IoT networks, called DeepChainIoT, which leverages overlooked potential in existing studies to enable anomaly detection, as well as secure storage and distribution of transactions across generic IoT networks.

The DeepChainIoT framework draws inspiration from the solution proposed by Sabina et al. [18], which integrates an LSTM autoencoder within the endorsement chaincode to effectively detect and filter anomalous IoT transactions, while encoding valid ones before submitting them to the orderer and

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

validating nodes for secure storage and distribution in a Hyperledger Fabric network. This method can improve IoT transaction security and improve on-chain transaction processing efficiency, but it has several limitations. First, it does not address the role of the consensus mechanism in reducing latency or improving throughput. Second, it focuses solely on detecting anomalous transactions, without considering anomalous node behavior, allowing malicious nodes to continue validating and propagating transactions.

DeepChainIoT addresses these limitations by integrating an optimized consensus mechanism alongside node behavior monitoring to enhance both security and performance in IoT networks.

4.1. Architecture

Figure 2 illustrates the architecture of DeepChainIoT, where transaction processing is handled by a hybrid blockchain. The centralized certificate authority (CA) is responsible for issuing cryptographic identities to all IoT devices, users, and validating nodes. The network administrator (admin) monitors network activities and takes actions when malicious activities are detected. IoT data from various registered devices is sent to an IoT gateway for aggregation and processing into a transaction format `<Digital signature, node data>`, which is then sent to the blockchain for validation and storage. Registered users can then request data from the blockchain validators. This hybrid approach combines centralized identity management of permissioned blockchains with the openness and transparency typically associated with public blockchains, ensuring data security and accessibility.

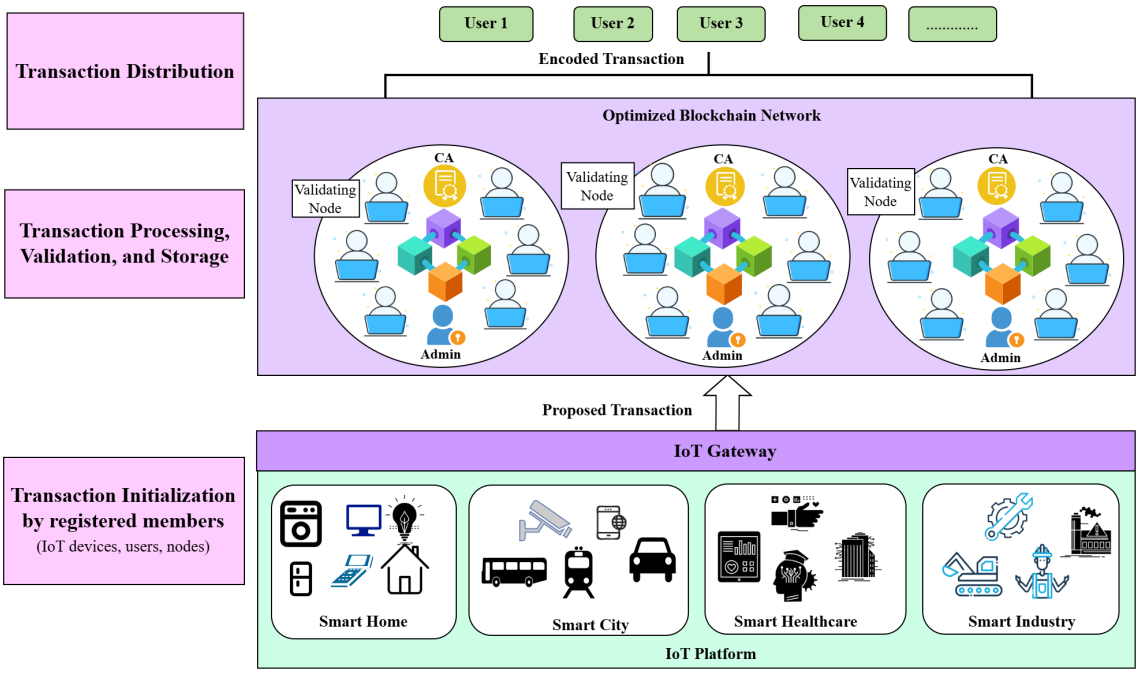

Figure 2. DeepChainIoT Architecture.

4.1.1. LSTM Autoencoder for Anomaly Detection and Transaction Encoding

One key component of DeepChainIoT is an LSTM autoencoder-enabled smart contract deployed on the validator nodes as shown in Figure 3. This smart contract contains the DNN models to detect anomalies in transactions, transaction senders, network traffic, and to encode the non-anomalous data.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

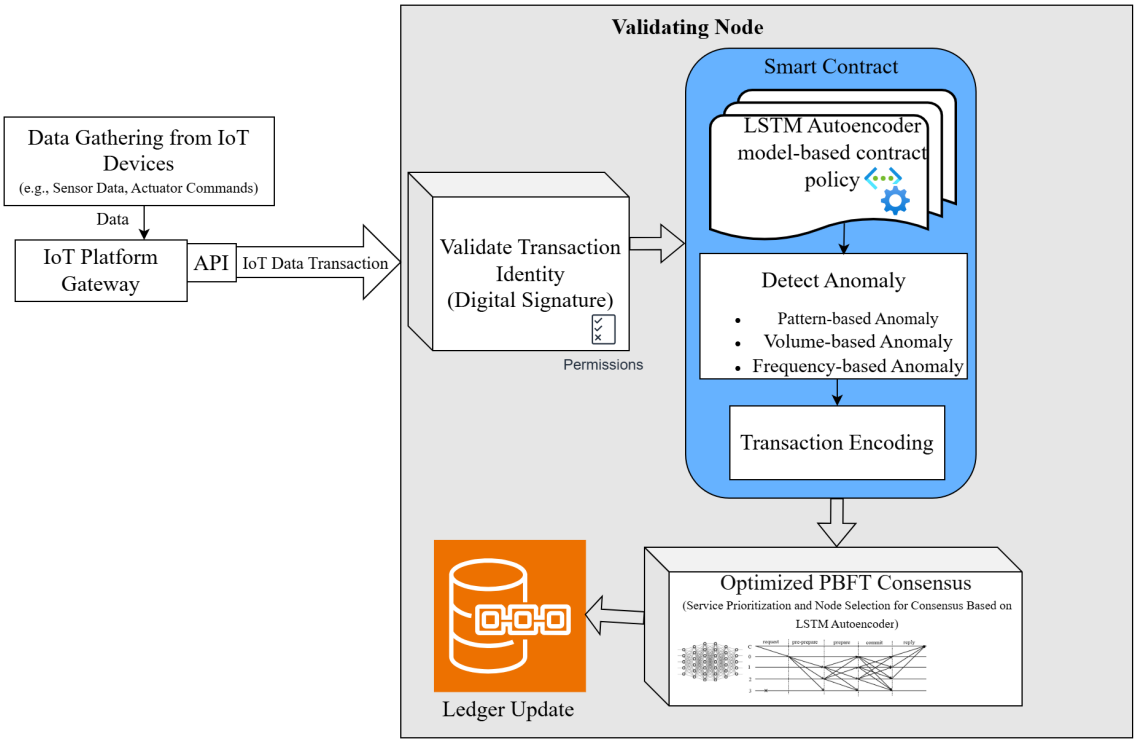

Figure 3. Internal Architecture of Validating Node.

LSTM, a supervised learning algorithm, is renowned for predicting future outcomes, even with long-term dependencies [151]. In contrast, an autoencoder is unsupervised and aims to replicate the input as output by retaining only the most relevant features [46]. The proposed LSTM autoencoder therefore consists of two main phases: the *LSTM* phase and the *autoencoder* phase. Firstly, an LSTM model is trained with labeled data to learn optimal weights and biases. The output, containing the learned temporal features, is then passed to the autoencoder. Next, the autoencoder compresses the input data into a lower-dimensional representation, or a latent variable by retaining the most relevant features while filtering out the noise (cf. Section 2). Instead of storing transactions in their original forms, DeepChainIoT stores the encoded transactions. After the encoding, non-anomalous transactions are sent to the blockchain network for validation and stored in the shared ledger if validated, while anomalous transactions are discarded and the details of their senders are sent to the admin, who can then take appropriate actions, such as removing malicious nodes to prevent their participation in consensus, or halting the network. During decoding, the autoencoder effectively removes noise as part of the reconstruction loss and converts the latent variables into a reconstructed version of the input. The threshold for loss is empirically determined. When the loss exceeds the threshold, the decoding cannot accurately reconstruct the input, and therefore the original data is regarded as anomalous and discarded.

4.1.2. LSTM Autoencoder Integrated PBFT Consensus

We propose an optimized PBFT consensus mechanism for our hybrid blockchain network, integrating an LSTM autoencoder to enhance performance and security. This integration aims to prioritize transaction processing based on urgency, detect malicious behaviors in real-time, and improve consensus efficiency by dynamically adjusting node participation.

The LSTM autoencoder is first trained on historical transaction data to learn temporal patterns associated with legitimate, high-priority transactions and potentially malicious activities. The autoencoder compresses transaction-related messages into latent feature representations, filtering out redundant data while retaining crucial patterns. The optimized PBFT consensus mechanism consists of the following key components:

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

- **Message Reception and Pre-processing**: Validating nodes receive transaction proposals, confirmations, and status updates, which are pre-processed into ordered sequences and fed into the LSTM autoencoder for pattern recognition and anomaly detection.
- **Feature Extraction & Dynamic Transaction Prioritization**: The LSTM autoencoder analyzes transaction characteristics to identify patterns related to transaction validity and importance. Based on these learned features, transactions are classified into different priority levels. High-priority transactions such as transaction validity confirmations, are processed immediately, while low-priority transactions, such as status updates, are processed later. The primary node (of PBFT consensus) then adjusts the message sequencing based on transaction priorities before broadcasting it.
- **Adaptive Node Rating & Malicious Node Detection**: Each node's behavior is continuously monitored through message sequences. Nodes that participate honestly in consensus are deemed honest. Nodes that frequently send conflicting or invalid transactions are flagged as suspicious or malicious nodes. The suspicious nodes are given a lower voting weight in consensus, while the malicious nodes are excluded. When a primary node is identified as malicious, an automated view change is triggered to replace it, switching to a new primary node.
- **Byzantine Fault Tolerance & Security Reinforcement**:If more than one-third of the nodes are flagged as malicious, an alert is triggered, and a notification is sent to the admin, who is responsible for initiating protocols and taking appropriate actions. One of the following actions will then be performed:
  - Network Halting: Temporary freeze on new transactions.
  - Consensus Reconfiguration: Adjusting voting weight of or completely excluding faulty nodes.

4.2. Workflow

The working of DeepChainIoT, as shown in Figure 3, consists of the following steps:

1. Node registration: Upon joining the network, each node responsible for sending transactions, must be registered with the CA to verify their identities and obtain unique private-public key pairs.
2. Transaction initialization: Data collected by various IoT devices is first aggregated and processed in an IoT gateway, where the data is transformed into a transaction format `<Digital signature, node data>`, before being sent to the blockchain.
3. Sender verification: Upon receiving a new transaction, each validating node verifies its digital signature, making sure it comes from a registered node.
4. Anomaly detection and transaction encoding: Validating nodes execute an LSTM autoencoder-embedded smart contract to discard anomalous transactions and encode the non-anomalous transactions.
5. On-chain transaction verification and storage: Using the optimized PBFT consensus, transactions are validated by the blockchain and appended to the shared ledger.
6. Data dissemination: When a node requests certain data from a validator, the validator verifies the requester's identity and only shares the encoded data if the requester is verified. The encoded data can only be decoded by the corresponding decoding algorithm so that only approved applications and requesters can access the original content.

5. Evaluation

This section provides a comprehensive evaluation of the proposed DeepChainIoT framework, analyzing its key features and performance across various aspects. Specifically, we assess the LSTM autoencoder's performance using multiple metrics on the pump sensor dataset for a smart water system and evaluate enhancements to the PBFT consensus mechanism. We also examine the framework's impact on user experience, trust, and data control, along with scalability, cost, and comparisons with

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

existing frameworks. Finally, we discuss the challenges of implementing DeepChainIoT in real-world IoT environments.

5.1. Features

Node Authentication

DeepChainIoT uses public-key based identity management to verify transaction senders' identities and prevent unauthorized access to data shared in the network. This achieves a similar authentication benefit as using a public key-based proxy in [152].

Anomaly Detection During Transaction Initiation

DeepChainIoT uses an LSTM autoencoder embedded smart contract to detect and discard malicious transactions and network traffic before further processing the transactions in the blockchain.

Node monitoring and alerts

DeepChainIoT monitors node behaviors to identify malicious activities and nodes. When a node is found malicious, an alert is sent to the admin to take appropriate actions.

Prioritization-Based Consensus

Enhances PBFT algorithm consensus efficiency by prioritizing critical transactions.

Reduced Forking

DeepChainIoT collects various real-time network data and analyzes them with the LSTM autoencoder-based DNN model to detect conflicts, predict network failures and dynamically adjust the consensus algorithm, which significantly reduces the likelihood of forking.

Privacy-Preserving and Reliable Data Storage and Dissemination

Transferring large volumes of data within a blockchain network poses challenges related to bandwidth, storage, and privacy. After the anomaly detection, only legitimate transactions undergo an LSTM autoencoder-based encoding. As a result, the reduction in transaction volume and size reduces the workload and storage space needed by the validating nodes. At the same time, by storing encoded transactions in the shared ledger, only authorized users with the corresponding decoding algorithm can access the original content. The blockchain provides immutability, making it hard for malicious players to tamper with the validated transactions.

Reduced Latency and Improved Throughput

The reduced transaction volume and size also reduces the latency for propagating transactions, leading to an improved transaction throughput.

Safeguarding the DNN Model and Training Data

The DNN model and the data used for training are immutable, as they are implemented within the blockchain network on validating nodes, preventing any unauthorized alterations and access.

Decentralization

Decentralization is provided by the blockchain network, which mitigates SPOF, synchronizes DNN model updates across all nodes, and ensures continuous operation of the network despite node churn.

5.2. Performance of the LSTM Autoencoder

To evaluate our proposed LSTM autoencoder for anomaly detection and secure communication, we use a pump sensor dataset collected from a smart water system [153].

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

The pump sensor dataset contains 220,320 timestamped entries, each capturing readings from 52 sensors that monitor key environmental and operational parameters such as pressure, flow rate, and temperature, etc. We first removed unnecessary columns, filled in missing NaN values with the mean of each column, and encoded the 'machine_status' labels, which categorized the data into 'Normal', 'Recovering', and 'Broken'. We considered both 'Recovering' and 'Broken' machine status as anomalous as the readings during these two status are unreliable and indicate potential pump malfunctions or deviations from expected behavior. In total, we identified 205,836 normal data entries and 14,484 anomalous entries. We split the normal dataset into training and testing subsets, using 70% of the data (144,085 entries) for training and 30% (61,751 entries) for testing. Next, we normalized the data using a standard scaler and reshaped it for input into the LSTM autoencoder, which consisted of two layers with 128 and 64 neurons, respectively. The model was fine-tuned using the Adam optimizer and trained for 10 epochs with a batch size of 512 and a dropout rate of 0.2 to prevent over-fitting. The model yielded a training loss of 0.5763 and a validation loss of 0.5751 during testing.

We derived the anomaly detection threshold of 0.79 from the mean absolute error (MAE) of the training data, which was calculated as the average of the absolute differences between the predicted and actual values across all 52 sensors. Data points in both the normal and anomalous datasets with an MAE exceeding this threshold were classified as anomalous. Figure 4 shows a comparison of MAE values between the normal and anomalous datasets, clearly indicating that the MAE value for anomalous data is significantly higher than for normal data, with most values exceeding the anomaly detection threshold of 0.79.

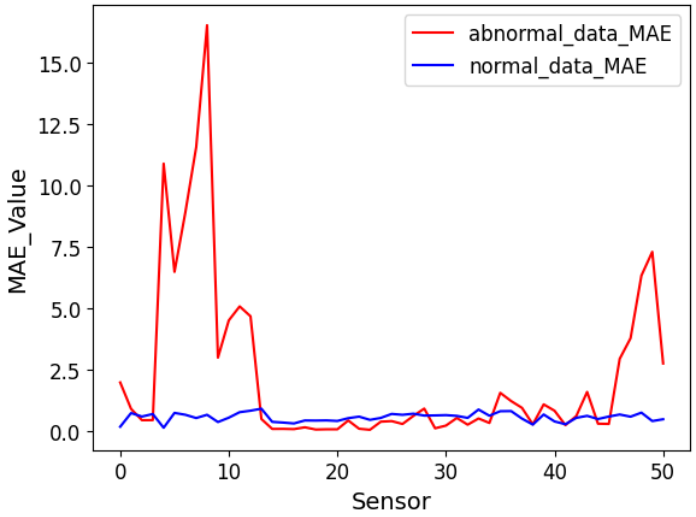

Figure 4. MAE Comparison of Normal and Anomalous Data in the Pump Sensor Dataset.

The model correctly classified 61,421 normal instances (true negatives) and 14,454 anomalous instances (true positives), achieving an accuracy of 99.6%, with a recall of 100%, indicating flawless detection of true positive cases. It also attained a precision of 97.95% and an F1 score of 98.97%, demonstrating a well-balanced and robust performance. Additionally, the autoencoder effectively compressed the data by a factor of 23.9, significantly reducing the data size while retaining key information for anomaly detection. We further analyze the anomalous dataset and identify the sensors in the network that exhibit anomalous behavior and pose potential threats, as illustrated in Figure 5.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

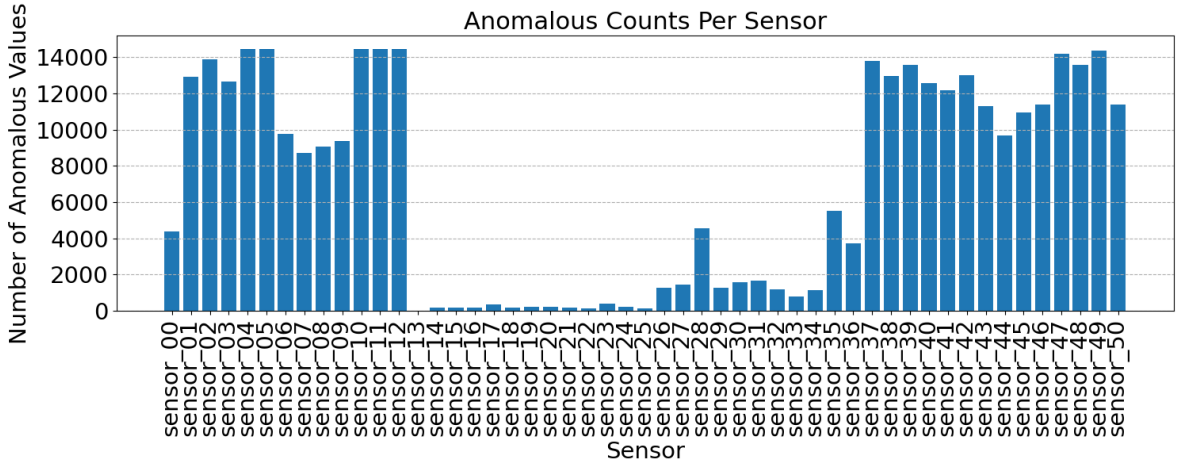

Figure 5. Anomaly Detection Counts for Each Sensor in the Smart Water System.

Sapkota et al. [18] also applied an autoencoder model to the same pump sensor dataset, derived the anomaly detection threshold as 0.781 and achieved a compression ratio of 47.81. However, their analysis did not account for other critical performance metrics such as accuracy, recall, precision, and the F1 score. In comparison, with a reduced compression ratio, our model achieves an overall better performance in detecting anomalies.

5.3. LSTM Enhanced PBFT Consensus

To detect malicious nodes in the network, we define a transaction window, representing a fixed number of transactions sent by a node before evaluation. Within this window, if the number of anomalous transactions sent by a node exceeds a predefined threshold, the node is classified as anomalous, and appropriate actions are taken. Once a transaction window is completed, a new window begins, resetting the anomalous transaction count to ensure that past behavior does not indefinitely affect the node's status. This approach prevents all nodes from eventually being classified as malicious while maintaining continuous monitoring.

In our pump sensor analysis, the LSTM autoencoder model continuously monitors transactions within a fixed transaction window of 14,454 total sensor entries, as shown in Figure 5 and categorizes sensors based on their anomalous transaction counts, summarized in Table 7. If a sensor exceeds 12,000 anomalous transactions within its transaction window, it is classified as critical, requiring immediate action. This method ensures a balanced evaluation by detecting malicious activity within a controlled scope. If a node is identified as anomalous, its transactions undergo additional review before processing, while normal transactions continue to be processed with minimal delay. This ensures that the network prioritizes legitimate transactions, maintaining efficiency and preventing bottlenecks caused by potentially malicious nodes.

Table 7. Categorization Framework for Sensor Anomalies.

Category	Range	Description
Normal	Count = 0	No anomalies. Sensor operating as expected
Minor	1 – 500	Low level anomalies. Monitor over time.
Moderate	501 – 4000	Noticeable anomalies. Requires investigation.
Major	4001 – 12000	Significant anomalies. Requires urgent actions.
Critical	$>$ 12000	Severe anomalies. Immediate action required.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Additionally, the model's ability to process 61,728 samples with a batch size of 16 (totaling 3,858 batches) demonstrates its efficiency, completing the prediction process in 134 seconds, with each batch taking approximately 35 milliseconds and each sample around 2.19 milliseconds. This performance meets the real-time constraints of our smart water system, where transactions must be processed within milliseconds. If needed, efficiency can be further improved through model quantization or hardware acceleration, ensuring fast transaction validation and anomaly detection without disrupting the consensus mechanism.

Integrating the LSTM autoencoder into PBFT introduces a DNN-driven consensus optimization that enhances both efficiency and security in blockchain networks. The adaptive node rating mechanism enhances PBFT's fault tolerance by continuously evaluating node behaviors and managing their participation in consensus. By dynamically prioritizing transactions, our approach ensures that critical operations are processed with minimal delays while transactions from suspicious nodes undergo additional review. Lastly, our framework optimizes real-time fraud and anomaly detection, making blockchain-based systems more secure and reliable for IoT applications.

5.4. Impact on User Experience, Trust, and Data Control

DeepChainIoT significantly enhances user experience by improving storage efficiency, response time, and strengthening data security and overall system trust. During evaluation (cf. Section 5.2), out of 220,320 data entries collected from 52 sensors, 14,484 anomalous entries (6.57%) were accurately identified and excluded by the LSTM autoencoder, preventing unnecessary processing of malicious data and reducing network congestion. This early rejection of anomalous transactions during the validation phase saves time, streamlines transaction processing, and improves system responsiveness. Furthermore, this targeted detection reduces the computational and validation load on normal, non-malicious nodes, enhancing the overall user experience.

Trust is further reinforced by an adaptive node rating mechanism that ensures only reliable nodes participate in the consensus process. The model groups data into 14,484 transaction windows (cf. Section 5.3), and any sensor exceeding 12,000 anomalous transactions within a window is classified as critical. Sensors such as sensor_0, sensor_05, sensor_10, sensor_11, and sensor_12 were flagged as high-risk, demonstrating the system's effectiveness in isolating compromised nodes. By identifying and removing malicious nodes, DeepChainIoT prevents collusion and Byzantine attacks, thereby strengthening network integrity. Additionally, by dynamically prioritizing transactions, users submitting high-priority requests experience significantly reduced delays compared to traditional PBFT networks, ensuring reliable communication even under heavy load.

The LSTM autoencoder also achieved a compression factor of 23.9, substantially reducing blockchain storage requirements. This not only cuts storage costs but also accelerates synchronization and query operations for both users and validators, resulting in a faster and lighter experience. Verified and encoded transactions stored on-chain guarantee data integrity and make it infeasible for unauthorized actors to manipulate or disrupt the consensus process.

When users request data, they receive encoded transactions from the blockchain rather than raw data. This allows them to perform decryption locally within their own applications, ensuring that sensitive information remains secure even if accessed by unauthorized parties. By allowing only legitimate, non-anomalous transactions into the ledger and continuously monitoring the network to remove malicious nodes, DeepChainIoT ensures that only trusted entities handle user data. As a result, it provides a secure, efficient, and trustworthy IoT data management environment.

5.5. Scalability Considerations

Scalability is a key concern when integrating blockchain and DNN for IoT security. DeepChainIoT tackles scalability challenges at multiple levels:

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

Efficient Storage and Transaction Handling

DeepChainIoT rejects anomalous transactions before processing them on-chain, reducing the time and resources that would otherwise be spent on them. DeepChainIoT utilizes an LSTM autoencoder to encode non-anomalous transactions, inherently compressing them before storing them in the ledger, which significantly reduces storage overhead and minimizes transaction size, improving the system's capacity to handle large-scale IoT data.

Optimized PBFT Consensus for High Throughput

Traditional PBFT suffers from communication overhead $O(n^2)$ [20], making it less scalable for large networks. DeepChainIoT overcomes this by prioritizing critical transactions and restricting malicious nodes from participating in consensus.

DNN Model Efficiency for Real-Time Anomaly Detection

The LSTM autoencoder is lighter-weight compared to traditional deep learning models. The LSTM autoencoder model processes transactions in real-time during the initial validation phase. Since only the non-anomalous subset of transactions is sent to the blockchain for validation, the workload of the blockchain network is reduced. Blockchain decentralization also enables the simultaneous update of the LSTM autoencoder model, ensuring accurate and efficient real-time anomaly detection and transaction encoding.

Hierarchical Processing for Large-Scale IoT Deployments

IoT data is first aggregated, processed, and validated by validating nodes before being forwarded to the entire blockchain network. This offloads computational burden and ensures only verified and essential transactions are added to the blockchain. This prevents network congestion, making the system scalable for large IoT deployments.

Dynamic Network Adjustments

As the number of IoT devices grows, our adaptive PBFT consensus mechanism scales dynamically by increasing the number of validator nodes while maintaining efficient transaction processing and using dynamic node rating to prevent consensus slowdowns.

5.6. Cost Analysis

Deploying blockchain technology in real-life smart water systems involves various cost factors that influence feasibility and scalability. DeepChainIoT is designed to optimize performance while addressing potential resource constraints.

Transaction Fees

DeepChainIoT can be deployed on a permissioned blockchain, e.g., using Hyperledger Fabric. This eliminates direct transaction fees.

Storage Requirements

Our LSTM autoencoder-based compression achieves a 23.9x data reduction, minimizing storage demands while preserving critical anomaly detection information.

Energy Consumption

DeepChainIoT employs PBFT consensus, which is more energy-efficient than PoW-based blockchains, making it suitable for IoT applications. The prioritization-based consensus mechanism further enhances efficiency by reducing redundant verifications, lowering overall energy consumption.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

5.7. Comparison With Existing Studies

DeepChainIoT optimally integrates DNN and blockchain into a single platform, allowing both technologies to complement each other, unlike many existing frameworks [50,74,83,92,128] where blockchain and DNN are implemented separately or not efficiently optimized together. Our DNN model detects anomalies in transactions, transaction senders and network traffic, discards anomalous transactions, and encodes non-anomalous transactions to reduce the transaction volume and size on the blockchain network, thus addressing storage management issues faced by other studies [89, 94,98,118]. As the encoded data in the blockchain can only be decoded with specialized decoding algorithms, DeepChainIoT also provides a higher level of privacy than many existing frameworks [49, 118]. Moreover, DeepChainIoT emphasizes the importance of the consensus algorithm, which is often overlooked in other studies [18,76]. By optimizing consensus with DNN, DeepChainIoT prioritizes important transactions and exclude malicious nodes, resulting in an effective node rating mechanism that enhances security and reliability. This is a significant improvement compared to the work of Sapkota et al. [18]. Additionally, our optimized smart contract and consensus mechanisms help reduce overall network latency and increase transaction throughput, addressing the performance issues that many existing works have struggled with [14,76,110]. In [19,20], PBFT consensus optimization with DNNs demonstrates good performance but focuses solely on honest nodes. In contrast, our solution incorporates not only node rating but also transaction prioritization, anomaly detection, and rule-based handling of detected anomalies, which is expected to improve reliability, transaction per second (TPS) performance, and security beyond the approach proposed in [19,20]. Overall, DeepChainIoT is a comprehensive framework that demonstrates enhanced security features and improved performance in securing generic IoT networks.

5.8. Challenges in Implementing DeepChainIoT in Real-Life IoT Environments

While DeepChainIoT offers a promising approach to securing IoT applications using blockchain and DNN, several practical challenges must be considered for real-life deployment.

Blockchain Overhead and Latency

Blockchain networks introduce transaction validation delays and storage overhead, particularly when processing large volumes of IoT data. DeepChainIoT reduces this by rejecting anomalous transactions and compressing non-anomalous transactions before storage. However, additional optimizations, such as pruning techniques, may be necessary for larger-scale deployments.

Scalability in Large IoT Networks

As the number of IoT devices grows, consensus mechanisms like PBFT may face performance bottlenecks. While DeepChainIoT optimizes PBFT using dynamic node selection, real-life implementation may require hybrid consensus mechanisms that balance security and scalability.

Real-Time Anomaly Detection

The effectiveness of anomaly detection depends on model training quality and dataset variations. IoT environments are highly dynamic, and unseen attack patterns could affect detection accuracy. Continuous model retraining and federated learning approaches could enhance adaptability to evolving threats.

Energy Consumption

Blockchain and deep learning models require significant energy resources, which may be impractical for battery-powered IoT devices. Techniques such as energy-efficient DNN models and lightweight cryptographic methods could help reduce power consumption.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

6. Conclusion and Future Work

This paper explores the integration of blockchain and DNNs to develop secure, efficient IoT networks. Our proposed framework–DeepChainIoT–utilizes DNNs to identify malicious transactions, nodes, and network traffic; an optimized PBFT consensus to improve transaction processing efficiency; and blockchain's decentralization and immutability to ensure the integrity of the DNN model and its training data. More specifically, using an LSTM autoencoder-based DNN algorithm, DeepChainIoT tackles challenges such as forking, latency, throughput, and storage limitations through anomaly detection, secure transaction encoding, and optimized consensus. The empirical evaluation on a pump sensor dataset demonstrated the effectiveness of the LSTM autoencoder-based anomaly detection model, with an accuracy of 99.6%, a recall of 100%, a precision of 97.95%, and an F1 score of 98.97%. The autoencoder also achieved significant data compression, reducing the data size by a factor of 23.9, significantly improving storage and transmission efficiency within the network. Additionally, it effectively identified malicious sensors and categorized them based on the number of anomalous transactions sent within a defined transaction window, enabling appropriate actions to be taken. This efficiently demonstrates the application of the LSTM autoencoder in node rating and prioritization within an optimized consensus mechanism. Despite being evaluated on a particular dataset, the architecture of DeepChainIoT can be adapted and applied to other types of IoT networks. The DNN-anomaly detection model can also be redesigned based on the unique characteristics of these IoT systems. Therefore, we conclude that DeepChainIoT can ensure secure, efficient data transmission and strengthens resilience against malicious activities in generic IoT networks. Compared to existing studies, DeepChainIoT demonstrates significant improvements with its ability to reduce forking, improve latency and throughput, authenticate and monitor node behaviors, detect anomalies, and efficiently store and disseminate data while preserving privacy.

Future research will focus on developing and deploying a functional prototype of DeepChainIoT on a small-scale Hyperledger Fabric network, consisting of multiple peer nodes, an orderer node, and smart contracts integrated with an LSTM autoencoder for anomaly detection. Efforts will also be made to incorporate an LSTM-optimized PBFT consensus mechanism. To evaluate system performance and feasibility in resource-constrained IoT environments, we will conduct benchmark tests focusing on key metrics such as transaction throughput, network latency, real-time anomaly detection accuracy, processing time, and resource usage (CPU, memory, and storage).

We will further address the practical challenges of deploying DeepChainIoT in real-life IoT applications by integrating techniques such as Federated Learning to support continual model retraining, thereby improving detection accuracy against evolving threats. Additionally, we plan to optimize deep learning architectures and employ lightweight cryptographic methods to reduce energy consumption on IoT devices. We will also explore off-chain storage strategies, including IPFS and sharding, to further minimize on-chain storage overhead.

Author Contributions: SS contributed to the conceptualization, data curation, formal analysis, investigation, methodology, project administration, resource management, validation, and visualization of the framework and model. SS was also a major contributor to manuscript writing. YH, the corresponding author, contributed to the methodology, supervision, framework validation, and manuscript writing. AG assisted with framework validation and manuscript review. FH contributed to the manuscript review. All authors read and approved the final manuscript.

Data Availability Statement: The dataset analysed during the current study is available in the: N. Phantawee. Pump sensor data. https://www.kaggle.com/datasets/nphantawee/pump-sensor-data

Conflicts of Interest: The authors declare that they have no competing interests.

Abbreviations

The following abbreviations are used in this manuscript:

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

IoTs	Internet of Things
DNNs	Deep Neural Networks
LSTM	Long-Short Term Memory
PBFT	Practical Byzantine Fault Tolerance
DoS	Denial of Service
DDoS	Distributed Denial of Service
SPOF	Single Points of Failures
IoVs	Internet of Vehicles
IIoTs	Industrial Internet of Things
IoMT	Internet of Medical Things
AI	Artificial Intelligence
P2P	Peer to Peer
PoW	Proof of Work
PoS	Proof of Stake
CNNs	Convolutional Neural Networks
RNNs	Recurrent Neural Network
RL	Reinforcement Learning
BiLSTM	Bidirectional Long Short-Term Memory
EPoW	Enhanced Proof of Work
ECC	Elliptic Curve Cryptography
IDS	Intrusion Detection Systems
IBE	Identity Based Encryption
GNN	Graph Neural Networks
ECDH	Elliptic Curve Diffie-Hellman
DRL	Deep Reinforcement Learning
CA	certificate authority

References

1. Shaaban, S.; El Badawy, H.M.; Hashad, A. Performance Evaluation of the IEEE 802.11 Wireless LAN Standards. In *Proceedings of the World Congress on Engineering*, London, UK, 2–4 July 2008; Volume 1.
2. Aldahmani, A.; Ouni, B.; Lestable, T.; Debbah, M. Cyber-security of embedded IoTs in smart homes: Challenges, requirements, countermeasures, and trends. *IEEE Open J. Veh. Technol.* **2023**, *4*, 281–292.
3. Touqeer, H.; Zaman, S.; Amin, R.; Hussain, M.; Al-Turjman, F.; Bilal, M. Smart home security: challenges, issues and solutions at different IoT layers. *J. Supercomput.* **2021**, *77*, 14053–14089.
4. Aldowah, H.; Ul Rehman, S.; Umar, I. Trust in IoT Systems: A Vision on the Current Issues, Challenges, and Recommended Solutions. In *Advances on Smart and Soft Computing: Proceedings of ICACIn 2020*; Springer: Singapore, 2021; pp. 329–339.
5. Cloudflare. Famous DDoS Attacks. 2024. Available online: https://www.cloudflare.com/learning/ddos/famous-ddos-attacks/ (accessed on 9 June 2024).
6. Gelgi, M.; Guan, Y.; Arunachala, S.; Samba Siva Rao, M.; Dragoni, N. Systematic Literature Review of IoT Botnet DDoS Attacks and Evaluation of Detection Techniques. *Sensors* **2024**, *24*, 3571.
7. The Guardian. DDoS attack that disrupted internet was largest of its kind in history, experts say. Available online: https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet (accessed on 30 July 2024).
8. Antonakakis, M.; April, T.; Bailey, M.; Bernhard, M.; Bursztein, E.; Cochran, J.; Durumeric, Z.; Halderman, J.A.; Invernizzi, L.; Kallitsis, M.; et al. Understanding the Mirai Botnet. In *Proceedings of the 26th USENIX Security Symposium (USENIX Security 17)*, Vancouver, BC, Canada, 16–18 August 2017; pp. 1093–1110.
9. Newman, S. Critical RCE Vulnerability Found in Over a Million GPON Home Routers. 2023. Available online: https://www.vpnmentor.com/blog/critical-vulnerability-gpon-router/ (accessed on 21 February 2024).
10. Joel, M.R.; Manikandan, G.; Bhuvaneswari, G. An Analysis of Security Challenges in Internet of Things (IoT) Based Smart Homes. In *Proceedings of the 2023 Second International Conference on Electronics and Renewable Systems (ICEARS)*, Tuticorin, India, 2–4 March 2023; pp. 490–497.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

11. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 3 March 2025).
12. Mangla, M.; Ambarkar, S.; Akhare, R.; Deokar, S.; Mohanty, S.N.; Satpathy, S. A proposed framework to achieve CIA in IoT networks. In Proceedings of the International Conference on Artificial Intelligence and Sustainable Engineering: Select Proceedings of AISE 2020, Volume 2, Jaipur, India, 18–19 December 2020; Springer: Cham, Switzerland, 2022; pp. 19–30.
13. Terumalasetti, S.; Reeja, S.R. A comprehensive study on review of AI techniques to provide security in the digital world. In Proceedings of the 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT), Kannur, India, 11–12 August 2022; IEEE: New York, NY, USA, 2022; pp. 407–416.
14. Hannah, S.; Deepa, A.J.; Chooralil, V.S.; BrillySangeetha, S.; Yuvaraj, N.; Raja, R.A.; Suresh, C.; Vignesh, R.; Srihari, K.; Alene, A.; et al. Blockchain-based deep learning to process IoT data acquisition in cognitive data. *BioMed Research International* **2022**, *2022*, 1–13.
15. Menon, S.; Anand, D.; Kavita; Verma, S.; Kaur, M.; Jhanjhi, N.Z.; Ghoniem, R.M.; Ray, S.K. Blockchain and machine learning inspired secure smart home communication network. *Sensors* **2023**, *23*, 6132.
16. Zhang, Z.; Song, X.; Liu, L.; Yin, J.; Wang, Y.; Lan, D. Recent advances in blockchain and artificial intelligence integration: Feasibility analysis, research issues, applications, challenges, and future work. *Security and Communication Networks* **2021**, *2021*, 1–15.
17. Shafay, M.; Ahmad, R.W.; Salah, K.; Yaqoob, I.; Jayaraman, R.; Omar, M. Blockchain for deep learning: review and open challenges. *Cluster Computing* **2023**, *26*, 197–221.
18. Sapkota, S.; Huang, H.; Hu, Y.; Hussain, F. A Deep Neural Network (DNN) Based Contract Policy on Hyperledger Fabric for Secure Internet of Things (IoTs). In Proceedings of the International Conference on Advanced Information Networking and Applications, Springer, 2024; pp. 313–325.
19. Goh, Y.; Yun, J.; Jung, D.; Chung, J.M. Secure Trust-Based Delegated Consensus for Blockchain Frameworks Using Deep Reinforcement Learning. *IEEE Access* **2022**, *10*, 118498–118511.
20. Papaioannou, D.; Mygdalis, V.; Pitas, I. Proof of Quality Inference (PoQI): An AI Consensus Protocol for Decentralized DNN Inference Frameworks. In *Proceedings of the 2024 IEEE Symposium on Computers and Communications (ISCC)*, Paris, France, 26–29 June 2024; pp. 1–7.
21. Bennet, D.; Maria, L.; Putri Ayu Sanjaya, Y.; Rahmania Az Zahra, A. Blockchain technology: revolutionizing transactions in the digital age. *ADI Journal on Recent Innovation* **2024**, *5*, 194–199.
22. Buterin, V.; et al. A next-generation smart contract and decentralized application platform. *White Paper* **2014**, *3*, 2–1.
23. Padmavathi, M.; Suresh, R.M. Secure P2P intelligent network transaction using litecoin. *Mob. Netw. Appl.* **2019**, *24*, 318–326.
24. Nani, A. The doge worth 88 billion dollars: A case study of Dogecoin. *Convergence* **2022**, *28*, 1719–1736.
25. Heo, H.; Woo, S.; Yoon, T.; Kang, M.S.; Shin, S. Partitioning Ethereum without Eclipsing It. In *NDSS*, 2023.
26. Ethereum development community. Ethereum wire protocol (eth). Available online: https://github.com/ethereum/devp2p/blob/master/caps/eth.md (accessed on 9 June 2024).
27. Nakaike, T.; Zhang, Q.; Ueda, Y.; Inagaki, T.; Ohara, M. Hyperledger Fabric performance characterization and optimization using goleveldb benchmark. In Proceedings of the 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Toronto, ON, Canada, 3–6 May 2020; IEEE: New York, NY, USA, 2020; pp. 1–9.
28. De Filippi, P.; Wray, C.; Sileno, G. Smart contracts. *Internet Policy Rev.* **2021**, *10*, 2.
29. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; De Caro, A.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger Fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference, Porto, Portugal, 23–26 April 2018; ACM: New York, NY, USA, 2018; pp. 1–15.
30. Saini, K.; Sharma, S.; Sarkar, U. Blockchain and cryptography. In Proceedings of the 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 16–17 December 2022; IEEE: New York, NY, USA, 2022; pp. 1863–1868.
31. Shifferaw, Y.; Lemma, S. Limitations of Proof of Stake Algorithm in Blockchain: A Review. *Zede J.* **2021**, *39*, 81–95.
32. Castro, M.; Liskov, B.; et al. Practical Byzantine fault tolerance. In Proceedings of the 3rd Symposium on Operating Systems Design and Implementation (OSDI), New Orleans, LA, USA, 22–25 February 1999; USENIX: Berkeley, CA, USA, 1999; pp. 173–186.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

33. Nandanwar, H.; Katarya, R. A systematic literature review: approach toward blockchain future research trends. In Proceedings of the 2023 International Conference on Device Intelligence, Computing and Communication Technologies (DICCT), Bhopal, India, 24–25 February 2023; IEEE: New York, NY, USA, 2023; pp. 259–264.
34. Jin, A.; Ye, Y.; Lee, B.; Qiao, Y. Topology analysis of the Ripple transaction network. *International Journal of Network Management* **2024**, *34*, e2253.
35. Tripathi, G.; Ahad, M.A.; Casalino, G. A comprehensive review of blockchain technology: underlying principles and historical background with future challenges. *Decision Analytics Journal* **2023**, 100344.
36. Bau, D.; Zhu, J.Y.; Strobelt, H.; Lapedriza, A.; Zhou, B.; Torralba, A. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences* **2020**, *117*, 30071–30078.
37. Liu, C.; Zhu, L.; Belkin, M. Toward a theory of optimization for over-parameterized systems of non-linear equations: the lessons of deep learning. *arXiv* **2020**, arXiv:2003.00307.
38. Montavon, G.; Samek, W.; Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digit. Signal Process.* **2018**, *73*, 1–15.
39. Silva, F.M.; Almeida, L.B. Acceleration techniques for the backpropagation algorithm. In *European Association for Signal Processing Workshop*; Springer: 1990; pp. 110–119.
40. Puig-Arnavat, M.; Bruno, J.C. Chapter 5 - Artificial Neural Networks for Thermochemical Conversion of Biomass. In *Recent Advances in Thermo-Chemical Conversion of Biomass*; Pandey, A., Bhaskar, T., Stöcker, M., Sukumaran, R.K., Eds.; Elsevier: Boston, 2015; pp. 133–156.
41. Yu, T.; Zhu, H. Hyper-parameter optimization: A review of algorithms and applications. *arXiv* **2020**, arXiv:2003.05689.
42. Bodiwala, S.; Nanavati, N. Efficient hardware implementations of deep neural networks: A survey. In Proceedings of the 2020 Fourth International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 8–10 January 2020; IEEE: New York, NY, USA, 2020; pp. 31–36.
43. Sakib, S.; Ahmed, N.; Kabir, A.J.; Ahmed, H. An overview of convolutional neural network: Its architecture and applications. *Preprints* **2019**.
44. Doya, K. Supervised learning in recurrent networks. In *Handbook of Brain Theory and Neural Networks*; MIT Press: 1995; pp. 796–800.
45. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480.
46. Yang, Z.; Xu, B.; Luo, W.; Chen, F. Autoencoder-based representation learning and its application in intelligent fault diagnosis: A review. *Measurement* **2022**, *189*, 110460.
47. Huang, Q. Model-based or model-free, a review of approaches in reinforcement learning. In *2020 International Conference on Computing and Data Science (CDS)*; IEEE: 2020; pp. 219–221.
48. Rokonuzzaman, M.; Akash, M.I.; Mishu, M.K.; Tan, W.-S.; Hannan, M.A.; Amin, N. IoT-based distribution and control system for smart home applications. In *2022 IEEE 12th Symposium on Computer Applications & Industrial Electronics (ISCAIE)*; IEEE: 2022; pp. 95–98.
49. Mishra, S.; Chaurasiya, V.K. Blockchain and IoT based infrastructure for secure smart city using deep learning algorithm with Dingo optimization. *Wireless Personal Communications* **2023**, *132*, 17–37.
50. Kumar, P.; Kumar, R.; Gupta, G.P.; Tripathi, R.; Jolfaei, A.; Islam, A.N. A blockchain-orchestrated deep learning approach for secure data transmission in IoT-enabled healthcare system. *Journal of Parallel and Distributed Computing* **2023**, *172*, 69–83.
51. Jameel, F.; Nabeel, M.; Jamshed, M.A.; Jäntti, R. Minimizing forking in blockchain-based IoT networks. In Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 7–11 June 2020; IEEE: New York, NY, USA, 2020; pp. 1–6.
52. Huang, J.; Tan, L.; Mao, S.; Yu, K. Blockchain network propagation mechanism based on P4P architecture. *Security and Communication Networks* **2021**, *2021*, 1–12.
53. Liu, Q.; Xu, Y.; Cao, B.; Zhang, L.; Peng, M. Unintentional forking analysis in wireless blockchain networks. *Digital Communications and Networks* **2021**, *7*, 335–341.
54. Tang, W.; Kiffer, L.; Fanti, G.; Juels, A. Strategic latency reduction in blockchain peer-to-peer networks. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* **2023**, *7*, 1–33.
55. Si, H.; Niu, B. Research on blockchain data availability and storage scalability. *Future Internet* **2023**, *15*, 212.
56. Vaigandla, K.K.; Karne, R.; Siluveru, M.; Kesoju, M. Review on blockchain technology: architecture, characteristics, benefits, algorithms, challenges and applications. *Mesopotamian Journal of CyberSecurity* **2023**, *2023*, 73–84.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

57. Werth, J.; Berenjestanaki, M.H.; Barzegar, H.R.; El Ioini, N.; Pahl, C. A review of blockchain platforms based on the scalability, security and decentralization trilemma. *ICEIS (1)* **2023**, 146–155.
58. Gracy, M.; Jeyavadhanam, B.R. A systematic review of blockchain-based system: Transaction throughput latency and challenges. In Proceedings of the 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA), Hyderabad, India, 29–30 January 2021; IEEE: New York, NY, USA, 2021; pp. 1–6.
59. Jabbar, S.; Abideen, Z.U.; Khalid, S.; Ahmad, A.; Raza, U.; Akram, S. Enhancing computational scalability in blockchain by leveraging improvement in consensus algorithm. *Frontiers in Computer Science* **2023**, *5*.
60. Sayeed, S.; Marco-Gisbert, H. Assessing blockchain consensus and security mechanisms against the 51% attack. *Applied Sciences* **2019**, *9*, 1788.
61. Aponte-Novoa, F.A.; Sandoval, A.L.; Villanueva-Polanco, R.; Wightman, P. The 51% attack on blockchains: A mining behavior study. *IEEE Access* **2021**, *9*, 140549–140564.
62. Douceur, J.R. The Sybil Attack. In *Proceedings of the International Workshop on Peer-to-Peer Systems*, Cambridge, MA, USA, 7–8 March 2002; pp. 251–260.
63. Platt, M.; Bandara, R.J.; Drăgnoiu, A.E.; Krishnamoorthy, S. Information privacy in decentralized applications. *Trust Models for Next-Generation Blockchain Ecosystems* **2021**, 85–104.
64. Bader, L.; Pennekamp, J.; Matzutt, R.; Hedderich, D.; Kowalski, M.; Lücken, V.; Wehrle, K. Blockchain-based privacy preservation for supply chains supporting lightweight multi-hop information accountability. *Information Processing & Management* **2021**, *58*, 102529.
65. Silviu, O. An overview of security issues in smart contracts on the blockchain. In Proceedings of the 21st International Conference on Informatics in Economy (IE 2022), Bucharest, Romania, 2–3 June 2022; Springer: Cham, Switzerland, 2023; pp. 51–63.
66. Wang, Y.; He, J.; Zhu, N.; Yi, Y.; Zhang, Q.; Song, H.; Xue, R. Security enhancement technologies for smart contracts in the blockchain: A survey. *Transactions on Emerging Telecommunications Technologies* **2021**, *32*, e4341.
67. Sanghami, S.V.; Lee, J.J.; Hu, Q. Machine-Learning-Enhanced Blockchain Consensus with Transaction Prioritization for Smart Cities. *IEEE Internet Things J.* **2022**, *10*, 6661–6672.
68. Hojjati, M.; Shafieinejad, A.; Yanikomeroglu, H. A Blockchain-Based Authentication and Key Agreement (AKA) Protocol for 5G Networks. *IEEE Access* **2020**, *8*, 216461–216476.
69. Khalid, M.I.; Ehsan, I.; Al-Ani, A.K.; Iqbal, J.; Hussain, S.; Ullah, S.S.; et al. A comprehensive survey on blockchain-based decentralized storage networks. *IEEE Access* **2023**, *11*, 10995–11015.
70. Alizadeh, M.; Andersson, K.; Schelén, O. Efficient decentralized data storage based on public blockchain and IPFS. In Proceedings of the 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Nadi, Fiji, 14–16 December 2020; IEEE: New York, NY, USA, 2020; pp. 1–8.
71. Talaei Khoei, T.; Ould Slimane, H.; Kaabouch, N. Deep Learning: Systematic Review, Models, Challenges, and Research Directions. *Neural Comput. Appl.* **2023**, *35*, 23103–23124.
72. Wang, J.; Zhang, Z.; Wang, M.; Qiu, H.; Zhang, T.; Li, Q.; Li, Z.; Wei, T.; Zhang, C. Aegis: Mitigating targeted bit-flip attacks against deep neural networks. In Proceedings of the 32nd USENIX Security Symposium (USENIX Security 23), Anaheim, CA, USA, 9–11 August 2023; USENIX Association: Berkeley, CA, USA, 2023; pp. 2329–2346.
73. Dong, T.; Zhang, Z.; Qiu, H.; Zhang, T.; Li, H.; Wang, T. Mind your heart: Stealthy backdoor attack on dynamic deep neural network in edge computing. In Proceedings of the IEEE INFOCOM 2023, New York, NY, USA, 17–20 May 2023; IEEE: New York, NY, USA, 2023; pp. 1–10.
74. Wang, J.; Jin, H.; Chen, J.; Tan, J.; Zhong, K. Anomaly detection in Internet of Medical Things with blockchain from the perspective of deep neural network. *Information Sciences* **2022**, *617*, 133–149.
75. Aljbali, S.; Roy, K. Anomaly detection using bidirectional LSTM. In Proceedings of the Intelligent Systems and Applications: IntelliSys 2020, Amsterdam, The Netherlands, 3–4 September 2020; Volume 1, pp. 612–619.
76. Shobana, M.; Shanmuganathan, C.; Challa, N.P.; Ramya, S. An optimized hybrid deep neural network architecture for intrusion detection in real-time IoT networks. *Transactions on Emerging Telecommunications Technologies* **2022**, *33*, e4609.
77. Tong, C.; Lan, T.; Shi, X. Double-layer ensemble monitoring of non-gaussian processes using modified independent component analysis. *ISA Trans.* **2017**, *68*, 181–188.
78. He, Y.; Wang, X. Group theory-based optimization algorithm for solving knapsack problems. *Knowl.-Based Syst.* **2021**, *219*, 104445.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

79. Dehghani, M.; Montazeri, Z.; Dehghani, A.; Malik, O.P.; Morales-Menendez, R.; Dhiman, G.; Nouri, N.; Ehsanifar, A.; Guerrero, J.M.; Ramirez-Mendoza, R.A. Binary spring search algorithm for solving various optimization problems. *Appl. Sci.* **2021**, *11*, 1286.
80. Sengupta, S.; Basak, S.; Peters, R.A. Particle Swarm Optimization: A Survey of Historical and Recent Developments with Hybridization Perspectives. *Mach. Learn. Knowl. Extract.* **2019**, *1*, 157–191.
81. University of New South Wales. Toniot Datasets. Available online: https://research.unsw.edu.au/projects/toniot-datasets (accessed on 23 June 2024).
82. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset. *Future Gener. Comput. Syst.* **2019**, *100*, 779–796.
83. Qi, P.; Chiaro, D.; Giampaolo, F.; Piccialli, F. A Blockchain-Based Secure Internet of Medical Things Framework for Stress Detection. *Inf. Sci.* **2023**, *628*, 377–390.
84. Hu, X.; Liu, T.; Hao, X.; Lin, C. Attention-Based Conv-LSTM and Bi-LSTM Networks for Large-Scale Traffic Speed Prediction. *J. Supercomput.* **2022**, *78*, 12686–12709.
85. Koblitz, N.; Menezes, A.; Vanstone, S. The State of Elliptic Curve Cryptography. *Des. Codes Cryptogr.* **2000**, *19*, 173–193.
86. Schmidt, P.; Reiss, A.; Duerichen, R.; Marberger, C.; Van Laerhoven, K. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, Boulder, CO, USA, 16–20 October 2018; pp. 400–408.
87. Pimple, J.F.; Sharma, A.; Mishra, J.K. Elevating Security Measures in Cyber-Physical Systems: Deep Neural Network-Based Anomaly Detection with Ethereum Blockchain for Enhanced Data Integrity. *J. Electr. Syst.* **2023**, *19*, 2.
88. Verma, A.; Ranga, V. On Evaluation of Network Intrusion Detection Systems: Statistical Analysis of CIDDS-001 Dataset Using Machine Learning Techniques. *Authorea Preprints* **2023**.
89. Demertzis, K.; Iliadis, L.; Tziritas, N.; Kikiras, P. Anomaly Detection via Blockchained Deep Learning Smart Contracts in Industry 4.0. *Neural Comput. Appl.* **2020**, *32*, 17361–17378.
90. Saravanan, V.; Madiajagan, M.; Rafee, S.M.; Sanju, P.; Rehman, T.B.; Pattanaik, B. IoT-Based Blockchain Intrusion Detection Using Optimized Recurrent Neural Network. *Multimed. Tools Appl.* **2024**, *83*, 31505–31526.
91. Boneh, D.; Franklin, M. Identity-Based Encryption from the Weil Pairing. In *Proceedings of the Annual International Cryptology Conference*, Santa Barbara, CA, USA, 19–23 August 2001; pp. 213–229.
92. 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.
93. Almomani, O.; Almaiah, M.A.; Alsaaidah, A.; Smadi, S.; Mohammad, A.H.; Althunibat, A. Machine learning classifiers for network intrusion detection system: comparative study. In Proceedings of the 2021 International Conference on Information Technology (ICIT), Amman, Jordan, 14–15 July 2021; pp. 440–445.
94. Dixit, A.; Trivedi, A.; Godfrey, W.W. IoT and machine learning based peer-to-peer framework for employee attendance system using blockchain. In Proceedings of the 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Greater Noida, India, 22–23 December 2022; IEEE: New York, NY, USA, 2022; pp. 1088–1093.
95. Pahlevi, R.; Setiaji, B. Analysis of Application Haar Cascade Classifier and Local Binary Pattern Histogram Algorithm in Recognizing Faces with Real-Time Grayscale Images Using OpenCV. *J. Tek. Inform. (Jutif)* **2023**, *4*, 179–186.
96. Bhardwaj, S.; Harit, S.; Shilpa; Anand, D. Message queuing telemetry transport-secure connection: a power-efficient secure communication. *Int. J. Sens. Netw.* **2023**, *42*, 29–40.
97. Rijmen, V.; Daemen, J. Advanced encryption standard. *Proc. Fed. Inf. Process. Stand. Publ. Natl. Inst. Stand. Technol.* **2001**, *19*, 22.
98. Albakri, A.; Alqahtani, Y.M. Internet of Medical Things with a blockchain-assisted smart healthcare system using metaheuristics with a deep learning model. *Appl. Sci.* **2023**, *13*, 6108.
99. Hussain, M.; Iqbal, N.; Bashir, Z. A chaotic image encryption scheme based on multi-directional confusion and diffusion operations. *J. Inf. Secur. Appl.* **2022**, *70*, 103347.
100. Kushwah, G.S.; Ranga, V. Voting Extreme Learning Machine Based Distributed Denial of Service Attack Detection in Cloud Computing. *J. Inf. Secur. Appl.* **2020**, *53*, 102532.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

101. Alqaralleh, B.A.Y.; Vaiyapuri, T.; Parvathy, V.S.; Gupta, D.; Khanna, A.; Shankar, K. Blockchain-Assisted Secure Image Transmission and Diagnosis Model on Internet of Medical Things Environment. *Pers. Ubiquitous Comput.* **2021**, *1*, 1–11.
102. Yacin Sikkandar, M.; Alrasheadi, B.A.; Prakash, N.B.; Hemalakshmi, G.R.; Mohanarathinam, A.; Shankar, K. Deep Learning Based an Automated Skin Lesion Segmentation and Intelligent Classification Model. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 3245–3255.
103. Rathore, S.; Park, J.H.; Chang, H. Deep Learning and Blockchain-Empowered Security Framework for Intelligent 5G-Enabled IoT. *IEEE Access* **2021**, *9*, 90075–90083.
104. Zhang, Y.; Cui, L.; Wang, W.; Zhang, Y. A Survey on Software Defined Networking with Multiple Controllers. *J. Netw. Comput. Appl.* **2018**, *103*, 101–118.
105. Cai, J.; Liang, W.; Li, X.; Li, K.; Gui, Z.; Khan, K. GTxChain: A Secure IoT Smart Blockchain Architecture Based on Graph Neural Network. *IEEE Internet Things J.* **2023**, *PP*, 1–1.
106. Moorthy, S.K.; Jagannath, J. Survey of Graph Neural Network for Internet of Things and NextG Networks. *arXiv* **2024**, arXiv:2405.17309.
107. Digitale, J.C.; Martin, J.N.; Glymour, M.M. Tutorial on Directed Acyclic Graphs. *J. Clin. Epidemiol.* **2022**, *142*, 264–267.
108. Revanesh, M.; Acken, J.M.; Sridhar, V. DAG Block: Trust Aware Load Balanced Routing and Lightweight Authentication Encryption in WSN. *Future Gener. Comput. Syst.* **2023**, *140*, 402–421.
109. Hua, Y.; Ding, L.; Chen, Z.; Wang, J.; Zhu, Z. Blockchain Construction and Query Method for Spatio-Temporal Data. *J. Comput. Appl.* **2022**, *42*, 3429.
110. Agrawal, R.; Singhal, S.; Sharma, A. Blockchain and Fog Computing Model for Secure Data Access Control Mechanisms for Distributed Data Storage and Authentication Using Hybrid Encryption Algorithm. *Clust. Comput.* **2024**, *1*, 1–16.
111. Ejim, S.; Gital, A.Y.; Chiroma, H.; Lawal, M.A.; Abubakar, M.Y.; Kubi, G.M. Data Encryption in Fog Computing Using Hybrid Cryptography with Integrity Check. In *Soft Computing for Problem Solving: Proceedings of the SocProS 2022*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 627–638.
112. Yu, S.; Mao, X.L.; Wei, W.; Huang, H. Unsupervised Deep Hashing via Adaptive Clustering. In *Proceedings of the Web and Big Data: 5th International Joint Conference, APWeb-WAIM 2021, Guangzhou, China, 23–25 August 2021*; Part II 5, Springer: Cham, Switzerland, 2021; pp. 3–17.
113. Fiege, U.; Fiat, A.; Shamir, A. Zero knowledge proofs of identity. In Proceedings of the 19th Annual ACM Symposium on Theory of Computing, New York, NY, USA, 25–27 May 1987; pp. 210–217.
114. Kaur, M.; Gupta, S.; Kumar, D.; Raboaca, M.S.; Goyal, S.B.; Verma, C. IPFS: An off-chain storage solution for blockchain. In Proceedings of the International Conference on Recent Innovations in Computing: ICRIC 2022, Volume 1, Jammu, India, 10–11 March 2022; pp. 513–525.
115. Lee, J.; Pak, J.G.; Lee, M. Network intrusion detection system using feature extraction based on deep sparse autoencoder. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 21–23 October 2020; pp. 1282–1287.
116. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp* **2018**, *1*, 108–116.
117. Sharafaldin, I. CIC-IDS2017 Datasets. 2017. Available online: http://205.174.165.80/CICDataset/CIC-IDS-2017/Dataset/ (accessed on 24 June 2024).
118. Połap, D.; Srivastava, G.; Jolfaei, A.; Parizi, R.M. Blockchain Technology and Neural Networks for the Internet of Medical Things. In *Proceedings of the IEEE INFOCOM 2020—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Toronto, ON, Canada, 6–9 July 2020; pp. 508–513.
119. Zhang, C.; Xie, Y.; Bai, H.; Yu, B.; Li, W.; Gao, Y. A Survey on Federated Learning. *Knowl.-Based Syst.* **2021**, *216*, 106775.
120. Jaeger, S.; Candemir, S.; Antani, S.; Wáng, Y.X.J.; Lu, P.X.; Thoma, G. Two Public Chest X-ray Datasets for Computer-Aided Screening of Pulmonary Diseases. *Quant. Imaging Med. Surg.* **2014**, *4*, 475.
121. Kumar, R.; Kumar, P.; Tripathi, R.; Gupta, G.P.; Islam, A.N.; Shorfuzzaman, M. Permissioned Blockchain and Deep Learning for Secure and Efficient Data Sharing in Industrial Healthcare Systems. *IEEE Trans. Ind. Inform.* **2022**, *18*, 8065–8073.
122. Bi, J.; Guan, Z.; Yuan, H.; Yang, J.; Zhang, J. Network Anomaly Detection with Stacked Sparse Shrink Variational Autoencoders and Unbalanced XGBoost. *IEEE Trans. Sustain. Comput.* **2024**.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

123. Jian, W.; Li, J.P.; Akbar, A.; Haq, A.U.; Khan, S.; Alotaibi, R.M.; Alajlan, S.A. SA-Bi-LSTM: Self Attention With Bi-Directional LSTM Based Intelligent Model for Accurate Fake News Detection to Ensure Information Integrity on Social Media Platforms. *IEEE Access* **2024**, *12*, in press.
124. Kumar, C.; Chittora, P. Deep-Learning and Blockchain-Empowered Secure Data Sharing for Smart Grid Infrastructure. *Arab. J. Sci. Eng.* **2024**, 1–14.
125. Doersch, C. Tutorial on Variational Autoencoders. *arXiv* **2016**, arXiv:1606.05908.
126. Rathod, T.; Jadav, N.K.; Tanwar, S.; Polkowski, Z.; Yamsani, N.; Sharma, R.; Alqahtani, F.; Gafar, A. AI and Blockchain-Based Secure Data Dissemination Architecture for IoT-Enabled Critical Infrastructure. *Sensors* **2023**, *23*, 8928.
127. Mankodiya, H.; Jadav, D.; Gupta, R.; Tanwar, S.; Alharbi, A.; Tolba, A.; Neagu, B.C.; Raboaca, M.S. XAI-FALL: Explainable AI for fall detection on wearable devices using sequence models and XAI techniques. *Mathematics* **2022**, *10*, 1990.
128. Malik, V.; Mittal, R.; Mavaluru, D.; Narapureddy, B.R.; Goyal, S.B.; Martin, R.J.; Srinivasan, K.; Mittal, A. Building a Secure Platform for Digital Governance Interoperability and Data Exchange Using Blockchain and Deep Learning-Based Frameworks. *IEEE Access* **2023**.
129. Das, A.K.; Pratihar, D.K. A New Bonobo Optimizer (BO) for Real-Parameter Optimization. In *Proceedings of the 2019 IEEE Region 10 Symposium (TENSYMP)*, Kolkata, India, 7–9 June 2019; pp. 108–113.
130. Ong, K.S.H.; Wang, W.; Niyato, D.; Friedrichs, T. Deep-reinforcement-learning-based predictive maintenance model for effective resource management in industrial IoT. *IEEE Internet Things J.* **2021**, *9*, 5173–5188.
131. Saleh, A.M.S. Blockchain for secure and decentralized artificial intelligence in cybersecurity: A comprehensive review. *Blockchain: Res. Appl.* **2024**, 100193.
132. Zhu, J.; Cao, J.; Saxena, D.; Jiang, S.; Ferradi, H. Blockchain-empowered federated learning: Challenges, solutions, and future directions. *ACM Comput. Surv.* **2023**, *55*, 1–31.
133. Frikha, T.; Ktari, J.; Zalila, B.; Ghorbel, O.; Ben Amor, N. Integrating blockchain and deep learning for intelligent greenhouse control and traceability. *Alexandria Eng. J.* **2023**, *79*, 259–273.
134. Luz, A.; Olaoye, O.J.G. Secure Multi-Party Computation (MPC): Privacy-preserving protocols enabling collaborative computation without revealing individual inputs, ensuring AI privacy. 2024.
135. Ghanem, S.M.; Moursy, I.A. Secure multiparty computation via homomorphic encryption library. In Proceedings of the 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), 2019; pp. 227–232.
136. Signorini, M.; Pontecorvi, M.; Kanoun, W.; Di Pietro, R. BAD: A blockchain anomaly detection solution. *IEEE Access* **2020**, *8*, 173481–173490.
137. Li, B.; Chenli, C.; Xu, X.; Shi, Y.; Jung, T. DLBC: A Deep Learning-Based Consensus in Blockchains for Deep Learning Services. *arXiv* **2020**, arXiv:1904.07349.
138. Yang, H.-F.; Lin, K.; Chen, C.-S. Supervised learning of semantics-preserving hash via deep convolutional neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 437–451.
139. Protogerou, A.; Papadopoulos, S.; Drosou, A.; et al. A graph neural network method for distributed anomaly detection in IoT. *Evolving Syst.* **2021**, *12*, 19–36.
140. Mao, H.; Alizadeh, M.; Menache, I.; Kandula, S. Resource management with deep reinforcement learning. In Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 2016; pp. 50–56.
141. Hisham, S.; Makhtar, M.; Aziz, A.A. Anomaly detection in smart contracts based on optimal relevance hybrid features analysis in the Ethereum blockchain employing ensemble learning. *Transactions* **2023**, *3*, 5.
142. Venkatesan, K.; Rahayu, S.B. Blockchain security enhancement: an approach towards hybrid consensus algorithms and machine learning techniques. *Sci. Rep.* **2024**, *14*, 1149.
143. Scicchitano, F.; Liguori, A.; Guarascio, M.; Ritacco, E.; Manco, G.; et al. A deep learning approach for detecting security attacks on blockchain. In *CEUR Workshop Proceedings*, vol. 2597; CEUR-WS: 2020; pp. 212–222.
144. Jiang, X.; Shi, Q.; Miao, H.; Cao, W.; He, H.; Chen, S.; Yang, J. Credible Link Flooding Attack Detection and Mitigation: A Blockchain-Based Approach. *IEEE Trans. Netw. Serv. Manag.* **2024**.
145. Haouari, W.; Hafid, A.S.; Fokaefs, M. Vulnerabilities of smart contracts and mitigation schemes: A Comprehensive Survey. *arXiv* **2024**, arXiv:2403.19805.
146. Sendner, C.; Chen, H.; Fereidooni, H.; Petzi, L.; König, J.; Stang, J.; Dmitrienko, A.; Sadeghi, A.-R.; Koushanfar, F. Smarter Contracts: Detecting Vulnerabilities in Smart Contracts with Deep Transfer Learning. To appear at the Network and Distributed System Security Symposium (NDSS), 2023.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 July 2025 doi:10.20944/preprints202507.0645.v1

147. Tang, X.; Du, Y.; Lai, A.; Zhang, Z.; Shi, L. Deep learning-based solution for smart contract vulnerabilities detection. *Sci. Rep.* **2023**, *13*, 20106.
148. Jiang, F.; Chao, K.; Xiao, J.; Liu, Q.; Gu, K.; Wu, J.; Cao, Y. Enhancing Smart-Contract Security through Machine Learning: A Survey of Approaches and Techniques. *Electronics* **2023**, *12*, 2046.
149. Chen, Z.; Xiong, X.; Wang, W.; Xiao, Y.; Alfarraj, O. A Blockchain-Based Multi-Unmanned Aerial Vehicle Task Processing System for Situation Awareness and Real-Time Decision. *Sustainability* **2023**, *15*, 13790.
150. Maskey, S.R.; Badsha, S.; Sengupta, S.; Khalil, I. Reputation-based miner node selection in blockchain-based vehicular edge computing. *IEEE Consum. Electron. Mag.* **2020**, *10*, 14–22.
151. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
152. Annane, B.; Alti, A.; Lakehal, A. Blockchain-Based Context-Aware CP-ABE Schema for Internet of Medical Things Security. *Array* **2022**, *14*, 100150.
153. Phantawee, N. Pump Sensor Data. 2018. Available online: https://www.kaggle.com/datasets/nphantawee/pump-sensor-data (accessed on 25 April 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.