City, University of London Institutional Repository

City Research Online

Article

An Optimized 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; superstar_wzc@zzu.edu.cn
2 School of Economics, Sichuan University, Chengdu 610065, 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, Zhengzhou 451460, China; shaoxinming@zzrvtc.edu
5 Information Systems Department, College of Computer Science and Information Systems, Najran University, Najran 55461, Saudi Arabia; msalrizq@nu.edu.sa (M.A.)
6 School of Cybersecurity, Old Dominion University, Norfolk, VA 23529, USA; mmunir@odu.edu
7 Computer Science and Digital Technologies Department, University of East London, University Way, London E16 2RD, UK
* Correspondence: xyliu@cauc.edu.cn (X.L.); sujitbiswas@ieee.org (S.B.)

Abstract: Consumer Internet of Things (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) models using end-user data. Federated learning (FL) uses privacy-preserving ML techniques to forecast customers' needs and consumption habits, and blockchain replaces the centralized 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 introduces a novel architecture, emphasizing gateway peer (GWP) in the blockchain network to address scalability, ledger optimization, and secure model transmission issues. In the architecture, we replace the centralized aggregator with the blockchain network, while GWP limits the number of local transactions to execute in BCN. Considering the security and privacy of FL processes, we incorporated differential privacy and advanced normalization techniques into ML processes. These approaches enhance the cybersecurity of end-users and promote the adoption of technological innovation standards by service providers. The proposed approach has undergone extensive testing using the well-respected Stanford (CARS) dataset. We experimentally demonstrate that the proposed architecture enhances network scalability and significantly optimizes the ledger. In addition, the normalization technique outperforms batch normalization when features are under DP protection.

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

MSC: 68Q11

1. Introduction

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 between end-users via the internet. CIoT plays a vital role in facilitating smart home (SH) functionality, where residents use wearable CIoT devices and appliances that provide smart services and impact the lives of end-users. CIoT devices are

interconnected in an 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 existing 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%. Meanwhile, total revenue is expected to grow from USD 83 billion in 2023 to USD 86 billion in 2027, a gain of 60%. In 2027, sales are expected to increase and reach USD 222.9 billion [2]. Wearable CIoT devices refer to gadgets worn by home users, which are used to process private data; home appliances (HAs) ensure smart services by connecting with each other through automation. Data from these devices generate 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 use service-specific centralized servers, such as cloud or edge servers, and are managed either autonomously or in tandem with third-party service providers [3]. When it comes to analyzing 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 [4].

1.1. Challenges of the Typical Intelligent System

Considering security, many IoT devices are designed to contain built-in security features (i.e., SHA-256, PoW, SD memory, etc.) with other peripheral electronic devices targeting blockchain feature integration with them [5]. However, every HA integration with BCN is costly. Likewise, it would be very expensive to set up a separate, secure, and self-sufficient control and management system for SHs. In addition, many novel IoT devices are consistently being introduced to provide ubiquitous services. The data types, message structures, and other features of these devices are very different, which makes it hard to manage them with standard systems, especially using relational databases [6]. However, the majority of SHs employ the services of third-party providers that commonly utilize a centralized control system [3]. The implementation of a comprehensive system utilizing centralized 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 [7]. The inclusion of extra obstacles complicates the task of guaranteeing the appropriate utilization of data for subsequent analysis. Several recent studies have proposed the use of ML techniques for the analysis and prediction of features. These studies have specifically focused on employing a stand-alone server [8] for this purpose. However, it is worth noting that the adoption of such a server poses concerns related to centralized control. The aforementioned complex problems can be effectively addressed by the strategic integration of FL and decentralized trustless service platforms like blockchain.

1.2. Autonomous Learning

Within the context of SHs, ML servers acquire knowledge from many aspects of user behaviors, feelings, and usage patterns in order to independently deliver a personalized environment for the individual. The task at hand involves the collection, processing, and analysis of environmental data within a SH system [9]. This procedure entails the implementation of a learning system and the establishment of a structured framework to handle these data. Typically, in addition to a hardware-based system control unit that facilitates communication between wireless electrical outlets and sensors, the gathered data are 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 performance evaluation is contingent upon the secure handling and processing of data.

In this study, the cloud server 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 user behaviors. However, combining different cloud services creates a lot of data and allows a lot of activity to occur all at once, which makes it possible to make accurate predictions. In the proposed system, a cloud server would independently train a local model by utilizing 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 are able to see the final prediction that the BC network generates. This idea utilizes a cloud server as a localized learning server, specifically referred to as a federated learning node. Additionally, a blockchain network is employed to fulfill 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, in 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 centralized entity, like a cloud server. The BC system conducts an autonomous verification process, commonly referred to as an endorsement, prior to granting approval for a transaction. This verification process is of utmost importance in maintaining the security of the system [10]. It also makes consortium-based transactions between organizations possible through smart contracts, which is a very important way to make communication between service providers easier. According to [11], the utilization of a blockchain as a service platform enables the management and transformation of current centralized servers into a decentralized 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 1 MB of data. However, in the context of federated learning (FL), the requirement is that each model exceeding 200 MB in size be accommodated within a single block. This presents a considerable challenge. Hence, we put forth a proposition for a GWP that can effectively manage ongoing transactions by utilizing a tailored block structure to transport the block containing the replica of the model. This study involves the use of a cloud server to build local models and afterward 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 a BCT that leverages an FL architecture for intelligent and secure analysis of daily use data in SHs. The architecture aims to address the issue of excessive local transactions in an SH by processing data supplied by the SH in a GWP. Within a permissioned blockchain-controlled global learning network, the GWP assumes the role of a federated learning server. The BC offers access control services for the entire ecosystem and aggregator services. It allows a CIoT manufacturer to anticipate client behaviors 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 networks.
- An optimal solution for handling substantial local transactions generated in a home.
- An optimized approach to manage the continuous transaction-generated Big Data.
- An effective testbed analysis based on the public Stanford (CARS) dataset.
- Finally, open research issues that present the technical challenges raised in real-life environments.

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 contributions are summarized 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 refined the selection of FL systems relevant to our research and uncovered notable distinctions compared to the current approach. Thus far, there have been numerous substantial proposals in the realm of smart home security. The majority of these systems employ a conventional centralized architecture, which gives rise to vulnerabilities in the form of single points of failure, as well as concerns regarding security and privacy [12]. In light of the constraints posed by the centralized system, there has been growing interest in the utilization of standalone BCT as a potential solution to address common difficulties faced in smart home environments, as discussed in various recent publications [13,14]. Numerous scholarly works have extensively discussed the concept of data as a valuable asset for autonomous learning and the use of BCT for enhancing cybersecurity [15,16]. However, in conventional ML, computers acquire knowledge from the unprocessed data provided by users, thereby giving rise to additional security concerns addressed by Google's FL technology [17]. The state of Florida permits the implementation of decentralized training by data owners, while simultaneously sharing the acquired learning outcomes with a centralized aggregator. The centralized aggregator is acknowledged as a drawback in terms of security [18]. The study conducted a decentralized approach to local gradient sharing, utilizing a blockchain-based system for storing models.

The authors of [19] 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 [20] introduced 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 trustworthiness of devices. The subject of device failure in IIoT is discussed at length in [21]. The authors proposed a decentralized, 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.

Reference [22] 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 analyzing experimental data. Using blockchain-assisted FL for intelligent edge computing, the authors 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 of [23] 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 only used for calculating the rewards and credit. There have been several articles discussing the broader concerns with FL, its limitations, and the potential benefits of combining FL with blockchain. However, most of these research contributions are aimed at addressing the problems caused by centralized aggregators by leveraging blockchain technology. Adding noise to the local model has been proposed in certain articles as a way to increase safety.

However, the management of large transactions created by SHs in the BCN, the constraints of blocks for storing a large model, and the security of the smart home ecosystem

as a whole were not considered. Moreover, instead of forwarding every transaction to the BCN, to the best of our knowledge, none of them considered how we could increase scalability by processing intra-organizational networks. Instead, this article seeks to provide a safe and intelligent learning approach to ensure the most advanced benefits in access control, blockchain scalability, and ledger optimization targeting the overburden to BCN.

3. Decentralized Learning Architecture

Smart home networks (SHNs) and BCNs are the two main components of this ecosystem. Traditionally, a SHN is managed by a centralized private or cloud server; in this study, the server also acts as a gateway between the SHN and BCN. The framework's main goal is fourfold: *security*, *scalability*, *ledger optimization*, and *accurate prediction*. First, blockchain ensures data security and secures remote access to home appliances. Second, *scalability and ledger optimization* are ensured by migrating the home server to a GWP by separating transactions into local and global transactions. 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**.
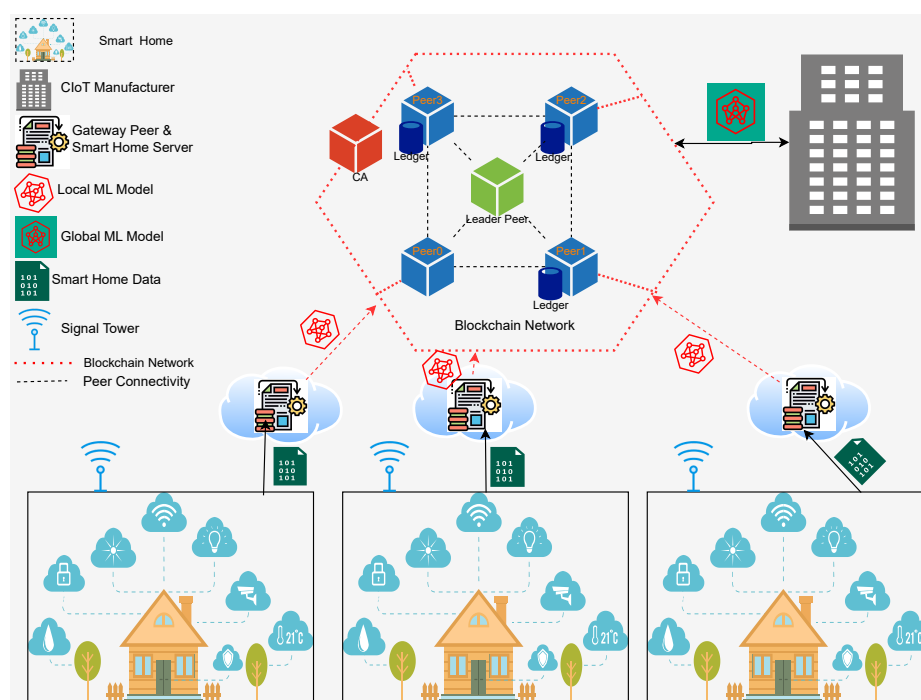

Figure 1. Decentralized federated learning platform.

3.1. Overview

Every device in an 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 the SHN and BCN based on the destination of the transaction, indirectly optimizing the ledger. Secondly, we 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 acts as a federated learning node 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 facilitates the session,

creates a global model by averaging the models, and organizes 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 devices (i.e., smart devices) and home automation services using standard networking technology. Figure 2 depicts the typical smart home application 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.
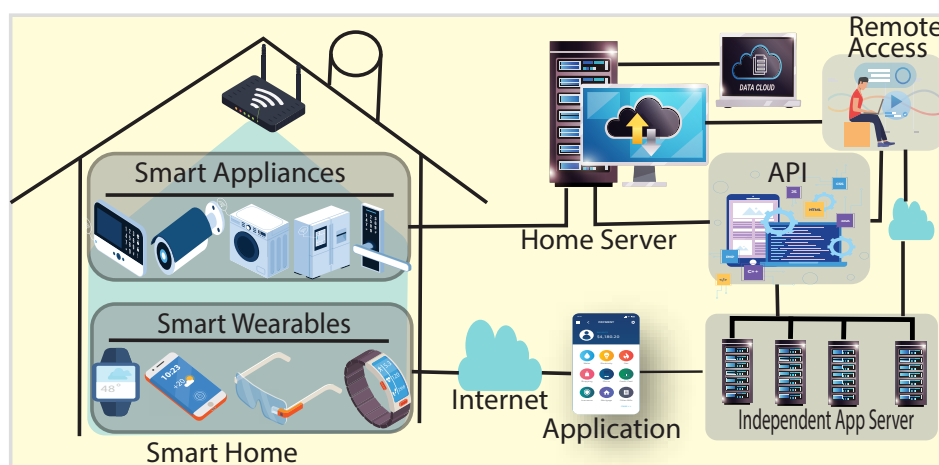

Figure 2. Typical smart home network.

3.2.1. Home Appliances Connectivity

A smart home consists of a variety of smart appliances (e.g., intelligent freezers, meters, 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 a 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 the ith 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 user 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 are sent from the APP server to the GWP using application programming interface (API) services so that 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.

Symbol	Meaning
T_i	Transaction from the GW_i weight without/with the 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 the global model
$\text{pk}_{\text{sign}}^{\rho_i}$	Public key with the signature of users
v	User
`sck`	Secret key of v
M^l	Local ML model
M^g	Global ML model

3.3. Smart Home 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 [24]. A GWP might be implemented at home or in the cloud.

Functionalities: GWP plays dual roles, such as (1) ledger optimization 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 use CPU for local transaction execution processes and GPU roles for local training, in parallel [25].

3.4. Blockchain Network (BCN)

The blockchain network (BCN) consists of interconnected, independent peers that maintain their ledger and hold related smart contracts (chaincodes). In the proposed framework, GWP communicates with peers on behalf of the 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 [26]. Peers receive the transactions from the 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 organizes 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 synchronization 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 by a peer. The leader peer collects all

simultaneously approved transactions and affixes them to a new block. The newly generated block is then forwarded to every peer in the network.

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

4. Technical Details

4.1. Scalability and Ledger Optimization

Every GWP should work as a localized peer for the home and interact with the BC network. Algorithm 1 illustrates transaction processing at the GWP. As shown, during the transaction execution, GWP verifies the source and destination of the transaction. If the transaction source and destination belong to the same entity, it is executed locally without interaction with the BCN; otherwise, it is forwarded to the BCN. All locally executable transactions initially invoke a smart contract (a pre-installed program chaincode). The chaincode reflects the terms and conditions between two devices. Regardless of 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 the GWP, its integrated application prepares the transaction to be executable in the BCN. This ultimately reduces transaction overloading in the BCN by up to 70% [27].

Algorithm 1: Transaction processing at the GWP.

> **Input** : $(\text{sck}, v_i, T_i, \text{pk}_{\text{sign}}^{v_i})$
> **Output**: Success/Failure
> **1** $GW\{T_i, T^{src}, T^{dst}\} \leftarrow \forall A_i \in i[1, n]$
> **2 if** T^{dst} *exists in* GW_i **then**
> **3** $\quad \overline{T_i} \leftarrow hash(T_i)$
> **4** $\quad \text{B}_\text{i} \leftarrow append \left(\overline{T_i}, (\text{sck}, \rho_i, \delta, \text{pk}_{\text{sign}}^{v_i})\right)$
> **5** $\quad L^{GW_i} \leftarrow B_i$
> **6 else**
> **7** $\quad \text{B}_\text{i} \rightarrow BCN \qquad \backslash\backslash$ *for block formation*
> **8** $\quad B_x \leftarrow \forall_{i=1}^n B_i$
> **9** $\quad$ **if** B_x *passes in consensus* **then**
> **10** $\quad \quad | \quad L^T \leftarrow B_x$
> **11** $\quad$ **end**
> **12 end**

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 the *n* GWP trains an accurate machine learning model using previously generated data $\{D_1, D_2, \ldots, D_N\}$. A GWP_i, on behalf of user *i*, chooses to process its local data (D_i) and download the initial model (M_i) training tasks from BCN for the fast training epoch. At the end of the training round, it generates a local model M_i^l. Before forwarding to the BCN, a differential privacy parameter (detailed in Section 4.3) is added to the local model to ensure the advanced security of the local model. Similarly, all other $\forall_{i=1}^n GWP_i$ generate their local models $\{M_1^l, M_2^l, \ldots, M_n^l\}$ belonging to $\{GWP_1, GWP_2, \ldots, GWP_n\}$. By leveraging federated learning, all users can forward their local models to the BCN for generating a global model (M^g) for knowledge sharing without exposing their sensitive data.
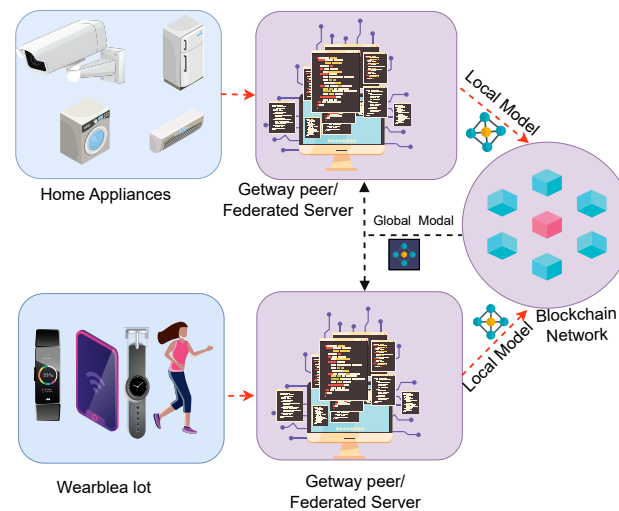
Figure 3. Federated training steps.

BCN initiates a consensus session and elects a leader for the averaging task. The leader creates a global model (M_i^g) using Equation (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, where 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 utilized inside the FL method. The output of the input X_{id} can be represented as Y_{id}. The output vector utilized 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 determined 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 minimize the training loss, the user designated as *i* strives to determine the ideal parameters for the learning model. Symbol w_i is used to represent a variable or parameter in the context of the training procedure of a FL algorithm, 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, with $\frac{1}{D}$ averaging the weights. Similarly, M^G and $f(w_i, x_{i,d}, y_{i,d})$ refer 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 optimization algorithm. The stochastic gradient descent (SGD) technique was employed to perform updates on the local FL model.

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)) [28]. 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

The basic objective of incorporating differential parameters into a model is to introduce stochastic perturbations to the local models, thereby generating imprecise models with limited diversity. While the implementation of DP does indeed alter the model, its influence on the pattern is expected to be minimal. Preserving the confidentiality of the underlying model is beneficial, as recent studies have demonstrated the potential for extracting source data from the machine learning model. The auditory stimuli have been meticulously crafted. Differential privacy enables technology companies to gather and disseminate aggregated data pertaining to user behaviors while safeguarding the privacy of individual users. In this research, DP guarantees the privacy of data during mutual learning with the active participation of multiple SHs. 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 [29] as a smaller privacy parameter. A randomized algorithm f provides (ϵ, δ) differential privacy if the neighboring 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 [30]. Y iterates through all subsets of the output range of mechanism f. When $\delta = 0$, the mechanism f becomes ϵ-deferentially private.

4.4. Normalization Technique

To ensure the confidentiality of user updates, we introduce perturbations to the extracted features within the normalization 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 $P_{i,j,n}$. Although CNN has many channels, a single channel is employed to avoid complexity. We employed $\hat{P}_{i,j,n}$ for $n \in B$ with a mean of 0 and variance of 1 instead of the typical batch normalization, $P_{i,j,n}$, for each batch B.

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

and

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

According to the Cauchy–Schwarz inequality [31] bounds $|B| = M$ and

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

where for any i, j, and 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 [30] is employed. Specifically, a zero-mean Laplace noise with a scale of $2\sqrt{M-1}/\epsilon$ 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 normalizes $\hat{P}_{i,j,n}$ 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

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

varies for image n, the sensitivity in

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

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 the zero-mean Laplace noise, which is a common approach for existing solutions. However, the feature distribution is modeled using a Gaussian distribution, which is widely employed in various real-world applications. The majority of feature values, following the application of batch normalization, 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 normalization strategy, the feature values are distributed more uniformly across the range of $[-\sqrt{M-1}, \sqrt{M-1}]$. Batch normalization approaches are more susceptible to perturbations in feature values compared to an equivalent quantity of the Laplace noise. For instance, consider the case when the batch size is set to $N = 32$ and the scale parameter of the Laplace distribution is given by $2\sqrt{M-1}/\epsilon$. The privacy parameter threshold is calculated for feature values after batch normalization, where noise is calculated as follows:

For batch normalization, we have

$$\frac{2\sqrt{M-1}}{\epsilon} \gg 3\sigma$$

$$\Rightarrow \frac{2\sqrt{M-1}}{\epsilon} \gg \cdot 3$$

$$\Rightarrow \epsilon \ll \frac{11}{3} \approx 3.71.$$

Here, real feature values are perturbed by noise when the privacy parameter $\epsilon \ll 3.71$ is used for batch normalization. Instead, using the following formula

$$\frac{2\sqrt{M-1}}{\epsilon} \gg \sqrt{M-1}$$

$$\Rightarrow \epsilon \approx\ll 2.$$

Our normalization technique generates privacy parameters $\epsilon \ll 2$, the true value to overcome the noise. It is widely acknowledged that a higher privacy parameter corresponds to a reduced level of noise, hence making feature values obtained from batch normalization more susceptible to vulnerability. Therefore, based on the aforementioned example, it can be inferred that the perturbation of features is more pronounced when employing batch normalization compared to our normalization technique. In summary, the use of our normalization technique in the trained model is expected to yield superior test accuracy compared to the use of batch normalization in the training process.

5. Evaluations and Analysis

In two testbeds, we evaluated our proposed blockchain-based FL framework. Initially, we evaluated the blockchain transaction for ledger optimization 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. Implementing a machine learning application in Hyperledger Fabric is hard, so we did it again in a Python environment to test how well the whole ecosystem worked (see Section 5.2 for more information).

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 was 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 the experimental evaluation on hyperledger fabric, we know that every trade is ≈5–10 KB on average, a block is formed with an average of 500 transactions per second, and a block header is 4.5 KB. It expands the ledger at a rate of approximately 50–100 KB/s, 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 10-K home network with 20 devices per home. Figure 4a presents a production environment synopsis for 1 K smart homes in a blockchain network where approximately 15 devices are contained per home. It shows three scenarios where transaction weight may vary from source to source depending on formats. In this experiment, we have considered three different sizes: 5 KB to 7 KB, 8 KB to 9 KB, and 10 KB to 12 KB. The ledger size is proportional to the transaction amount and size. The core impact of GWP has been presented in Figure 4b.
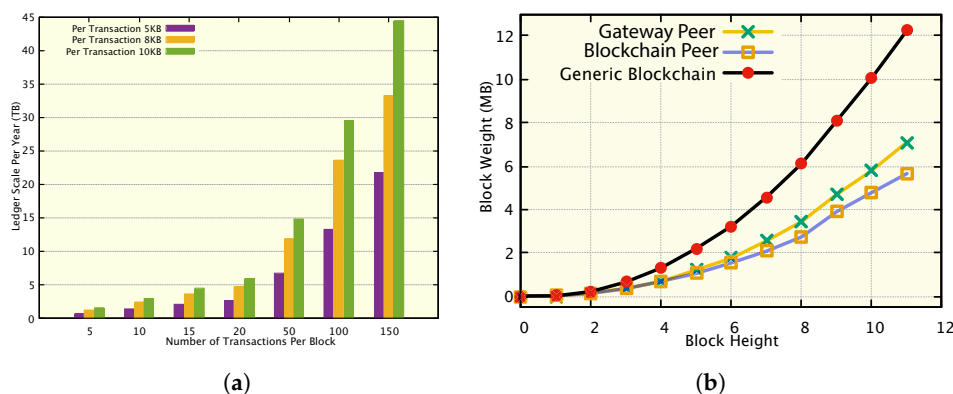

Figure 4. Ledger optimization effect on gateway peer implementation. (**a**) Memory optimization effect, (**b**) ledger scalability in the proposed framework.

Figure 4b presents the ledger optimization 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 recognized public Stanford (CARS) dataset [32], which comprises 16,185 photos representing 196 distinct automobile classes. The dataset comprises 8144 photos for training purposes and 8041 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 re-

spective 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 utilization 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 generalized weighted pooling (GWP) of each individual local organization, also referred to as a learning node. The NVIDIA GeForce RTX 2080 GPU is utilized 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 generalized 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 were implemented using Python 3.8.

The CNN network that we 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 was lowered through the utilization of the max-pooling layer. Hence, the utilization of max-pooling layers enhances the pace of learning in neural networks. Following the normalization 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 regularize 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 using 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 benchmarking. Figure 5 illustrates the training progress and accuracy. For the experiment, we ran 100 rounds to train the model in six federated learning nodes in parallel. Training success is shown in Figure 5a, where the loss declines gradually. Loss decreases firstly in the baseline and typical ML, then in our proposed method. However, they reach a convergence point almost at the end of the same round. Figure 5b demonstrates the learning accuracy through training the model for object detection compared to the baseline and typical approaches. The figure shows that the proposed framework converges with typical approaches almost at the same time.
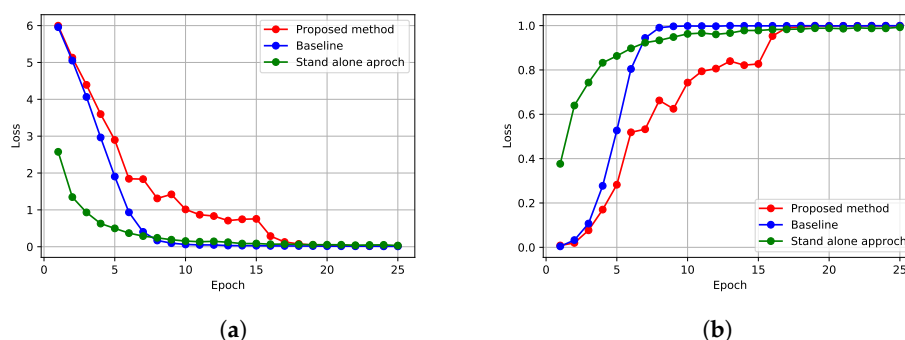

| (a) | (b) |

Figure 5. Training outcomes.(**a**) Learning Progress, (**b**) Learning accuracy.

The classification performance of the proposed system (in comparison to the 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 regard 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 6b. Various global models, derived by averaging the FL models, were employed to assess their efficacy in real-world situations. The presented data illustrate 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.
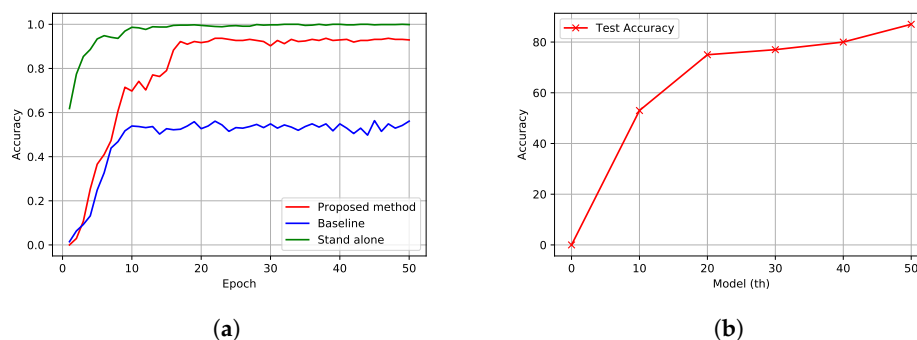

(a)	(b)

Figure 6. Validation and test evaluations. (**a**) Validation accuracy, (**b**) test accuracy.

5.4. Application Challenges and Future Direction

The implementation of distributed learning and information-sharing components can extend to several sectors beyond the realm of smart homes. In several application fields, such as e-healthcare and the interoperability of fintech firms, the utilization of this concept can be employed for the purpose of secure knowledge transfer, as opposed to the sharing of raw data. The presence of security measures may have an impact on the willingness of participants to join a collaboration aimed at constructing a diverse Big Data repository, a crucial component for conducting research in machine learning. The challenge is to align diverse and various structured data to make them suitable for ML processes. Automatic feature selection and alignment from various data sources can help build quality and diverse data, which can influence the learning process of ML more accurately in any specific application domain.

6. Conclusions

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 updated management. Blockchain is being used to address 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 results of the testbed show that the contribution solves important issues that might arise when HAN is combined 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 reducing ledger overhead by more than 60% compared to conventional procedures. It devises a workable and secure method of dealing with the continuous data provided by smart homes.

References

1. Caviglione, L.; Wendzel, S.; Mazurczyk, W. The Future of Digital Forensics: Challenges and the Road Ahead. *IEEE Secur. Priv.* **2017**, *15*, 12–17. [CrossRef]
2. Holst, A. Smart Home Report 2021. Available online: https://www.statista.com/topics/2430/smart-homes/ (accessed on 22 November 2023).
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 Trans. Consum. Electron.* **2016**, *62*, 1–9. [CrossRef]
4. Thantharate, P.; Thantharate, A. ZeroTrustBlock: Enhancing Security, Privacy, and Interoperability of Sensitive Data through ZeroTrust Permissioned Blockchain. *Big Data Cogn. Comput.* **2023**, *7*, 165. [CrossRef]
5. Gonzalez-Amarillo, C.; Cardenas-Garcia, C.; Mendoza-Moreno, M.; Ramirez-Gonzalez, G.; Corrales, J.C. Blockchain-IoT Sensor (BIoTS): A Solution to IoT-Ecosystems Security Issues. *Sensors* **2021**, *21*, 4388. [CrossRef] [PubMed]
6. Irwin, D.; Albrecht, J. Smart Homes: Implemented. *IEEE Pervasive Comput.* **2019**, *18*, 91–95. [CrossRef]
7. Mukherjee, A.; Balachandra, M.; Pujari, C.; Tiwari, S.; Nayar, A.; Payyavula, S.R. Unified smart home resource access along with authentication using Blockchain technology. *Glob. Transitions Proc.* **2021**, *2*, 29–34. [CrossRef]
8. Yang, J.; Zou, H.; Jiang, H.; Xie, L. Device-Free Occupant Activity Sensing Using WiFi-Enabled IoT Devices for Smart Homes. *IEEE Internet Things J.* **2018**, *5*, 3991–4002. [CrossRef]
9. Wu, Q.; Chen, X.; Zhou, Z.; Zhang, J. FedHome: Cloud-Edge Based Personalized Federated Learning for In-Home Health Monitoring. *IEEE Trans. Mob. Comput.* **2022**, *21*, 2818–2832. [CrossRef]
10. Jung, S.S.; Lee, S.J.; Euom, I.C. Delegation-Based Personal Data Processing Request Notarization Framework for GDPR Based on Private Blockchain. *Appl. Sci.* **2021**, *11*, 10574. [CrossRef]
11. 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 Trans. Eng. Manag.* **2020**, *67*, 1363–1376. [CrossRef]
12. 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. [CrossRef]
13. Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaram, P. Blockchain for IoT security and privacy: The case study of a smart home. In Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kona, HI, USA, 13–17 March 2017; pp. 618–623. [CrossRef]
14. 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 Things J.* **2020**, *7*, 818–829. [CrossRef]
15. 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 Netw.* **2021**, *35*, 223–229. [CrossRef]
16. Kim, D. A Reverse Sequence Hash Chain-based Access Control for a Smart Home System. In Proceedings of the 2020 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 4–6 January 2020; pp. 1–4. [CrossRef]
17. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.Y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; Proceedings of Machine Learning Research; Volume 54, pp. 1273–1282.

18. Ramanan, P.; Nakayama, K. BAFFLE: Blockchain Based Aggregator Free Federated Learning. In Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain), Rhodes, Greece, 2–6 November 2020; pp. 72–81. [CrossRef]

19. Sun, J.; Wu, Y.; Wang, S.; Fu, Y.; Chang, X. Permissioned Blockchain Frame for Secure Federated Learning. *IEEE Commun. Lett.* **2022**, *26*, 13–17. [CrossRef]

20. Otoum, S.; Ridhawi, I.A.; Mouftah, H. Securing Critical IoT Infrastructures With Blockchain-Supported Federated Learning. *IEEE Internet Things J.* **2022**, *9*, 2592–2601. [CrossRef]

21. 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 Things J.* **2021**, *8*, 5926–5937. [CrossRef]

22. 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 Things J.* **2021**, *8*, 2265–2275. [CrossRef]

23. 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 Things J.* **2022**, *9*, 18378–18391. [CrossRef]

24. Ammi, M.; Alarabi, S.; Benkhelifa, E. Customized blockchain-based architecture for secure smart home for lightweight IoT. *Inf. Process. Manag.* **2021**, *58*, 102482. [CrossRef]

25. 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. [CrossRef]

26. Alghamdi, A.; Zhu, J.; Yin, G.; Shorfuzzaman, M.; Alsufyani, N.; Alyami, S.; Biswas, S. Blockchain Empowered Federated Learning Ecosystem for Securing Consumer IoT Features Analysis. *Sensors* **2022**, *22*, 6786. [CrossRef] [PubMed]

27. Biswas, S.; Sharif, K.; Li, F.; Nour, B.; Wang, Y. A Scalable Blockchain Framework for Secure Transactions in IoT. *IEEE Internet Things J.* **2019**, *6*, 4650–4659. [CrossRef]

28. Pappas, C.; Chatzopoulos, D.; Lalis, S.; Vavalis, M. IPLS: A Framework for Decentralized Federated Learning. In Proceedings of the 2021 IFIP Networking Conference (IFIP Networking), Espoo, Finland, 21–24 June 2021; pp. 1–6. [CrossRef]

29. Erlingsson, U.; Pihur, V.; Korolova, A. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, 3–7 November 2014; Association for Computing Machinery: New York, NY, USA, 2014; pp. 1054–1067. [CrossRef]

30. Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, 4–7 March 2006*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 265–284. [CrossRef]

31. Jiang, L.; Lou, X.; Tan, R.; Zhao, J. Differentially Private Collaborative Learning for the IoT Edge. In Proceedings of the Wireless Systems and Networks, EWSN '19, Beijing, China, 25–27 February 2019.

32. Krause, J.; Stark, M.; Deng, J.; Fei-Fei, L. 3D Object Representations for Fine-Grained Categorization. In Proceedings of the 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney, NSW, Australia, 2–8 December 2013.