Blockchain Empowered Secure Federated Learning for Consumer IoT Applications in Cloud-Edge Collaborative Environment

Mohit Kumar, Jitendra Kumar Samriya, *Member, IEEE*, Guneet Kaur Walia, *Graduate Student Member, IEEE*, Prabal Verma, *Member, IEEE*, Huaming Wu, *Senior Member, IEEE*, and Sukhpal Singh Gill

***Abstract*—The growing number of consumer Internet of Things (IoT) gadgets, including smart homes, fitness trackers, connected appliances, and home security systems, is transforming the way we live our daily lives. This has led to the emergence of a collaborative cloud-edge paradigm to leverage resources and services near the end-user, thereby providing prompt response to delay-sensitive real-time applications. Nevertheless, the tremendous amount of data generated by various IoT devices and sent over the network is always an open security challenge. The introduction of Federated Learning (FL) addresses the security and data privacy shortcomings of traditional centralized machine learning. Despite FL's use for data privacy, it must overcome a number of significant challenges, such as privacy concerns, communication overhead, stragglers, and heterogeneity. To solve these challenges, this paper proposes a novel technique for enhancing security in IoT-enabled edge cloud computing networks, utilizing blockchain-driven FL and Gaussian Bayesian transfer convolutional neural network architectures for data analysis. Blockchain-driven FL ensures the security and privacy of consumer IoT applications. In comparison to state-of-the-art works, the experimental results achieved throughput of up to 89%, latency of 71%, training accuracy of 91%, validation accuracy of 96%, and network security of 92%.**

***Index Terms*—Consumer IoT, edge computing, cloud computing, blockchain, federated learning.**

I. Introduction

THE PROLIFERATION of the Internet of Things (IoT) paradigm, coupled with advanced technologies such as Artificial Intelligence (AI) has significantly paved the way towards more sophisticated Next Generation Networks (NGNs) [1]. Consumer IoT gadgets, such as smart homes, fitness trackers, connected appliances, and home security systems, are transforming our daily lives. However, the IoT workload is characterized by low latency and high responsiveness [2]. Utilizing the centralized processing, analysis and storage capabilities of cloud datacenters for serving geographically distributed IoT devices leads to increased response time. As a result, the collaborative cloud-edge paradigm is an optimal choice because it leverages resources and services in close proximity to the end-user, providing prompt response and improvising overall network bandwidth [3]. The significance of IoT devices has increased, contributing to the development of new consumer applications that create intelligent environments for people, enhancing their quality of life [4]. Nevertheless, the security of IoT devices and data over networks remains vulnerable due to a myriad of modern attacks and the inherent heterogeneity within the IoT landscape [5]. In addition, the integration of AI in centralized cloud data centers gives rise to significant challenges such as privacy and data leakage as models are deployed on centralized cloud data centers using data aggregated from numerous IoT devices. A distributed learning model known as Federated Learning (FL) provides hyper-personalized space, ensuring data localization with less dependency on cloud infrastructure. In FL, an initialized global model is transmitted by the server to the end devices, which subsequently utilize their local data for training their models [6]. Each device computes the weights and sends them back to the Mobile Edge Servers (MES) within a single epoch. The server then receives the trained local models. This iterative process continues until the training accuracy of the global model satisfies the server's criteria [7]. Hence, incorporating FL into the collaborative cloud-edge paradigm retains data privacy, optimizes bandwidth utilization and improves model accuracy [8].

However, there are still some challenges with this integration, such as privacy concerns, communication overhead, stragglers, heterogeneity, etc. For example, it's not easy to keep track of model updates; it's open to security attacks like adversarial, model inversion, or membership inference; the integrity of the data has been compromised, and there are

Received 3 September 2024; revised 20 December 2024; accepted 19 January 2025. Date of publication 22 January 2025; date of current version 14 August 2025. This work was supported by the National Natural Science Foundation of China under Grant 62071327. *(Corresponding author: Huaming Wu.)*

Mohit Kumar and Guneet Kaur Walia are with the Department of Information Technology, National Institute of Technology Jalandhar, Jalandhar 144011, India (e-mail: kumarmohit@nitj.ac.in; guneetkw.it.22@nitj.ac.in).

Jitendra Kumar Samriya is with the Computer Science and Engineering, Indian Institute of Information Technology Sonepat, Sonipat 131001, India (e-mail: jitu.samriya@gmail.com).

Prabal Verma is with the Department of Information Technology, National Institute of Technology Srinagar, Srinagar 190006, India (e-mail: prabalverma357@gmail.com).

Huaming Wu is with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China (e-mail: whming@tju.edu.cn).

Sukhpal Singh Gill is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, E1 4NS London, U.K. (e-mail: s.s.gill@qmul.ac.uk).

Digital Object Identifier 10.1109/TCE.2025.3532676

trust issues [9]. Therefore, it becomes crucial to have a secure and robust solution in order to avoid malicious trust estimates and cope with attacks tailored toward trust management. Privacy concerns pertain to the risk of exposing sensitive data during model training [10]. To reduce the risk of data leakage, our suggested approach incorporates cutting-edge privacy-preserving methods like differential privacy and secure multi-party computation. Additionally, we addressed effective aggregate procedures to enhance the Quality of Service (QoS) parameters [11]. Communication overhead involves the cost of transmitting model updates across the network. Stragglers are slow or delayed participants that can hinder training efficiency. Heterogeneity Clients' data distribution and system capabilities vary, raising concerns. Blockchain has been introduced to enable distributed, immutable, transparent, verifiable, and decentralized solutions by empowering consumer IoT devices to execute workloads collaboratively [12]. Blockchain when integrated with FL, balances the workload across IoT devices and overcomes the above-mentioned issues [13]. It can implement trusted distributed authentication and authorization for devices belonging to specific IoT use cases. Apart from this, it ensures data reliability by facilitating data traceability and accountability for tracking billions of IoT devices, transaction processes, and intra-device coordination [14]. Above all, it guarantees the exchange of messages as transactions, validated by smart contracts. Furthermore, we manage data access using the following principles: Some of the most resilient encryption standards support an additional layer of security that intruders must get around [15]. These standards ensure the secure storage of locally trained data from various transactions. Concerns about single-point failures are alleviated, as there is no centralized authority. It provides a secure platform for IoT consumer devices, fostering a high level of trust because the majority of network participants must reach a consensus to validate each device's transactions. The main contributions of this research article are:

- We propose a decentralized and distributed solution to the data-sharing challenges of machine learning models, safeguarding data privacy in critical real-time consumer IoT use cases.
- We design a secure blockchain-empowered FL model that helps to mitigate potential attacks such as malware and ransomware, device spoofing, Denial of Service (DoS), data integrity and unauthorized access, data privacy and eavesdropping.
- We develop a novel framework for securing network-based data analysis using a new Gaussian Bayesian transfer convolutional neural network method.
- We evaluate the effectiveness of the proposed work with benchmark real-world datasets and it achieves superior performance compared with state-of-the-art works at influential parameters.

The rest of this paper is organized as follows: Section II presents a brief review of state-of-the-art approaches. Section III discusses a system framework model and the proposed technique. Section IV presents the illustrative experimental results of the proposed technique. Finally, Section V concludes this paper.

II. Related Work

Machine learning models play a major role in IoT security to detect illegitimate users and monitor the anonymous behavior in the network, but using a centralized system doesn't serve as an optimal solution [24]. Therefore, the FL approach, which updates the model rather than the user's data, addresses these issues. The server side performs the aggregation based on the received weight and bias, ensuring the privacy of end users' data. For the participants who want to train a cooperative model, FL offers a promising solution with synchronous and asynchronous approaches. By learning from dispersed data, FL delivers edge intelligence. Federated Averaging (FedAvg) is one of the most widely used algorithms in FL [25] which allows for distributed machine learning where data remains local on client devices, and the model is trained collaboratively across multiple devices. The FedAvg algorithm works by performing local training on each client device, and then aggregating the model updates (using a weighted average) on a central server. Prominent industries have widely adopted the collaborative concept, and in FL, it can reduce the privacy risk associated with direct data exchange [26]. However, FL faces some security challenges especially modern attacks that make it difficult to track and store the data.

Blockchain is a decentralized and distributed ledger technology that provides secure and transparent services, where traditional data-centric failed. The developed approach integrates blockchain and FL for collaborative mode training at local sites, ensuring data privacy for healthcare applications [16]. In the healthcare domain, the authors have used the FL-driven blockchain approach to secure patent records. Blockchain technology offers the smart contracts concept between the users and sensors to enable the transfer of healthcare records over the networks [17]. Furthermore, FL provides collaborative training and learning mechanisms that facilitate coordination with multiple mobile edge devices, thereby enhancing network privacy and preventing any potential data leakage [18]. Blockchain incorporates decentralization and transparency into the network, overcoming the limitations of the central server and ensuring data security. The authors depict the need for blockchain in AI-enabled edge networks and integrate blockchain with AI over the edge network for IoT applications [19]. Several challenges in terms of energy efficiency, model optimization, data administration and existing solutions along with their limitations have been discussed by the authors in the article. The aggregation process in federated learning suffers from Byzantine attacks and other security issues where attackers try to modify the weights of the model. A convolutional Kernel-based defense aggregation (CKADA) approach has been proposed by authors that use the angle between convolutional kernels to avoid the mentioned attacks and enhance the performance in terms of accuracy and loss parameters [20]. Interpretability and computational constraints are major issues with traditional approaches due to which data privacy and security are compromised in cyber-physical systems. The authors have proposed an Interpretation-based Privacy-Preserving FL technique (IP2FL) that mitigates the mentioned privacy issues by integrating IP2FL with Additive

TABLE I
COMPARISON OF OUR PROPOSED WORK WITH EXISTING STUDIES

Work	Techniques	Aims and Objectives	Performance	Limitations
[16]	Blockchain empowered FedAvg algorithm	To store and process the patient data securely and transparently	Enhance accuracy, precision, recall, and F1-score	Communication and computational costs are high
[17]	Blockchain-based FL	To ensure scalability, trust, and security of patient data	Mean Square Error and peak signal-to-noise ratio is improved	Latency is an issue with high-volume data
[18]	Cognitive trust management approach	To adopt the program for cross-authentication mechanism	The normal decryption and authenticating times are 18 and 10 milliseconds, respectively	Only Denial of Service (DoS) attack is considered and there is no real implementation to enhance security
[19]	Integrating Blockchain with Edge Intelligence (EI)	To optimize computing power management, and model optimization	Discuss the role of blockchain and EI in real-time IoT applications	Not Available
[20]	CKADA approach	To avoid Byzantine attacks and other security issues	Enhance the accuracy and loss parameters	Communication cost and other significant parameters are not considered to measure the training performance
[21]	Secure data transmission techniques	To identify and mitigate DoS, reply and master key attacks	Enhance confidentiality, integrity and authentication	Neglect the latency as security enhances
[22]	Convolutional neural network with transformer encoder	To improve the Mean and median localization error	The cutting edge technologies can be used to enhance it	The performance of proposed work can vary based upon dataset size and sample size
[23]	Blockchain-based FL for Consumer IoT	To enhance the accuracy and time	Validation and training accuracy is improved	Computational complexity, computational cost and latency are need to be optimized.
This work	Gaussian Bayesian transfer convolutional neural network approach, Blockchain and FL	To propose a decentralized framework for securing network-based data analysis in edge-cloud paradigm for real-time consumer IoT applications	Improved throughput, latency and accuracy and enhanced the network security and robustness	Computational complexity need to be optimized for a large data to train the model.

Homomorphic Encryption (AHE) to reduce the overhead and guarantee the privacy of data [27]. IoT data is transmitted through open channels that attract attackers to gain the confidential information of end users and compromise authentication, integrity and confidentiality. Hence, an enhanced secure mechanism is needed to address the mentioned challenges in IoT environments where the security-based algorithm is implemented over an edge server. The authors have proposed a secure approach for smart city applications where IoT devices and base station prove their legitimacy before transmitting the data [21]. The communication between the sensor and receiver is established in three phases to evade the user anonymity, and forging attacks.

A hybrid Convolutional Neural Network (CNN)-based approach has been developed by authors to localize the node using features and enhance the influential parameters like mean errors, median and efficiency in LoRaWAN networks [22]. Trust, transparency and immutability are two major issues with federated learning-based solutions for IoT applications. The authors have proposed a blockchain-enabled FL-based solution that transparently provides the security of data and replaces the centralized aggregator leading to enhanced security as well as trustability [23]. The proposed work uses an edge server to update the models and avoid the possibility of data leakage in the gateway. Blockchain technology uses a consensus process to enhance the security in model sharing for consumer IoT applications. Blockchain technology is a developing platform that provides services and opportunities for traditional data-centric networks [28]. Another research [29] provided in-depth analyses of cloud-edge collaborative architecture, emphasizing collaborative learning mechanisms such as pre-training models, graph neural networks, and reinforcement learning. However, they did not address the promise of decentralized FL in collaborative architecture. Numerous studies have explored FL in an edge computing context, primarily focusing on edge computing-enabled methodologies and ignoring system entity collaboration.

Security and privacy are major concerns with AI-enabled IoT devices, as they use the centralized model to process the data. The authors have proposed a Digital Twin (DT) based on automated consumer electronic devices that send only weights and bais to train the model using FL [30]. The proposed approach utilizes blockchain technology to securely store data after each training cycle, resulting in enhanced performance metrics, including accuracy, precision, and recall. In another research work [31], the authors discussed and addressed the three primary concerns of FL at mobile edge networks: communication cost, optimal resource scheduling, privacy, and security of IoT applications. This work also established the foundational elements and distinctive qualities of FL. In [32], authors review the security and privacy issues associated with FL systems. Additionally, it provides several functions commonly used in cellular and ad hoc networks, including resource management and content caching. The authors employ the Mobile Edge Computing (MEC) technique to address the limitations of the existing radio access network, utilizing the cellular network edge [33]. The authors suggest that MEC provides a collaborative, real-time, context-aware framework that interacts with an underlying communication network. The authors suggested a two-tier compute offloading system to cut down on network latency using MEC. This would improve energy efficiency or power consumption in heterogeneous networks. The authors also suggested making dynamic offloading decisions based on the anticipated IoT workload in the collaborative edge cloud to reduce task time. They also took into account the battery capacity of User Equipment (UE) in software to create an ultra-dense network, utilizing the concept of a software-defined network. Table I compares our proposed work with existing studies based on techniques, aims and objectives, performance, and limitations to demonstrate its uniqueness and novelty.

III. SYSTEM MODEL

The proposed model aims to improve consumer IoT applications' security in an integrated cloud-edge computing paradigm using a blockchain-driven FL architecture by analyzing network data, as depicted in Fig. 1. The workflow originates from the IoT layer, which constitutes various use cases such as autonomous vehicles, predictive maintenance, Unmanned Aerial Vehicles (UAVs), smart healthcare, maritime engineering, smart homes and Augmented Reality (AR)/Virtual Reality (VR) applications. Massive amounts of data are generated by consumer IoT devices, for which ensuring privacy and safeguarding security becomes of paramount importance, such that sensitive or confidential information remains protected from unauthorized access or breaches.

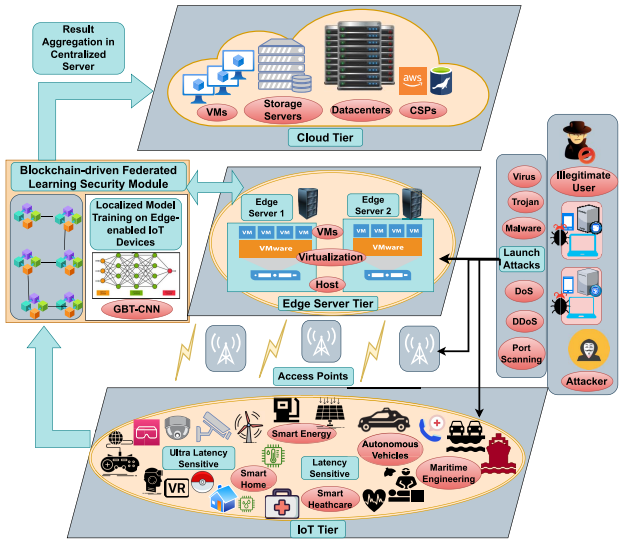

Fig. 1. Proposed Framework for Blockchain enabled FL Cloud Edge Environment.

Furthermore, we assume that all these devices possess the capability to perform model training using device local data. These devices need to possess computational capabilities for training and developing localized models, for which edge servers are proposed to cater to the resource-constrained nature of consumer IoT devices. Finally, the global model is attained by aggregating the results on a centralized cloud server.

The goal of this work is to address the security concerns of 4GNet [34]. Attackers inside or outside 4GNet may use wireless or wired channels in the backbone or access networks to launch attacks like DoS, port scanning, data integrity and unauthorized access, malware distribution attacks, etc. Three distinct tiers within 4GNet disperse the nodes to detect attacks: cloud, edge, and IoT devices. Various layers cooperate to improve the attack detection model. Within the FL system, each participant shares the same identity and status. We retain all parties' data locally, ensuring privacy and legal compliance. Apart from this, transfer learning offers a mechanism for knowledge migration, even in cases where users or features lack alignment, by transferring cryptographic parameters across datasets. FL addresses the challenge of data silos, fostering collaboration between two or more entities that utilize data.

FL is one of the machine learning paradigms where multiple parties, such as devices or organizations, collaboratively train a global model while keeping their local data private. Rather than centralizing data in one location, FL keeps data decentralized and ensures it remains on the participants' devices or nodes [25]. However, this decentralized approach introduces several security and privacy concerns, such as data privacy, model poisoning, byzantine faults and trust issues [30]. Blockchain technology has been introduced to overcome the mentioned issues with its decentralized, immutable, and transparent characteristics that can help mitigate these issues and enhance the security and robustness of federated learning systems [23]. Blockchain addresses the challenges of FL, particularly in securing data from modern attacks by enabling a distributed and decentralized solution. The cryptographic features of blockchain provide a guaranteed solution to achieve data consistency, secure transmission, and data storage. Every node connects to the hash of the previous node, ensuring that no one can alter the information. In addition, blockchain can monitor new devices that have joined the network and validate authentication and authorization based on transaction records [16]. This mechanism improves secure communication between devices and provides more transparent transactions, as well as validations in the network. This collaboration occurs seamlessly without the need for data to traverse beyond local boundaries, thereby enhancing efficiency and mitigating concerns associated with centralized data sharing. Furthermore, we suggest a mechanism for evaluating each participant's trustworthiness based on the assumption that nodes will use blockchain to validate a locally trained model. Therefore, only after reaching a consensus does the cloud receive an updated global model.

Blockchain enhances the security, privacy and reliability of FL by addressing critical vulnerabilities that threaten its effectiveness in decentralized environments [23]. Attacks like byzantine faults, where malicious participants send corrupted updates and model poisoning, where adversaries inject backdoors into the global model, are mitigated through blockchain's decentralized consensus mechanisms, such as Practical Byzantine Fault Tolerance (PBFT) or Proof-of-Stake (PoS), ensuring that only valid updates contribute to the model. Similarly, data poisoning attacks, which degrade model quality by manipulating local datasets, are countered through blockchain's immutable ledger, which maintains a secure and auditable record of data provenance. Privacy threats in FL, such as gradient inversion attacks (reconstructing private data from shared gradients) and membership inference attacks (determining if specific data was used in training), are mitigated by integrating blockchain with advanced privacy-preserving techniques like homomorphic encryption, secure multi-party computation and differential privacy, ensuring gradients are securely aggregated without exposing sensitive information [16]. Blockchain also eliminates the risk of single points of failure in centralized systems by decentralizing aggregation, making FL systems fault-tolerant and resilient to targeted attacks. Moreover, blockchain's transparent and auditable framework enhances accountability, deterring inference attacks and ensuring secure management of model updates and contributions [12]. The convergence between blockchain and FL creates a robust, privacy-preserving and scalable solution suitable for deployment in adversarial and untrusted settings, such as healthcare, IoT and financial applications. Therefore, addressing security and trust challenges, blockchain strengthens FL's potential for real-world adoption.

In this paper, we propose a novel blockchain-driven FL framework that prioritizes data privacy during model training, while blockchain manages integrity, traceability, and decentralization at the network's edge to guarantee data integrity, transparency, and decentralization for real-time consumer IoT use cases. In addition, the proposed work helps to mitigate potential attacks such as malware and ransomware, device spoofing, DoS, data integrity and unauthorized access, data

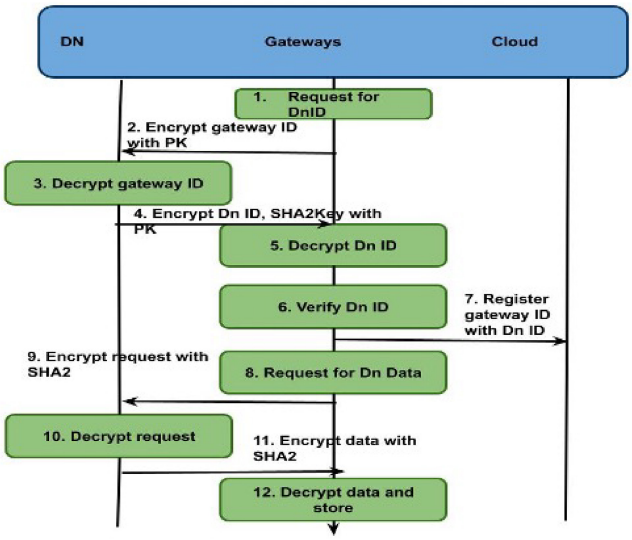

Fig. 2. Robust and Secure Connection in IoT ecosystem.

privacy and eavesdropping. It also secures network-based data analysis using a novel Gaussian Bayesian transfer convolutional neural network approach.

A. System Model Workflow

The cloud layer stores the gateway IDs and blockchain information associated with each gateway, ensuring universal and continuous access for end users. In any situation, the exchange of blocks facilitates seamless knowledge availability. Each device is characterised by a unique identification (ID) that distinguishes it from other entities within the system. In addition, the devices possess computational capabilities, which assist them in carrying out Public Key Infrastructure (PKI) and Secure Hash Algorithm (SHA2) encoding and decoding operations. Devices and gateways' interconnection operations include certificate registration and data storage.

In Fig. 2, the gateway acts as an intermediary between IoT devices and the cloud, where Blockchain helps store gateway IDs (unique identifiers for gateways) in the cloud in a secure, tamper-proof manner. These IDs are linked to the blockchain to ensure that only authorized gateways can interact with the IoT devices. The device can be issued a unique cryptographic identity (often in the form of public/private keys or digital certificates) using blockchain that is stored in a distributed ledger. The blockchain records every authentication event and device interaction in a secure, immutable ledger. Once a device's identity is registered on the blockchain, it cannot be altered, ensuring that the device's identity remains verifiable over time. This makes it harder for malicious actors to spoof or impersonate devices. The cloud layer may store essential blockchain-related information, such as transaction records, device identities, and logs of interactions between IoT devices and gateways. This ensures that the authentication data remains available to users and IoT devices even if some of the devices go offline or the blockchain network experiences failures. The entire workflow process is discussed below:

- *Routine Validation at Gateway:* Devices authorized by the gateway consistently undergo routine validation. At the IoT tier, devices D_n initiates sign-up or login attempts, connecting to the gateway instantly. Then, the gateway prompts for a linked device ID or requests information about the connected devices.
- *Cryptographic Encryption by Device Gateway:* The device gateway employs cryptographic technology to encrypt device data, which is then sent to D_n, thus establishing a secure foundation for further communication.
- *Decoding Unregistered or Non-Encrypted Gateway Messages:* Encrypted messages received via an unregistered or non-encrypted gateway are decoded and requested.
- *Service Request to Gateway at IoT Tier:* At the IoT tier, the request is sent for services to the gateway that includes Device ID and SHA2 key to communicate securely between devices and the gateway.
- *Validation of Information Received:* The information received in the form of an encrypted packet is decrypted and verified to check if it is coming from a legitimate user or not.
- *Registration of Legitimate Devices:* After successful validation, legitimate devices are registered over the cloud platform and become eligible to use the services from the cloud.
- *Robust and Secure Communication:* Gateways communicate as per requirement with the cloud to update the information like ID lists, and other attributes in the incoming packet, hence, enabling robust and secure communication.

After validation, the device sends the data in encrypted form to a gateway which stores and processes it after decryption using a public key algorithm. The secrecy capacity is defined as the maximum achievable data transmission rate at which secure communication can occur, ensuring that unauthorized parties are unable to decipher or access the transmitted information. This rate represents the highest level of data throughput that maintains confidentiality and prevents eavesdroppers from obtaining any useful information from the communication channel [35], [36].

The hash key (H) of the i^{th} layer can be calculated by: $\psi(H_{i-1}, p_i, p_{i+1})$, where ψ stands for an appropriate hashing method, such as SHA256, and p stands for the layer's settings. A ledger block is used to keep the hash of all the layers. For security purposes, we use the block to monitor and test the compromised and tempered layers [35], [36]. To determine the order of ledger block input, we selected a random number between 1 and the entire number of layers. The layer signature reflects the layer's private key. The complete model examines each updated ledger block to determine whether the layer above it has approved the update or not.

B. Gaussian Bayesian Transfer Convolutional Neural Network

The mixture density is expressed as a weighted sum of k components, where the density of the j^{th} component is

represented by $p(x; \theta_j)$, with θ_j being the component-specific parameters. The function $p(x)$ represents the probability that a given data sample belongs to the j^{th} mixture component. The component mixture density can be defined as follows:

$$p(x) = \sum_{j=1}^{K} \pi_j p(x; \theta_j), \tag{1}$$

$$p(x) = \sum_{c=1}^{C} \pi_c f_c(x \mid \theta). \tag{2}$$

A vector of parameters, $\underline{\theta} = \{\theta_1, \ldots, \theta_k, \pi_1, \ldots, \pi_k\}$, is represented by a Mixture model, where Z is a hidden variable. There are K discrete sets that accept the value 1 based upon the condition $z_k \in \{0, 1\}$ and $\sum_z z_k = 1$.

$$p(z, x) = p(z)p(x \mid z), \tag{3}$$

where $p(x|z)$ is a conditional and marginal distribution from a multinomial distribution that relies on z.

Mixing coefficients π_k are used to specify the marginal distribution across z, illustrated as follows:

$$p(z_k = 1) = \pi_k. \tag{4}$$

The probability density functions of X can be defined as follows:

$$p(x \mid \mu_k, \Sigma_k) = \frac{1}{\sqrt{2\pi |\Sigma^{-1}|}} e^{-\frac{1}{2}(x-\mu_x)^T \Sigma_x^{-1} (x-\mu_x)^T}, \tag{5}$$

$$f_c(x \mid \mu_c, \Sigma_c) = \frac{1}{(2\pi)^{\frac{1}{7}} |\Sigma_c|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_c)^T \Sigma_c^{-1} (x-\mu_c)} \tag{6}$$

where μ_x is a vector of means, $(\mu_{x1}, \ldots, \mu_{xN})$, and Σ_x is an $N \times N$ covariance matrix.

The linear superpositions of Gaussians, can be used to represent a Gaussian mixture distribution:

$$p(x) = \sum_{k=1}^{K} \pi_k p(x \mid \mu_k, \Sigma_k) \tag{7}$$

$$\hat{\pi}_c = \frac{n_c}{n} \tag{8}$$

$$\dot{\mu}_c = \frac{1}{n_c} \sum_{(i, y_i = c_j)} x_i, p(x \mid z_k = 1) = p(x \mid \mu_k, \Sigma_k) \tag{9}$$

$$\hat{\Sigma}_c = \frac{1}{n_c - 1} \sum_{(i \mid y_i = c)} (x_i - \mu_c)(x_i - \mu_c)^t \tag{10}$$

where x is modeled as a Gaussian distribution for a specific value of z, and the probability density function is given by: $p(x \mid z) = \prod_{k=1}^{K} p(x \mid \mu_k, \Sigma_k)^{z_k}$.

The marginal distribution of x is determined by adding the all-possible states of z, given as follows:

$$p(x) = \sum_z p(z)p(x \mid z) = \sum_{k=1}^{K} \pi_k p(x \mid \mu_k, \Sigma_k). \tag{11}$$

For specific data vectors, the "posterior probability" is represented by:

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n \mid \mu_j, \Sigma_j)}, = \frac{p(z_k = 1)p(x \mid z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(x \mid z_j = 1)}. \tag{12}$$

Set the means μ_k, covariances Σ_k, and mixing coefficients π_k to their starting values, then calculate the logarithmic likelihood. To evaluate the duties, we can use the current parameter values as follows:

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n \mid \mu_j, \Sigma_j)}. \tag{13}$$

We utilize the current responsibility equation to re-estimate the parameters as follows:

$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n, \tag{14}$$

$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \left(x_n - \mu_k^{\text{new}}\right)\left(x_n - \mu_k^{\text{new}}\right)^T. \tag{15}$$

The training phase, utilizing Bayesian neural networks, requires posterior assumptions, which represent the probabilistic representation of uncertainty about the true values of the model. However, the accurate inference of the model posterior poses a computational challenge and hence becomes impractical, especially for moderately large models. Therefore, the model posterior is typically estimated. Variational inference is an efficient and well-liked approximation technique. The function $f(X) = y$ estimates the output y from the inputs X given the input set $X = \{x_1, x_2, \ldots, x_N\}$ and a matching output set $y = \{y_1, y_2, \ldots, y_N\}$. Using Bayesian learning, one can extract the model posterior $p(f|X, y)$ in a principled manner. The posterior can only be calculated using two components. A prior distribution $p(f)$, which reflects a previous belief, first represents the estimator functions. In addition, a probability function $p(y|f, X)$ is provided to show how likely it is for the model f to correctly anticipate the output y in light of the observations X. More specifically, the posterior is produced from an unknown set of data (x, y) by integrating over all feasible estimator functions (f), which are parametric models with parameters θ determined by:

$$p\left(y^* \mid x^*, X, y\right) = \int p\left(y^* \mid f\right)\, p\left(f \mid x^*, X, y\right)\, df = \int p\left(y^* \mid f\right)\, p\left(f \mid x^*, \theta\right) p(\theta \mid X, y)\, dfd\theta. \tag{16}$$

The log evidence lower bound with regard to the parameter set θ is maximized when the aforementioned KL divergence is minimized, according to:

$$\text{KL}_{\text{V1}} = \int q(\theta) p(F \mid X, \theta) \log_p(y \mid F)\, dFd\theta - \text{KL}(q(\theta) \| p(\theta)). \tag{17}$$

The variational function that is produced by maximizing KLVI closely resembles the posterior. Using the approximation $q(\theta)$, the following formula can be simplified:

$$q\left(y^* \mid x^*\right) = \int p\left(y^* \mid f\right)\, p\left(f \mid x^*, \theta\right) q(\theta)\, df\, d\theta. \tag{18}$$

The network samples the network parameters θ from $q(\theta)$ when performing inference. The stage l feature extraction module, denoted as $g(l)$, extracts the features $H(l)$ as specified by:

$$H^{(n)} = g^{(l)}\left(H^{(l-1)}; W^{(l)}, b^{(l)}\right) = \text{normalize}\left(\text{pool}\left(\text{ReLu}\left(W^{(l)} \cdot H^{(l-1)} + b^{(l)}\right)\right)\right). \quad (19)$$

where the operator denotes convolution. A few more convoluted networks make up the fully connected layers, which are considered an extracting features layer. The size of the network will affect the size of the resulting feature map. After the scaling factor has been combined and transmitted, the length of the feature space is determined using the equation as follows:

$$N'_x = \frac{N_x^{(l-1)} - K'_x + 2P'_x}{S'_x}, \quad (20)$$

$$N'_y = \frac{N'_y - K'_y + 2P'_y}{S'_y}, \quad (21)$$

where K is the size of the ConvNet, S is the scale parameter, and P is the fill total number of pixels. After the pooling technique, there is a quadratic modification in the kernel function. Cross-validation is used to determine the masses and properties of each layer, and we yield the equation of a connected neuron as follows:

$$x'_j = \text{Relu}\left(\sum_{i \in M_j} x^{(l-1)} w'_{ij} + b'_j\right). \quad (22)$$

where M is the filtering diameter, w and b represent the connecting load as well as distance, respectively. The pooling layer, often positioned between two fully connected layers, is used for network segmentation. Optimum and median aggregating methods are the two varieties. In order to decrease the number of layers without maintaining the consistency of the characteristics, evaluations of certain attributes in the input neurons are assessed and combined. This is accomplished by using a quantization phase to decrease the variance of the converted data. The calculation for the convolutional is provided as follows:

$$y = \max(x_i), \quad x_i \in x \quad (23)$$

where x_i is the molecule's function in the area marked on the convolution layer by the character x. Layers with full connectivity connect all the layers from the previous convolutional layer to the input layers, turning all localized properties into feature sets. Our network's neural component contains three completely connected layers [35], [36]. Convolution layers are more susceptible to overloading problems. To address this problem, we use the dropout mechanism to reduce the regularization of the initial two phases. The output layer is the last completely linked layer, and we replace it with an output layer that has two neurons to represent the likelihood that output will occur. By the SoftMax function, the probability is given as follows:

$$y_j = \frac{\exp(f_j)}{\sum_{i=1}^{2} \exp(f_i)}, \quad j = 1, 2. \quad (24)$$

where y_j is the output probability of the jth neuron. Let $x_1, x_2, x_3, \ldots, x_n$ be independent random variables defined over X, we have:

$$I_n(g) = \frac{1}{n} \sum_{i=1}^{n} g(x_i), \quad (25)$$

which describes the results calculation

$$E(I(g) - I_n(g))^2 = \frac{\text{Var}(g)}{n}, \quad (26)$$

$$\text{Var}(g) = \int_X g^2(x)\, dx - \left(\int_X g(x)\, dx\right)^2. \quad (27)$$

A neural network (NN) consists of input, output, and hidden layers. Several types of activation functions are possible in NN, represented by n_f. Input neurons are denoted by j, hidden layers by I in the equations, and bias weight is $w_{\bar{\epsilon}^{\wedge(H)}}$.

The output of the hidden layer $e^{(H)}$ can be calculated by:

$$e^{(H)} = n_f\left(W^{((i))}_{(\bar{R}^{\wedge(i)})} + \sum_{j=1}^{n} w^{((i))}_{(ji)} F_D\right), \quad (28)$$

$$\hat{G} = n_f\left(W^{(G)}_{(\widehat{BO})} + \sum_{i=1}^{n_0} W^{(G)}_{(i^-)} e^{(i)}\right) \quad (29)$$

The biases B_n and weight matrices W_n are calculated by:

$$W_n = U_n = \sum_{n=1}^{N} a \cdot \left(\text{rand} \lhd -\frac{1}{2}\right), \quad (30)$$

$$B_n = \sum_{n=1}^{N} a \cdot \left(\text{rand} - \frac{1}{2}\right), \quad (31)$$

which are used for Error Reduction Network (ERN) optimization, as given:

$$\left|R(\hat{f}) - \hat{R}_n(\hat{f})\right| \le \sup_{f \in H_m} \left|R(f) - \hat{R}_n(f)\right| = \sup_{f \in H_m} |I(g) - I_n(g)|. \quad (32)$$

IoT devices have limited resources, making it impractical to train machine learning models directly on them. This is because neural network-based models require extensive datasets and computational capacity that IoT devices typically lack. We train the model on edge servers, which offer sufficient computational resources and deliver faster response times compared to cloud data centers. Hence, training the model over the edge servers would not create an extra delay in the network. The computational complexity of deep learning-based models, such as Gaussian Bayesian transfer convolutional neural networks, is influenced by several parameters and is consistently high due to their reliance on large datasets for training. The complexity for training a CNN is generally $\mathcal{O}(N \times D \times K^2 \times M \times m)$, where N is the number of filters, D is the number of input channels, K is the kernel size, M is the number of operations (e.g., activations) per filter, and m is the number of samples.

IV. Performance Evaluation

In this section, we discuss the experimental setup, dataset description and experimental results.

TABLE II
COMPARATIVE ANALYSIS OF PROPOSED IoT_BFLA_ML WITH BASELINE APPROACHES BASED ON QoS PARAMETERS

Dataset	Techniques	Throughput	Latency	Training accuracy	Validation accuracy	Security	Robustness
LFW	FL-BC [6]	79	59	82	85	81	77
	DRL-BC [26]	81	61	85	89	83	79
	IoT_BFLA_ML	83	63	86	91	85	81
CelebA	FL-BC [6]	82	63	83	88	82	83
	DRL-BC [26]	84	65	85	92	86	85
	IoT_BFLA_ML	86	68	88	94	88	88
CASIA-WebFace	FL-BC [6]	85	66	85	92	85	86
	DRL-BC [26]	88	69	89	94	89	91
	IoT_BFLA_ML	89	71	91	96	92	93

A. Experimental Setup

To evaluate the performance and effectiveness of our proposed approach, we use the MATLAB simulation toolkit to set up a simulation environment. The system is configured with 16 GB of RAM, an 11^{th} generation Intel Core i7 CPU (i7-7700) running at 3.60 GHz, and 500 GB of secondary storage. Our study focuses on three key techniques: decentralized ledger, distributed learning, and 4GNet. Together, these techniques contribute to the advancement of blockchain-enabled 4GNet edge networks and enhance their capabilities for cross-silo FL.

B. Dataset Description

The work has been evaluated over 3 datasets [37]: Labeled Faces in Wild (LFW), CelebA, and CASIA-WebFace.

- *LFW:* It is a standardized dataset tailored for facial recognition, comprising 13,233 photos featuring 5,749 individuals [37]. Each photograph is labeled with attributes such as race, age, gender, hair color, and eyewear. LFW1 focuses on the "race: black" target property, functioning predominantly as a race classifier, whereas LFW2 is primarily designed as a race classifier, emphasizing the target property "male" for LFW1.
- *CelebA (CelebFaces):* It contributes attributes such as race, smile, and black hair, which are the target properties for FL on CeleA represented as CelebA1, CelebA2, CelebA3, and CelebA4, respectively [37]. The primary tasks assigned to FL in the context of CelebA involve the classification of gender and smiles. The dataset comprises a total of 128,000 photos with 64 photographs per participant.
- *CASIA-WebFace:* It comprises more than 400,000 facial photos representing 10,575 individuals [37]. In CASIA1, the target property is black race, and the main task is gender classification, whereas in CASIA2, the major work is race classification, and the target property is male.

C. Baseline Approaches

We have chosen two prominent state-of-the-art techniques as baselines from the literature: blockchain-enabled federated learning (FL-BC) [6] and blockchain-enabled deep reinforcement learning (DRL-BC) [26] to evaluate the performance of our proposed work (IoT_BFLA_ML). Despite being one of the most widely used algorithms in FL [25], we did not consider FedAvg for performance comparisons due to the following limitations [15], [30], [38]:

1) Data Heterogeneity: Each client often has non-IID (Independent and Identically Distributed) data, meaning that the distribution of the data varies across clients. FedAvg can struggle in this scenario because the aggregated model may not generalize well if the client's data distributions are significantly different.

2) Model Bias: In cases of severe data imbalance, the global model may become biased toward the data distribution seen by the majority of clients, potentially resulting in poor performance for clients with minority data distributions.

3) Model Update Size: While FedAvg reduces the need for transferring raw data, the size of model updates (weights or gradients) can still be substantial, especially with large models or large numbers of clients.

4) Straggler Problem: In FL, some clients may take much longer than others to compute updates, leading to delays in the global aggregation step. Environments with varying computational resources or unreliable connectivity exacerbate this issue.

5) Slow Convergence rate: FedAvg can suffer from slow convergence, especially when clients' models are significantly different or when clients have insufficient local training (due to limited data or computation). This can make the training process longer when compared to centralized methods.

6) Model Overhead on Clients: The local training process requires clients to run machine learning models, which can be resource-intensive, especially on devices with limited computational power, such as smartphones or IoT devices.

7) Poor Data Quality: Clients with low-quality data, such as noisy or incomplete data, may train the global model on unreliable data, thereby affecting its performance. However, centralized learning allows for systematic data preprocessing, which FL frequently finds challenging to control.

8) Trust Issues: The central server holds the role of aggregating client updates, which can be problematic in scenarios where clients do not trust the central server, leading to concerns about data manipulation or model poisoning.

D. Results and Discussions

This section presents a comparative analysis of the proposed work (IoT_BFLA_ML) and baseline approaches (FL-BC [6] & DRL-BC [26]) evaluated across the mentioned datasets as

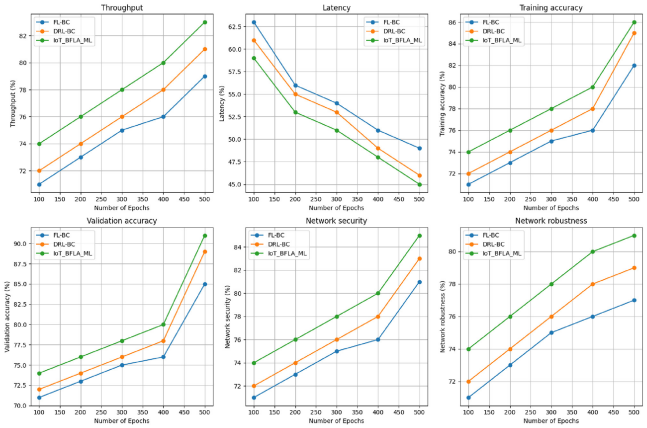

Fig. 3. Comparison between IoT_BFLA_ML and baselines for the LFW dataset in terms of throughput, latency, training accuracy, validation accuracy, network security, and network robustness.

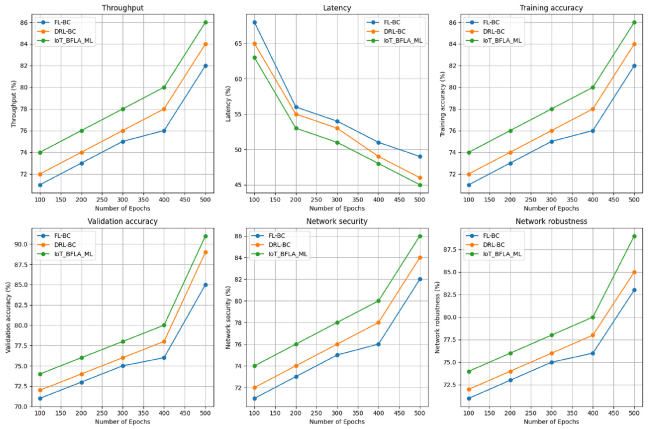

Fig. 4. Comparison between IoT_BFLA_ML and baselines for the CelebA dataset to improve several QoS parameters including throughput, latency, and network robustness.

illustrated in Table II. We have evaluated all the test cases in the edge cloud environment to thoroughly investigate the LFW, CelebA, and CASIA-WebFace datasets, and ultimately evaluate crucial performance metrics such as training and validation accuracy, latency, network security, throughput, and network robustness.

1) Test Case 1: LFW Dataset: In the first test case, we have chosen the LFW dataset to assess the performance of proposed work (IoT_BFLA_ML) and baseline approaches (FL-BC [6] & DRL-BC [26], as illustrated in Fig. 3. LFW includes images captured in diverse and uncontrolled conditions, closely mimicking the real-world environments in which IoT devices operate. This variability helps test the robustness and adaptability of the FL models when deployed in real-world IoT applications, ensuring they can handle diverse and unpredictable inputs.

The proposed IoT_BFLA_ML technique secures the data from outsider attacks and plays a vital role in ensuring the security and privacy of data in a collaborative cloud-edge environment. The proposed model is trained over the cloud and can detect anomalies or all kinds of modern attacks that usually occur in IoT networks. The LFW dataset's diverse and unconstrained images provide a comprehensive test bed for assessing the robustness and accuracy of the proposed model in various conditions, closely mirroring the environments where IoT devices operate. The proposed approach is applied to the mentioned dataset for 100 epochs initially to measure the performance, then the number of epochs is increased to 100 every time up to 500. In addition, the proposed approach analyzed the data request using a convolutional neural network and improved the latency by up to 63%, training and validation accuracy by up to 86% and 91%, throughput up to 83%, network security up to 85%, and network robustness of 81%. In the baseline approach, FL-BC achieved a latency of 59%, training accuracy of 82%, throughput of 79%, validation accuracy of 85%, and network security of 81%, network robustness of 77%; DRL-BC attained latency of up to 61%, training & validation accuracy of 85-89%, throughput of 81%, network security of 83%, and network robustness of 79%. Hence, By leveraging the LFW dataset, the proposed approach ensures robust model training and validation in diverse, real-world conditions, making it a viable solution for secure and efficient IoT data processing.

2) Test Case 2: CelebA Dataset: In addition to this, the work evaluates the performance of the proposed technique on the CelebA dataset as shown in Fig. 4. Its many annotations and large size make it a complete test for figuring out how well the blockchain-enabled FL model works, especially at complex, multi-attribute facial recognition tasks. The model achieved enhanced multi-label classification accuracy, showcasing its ability to handle complex attribute recognition tasks. The dataset's diverse conditions, including variations in pose, lighting, and occlusions, consistently validated the model's robustness. Furthermore, the proposed technique exhibited superior scalability and efficiency, as evidenced by improved latency and throughput figures, making it well-suited for real-time IoT applications. We also validated the model's privacy-preserving aspects, using blockchain to ensure secure and verifiable model updates, thereby preventing unauthorised access and data breaches.

3) Test Case 3: CASIA-WebFace Dataset: The third test case has been conducted to evaluate the proposed work performance over the CASIA-WebFace dataset. The authors have applied the blockchain-driven FL approach with a Gaussian Bayesian transfer convolutional neural network to secure the network and detect any malware or port scanning types of attacks. The integration of GBT-CNN in existing frameworks offers significant advantages in securing networks and detecting various types of attacks, such as malware and port scanning. The ability to quantify uncertainty, leverage transfer learning, enhance robustness, improve model confidence, and scale effectively makes GBT-CNN a powerful tool for maintaining robust and reliable network security in dynamic and evolving threat landscapes. The simulation-based outcome of the proposed and baseline techniques is shown in Fig. 5. The proposed approach achieved maximum throughput of up to 89%, latency of up to 71%, training and validation accuracy of the model is up to 91% to 96%, network security of 92%, and network robustness of 93%. The performance of the existing FL-BC is assessed in the same simulation

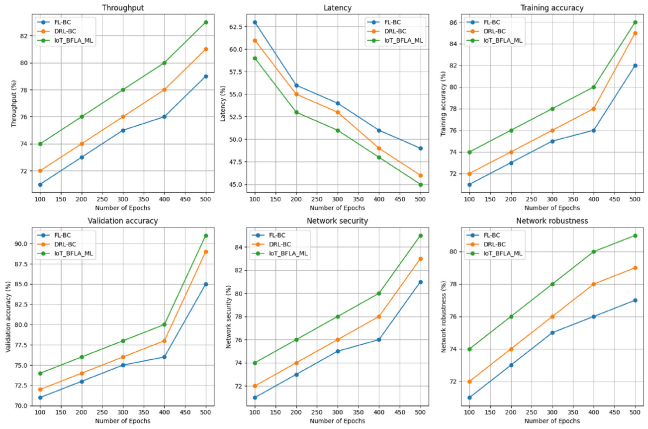

Fig. 5. Comparison between IoT_BFLA_ML and baselines for the CASIA-WebFace dataset in terms of throughput, latency, training accuracy, validation accuracy, network security, and network robustness.

environment, but the efficiency of parameters was not up to the proposed approach. DRL-BC attained throughput of 88%, latency of 69%, training accuracy of 89%, validation accuracy of 94%, network security of 89%, and network robustness of 91%. Our proposed work (IoT_BFLA_ML) outperforms the baseline approaches (FL-BC [6] & DRL-BC [26]) in terms of QoS parameters, as demonstrated by the experimental results using three different datasets (LFW, CelebA, and CASIA-WebFace).

V. Conclusion

IoT has permeated many aspects of our daily lives in recent years, and AI-powered intelligent services have proliferated. However, the centralized processing of data does not serve as a simple solution to the high scalability and robust IoT networks, as well as the growing data privacy concerns in consumer applications. However, there is still a constant risk that unauthorised and dishonest entities will target sensitive data generated by important IoT applications. Hence, ensuring data privacy and protecting data against manipulation or misuse become of paramount importance. FL emerged as a distributed solution, enabling model training at local devices and avoiding the need for data sharing. However, the data remains susceptible to attacks by various malicious entities, highlighting the dire need for a trustworthy and efficient framework to secure the IoT network for better service delivery to end users with consumer IoT applications. Hence, our work proposes a novel approach for enhancing security in underlying cloud edge networks by incorporating a blockchain-driven FL architecture. Integration of blockchain with FL offers a promising avenue to facilitate secure and intelligent data sharing while maintaining data integrity. For secure network-based data analysis, the proposed framework uses a Gaussian Bayesian transfer convolutional neural network. The proposed work is evaluated against various parameters such as throughput, latency, training and validation accuracy, and network robustness up to a significant value.

Future work can address the issue of resource management when implementing AI models in a 5G-enabled cloud-edge environment [8]. Moreover, explainable artificial intelligence (XAI) can enhance the transparency and explainability of existing AI models [24].

References

[1] M. Kumar, G. K. Walia, H. Shingare, S. Singh, and S. S. Gill, "AI-based sustainable and intelligent offloading framework for IIoT in collaborative cloud-fog environments," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1414–1422, Feb. 2024.

[2] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[3] J. K. Samriya, C. Chakraborty, A. Sharma, M. Kumar, and S. K. Ramakuri, "Adversarial ML-based secured cloud architecture for consumer Internet of Things of smart healthcare," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 2058–2065, Feb. 2024.

[4] W. Z. Khan, M. Y. Aalsalem, and M. K. Khan, "Communal acts of IoT consumers: A potential threat to security and privacy," *IEEE Trans. Consum. Electron.*, vol. 65, no. 1, pp. 64–72, Feb. 2019.

[5] B. Li, Y. Feng, Z. Xiong, W. Yang, and G. Liu, "Research on AI security enhanced encryption algorithm of autonomous IoT systems," *Inf. Sci.*, vol. 575, pp. 379–398, Oct. 2021.

[6] S. Otoum, I. A. Ridhawi, and H. Mouftah, "Securing critical IoT infrastructures with blockchain-supported federated learning," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2592–2601, Feb. 2022.

[7] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda, and Y. Kato, "Anomaly detection in smart home operation from user behaviors and home conditions," *IEEE Trans. Consum. Electron.*, vol. 66, no. 2, pp. 183–192, May 2020.

[8] G. K. Walia, M. Kumar, and S. S. Gill, "AI-empowered fog/edge resource management for IoT applications: A comprehensive review, research challenges, and future perspectives," *IEEE Commun. Surveys Tuts.*, vol. 26, no. 1, pp. 619–669, Feb. 2024.

[9] Q. Lai et al., "Improved transformer-based privacy-preserving architecture for intrusion detection in secure V2X communications," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1810–1820, Feb. 2024.

[10] H. An, D. He, C. Peng, M. Luo, and L. Wang, "Efficient certificateless online/offline signcryption scheme without bilinear pairing for smart home consumer electronics," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 4005–4015, Feb. 2024.

[11] S. Singh, I. Chana, and M. Singh, "The journey of QoS-aware autonomic cloud computing," *IT Prof.*, vol. 19, no. 2, pp. 42–49, 2017.

[12] J. K. Samriya et al., "Blockchain and reinforcement neural network for trusted cloud-enabled IoT network," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 2311–2322, Feb. 2024.

[13] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in Industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.

[14] D. Javeed, M. S. Saeed, I. Ahmad, P. Kumar, A. Jolfaei, and M. Tahir, "An intelligent intrusion detection system for smart consumer electronics network," *IEEE Trans. Consum. Electron.*, vol. 69, no. 4, pp. 906–913, Nov. 2023.

[15] J. Yang, T. Baker, S. S. Gill, X. Yang, W. Han, and Y. Li, "A federated learning attack method based on edge collaboration via cloud," *Softw. Pract. Exp.*, vol. 54, no. 7, pp. 1257–1274, 2024.

[16] M. Gupta, M. Kumar, and Y. Gupta, "A blockchain-empowered federated learning-based framework for data privacy in lung disease detection system," *Comput. Human Behav.*, vol. 158, Sep. 2024, Art. no. 108302.

[17] M. Guduri, C. Chakraborty, U. Maheswari, and M. Margala, "Blockchain-based federated learning technique for privacy preservation and security of smart electronic health records," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 2608–2617, Feb. 2024.

[18] D. C. Nguyen et al., "Federated learning meets blockchain in edge computing: Opportunities and challenges," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12806–12825, Aug. 2021.

[19] X. Wang, X. Ren, C. Qiu, Z. Xiong, H. Yao, and V. C. M. Leung, "Integrating edge intelligence and blockchain: What, why, and how," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 4, pp. 2193–2229, 4th Quart., 2022.

[20] T. Zhu et al., "Byzantine-robust federated learning via cosine similarity aggregation," *Comput. Netw.*, vol. 254, Dec. 2024, Art. no. 110730. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128624005620

[21] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, "Deep-learning-enhanced multitarget detection for end–edge–cloud surveillance in smart IoT," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12588–12596, Aug. 2021.

[22] A. Lutakamale, H. Myburgh, and A. De Freitas, "A hybrid convolutional neural network-transformer method for received signal strength indicator fingerprinting localization in long range wide area network," *Eng. Appl. Artif. Intell.*, vol. 133, Jul. 2024, Art. no. 108349.

[23] A. Alghamdi et al., "Blockchain empowered federated learning ecosystem for securing consumer IoT features analysis," *Sensors*, vol. 22, no. 18, p. 6786, 2022.

[24] M. Golec, S. S. Gill, R. Bahsoon, and O. Rana, "BioSec: A biometric authentication framework for secure and private communication among edge devices in IoT and industry 4.0," *IEEE Consum. Electron. Mag.*, vol. 11, no. 2, pp. 51–56, Mar. 2022.

[25] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Stat.*, 2017, pp. 1273–1282.

[26] B. Sellami, A. Hakiri, and S. B. Yahia, "Deep reinforcement learning for energy-aware task offloading in join SDN-blockchain 5G massive IoT edge network," *Future Gener. Comput. Syst.*, vol. 137, pp. 363–379, Dec. 2022.

[27] D. Namakshenas, A. Yazdinejad, A. Dehghantanha, R. M. Parizi, and G. Srivastava, "IP2FL: Interpretation-based privacy-preserving federated learning for industrial cyber-physical systems," *IEEE Trans. Ind. Cyber Phys. Syst.*, vol. 2, pp. 321–330, 2024.

[28] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, and A. Y. Zomaya, "Federated learning for COVID-19 detection with generative adversarial networks in edge cloud computing," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 10257–10271, Jun. 2022.

[29] A. M. Alwakeel, "An overview of fog computing and edge computing security and privacy issues," *Sensors*, vol. 21, no. 24, p. 8226, 2021.

[30] A. Islam, H. Karimipour, T. R. Gadekallu, and Y. Zhu, "A federated unlearning-based secure management scheme to enable automation in smart consumer electronics facilitated by digital twin," *IEEE Trans. Consum. Electron.*, early access, May 3, 2024, doi: 10.1109/TCE.2024.3396723.

[31] J. Gong and N. J. Navimipour, "An in-depth and systematic literature review on the blockchain-based approaches for cloud computing," *Clust. Comput.*, vol. 25, no. 1, pp. 383–400, 2022.

[32] Y.-A. Daraghmi, E. Y. Daraghmi, R. Daraghma, H. Fouchal, and M. Ayaida, "Edge-fog-cloud computing hierarchy for improving performance and security of NB-IoT-based health monitoring systems," *Sensors*, vol. 22, no. 22, p. 8646, 2022.

[33] L. Ismail and H. Materwala, "IoT-edge-cloud computing framework for QoS-aware computation offloading in autonomous mobile agents: Modeling and simulation," in *Proc. 6th Int. Conf. Mobile Secure Program. Netw. (MSPN)*, 2021, pp. 161–176.

[34] S. K. Mohapatra, B. R. Swain, and P. Das, "Comprehensive survey of possible security issues on 4G networks," *Int. J. Netw. Security Appl.*, vol. 7, no. 2, p. 61, 2015.

[35] Y. Luo, W. You, C. Shang, X. Ren, J. Cao, and H. Li, "A cloud-fog enabled and privacy-preserving IoT data market platform based on blockchain," *Comput. Model. Eng. Sci.*, vol. 139, no. 2, pp. 2237–2260, 2024.

[36] S. Sciancalepore, "PARFAIT: Privacy-preserving, secure, and low-delay service access in fog-enabled IoT ecosystems," *Comput. Netw.*, vol. 206, Apr. 2022, Art. no. 108799.

[37] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-CELEB-1m: A dataset and benchmark for large-scale face recognition," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 87–102.

[38] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "FedAvg with fine tuning: Local updates lead to representation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 10572–10586.