Article

An Optimised and Scalable Blockchain-Based Distributed Learning Platform for Consumer IoT

Zhaocheng Wang , Xueying Liu * , Xinming Shao , Abdullah Alghamdi , And Mesfer Alrizq , Md Shirajum Munir , Sujit Biswas *

Posted Date: 6 November 2023

doi: 10.20944/preprints202311.0273.v1

Keywords: Blockchain; IoT; Security and Privacy; Smart Home; Distributed Ledger Technology

Article

An Optimised and Scalable Blockchain-Based Distributed Learning Platform for Consumer IoT

Zhaocheng Wang [1,2], Xueying Liu [3,*], Xinming Shao [4], Abdullah Alghamdi [5], Mesfer Alrizq [5], Md. Shirajum Munir [6] and Sujit Biswas [7,*]

[1] School of Cyber Science and Engineering, Zhengzhou University, Zhengzhou 450001, China
[2] School of Economics of Sichuan University, Sichuan, China
[3] Cabin Attendant College, Civil Aviation University of China, Tianjin 300300, China
[4] Computer Science and Technology Department, Zhengzhou Railway Vocational and Technical College, Henan 451460, China
[5] Information Systems Department, College of Computer Science and Information Systems, Najran University, Najran, Saudi Arabia
[6] School of Cybersecurity, Old Dominion University, Norfolk, Virginia, USA
[7] Computer Science and Digital Technologies Department, University of East London, University Way, London, UK
* Correspondence: xueying_liu2000@outlook.com (X.L.); sujitbiswas@ieee.org (S.B.)

Abstract: Consumer IoT (CIoT) manufacturers seek customer feedback to enhance their products and services, creating a smart ecosystem like a smart home. Due to security and privacy concerns, Blockchain-based federated learning (BCFL) ecosystems can let CIoT manufacturers update their Machine Learning (ML) model using end-user data. FL uses privacy-preserving ML techniques to forecast customers' needs and consumption habits, and blockchain replaces the centralised aggregator to safeguard the ecosystem. However, Blockchain technology (BCT) struggles with scalability and quick ledger expansion. In BCFL, local model generation and secure aggregation are other issues. This research contributes a novel architecture emphasising Gateway Peer (GWP) in blockchain network to resolve scalability, ledger optimisation and secure model transmission issues. In the architecture we replace the centralised aggregator by the blockchain network, while GWP restricts the number of local transactions to execute in BCN. Considering the security and privacy of FL processes, we have added differential privacy and advanced normalisation techniques to ML processes. The approaches strengthen end-users' cyber security and encourage the adoption of technological innovation standards by service providers. The proposed approach has been tested extensively using a well-respected Stanford Cars dataset. We experimentally demonstrate that the proposed architecture makes the network scalable and optimises the ledger significantly. In addition, the normalisation technique outperforms batch normalisation when features are under DP protection.

Keywords: blockchain; IoT; security and privacy; smart home; distributed ledger technology

1. Introduction

Consumer Internet of Things (CIoT) devices are intended to make our lives more convenient, efficient, and connected. CIoT refers to the network of interconnected physical devices, wearables, appliances, and other objects embedded with sensors, software, and connectivity, enabling them to collect and exchange data from end-users via the internet. CIoT plays a vital role in facilitating smart home (SH) functionality, where residents use wearable CIoT and appliances that provide smart services and impact the lives of end-users. The CIoTs are interconnected in a SH ecosystem, allowing householders to monitor and control them via a central hub or smartphone app. This integration offers occupants convenience, energy savings, enhanced security, and an overall improvement in quality of life. Every day, the capabilities of these technologies advance and expand. In the near future, it is anticipated that this trend will surpass all extant market demand data [1].

According to forecasts in [2] , the number of SHs will reach 672.57 million globally by 2027, and their penetration rate will rise to 86.47 percent. Meanwhile, total revenue is expected to grow from 83 billion USD in 2023 to 86 billion USD in 2027, a gain of 60%. In 2027, sales are expected to increase and reach 222.9 billion USD [2]. Wearable CIoT refers to gadgets worn by home users and used to process private data, while home appliances (HA) ensure smart services by connecting with each other through automation. Data from these devices generates massive amounts of information with a wide range of characteristics, including user emotions, actions, and satisfaction, which may be used for real-time intelligent analysis, service demand analysis, and forecasting. Devices typically employ a service-specific centralised server, such as a cloud or edge server, and are managed either autonomously or in tandem by third-party service providers [3]. When it comes to analysing SH user data for insights into customer service expectations, future market analyses, etc., service providers frequently turn to machine learning (ML) and statistical methods. They also store information and run the system mostly from a central server.

1.1. *Typical Intelligent System's Challenges*

The implementation of a dedicated, autonomous, and secure control and management system for SH is associated with significant costs, making it an impractical solution for future projections. In addition, a multitude of novel IoT devices are consistently being introduced to provide ubiquitous services. These devices exhibit significant diversity in terms of data kinds, message structure, and other relevant characteristics [4]. Managing devices that are plug-and-play with a centralised server supported by a relational database may present challenges that are not easily resolved. However, the majority of SHs employ the services of third-party providers that commonly utilise a centralised control system [3]. The implementation of a comprehensive system utilising centralised servers has significant challenges, including but not limited to access control, the presence of a single point of failure and vulnerability, data security concerns, and the management of large volumes of data [5]. The inclusion of extra obstacles complicates the task of guaranteeing the appropriate utilisation of data for subsequent analysis. Several recent studies have proposed the utilisation of Machine Learning (ML) techniques for the analysis and prediction of features. These studies have specifically focused on employing a stand-alone server [6] for this purpose. However, it is worth noting that the adoption of such a server poses concerns related to centralised control. The aforementioned complex problems can be effectively addressed by the strategic integration of federated machine learning and decentralised trust-less service platforms like Blockchain.

1.2. *Autonomous Learning*

Within the context of SHs, ML servers acquire knowledge from many aspects of users' behaviour, feelings, and usage patterns in order to independently deliver a personalised environment for the individual. The task at hand involves the collection, processing, and analysis of environmental data within a SH system [7]. This procedure entails the implementation of a learning system and the establishment of a structured framework for handling these data. Typically, in addition to a hardware-based system control unit that facilitates communication between wireless electrical outlets and sensors, the gathered data is processed and managed via a third-party cloud service. The cloud system employs adaptive decision-making mechanisms to effectively cater to the requirements of its users within the given environment. The evaluation of performance is contingent upon the secure handling and processing of data gathering.

In this study, the cloud server is functions as a gateway to each individual SH, fulfilling a dual function of operating the SH and acquiring knowledge about various properties from its local data. A cloud server has the capability to manage many households, while ensuring the maintenance of distinct storage pathways for each home. The use of a solitary home dataset can facilitate the prediction of individual users' behaviour. However, the aggregation of various cloud services generates extensive data and enables pervasive activity, hence rendering it conducive for comprehensive prediction. In the proposed system, a cloud server would independently train a local model by utilising the data from its service providers. The resulting knowledge will then be disseminated to the global blockchain network to facilitate further progress. Manufacturers and other stakeholders have the ability to observe the ultimate prediction generated by the BC network. This idea utilises a cloud server as a localised learning server, specifically referred to as a federated learning node. Additionally, a blockchain network is employed to fulfil the role of a global network, serving as an aggregator.

1.3. Decentralized Aggregator

Blockchain (BC) is a cryptographic and distributed ledger technology (DLT) that facilitates secure data movement among many parties. It facilitates value exchange, commonly referred to as transactions, with the absence of reliance on a central authority for trust. The transactions are recorded in a ledger that is managed by a network of interconnected computers, known as peers, as opposed to a centralised entity like as a cloud server. The BC system conducts an autonomous verification process, commonly referred to as endorsement, prior to granting approval for a transaction. This verification process is of utmost importance in maintaining the security of the system [8]. Furthermore, it facilitates consortium-based inter-organizational transactions through smart contracts, which holds significant importance in facilitating communication among service providers. According to [9], the utilisation of a blockchain as a service platform enables the management and transformation of current centralised servers into a decentralised distributed ledger technology (DLT) system. One of the primary difficulties lies in effectively managing the ongoing transactions inside a Blockchain Network (BCN). Moreover, it is customary for a block to have a capacity of up to 1MB of data. However, in the context of federated learning (FL), the requirement is that each model, exceeding 200MB in size, must be accommodated within a single block. This presents a considerable challenge. Hence, we have put forth a proposition for a GWP that can effectively manage ongoing transactions by utilising a tailored block structure to transport the block containing the replica of the model. This study involves the utilisation of a cloud server to build local models and afterwards construct a global model within the context of a BCN. The network also assumes the responsibility of managing access control for cloud servers.

This article proposes an BCT leverages FL architecture for intelligent and secure analysis of daily uses data in SHs. The architecture aims to address the issue of excessive local transactions in a SH by processing data supplied by the SH in a GWP. The GWP assumes the function of a federated learning server inside a global learning network that is governed by a permissioned blockchain. The BC offers access control services for the entire ecosystem and aggregator services. It allows an CIoT manufacturer to anticipate client behaviours through the use of intelligent analysis. The article contributes –

- blockchain-controlled federated machine learning architecture for intelligent analysis of CIoT data produced in smart home network.
- an optimum solution for handling substantial local transactions generated in a home.
- an optimization way to handle the continuous transaction generated Big Data.
- an effective testbed analysis based on the public Stanford cars Dataset.
- Finally, open research issues present the technical challenges raised in the real-life environment.

The subsequent sections of this article provide further elaboration on the specifics of implementation. Section 2 provides a comprehensive overview of recent contributions in the field of smart home implementation terminology. The architectural specifics are illustrated in Section 3. Section 5 provides a comprehensive overview of the implementation settings, findings, and security analysis. Lastly, the overall contribution is summarised in Section 6.

2. Related Works

This section presents a thorough overview of recently suggested blockchain-backed machine learning technologies aimed at enhancing SH security and privacy. Additionally, we have refined the selection of FL systems that are relevant to our research and have uncovered notable distinctions in comparison to the present approach. Thus far, there have been numerous substantial proposals put out in the realm of smart home security. The majority of these systems employ a conventional centralised architecture, which gives rise to vulnerabilities in the form of single points of failure, as well as concerns regarding security and privacy [10]. In light of the constraints posed by the centralised system, there has been a growing interest in the utilisation of stand-alone BCT as a potential solution to address the common difficulties faced in smart home environments, as discussed in various recent publications [11,12]. In numerous scholarly works, the concept of data being considered a valuable asset for autonomous learning and the use of BCT for enhancing cyber security have been extensively discussed [13,14]. However, in conventional ML, computers acquire knowledge from the unprocessed data provided by users, so giving rise to additional security concerns that are addressed by Google's FL technology [15]. The state of Florida permits the implementation of decentralised training by data owners, while simultaneously sharing the acquired learning outcomes with a centralised aggregator. The centralised aggregator is acknowledged as a drawback in terms of security [16]. the study conducted a decentralised approach to local gradient sharing, utilising a blockchain-based system for storing models.

The authors of [17] suggested encrypting local updates before adding them to the BC ledger as part of a permissioned blockchain-supported FL platform. Healthcare, transport networks, the energy industry, etc., are just a few of the many potential application domains for FL and BCT. The authors of [18] contributed a BC-based FL that enabled an adaptable framework to guarantee the reliability and safety of networks. It took into account user-specific trust factors (i.e., prior positive experiences, guarantees, transparency, and accountability) to make predictions about the devices' trustworthiness. The subject of device failure in IIoT is discussed at length in [19]. The authors proposed a decentralised, FL platform using BCT to ensure the authenticity of user information. The proposal's novelty lies in its potential to implement a blockchain-based system for the periodic storage of client data records in tree and tree root stores.

The paper [20] examined the issue of data leaking from a model created by local members in a BC-based FL network. The authors launched an inference assault for the purpose of analysing experimental data. Using blockchain-assisted FL for intelligent edge computing, they took advantage of an accidental property leakage to single out a group of users that had a particular characteristic. In addition, a weighted fair data sampler technique has been implemented to improve training quality by increasing data fairness. The author [21] offered a blockchain-based system for incentivizing FL data owners to maintain data quality. In a technical sense, the blockchain-centered reputation system transparently aggregates high-quality models. Like other contributions, BC is used only for calculating the rewards and credit. There have been a number of articles discussing the broader concerns with FL, its limitations, and the potential benefits of combining FL with blockchain. However, most of these research contributions targeted to fix the problems caused by centralised aggregators using leveraging blockchain technology. Adding noise to the local model has been proposed in certain articles as a way to increase safety.

However, how big transactions created by SHs will be managed in the BCN, the constraints of blocks for storing a large model, etc., were not taken into account, and neither was the security of the smart home ecosystem as a whole. Moreover, instead of forwarding every transaction into BCN, to the best of our knowledge, none of them considered how we could increase scalability by processing intra-organisational networks. Instead, this article seeks to give a safe and smart learning approach to guarantee the most cutting-edge advantages in access control, blockchain scalability, and ledger optimisation targeting overburden to BCN.

3. Decentralised Learning Architecture

Smarthome Networks (SHN) and BCN are the two main components of this ecosystem. Traditionally, SHN is managed by a centralised private or cloud server; in this study, the server also acts as a gateway between SHN and BCN. The framework's main goal is fourfold: *Security*, *Scalability*, *ledger optimisation*, and *Accurate prediction*. First, blockchain ensures data security and secures remote access to home appliances. Second, *Scalability and ledger optimisation* are ensured by migrating the home server to a GWP by separating transactions into local and global. Finally, the Federated Machine learning process ensures accurate predictions through an intelligent process of diverse data. Blockchain-controlled Federated Learning architecture for CIoT data from SHN is depicted in Figure 1. The framework comprises three layers: the top layer provides the pervasive *CIoT-integrated SHN*; the middle layer depicts the **Gateway Peer (GWP)**, which is an additional peer of BCN; and the bottom layer depicts the **Blockchain Network**.

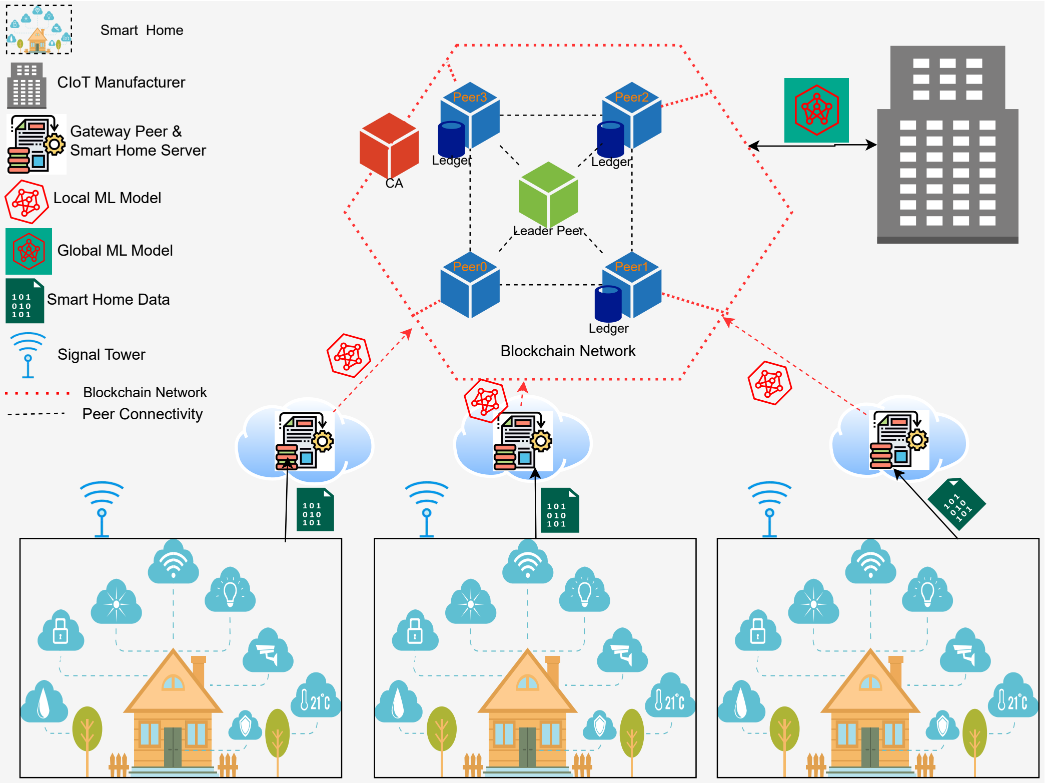


Figure 1. Decentralised Federated Learning Platfrom.

3.1. Overview

Every device in a SHN executes transactions on its local server, which is connected to the ecosystem through GWP. GWP serves two important functions: *transaction segregation* and *local training* for advanced ML model generation. Firstly, GWP divides the transactions between SHN and BCN based on the destination of the transaction, indirectly optimising the ledger. Secondly, continue local training based on the created time series data. GWP is interconnected with a BCN via the Certificate Authority (CA) registration procedure. The GWP serves as federated learning nodes and gathers an initial model from the blockchain network, which is generated by the network controller or manufacturer. After training, the model is transmitted to the BCN responsible for global model generation. The BCN accumulates all contemporary models from each GWP during a consensus session, a specific time session. During the consensus session, a leader peer facilitated the session, created a global model by averaging the models, and organised the session. Additionally, the leader collects each peer's vote regarding the global model and waits for 51% of all participants to endorse the global model. The global model is then transmitted to each GWP for the subsequent round of training, and the process is repeated until the final prediction objective is achieved.

3.2. Typical Smart Home Network Architecture

The SH is equipped with CIoT (i.e., smart devices) and home automation services using standard networking technology. Figure 2 depicts a typical smart home applications administration architecture that integrates smart devices, gateways, and back-end networking components into a HAN. Successful integration of these components with a network service provider or server enables global access to a SH.

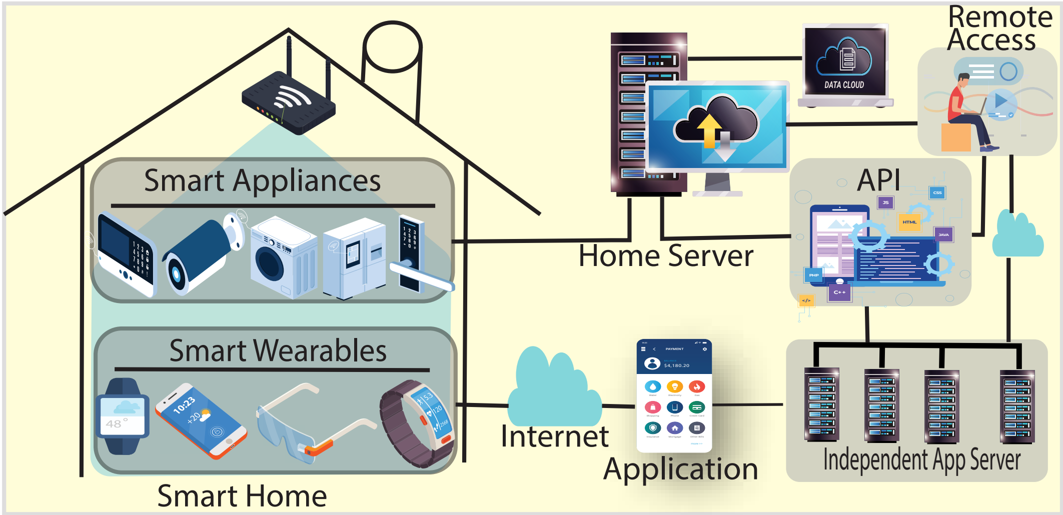


Figure 2. Typical Smart Home Network.

3.2.1. Home Appliances Connectivity

A smart home consists of a variety of smart appliances (e.g., intelligent freezers, metres, air conditioners, smart fans, etc.) connected to a local server. Due to the avoidance of server maintenance complexity, many SH users use third-party services that provide cloud-based services. Typically, appliance-generated transactions are stored and controlled on a local server or cloud server. In the proposed architecture, we use a gateway to connect every SH. In real-life implementation, an existing local server or cloud server can function as GW. Any transaction originating from HAN devices is processed by the GW. Consider $\{a_1, a_2, a_3, \ldots, a_n\} \in A_i$, where i^{th} smart home has n appliances controlled by the gateway GW_i. It is presumed that wearable devices used by family members are also managed by a GW-functioning home server. The GW is responsible for the interoperability of home services and functions as a GWP when connected to the BCN.

3.2.2. Independent App Server

SH users' wearables, like smart watches, glasses, shoes, and more, generally have limited resources and use third-party cloud services provided by the vendor, like the APP server. For learning reasons, data is sent from the APP server to the GWP using Application Programming Interface (API) services so that the autonomous services can be used. The trades can be sent to GW using the API. There is a BCN access control strategy that all cloud servers follow.

Table 1. Symbols With Labels.

Symbol	Meaning
T_i	Transaction from GW_i weight without/with Digital Asset
B_i	Block generated at GW_i
L^{GW}	BC ledger at GW
T^{dst}	Transaction Destination address
T^{src}	Transaction Source address
L^T	Consensus leader for global transactions
L^M	Consensus leader for global model
$\mathrm{pk}_{\mathrm{sign}}^{\rho_i}$	Public key with signature of users
v	User
`sck`	Secret key of v
M^l	Local ML Model
M^g	Global ML Model

3.3. Smarthome Gateway

Gateway(GW) replaces a typical home server, and by enabling blockchain functionalities, it creates a Gateway Peer (GWP) that works as a local peer. It is the key player in ledger optimization. Figure 1 represents the connectivity of GWP and BCN. All transactions must be executed through a GWP that interconnects the smart home with a BC network. Hence, we propose a GWP that comprises the full functionality of a solo-peer BCN [22]. A GWP might be implemented at home or in the cloud.
Functionalities: GWP plays dual roles, such as 1) ledger optimisation by segregating local transactions from external transactions, and 2) local training for intelligent automation processes. It is assumed GWP is fully functional with CPU and GPU, where we have used CPU for local transaction execution processes and GPU roles for local training in parallel [23].

3.4. Blockchain Network (BCN)

The Blockchain Network (BCN) consists of interconnected, independent peers that maintain their ledger and hold related smart contracts (chaincode). In the proposed framework, GWP communicates with peers on behalf of SHN. However, every SHN can be executed remotely with proper BCN endorsement. The following sections present details of the network components and the working procedure.

- **Peers:** The BCN comprises multiple peers (i.e., more than three) to ensure consensus and distributed ledger management. Peers receive the transactions from GWP and verify the source and credentials for the next processing round. A randomly selected leader leads the validation process through consensus. Similarly, another random peer organises the local model aggregation services and related consensus sessions (details in consensus). Every peer holds related smart contracts and separate ledgers for global models and IoT transactions.
- **Consensus:** During consensus, BCN initially creates a consensus session leader panel randomly. A particular leader (L^T) from the panel leads global transactions for smart homes, and another leader (L^M) handles the global model generation process (details in Section 4.2). One peer can lead only one consensus session at a time. The internal policy of the system controls the creation of leader panels and the synchronisation of responsibilities. Based on the PBFT consensus algorithm, leaders collect the maximum number of positive concerns from participating peers before approving the transactions. All global transactions from every GW to BCN are led by L^T and collected by Consensus for global transactions: positive voting depends on smart contract

validation, which was previously stored in peer The leader peer collects all simultaneous approved transactions and affixes them to a new block. The newly generated block is forwarded to every peer in the network.

- **Certificate Authority (CA):** The CA is responsible for generating a unique certificate and keys for every network component, including users. During transaction execution, peers justify the validation of the source and destination devices' and users' certificates.

Algorithm 1: Transaction Processing at GWP.

```
Input  : $(\mathtt{sck}, v_i, T_i, \mathtt{pk}^{v_i}_{\text{sign}})$
Output : $\mathtt{Success/Failure}$
$GW\{T_i, T^{src}, T^{dst}\} \leftarrow \forall A_i \in i[1, n]$
if $T^{dst}$ $\mathit{existsin}GW_i$ then
    $\overline{T_i} \leftarrow \mathtt{hash}(T_i)$
    $\mathtt{B_i} \leftarrow \mathtt{append}(\overline{T_i}, (\mathtt{sck}, \rho_i, \delta, \mathtt{pk}^{v_i}_{\text{sign}}))$
    $L^{GW_i} \leftarrow B_i$
else
    $\mathtt{B_i} \rightarrow$ BCN        \\ for block formation
    $B_x \leftarrow \forall^{n}_{i=1} B_i$
    if $B_x$ pass in Consensus then
        $L^T \leftarrow B_x$
    end
end
```

4. Technical Details

4.1. Scalability and Ledger Optimization

Every GWP should work as a localised peer for the home and interact with the BC network. The 1 illustrates transaction processing at GWP. As shown, during transaction execution, GWP verifies the source and destination of the transaction. If the transaction source and destination belong to itself, it executes locally without interaction with BCN; otherwise, it is forwarded to BCN. All locally executable transactions initially invoke a smart contract (a pre-installed programme chaincode). Chaincode reflects the terms and conditions between two devices. Whether the chaincode invocation result is positive or negative, transactions are executed and stored in a local ledger specific to the home network. If the incoming transaction destination does not belong to GWP, its integrated application prepares the transaction to be executable in the BCN. It ultimately reduces the transaction overloading in BCN up to 70% [24].

4.2. Federated Learning

Federated learning enables multiple users to train (i.e., local model) a shared global model without sharing their private data. Deep neural networks (DNNs) in this proposed architecture are capable of learning both global and local models. Figure 3 presents the communication flow of training models in different network components. It is assumed that n GWP trains an accurate machine learning model using their previously generated data $\{D_1, D_2, \dots, D_N\}$. A GWP_i, on behalf of the user i chooses to process its local data (D_i) and download initial model (M_i) training tasks from BCN for fast training epoch. At the end of the training round, it generates a local model M^l_i. Before forwarding to the BCN, a differential privacy parameter (details in Section 4.3) is added to the local model to ensure the advanced security of the local model. Similarly, all other $\forall^n_{i=1} GWP_i$ generate their local model $\{M^l_1, M^l_2, \dots, M^l_n\}$ belongs to $\{GWP_1, GWP_2, \dots, GWP_n\}$. By leveraging federated learning, all users can forward their local models into BCN for generating a global model (M^g) for knowledge sharing without exposing their sensitive data.

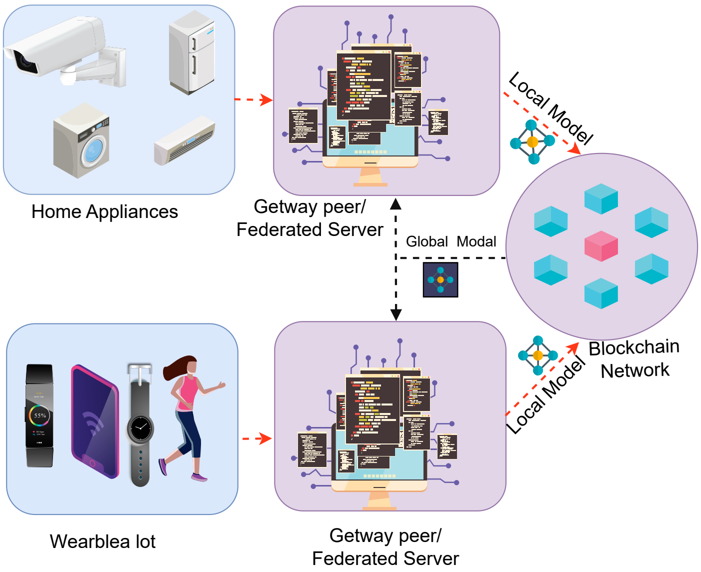


Figure 3. Federated Training Steps.

BCN initiates a consensus session and elects a leader for the averaging task. The leader creates a global model (M_i^g) using Eq. (1) at the end of the i^{th} training round.

$$M_i^G = \frac{1}{D} \sum_{i=1}^{n} M_i^l \tag{1}$$

In the context of the learning process, a generic FL model is formulated, in which a user denoted as i acquires and processes an input matrix denoted as $X_i = [x_{i1}, x_{i2}, \ldots x_{id_i}]$. Here, x_{id} represents an input vector utilised inside the FL method. The output of the input X_{id} can be represented as Y_{id}. The output vector utilised for training, acquired by the FL algorithm, by a specific user designated as GWP_i, is represented as $y_i = [y_{i1}, y_{i2}, \ldots\ldots y_{id_i}]$. The local FL model's (i.e., M^l) parameters are determines by the vector w_i. The projected output of a linear regression approach can be represented by the expression $x_{id}^T w_i$, where the weight vector (w_i) determines the efficacy of the linear regression learning process. In the quest to minimise the training loss, the user designated as i strives to determine the ideal parameters for the learning model. The symbol w_i is used to represent a variable or parameter in the context of the training procedure of a FL algorithm is conducted by

$$M^G = \frac{1}{D} \sum_{i=1}^{u} \sum_{d=1}^{d_i} f(w_i, x_{i,d}, y_{i,d}) \tag{2}$$

Here, D represents the summation of the training data from all users included in the study where $\frac{1}{D}$ averaging the weights. Likely, M^G and $f(w_i, x_{i,d}, y_{i,d})$ refers to the global model and loss function respectively.

The effectiveness of FL algorithms is contingent upon the values of both M^G and M^l, particularly following the initiation phase. The weight parameter w_i of each user is updated based on M^G, whereas the update of M^G is influenced by the M^l of all users. The modification of the local FL model w_i is contingent upon the selection of the learning algorithm and optimisation algorithm. The Stochastic Gradient Descent (SGD) technique was employed to perform updates on the local FL model.

doi:10.20944/preprints202311.0273.v1

4.2.1. Local Training

As mentioned earlier, GWP acts as a Federated Learning Server (FLS). During the local training, FLS initially collects an initial model from BCN and continues training using its local data. While training is completed, the updated model is stored in off-chain storage (e.g., Interplanetary File System (IPFS)) [25]. Then, the model's file reference and location pointer are fitted in a block with other meta-data (i.e., block-hash, source, destination, sign, etc.) and forwarded to BCN for global model generation.

4.3. *Differential Privacy*

Differential Privacy (DP) guarantees the privacy of data during mutual learning with the active participation of multiple smart homes. We incorporate a DP-enabled FL that protects data from external and internal sources (e.g., analysts) at training stages. Due to its advanced security features, it is well recommended in both academia and industry. For example, RaPPOR used DP in the Google Chrome browser [26] as a smaller privacy parameter. A randomised algorithm f provides (ϵ, δ) differential privacy if their neighbouring datasets D and $\acute{D}$ and f confirm that

$$Pr[f(D) \in Y] \leq e^{\epsilon} Pr[f(\acute{D}) \in Y] + \delta$$

Here δ is introduced to account for the probability (Pr) that plain ϵ-DP is broken [27]. Y iterates through all subsets of the output range of mechanism f. When $\delta = 0$, the mechanism f becomes $\epsilon-$ deferentially private.

4.4. *Normalization Technique*

To ensure the confidentiality of users' updates, we introduce perturbations to the extracted features within the normalisation layer. In the case of a singular channel, it is assumed that the convolutional layers produce an output with dimensions $L_f \times W_f$. The value at position (i, j) for the feature of image n is denoted as $P_{i,j,n}$.

We employed $\hat{P}_{i,j,n}$ for $n \in B$ with a mean 0 and variance 1 instead of typical batch normalisation, $P_{i,j,n}$

$$\frac{1}{|B|} \sum_{n \in B} \hat{P}_{i,j,n} = 0$$

and

$$\frac{1}{|B|} \sum_{n \in B} (\hat{P}_{i,j,n})^2 = 1$$

According to the Cauchy-Schwarz inequality [28] bounds $|B| = M$ and

$$\hat{P}_{i,j,n} \in (-\sqrt{M-1}, \sqrt{M-1})$$

where for any i, j, n while a single value of features

$$\{\hat{P}_{i,j,n}, | i \in \{1, 2, \ldots, L_f\} \text{ and } j \in \{1, 2, \ldots, W_f\}\}$$

of image n varies, the sensitivity of

$$\{\hat{P}_{i,j,n}, | i \in \{1, 2, \ldots, L_f\} \text{ and } j \in \{1, 2, \ldots, W_f\}\}$$

can be at most $2\sqrt{M-1}$.

To ensure ϵ-differential privacy, the Laplace mechanism [29] is employed. Specifically, a zero-mean Laplace noise with a scale of $2\sqrt{M-1}/\in$ is independently added to each $\hat{P}_{i,j,n}$, where i ranges from 1 to L_f and j ranges from 1 to W_f. This measure is taken to secure the privacy of $\hat{P}_{i,j,k}$.

This research normalize $\hat{P}_{i,j,n}$ for $i \in \{1, 2, \ldots, L_f\}$ and $j \in \{1, 2, \ldots, W_f\}$ as,

$$\hat{P}_{i,j,n} \in (-\sqrt{M-1}, \sqrt{M-1})$$

while if one value in the feature

$$\{\hat{P}_{i,j,n}, |i \in \{1,2,\ldots\ldots, L_f\} and\ j \in \{1,2,\ldots\ldots, W_f\}\}$$

varies for image n, the sensitivity in

$$\{\hat{P}_{i,j,n}, |i \in \{1,2,\ldots\ldots, L_f\} and\ j \in \{1,2,\ldots\ldots, W_f\}\}$$

is $2\sqrt{M-1}$.

Our normalization technique requires only

$$\hat{P}_{i,j,n} \in [-\sqrt{M-1}, \sqrt{M-1}]$$

without any constraints on the mean and variance.

In this experiment, the input layer is augmented using zero-mean Laplace noise, which is a common approach of existing solutions. However, the feature distribution is modelled using a Gaussian distribution, which is widely employed in various real-world applications. The majority of feature values, following the application of batch normalisation, are often within the range of $[-3\sigma, 3\sigma]$, where σ represents the standard deviation. This range is in contrast to the previously assumed range of $[-\sqrt{M-1}, \sqrt{M-1}]$, where M denotes the number of features. On the other hand, when employing this normalisation strategy, the feature values are distributed more uniformly across the range of $[-\sqrt{M-1}, \sqrt{M-1}]$. Batch normalisation approaches are more susceptible to perturbations in feature values compared to an equivalent quantity of Laplace noise. As an illustration, in the case when the batch size is set to $N = 32$ and the scale parameter of the Laplace distribution is given by $2\sqrt{M-1}/\in$.

5. Evaluations and Analysis

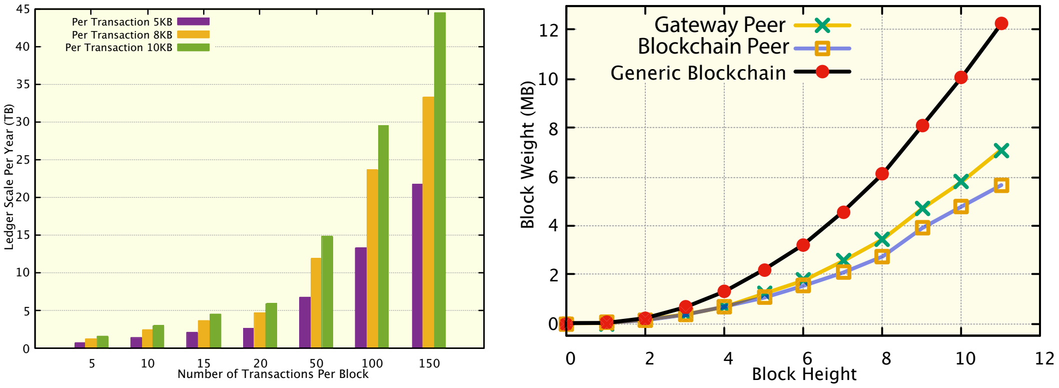
In two testbeds, we evaluated our proposed blockchain-based FL framework. Initially, we evaluated the blockchain transaction for ledger optimisation issues without ML using the Hyperledger Fabric (v2.0) platform within a Docker container. It aids in predicting the software-based implementation of a real-world application. As machine learning application implementation in Hyperledger Fabric is a complex undertaking, we re-simulated it in a Python environment to evaluate the performance of the entire ecosystem (details in Section 5.2).

5.1. Stand-alone Blockchain Applications

In order to incorporate blockchain with machine learning, two physical systems were utilised: *i*) an Intel i5 processor running at 3 GHz, equipped with 8 GB of 1600 MHz DDR3 RAM, and *ii*) an Intel i7 processor running at 2.7 GHz, equipped with 16 GB of 1600 MHz DDR3 RAM. The prototype is implemented with four peers, with a node-red based application employed for transaction generation.

Figure 4 presents the ledger growth and ledger scalability synopsis. It evaluates the execution of the continuous transactions in the BC network and the impact on ledger expansion in the BC Ledger. From experimental evaluation on Hyperledger Fabric, we know that every trade is $\approx 5 - 10KB$ on average, a block is formed with an average of 500 transactions per second, and a block header is $4.5KB$. It expands the ledger at a rate of approximately 50–100 KB/sec, or approximately 4–8 GB/day, or approximately 1.5–3 TB/year. Although this does not seem very high for a single node, it becomes impractical in a 10K home network with 20 devices per home. Figure 4(a) presents a production environment synopsis for 1K smart homes in a blockchain network where approx. 15 devices are contained per home. It shows three scenarios for whether transaction weight may vary from source to source depending on formats. In this experiment, we have considered three different sizes: 5KB to

7KB, 8KB to 9KB, and 10KB to 12KB. Ledger size is proportional to the transaction amount and size. The core impact of GWP has been presented in Figure 4(**b**).



(**a**) Memory Optimization Effect

(**b**) Leger Scalability in Proposed framework

Figure 4. Ledger Optimization Effect on Gateway Peer Implementation.

Figure 4(**b**) presents the ledger optimisation implemented in the Hyperledger Fabric platform in the proposed network. It evaluates the consecutive ten blocks both in the GWP and blockchain networks. The figure shows that GWP and BC peer carry almost 60% and 40% of the total required ledger present in generic blockchain lines, respectively.

5.2. Prediction Analysis

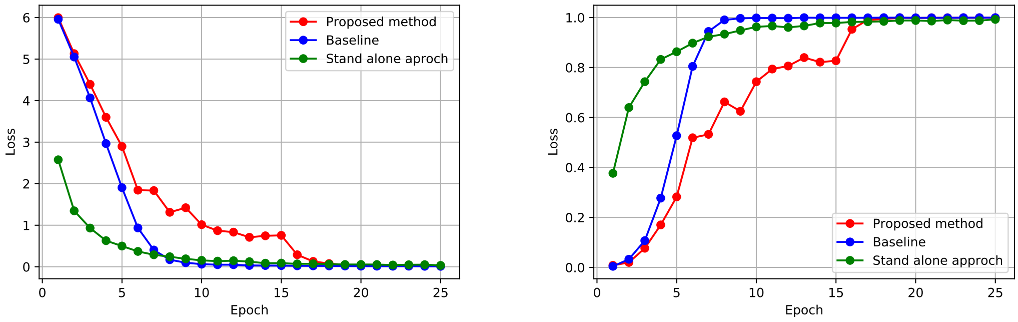
To forecast the comprehensive performance of the ecosystem, we employed the widely recognised public Stanford Cars Dataset [30], which comprises 16,185 photos representing 196 distinct automobile classes. The dataset comprises 8,144 photos for training purposes and 8,041 images for testing purposes. In order to provide a well-balanced dataset for each user, we evenly divide the overall training and test sets according to their respective classes. In conventional FL without blockchain, a comparable tailored dataset is employed to establish a baseline comprehension. Moreover, the experiment expands upon the identical experimental configuration by employing a stand-alone methodology to obtain the baseline outcome by conventional machine learning techniques.

The models are trained using the PyTorch library, employing the SGD with a learning rate of 0.01. The utilisation of a pre-trained ResNet50 model is employed for the purpose of conducting traditional image classification and carrying out the local training procedure within the Generalised Weighted Pooling (GWP) of each individual local organisation, also referred to as a learning node. The NVIDIA GeForce RTX 2080 GPU is utilised for each learning node. In the initial phase, a configuration consisting of four private servers, each equipped with four GPUs, is employed to establish numerous local training settings, resulting in a total of 16 GPUs. In order to facilitate experimentation, each GPU operates as an autonomous learning node. Concurrently, the CPU of a dedicated server functions as a generalised work processor to assess the execution process of the blockchain. The Blockchain network comprises six peers distributed among four remote servers. Each server is equipped with an Intel Xeon E7 v3 processor and a Core(TM) i7-5960X CPU running at a clock speed of 3.00 GHz, featuring 8 cores. Additionally, each server is equipped with 125 GB of RAM. The simulation of the blockchain network and consensus process is implemented using Python 3.8.

The CNN network that we have created incorporates hidden layers to facilitate the process of feature extraction, as well as fully connected layers to enable classification. In our network architecture, we incorporated two hidden layers, each consisting of 30 and 80 channels, respectively. The dimensionality of the output is lowered through the utilisation of the max-pooling layer. Hence, the utilisation of max-pooling layers enhances the pace of learning in neural networks. Following the normalisation of each CNN layer, various benefits are observed. Firstly, it facilitates the computation of sensitivity, which aids in determining the appropriate level of noise to be added. Additionally, it contributes to the acceleration of the learning rate and serves to regularise gradients, hence mitigating the impact of distractions and outliers.

5.3. Result and Discussion

This section provides a comprehensive overview of the learning outcomes associated with the proposed framework for federated learning utilising blockchain technology. We considered three scenarios where the typical FL learning approach is used as a baseline compared with our proposed method, and the typical ML method is used for overall bench marking. Figure 5 illustrates the training progress and accuracy. For the experiment, we ran 100 rounds to train the model in six federated learning nodes parallel. Training success is shown in Fig. 5(**a**) where the loss declines gradually. Loss decreases firstly in baseline and typical ML than our proposed method. However, they reach a convergence point almost at the end of the same round. The Figure 5(**b**) demonstrates the learning accuracy through train the model for the object detection model compared to baseline and typical approaches. The figure demonstrates the proposed framework converges with typical approaches almost at the same time.

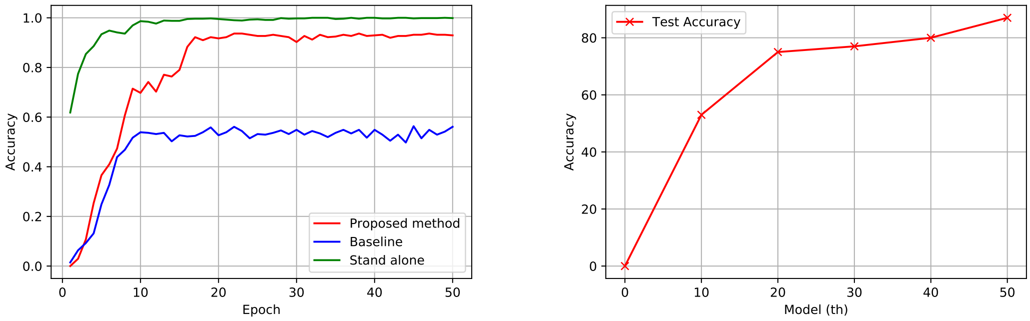


(**a**) Learning Progress

(**b**) Learning accuracy

Figure 5. Training Outcomes.

The classification performance of the proposed system in comparison to conventional FL and stand-alone ML systems is illustrated in Figure 1. The proposed methodology is assessed using both a validation dataset and a test set. The validation accuracy based on the validation dataset is depicted in Figure 1. The results indicate that the suggested scheme has an image recognition accuracy of approximately 88%, which is approximately 30% higher than the average performance of traditional federated learning methods. While the proposed system does not outperform stand-alone approaches much, there is still a noticeable difference in performance. Furthermore, the primary objective of this project is to enhance federated learning (FL) with regards to the security and privacy of users' data. Several test photos were evaluated using the models created by the proposed framework, as depicted in Figure 6(**b**). Various global models, derived by averaging the FL models, were employed to assess their efficacy in real-world situations. The presented data illustrates that the 10th global model exhibits a classification accuracy of around 56% for the photos, while the final model (50th) demonstrates a successful classification rate of 86% for the test images. Hence, it can be concluded that the performance of the proposed system is comparatively superior to that of classic federated learning systems. Additionally, the incorporation of blockchain technology in the aggregator strengthens the security measures of the entire ecosystem, hence demonstrating the effectiveness of the proposed system design.



(**a**) Validation accuracy

(**b**) Test accuracy

Figure 6. Validation and Test evaluations

6. Conclusion

Users of CIoT especially in smart homes seek to reap the benefits of automation without giving up their personal safety or confidentiality. For the safety of the entire ecosystem, having the most recent system is essential. Additionally, standard, external services should have robust regulation. The proposed architecture includes safety measures for automated prediction and update management. Blockchain is being used to tackle problems with secure automation, and our gateway peer helps to alleviate some of the current scaling problems in blockchain. Federated learning also prevents data sharing for machine learning, which is a significant improvement to the security policy. The testbed outcome demonstrates that the contribution provides solutions to significant problems that may occur as a result of combining HAN with blockchain and an intelligent automation system. In addition to addressing security concerns, the suggested GWP method has the potential to dramatically improve scalability by doubling throughput (TPS) and decreasing ledger overhead by more than 60% compared to conventional procedures. It devises a workable and safe method of dealing with the continual data provided by smart homes.

References

1. Caviglione, L.; Wendzel, S.; Mazurczyk, W. The Future of Digital Forensics: Challenges and the Road Ahead. *IEEE Security Privacy* **2017**, *15*, 12–17. doi:10.1109/MSP.2017.4251117.
2. Holst, A. *Smart Home Report 2021*; 2021. Access On: 05 November 2021.
3. Lee, Y.t.; Hsiao, W.h.; Huang, C.m.; Chou, S.c.T. An integrated cloud-based smart home management system with community hierarchy. *IEEE Transactions on Consumer Electronics* **2016**, *62*, 1–9. doi:10.1109/TCE.2016.7448556.
4. Irwin, D.; Albrecht, J. Smart Homes: Implemented. *IEEE Pervasive Computing* **2019**, *18*, 91–95. doi:10.1109/MPRV.2019.2912258.
5. Mukherjee, A.; Balachandra, M.; Pujari, C.; Tiwari, S.; Nayar, A.; Payyavula, S.R. Unified smart home resource access along with authentication using Blockchain technology. *Global Transitions Proceedings* **2021**, *2*, 29–34. 1st International Conference on Advances in Information, Computing and Trends in Data Engineering (AICDE - 2020), doi:https://doi.org/10.1016/j.gltp.2021.01.005.
6. Yang, J.; Zou, H.; Jiang, H.; Xie, L. Device-Free Occupant Activity Sensing Using WiFi-Enabled IoT Devices for Smart Homes. *IEEE Internet of Things Journal* **2018**, *5*, 3991–4002. doi:10.1109/JIOT.2018.2849655.
7. Wu, Q.; Chen, X.; Zhou, Z.; Zhang, J. FedHome: Cloud-Edge Based Personalized Federated Learning for In-Home Health Monitoring. *IEEE Transactions on Mobile Computing* **2022**, *21*, 2818–2832. doi:10.1109/TMC.2020.3045266.
8. Jung, S.S.; Lee, S.J.; Euom, I.C. Delegation-Based Personal Data Processing Request Notarization Framework for GDPR Based on Private Blockchain. *Applied Sciences* **2021**, *11*.

9. Biswas, S.; Sharif, K.; Li, F.; Latif, Z.; Kanhere, S.S.; Mohanty, S.P. Interoperability and Synchronization Management of Blockchain-Based Decentralized e-Health Systems. *IEEE Transactions on Engineering Management* **2020**, *67*, 1363–1376. doi:10.1109/TEM.2020.2989779.
10. Arif, S.; Khan, M.A.; Rehman, S.U.; Kabir, M.A.; Imran, M. Investigating Smart Home Security: Is Blockchain the Answer? *IEEE Access* **2020**, *8*, 117802–117816. doi:10.1109/ACCESS.2020.3004662.
11. Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaram, P. Blockchain for IoT security and privacy: The case study of a smart home. 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2017, pp. 618–623. doi:10.1109/PERCOMW.2017.7917634.
12. Lin, C.; He, D.; Kumar, N.; Huang, X.; Vijayakumar, P.; Choo, K.K.R. HomeChain: A Blockchain-Based Secure Mutual Authentication System for Smart Homes. *IEEE Internet of Things Journal* **2020**, *7*, 818–829. doi:10.1109/JIOT.2019.2944400.
13. Khan, M.A.; Abbas, S.; Rehman, A.; Saeed, Y.; Zeb, A.; Uddin, M.I.; Nasser, N.; Ali, A. A Machine Learning Approach for Blockchain-Based Smart Home Networks Security. *IEEE Network* **2021**, *35*, 223–229. doi:10.1109/MNET.011.2000514.
14. Kim, D. A Reverse Sequence Hash Chain-based Access Control for a Smart Home System. 2020 IEEE International Conference on Consumer Electronics (ICCE), 2020, pp. 1–4. doi:10.1109/ICCE46568.2020.9043090.
15. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.y. Communication-Efficient Learning of Deep Networks from Decentralized Data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics; Singh, A.; Zhu, J., Eds. PMLR, 2017, Vol. 54, *Proceedings of Machine Learning Research*, pp. 1273–1282.
16. Ramanan, P.; Nakayama, K. BAFFLE : Blockchain Based Aggregator Free Federated Learning. 2020 IEEE International Conference on Blockchain (Blockchain), 2020, pp. 72–81. doi:10.1109/Blockchain50366.2020.00017.
17. Sun, J.; Wu, Y.; Wang, S.; Fu, Y.; Chang, X. Permissioned Blockchain Frame for Secure Federated Learning. *IEEE Communications Letters* **2022**, *26*, 13–17. doi:10.1109/LCOMM.2021.3121297.
18. Otoum, S.; Ridhawi, I.A.; Mouftah, H. Securing Critical IoT Infrastructures With Blockchain-Supported Federated Learning. *IEEE Internet of Things Journal* **2022**, *9*, 2592–2601. doi:10.1109/JIOT.2021.3088056.
19. Zhang, W.; Lu, Q.; Yu, Q.; Li, Z.; Liu, Y.; Lo, S.K.; Chen, S.; Xu, X.; Zhu, L. Blockchain-Based Federated Learning for Device Failure Detection in Industrial IoT. *IEEE Internet of Things Journal* **2021**, *8*, 5926–5937. doi:10.1109/JIOT.2020.3032544.
20. Shen, M.; Wang, H.; Zhang, B.; Zhu, L.; Xu, K.; Li, Q.; Du, X. Exploiting Unintended Property Leakage in Blockchain-Assisted Federated Learning for Intelligent Edge Computing. *IEEE Internet of Things Journal* **2021**, *8*, 2265–2275. doi:10.1109/JIOT.2020.3028110.
21. Qi, J.; Lin, F.; Chen, Z.; Tang, C.; Jia, R.; Li, M. High-quality Model Aggregation for Blockchain-based Federated Learning via Reputation-motivated Task Participation. *IEEE Internet of Things Journal* **2022**, pp. 1–1. doi:10.1109/JIOT.2022.3160425.
22. Ammi, M.; Alarabi, S.; Benkhelifa, E. Customized blockchain-based architecture for secure smart home for lightweight IoT. *Information Processing & Management* **2021**, *58*, 102482. doi:https://doi.org/10.1016/j.ipm.2020.102482.
23. Keckler, S.W.; Dally, W.J.; Khailany, B.; Garland, M.; Glasco, D. GPUs and the Future of Parallel Computing. *IEEE Micro* **2011**, *31*, 7–17. doi:10.1109/MM.2011.89.
24. Biswas, S.; Sharif, K.; Li, F.; Nour, B.; Wang, Y. A Scalable Blockchain Framework for Secure Transactions in IoT. *IEEE Internet of Things Journal* **2019**, *6*, 4650–4659. doi:10.1109/JIOT.2018.2874095.
25. Pappas, C.; Chatzopoulos, D.; Lalis, S.; Vavalis, M. IPLS: A Framework for Decentralized Federated Learning. 2021 IFIP Networking Conference (IFIP Networking), 2021, pp. 1–6. doi:10.23919/IFIPNetworking52078.2021.9472790.
26. Erlingsson, U.; Pihur, V.; Korolova, A. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security; Association for Computing Machinery: New York, NY, USA, 2014; CCS '14, p. 1054–1067. doi:10.1145/2660267.2660348.

27. Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating Noise to Sensitivity in Private Data Analysis. Theory of Cryptography; Halevi, S.; Rabin, T., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2006; pp. 265–284.
28. *EWSN '19: Proceedings of the 2019 International Conference on Embedded Wireless Systems and Networks*, USA, 2019. Junction Publishing.
29. *Calibrating noise to sensitivity in private data analysis*, 2006.
30. Krause, J.; Stark, M.; Deng, J.; Fei-Fei, L. 3D Object Representations for Fine-Grained Categorization. 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13); , 2013.

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.