Blockchain-Based Privacy Preservation Scheme for Misbehavior Detection in Lightweight IoMT Devices

Sandi Rahmadika, Philip Virgil Astillo, Gaurav Choudhary, Daniel Gerbi Duguma, *Student Member, IEEE*, Vishal Sharma, *Senior Member, IEEE*, and Ilsun You, *Senior Member, IEEE*

***Abstract*—The Internet of Medical Things (IoMT) has risen to prominence as a possible backbone in the health sector, with the ability to improve quality of life by broadening user experience while enabling crucial solutions such as near real-time remote diagnostics. However, privacy and security problems remain largely unresolved in the safety area. Various rule-based methods have been considered to recognize aberrant behaviors in IoMT and have demonstrated high accuracy of misbehavior detection appropriate for lightweight IoT devices. However, most of these solutions have privacy concerns, especially when giving context during misbehavior analysis. Moreover, falsified or modified context generates a high percentage of false positives and sometimes causes a by-pass in misbehavior detection. Relying on the recent powerful consolidation of blockchain and federated learning (FL), we propose an efficient privacy-preserving framework for secure misbehavior detection in lightweight IoMT devices, particularly in the artificial pancreas system (APS). The proposed approach employs privacy-preserving bidirectional long-short term memory (BiLSTM) and augments the security through integrating blockchain technology based on Ethereum smart contract environment. The effectiveness of the proposed model is bench-marked empirically in terms of sustainable privacy preservation, commensurate incentive scheme with an untraceability feature, exhaustiveness, and the compact results of a variant neural network approach. As a result, the proposed model has a 99.93% recall rate, showing that it can detect virtually all possible malicious events in the targeted use case. Furthermore, given an initial ether value of 100, the solution's average gas consumption and Ether spent are 84,456.5 and 0.03157625, respectively.**

***Index Terms*—Blockchain, federated learning, Internet of Medical Things (IoMT), misbehavior detection, privacy preservation, smart contract.**

I. Introduction

ALONG with the exponential growth of the Internet of Things (IoT) in the medical industry, new security problems are continually emerging while existing security threats become more apparent. Additionally, privacy issues that may arise from the various data collected from IoT devices can negatively impact users. Notably, the collected data may be used to determine the users' location, ID, role, and other important information, posing a valid concern about data privacy and security. Data access is crucial to a secure system by controlling who accesses such information, as IoT devices store sensitive information in their memory or controllers. Unnecessary access control should be avoided, and all data should be verified as there are circumstances where a single vulnerability can be exploited owing to assumptions made at component interfaces in misbehavior detection approaches [1]–[3].

Misbehavior detection in IoMT devices is a challenging task given their severely constrained resources. To make matters worse, data transfer between IoMT devices and monitoring agents is inconsistent and vulnerable to insider assaults. As a result, IoMT devices must be more secure and sensitive in terms of misbehavior detection, because altered data decreases the effectiveness of misbehavior detection approaches [18], [19]. Despite the existence of numerous communication protocols that address the issues related to trust and security of context during message exchanges [20], there is still a deficiency concerning privacy preservation mechanisms [21], [22]. Blockchain-based privacy protection is an alternative solution that has recently gained popularity in the domain of IoT context sharing. [23], [24].

Blockchain is inherently tamper-proof and does not require the presence of a middleman during transactions, making it a feasible solution to various problems in the IoT ecosystem. Blockchain is used extensively in healthcare applications to provide the security, transparency, and immutability of data

Manuscript received 15 February 2022; revised 18 June 2022; accepted 21 June 2022. Date of publication 28 June 2022; date of current version 6 February 2023. This work was supported by the National Research Foundation of Korea funded by the Ministry of Education through Basic Science Research Program under Grant NRF-2020R1I1A2073603. *(Sandi Rahmadika and Philip Virgil Astillo contributed equally to this work.) (Corresponding author: Ilsun You.)*

Sandi Rahmadika is with the Department of Electronic Engineering, Universitas Negeri Padang (UNP), Sumatera Barat 25171, Indonesia (e-mail: ndiikaa@gmail.com).

Philip Virgil Astillo is with the Department of Computer Engineering, University of San Carlos, Cebu City 6000, Philippines (e-mail: pvbastillo@usc.edu.ph).

Gaurav Choudhary is with the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), 2800 Kgs. Lyngby, Denmark (e-mail: gauravchoudhary7777@gmail.com).

Daniel Gerbi Duguma is with the Department of Information Security Engineering, Soonchunhyang University, Asan 31538, South Korea (e-mail: danielgerbi2005@gmail.com).

Vishal Sharma is with the School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast (QUB), BT7 1NN Belfast, U.K. (e-mail: v.sharma@qub.ac.uk).

Ilsun You is with the Department of Information Security, Cryptology, and Mathematics, Kookmin University, Seoul 02707, South Korea (e-mail: ilsunu@gmail.com).

Digital Object Identifier 10.1109/JBHI.2022.3187037

records via autonomous contracts [25]. However, for the application of private data that is confidential, transparency needs to be considered further. This is the most important aspect of incorporating blockchain into our research. As a result, we use the Ethereum smart contract to control how data is kept and accessed safely, in conjunction with our security measures built inside the contracts. We also exert the merits of the smart contract feature as an incentive mechanism for the data provider. Our proposed scheme disguises the values of the transaction, and it cannot be linked to the corresponding entity. Specifically, the useful data is hidden from the observers.

Problem Statement and Our contribution: The current strategies for misbehavior detection include continuous monitoring and Machine Learning (ML)-based solutions, which are inefficient for lightweight devices. In such instances, specification-based misbehavior detection is preferable due high accuracy with low resource requirement. However, because IoT device communication is unreliable, context sharing can be altered, resulting in substantial false-positive rates. When used at resource-constrained IoT sites, traditional security and privacy protections will be prohibitively costly in terms of computing overhead. Motivated by the aforementioned disadvantages, we present our contributions as follows:

1) We study and present various state-of-the-art Misbehavior Detection Systems (MDS) for IoT in general and IoMT in particular with a focus on security and privacy.
2) We construct untraceable transaction protocols by expanding an application layer protocol called CryptoNote.
3) We propose a lightweight, privacy-preserved, and secure MDS leveraging blockchain and FL for IoMT.
4) We implement the proposed system using Raspberry-based APS Controller and an Ethereum blockchain.
5) We carryout various performance measurements such as accuracy of estimation model, gas usage per block, and percentage of transaction in a block.

II. Related Work

The security, privacy, and assurance of healthcare gadgets continue to be an open concern. Patients' privacy and safety may be threatened owing to a lack of a systematic and accurate line of defense against a wide range of security threats. The shortcomings in existing medical Cyber-Physical Systems (MCPS) security make it unbearable for the medical services sector and healthcare providers to verify and ensuring gadgets.

There are several important research works related to misbehavior detection in medical IoT that address diverse issues such as security, privacy, dependability, and attack detection techniques. Meng et al. [26], for instance, focused on trust-based interruption identification using social profiling and used Euclidean distance between two behavioral profiles. Celdrán et al. [27] emphasized the current MCPS security problems and presented Virtual Medical Device (VMD). In the paper, the mobile edge and fog computing models are employed to maintain an automated and reasonable framework used by Network Function Virtualization (NFV) and Software-Defined Network (SDN) procedures to enable a consistent association of MCPS security. Nithya et al. [28] concentrated on a variety of challenges, including security, dependability, connectivity, and privacy. Lee and Sokolsky [29] presented the current technology advancements in medical CPS, as well as numerous research trends and challenges in the area. Mitchell and Chen [30] provide an intrusion detection approach for MCPS based on behavior-rule formulation. Schneble and Thamilarasu [31] propose an attack detection technique leveraging an FL-based Intrusion Detection System (IDS) to thwart against multitude of security breaches. In general, the current researches mainly encompasses security controls, security design issues, countermeasure, and government legislations.

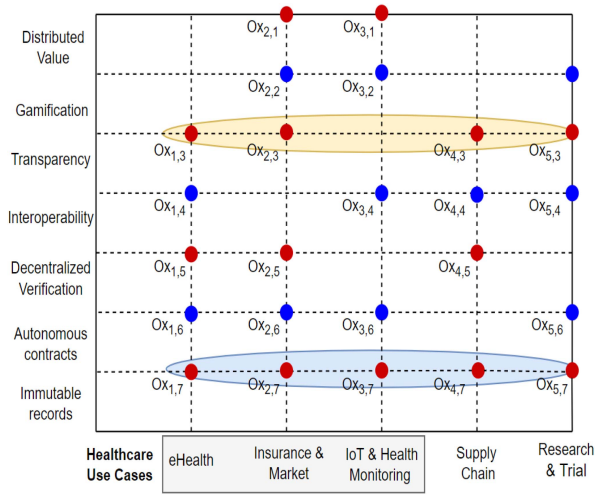

Fig. 1. Blockchain objectives for many different use cases. The advantages of transparency are utilized in the healthcare system in general (non-sensitive data).

Blockchain-based privacy preservation in IoT devices has emerged as a prominent solution, as shown in Fig. 1. The figure illustrates the essence of using blockchain technology in various use cases. With this regard, researchers like Loukil et al. [18] proposed a privacy-preserving IoT device management framework by highlighting data privacy. The proposed solution Judges the misbehavior of intelligent devices and determines the corresponding penalty. Zhu and Yu [17] designed a privacy-preserving scheme leveraging Deep Learning on sensor data. In the scheme, the privacy of the data is used for learning a model or as input to an existing model. Dwivedi et al. [15] proposed a system of adjusted blockchain models to fit for IoT devices. The authors constructed several supplementary cryptographic primitives' protocols to tackle the drawbacks in IoT applications running on a top blockchain-based network. A similar objective was introduced by Kuo et al. [32] that combines multiple techniques such as level-wise model learning, blockchain, and a new consensus algorithm for the model ensemble to preserve privacy modeling on the distributed ledger. Furthermore, various other researches (such as [33]–[35]) make use of FL to enhance privacy in different application areas. The existing state-of-the-art blockchain-based privacy preservation and IoT misbehavior detection are shown in Table I.

III. Core System Components and Models

A. Architectural Framework and the Essence of Decentralized Approach

In this research, blockchain-based privacy preservation techniques are merged with lightweight IoT devices to secure malicious behavior detection over the wireless network. Our system is designed for the insulin pump case with the respective controller to continuously monitor patient glucose levels within a specific time. It is well known as a compact medical system called continuous glucose monitors (CGM). The communication

TABLE I
EXISTING STATE-OF-THE-SOLUTIONS OF BLOCKCHAIN-BASED PRIVACY PRESERVATION AND IOT MISBEHAVIOR DETECTION

Authors	Category	Scheme	Mechanism	R1	R2	R3	R4	R5
Astillo et al. [8]	Behavior-based-IDS	Trust Management	Smoothened-trust-based scheme	No	Yes	Yes	Yes	No
Sedjelmaci et al. [9]	UAV-IDS	Hierarchical Detection Scheme	Combines rules-based detection and anomaly detection techniques	No	Yes	No	No	No
Sedjelmaci et al. [10]	UAV-IDS	Ejection framework against lethal attacks	Use Bayesian game Model for detection	No	Yes	No	No	No
Choudhary et al. [11]	IoT- IDS	Lightweight misbehavior detection in Medical IoT	Formally verified Behavior Rule based IDS	No	Yes	Yes*	Yes*	No
Khan et al. [12]	Behavior-based-IDS	Secure ICSs based on the behavior of their computational resources	Detect FDI attacks	No	Yes	No	No	No
Sharma et al. [13]	Behavior-based-IDS	Fuzzy based HCAPN	Detect zero day attacks	No	Yes	No	No	No
Jokar et al. [14]	Specification-based IDS	IDS for Home Area Networks	Used feature space for IDS targeting the IEEE 802.15.4 standard covering the PHY and MAC layers of the ZigBee technology	No	Yes	No	No	No
Salem et al. [15]	IoT-IDS	Anomaly Detection in Wireless Body Area Networks for Reliable Healthcare Monitoring	The approach is based on Haar wavelet, Non-Seasonal Holt-Winters (NSHW)	No	No	Yes	No	No
Dwivedi et al. [16]	Data sharing privacy in healthcare IoT	Privacy-Preserving Healthcare	Modified block chain models	Yes	No	No	Yes	No
Saeed et al. [17]	IoT-IDS	Intelligent Intrusion Detection in Low-Power IoTs	Intelligent security architecture using random neural networks (RNNs)	No	Yes	Yes	No	No
Zhu and Yu [18]	Sensor data privacy	Privacy-Preserving in Deep Learning	Privacy of the data used for learning a model or as input to an existing model	Yes	No	Yes	Yes	Yes*
Loukil et al. [19]	Data Privacy in IoT	Privacy-preserving IoT device management framework	Judging the misbehavior of the smart device and determines the corresponding penalty	Yes	No	No	Yes	Yes

R1: Block Chain Based Privacy for Decisions, R2: Rule Based Detection, R3: Deep Learning, R4: Secure Data Sharing Considerations, R5: Secure Misbehavior Detection,* Represents Only Theoretical Discussions.

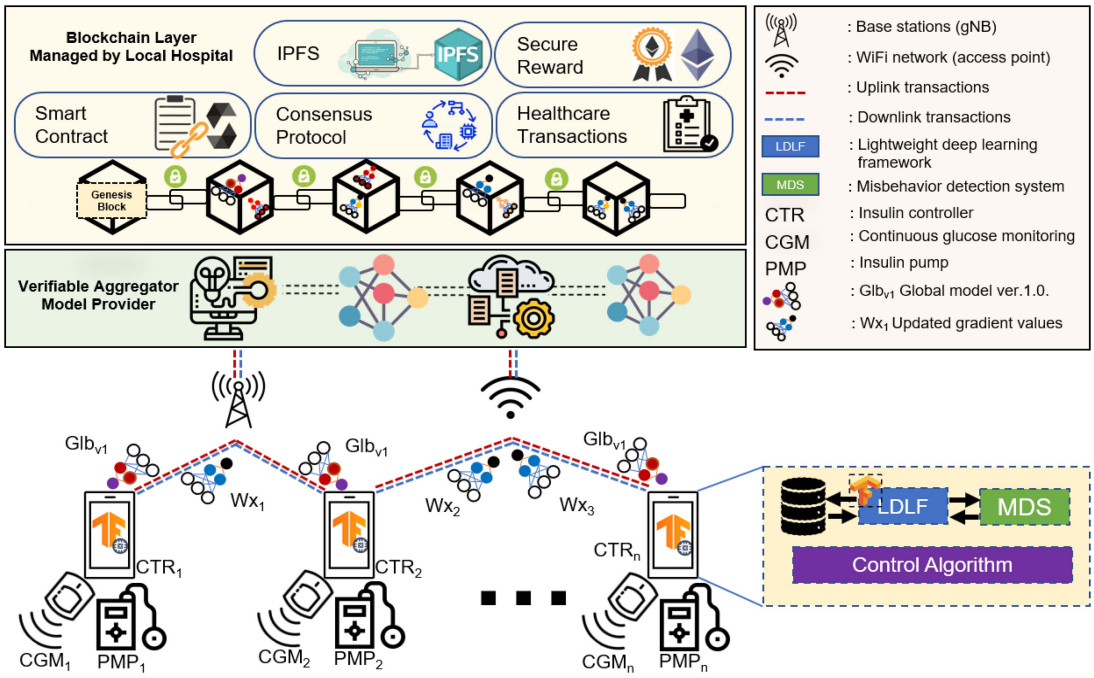

Fig. 2. System overview of our proposed scheme. Decentralized ledgers manage the outputs from insulin controller (CTR) within devices by integrating the lightweight deep learning framework (LDLF) and MDS. The outputs are stored securely, and the rewards are distributed equitably.

is conducted through a distributed information technology and edge computing manner. We leverage a variant of recurrent neural network (RNN), namely bidirectional long-short term memory (BiLSTM). On the other hand, blockchain technology can provide an immutable data record with several cryptography protocols embedded into smart contracts. The overall overview of our suggested method is depicted in Fig. 2. In the diagram, each APS controller collects critical information from the CGM and Insulin Pump, such as blood glucose level and remaining insulin quantity. Normally, the controller would be a smartphone that communicates with the blockchain layer and the aggregator through a cellular network or WiFi network. The controller is made up of several components, such as the lightweight deep-learning framework (LDLF), MDS, data storage, controller algorithms, and so on. Instead of sending it immediately to the verifiable aggregator model provider, the private information held in the controller is utilized for training the first global model received. When the relevant weight parameters with the greatest accuracy value are identified, the controllers send them to the blockchain-enabled aggregator. After the blockchain validates the aggregated weight parameters, they are averaged and sent back to each of the participating APSs. Further details of the entire process are described in the subsequent subsections. Meanwhile, it is important to note that the proposed framework

can be adapted to all subdomain of IoMT, but each subdomain may have a specific structure of the deep learning model that is more suitable for the environment.

One of the blockchain merits that need to be considered in this research is the transparency properties that are inherent in the decentralized approach. Every entity that is incorporated in a blockchain network can access information or transaction records. This transparency trait is beneficial in some cases. Still, it is not desirable for some scenarios, for instance, any systems that manage sensitive data [36]. Some examples of sensitive data are private information revealing racial or ethnic origin, religious or philosophical beliefs, sexual orientation, health-related data, etc. Therefore, blockchain-based applications require several additional dynamic protocols. By doing so, the system can obscure the information stored in the database. In our cases, the log of transactions is available to every member. However, the actual identity or data remains secret since we construct several protocols embedded into the system. We detail the decentralized privacy-preserving protocols in Section III-C.

B. Federated Learning (FL)-Based Misbehavior Detection

The massive proliferation of the Internet of Things (IoT) has opened more opportunities for an adversary to compromise, especially by exploiting newly discovered vulnerabilities, of a target system. This situation is critical, particularly in the healthcare industry where the integration of IoT for remote medical services is increasing [37].

Among the variety of promising solutions, the network MDS showed high efficiency and effectiveness in detecting malicious actions of devices as the consequence of both internal and external attacks [7], [10], [38]. This scheme requires careful analysis of a large number of operational data, or the solution provider must possess extensive domain knowledge to differentiate benign network events from malicious ones.

In the healthcare industry, most of this data contains sensitive personal health information of the patients. FL, an unconventional learning paradigm, is desirable and appropriately augments privacy-preservation of the patients since training data stays within the digital space of the owner and global model is built out of sub-models, which are trained locally by participating devices. This paper proposes a deep neural network-based MDS trained under an FL paradigm as applied to the APS.

In this case, APS controllers participate in the model-building process at a predefined iteration or communication round with the server, which serves as the aggregator. At every round, the participating devices receive the global model (GM) from the server and train it using their locally stored data. Subsequently, all devices submit their trained sub-models back to the server for aggregation using (1), in which the updated parameters of the global model, carried out to the next round, is the weighted average. The weight is defined by the cardinality of respective local dataset stored in device i over the cardinality of the union of local datasets ($\cup D_i$). Afterwards, the next round starts wherein the server distributes the aggregated model back to the devices.

$$GM \leftarrow \sum_{i=1}^{P} \frac{|D_i|}{|\cup D_i|} \times W_i \tag{1}$$

$$where W_i : (weights,\ bias)\ of\ device\ i$$

$$D_i :\ local\ dataset\ stored\ at\ device\ i$$

$$P :\ number\ of\ participating\ devices$$

To illustrate the concept of FL and show its advantages in misbehaviour detection, we used state-based modelling where states are connected to each other via weights. Each local device can generate state machines that will get updated via weights only, say, w. Based on the concept of FL, only weights get transferred, and the number of states cannot be predicted by an adversary trying to dodge the detection or prevent misbehaviour from getting caught. With k number of states, an adversary would have to check $\frac{k(k+1)}{2}$ transitions in a given span of time, which by the property of determinism cannot be completed effectively. Thus, maintaining the system's privacy and contents anonymous.

If S_i is the set of states for ith device such that $S_i = \{S_{1,i}, S_{2,i}, \ldots, S_{k,i}\}$ having n number of states and W_i be the set of weights for the ith device, such that $(S_{m,i}, S_{k,i})$ are the transitions connecting the two states, m and n, with weight $W_{mk,i}$, then the task is to effectively offload the weights and share them across devices through aggregation before the adversary can identify the sequence of transitions to avoid misbehaviour detection. An interesting observation is that if the state machines have near to complete connected graph, the attack prediction for an adversary will increase, which can be prevented by considering a secondary constraint, i.e., keeping the number of states too high than the number of transitions to delay the adversaries from avoiding the detection. Thus, if τ is the time for updating the weights, and the prediction rate of states is ϑ, then the number of states that can be predicted within τ should be $<< |S_i|$ for the ith device. This situation for state-based FL allows understanding the performance of the system by exploiting the model through an appropriate game theory where the situation between the adversaries and the honest device can be evaluated subject to the properties of FL.

To model this, let $L(|\mathcal{N}| \geq |\mathcal{M}|)$ be the function representing convergence cost as the divisibility of the weights across different state machines based on the properties of FL for k states of honest device and m states of adversarial device. Here, $L(|\mathcal{N}| \geq |\mathcal{M}|)$ must be followed such that updates of the states and weights should be done before the adversary can predict the weights and transitions across the states leading to the high possibility of attacks. To understand such a situation, we rely on the Stackelberg game [39] formation between the devices and the adversaries, specifically when the adversaries can collate to attack the system. This game formation expresses two advantages - the first is that it helps to understand when the weights must be updated, and the second is that it allows understanding actions points for the adversaries ensuring the utility of FL for securing the operations of misbehaviour detection. Following it, $L(|\mathcal{N}| \geq |\mathcal{M}|)$ is defined as the difference between the honest players and the collated adversaries operating against the honest players, such that H is the operational function for honest devices and A is the action function for the adversary, defined as:

$$L(\underset{k \geq m}{|\mathcal{N}| \geq |\mathcal{M}|}) = \sum_{i=1}^{|\mathcal{N}|} H_i - \sum_{j=1}^{|\mathcal{M}|} A_j, \tag{2}$$

where $|\mathcal{N}|$ denotes the number of honest devices and $|\mathcal{M}|$ denotes the adversarial devices, such that H is the operational

function for honest devices and A is the action function for the adversary, defined as:

$$H_i = P_{H_i}[w] \cdot \left(\frac{\frac{\Delta W_{H_i} \varphi_{|S_i|}}{k_i(k_i+1)}}{2} \right), \tag{3}$$

$$A_j = P_{A_j}[w] \cdot \Delta W_{A_j}, \tag{4}$$

where $P_{H_i}[w]$ and $P_{A_j}[w]$ define the probability of weights being updated accurately and traced by the adversary accurately, respectively using Poisson distribution, such that $P_{Hi}[w] = \frac{\Lambda^{U_{\tau+1-\tau}} e^{-\Lambda}}{U_{\tau+1-\tau}!}$, where $U_{\tau+1-\tau}$ is the number of times the weights are updated for transitions between $\tau+1$ and τ, Λ is the rate of changes occurring in states to the total number of states for ith device. $\varphi_{|S_i|}$ is the total transitions for the actual $|S_i|$ states of ith device. Similarly, for the adversary, $P_{A_i}[w] = \frac{\Lambda^{P_{(\tau+1)-\tau}} e^{-\Lambda}}{P_{(\tau+1)-\tau}!}$, where $P_{(\tau+1)-\tau}$ is the number of times the weights are accurately identified by the adversary. Here, $\Delta W_{H_i} = ||W_{i,\tau+1} - W_{i,\tau}||$ and $||W_{j,\tau+1} - W_{j,\tau}||$ denote the weight difference for the honest devices and the adversaries between two intervals denoted by $\tau+1$ and τ. Based on the Stackelberg game, the requirement is converted into an optimization problem, where the objective is:

$$P1: \max_{\forall i \neq j} L(x), \tag{5}$$

$$C1: \Delta W_{H_i} \neq 0 \text{ and } \Delta W_{A_j} \geq 0, i \neq j, \forall i, \forall j$$

$$C2: \Lambda \geq 0, U_{(\tau+1)-\tau} > 0, \vartheta \geq 0$$

$$C3: \varphi_{|S_i|} << \frac{k_i(k_i+1)}{2} \tag{6}$$

There exists a trade-off between the FL operations, anonymity and the connectivity of the graph, which can improve the traceability of misbehaviour - if the number of transitions for the given number of states, $\varphi_{|S_i|}$, is too close to $\frac{k_i(k_i+1)}{2}$ number of transitions (total), then the adversaries will require to match weights for lesser number of combinations, thus, making it easier to predict. Now, the system can operate using FL for the devices in its control; hence the problem deduces to:

$$P2: max \left(P_{H_i}[w] \left(\frac{\frac{\Delta W_{H_i} \varphi_{|S_i|}}{k_i(k_i+1)}}{2} \right) \right) \tag{7}$$

Following the constraints in $C1$ to $C3$. The weights follow a standard normal distribution if the model is operated for a longer duration as the converging rate will become constant, resulting in a zero mean and a unit standard deviation ($\mu = 0$ and $\sigma = 1$), which can be used to represent as $P(\Delta W_H) = \frac{e^{\frac{-\Delta W_{Hi}{}^2}{2}}}{\sqrt{2\pi}}$. Now, it can be used to understand the bounds for the behaviour of the problem, P2. If such is the case, then P2 will be operating with two controlling variables P_{H_i} and $P(\Delta W_{H_i})$, both of which will depend on the number of times the weights are adjusted in the setup. If $U_{\tau+1-\tau}$ becomes too large, $P_{(\tau+1)-\tau}$ will decrease considerably, however, it will also affect the convergence rate of the model and can cause latency, which is not an ideal situation for a setup operating with IoMT. Hence, the model can further divide itself to reduce the burden on performance while keeping intact the data privacy. If $U_{(\tau+1)-\tau} = \Lambda$, then the function will be bounded by values ≥ 1, which is the non-complex minimum

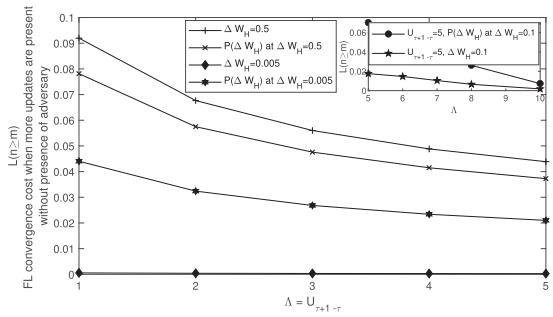

Fig. 3. Convergence cost using an FL scenario enabled with Stackelberg game formation for honest devices.

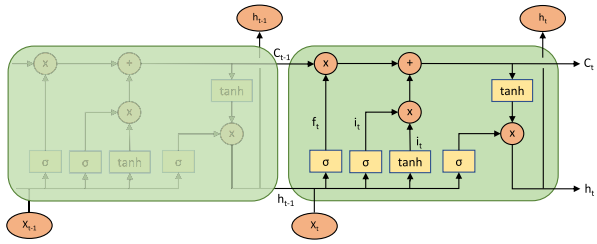

Fig. 4. Illustration of sequential processing in LSTM.

calculable when both the rates are equal. Alternatively, the function will attain its minimum when $U_{(\tau+1)-\tau} = \Lambda$. This helps to understand the impact of using FL and how it can be used for identifying misbehavior by finding values of weights leading to a minimum of the function and giving a minimum chance to an adversary for avoiding this detection. This has been illustrated using Fig. 3. It can be observed that the convergence rate will be affected if too frequent updates are performed to the weights, which will give more chances to an adversary to avoid the misbehaviour detection as it can predict the states and avoid getting caught by the detection mechanism.

Following the understanding of the utility of FL, it is supported by two layers of Bidirectional Long-Short Term Memory (BiLSTM) to help with the misbehaviour detection and ensure the privacy of the user relying on the properties of FL. The details are as follows:

1) Bidirectional Long-Short Term Memory (BiLSTM): LSTM is a variant of recurrent neural network (RNN). Because of its success, LSTM was specifically claimed as a solution for the technical issues of the classical RNN structure. In addition, LSTM practically remembers information for a long term, apart from learning the patterns of a given data. In consequence, its design feature makes LSTM appropriate for building models for those data that are collected in time order. Moreover, unlike classical RNN, LSTM can operate with flexible time steps of time-series data and resolves the vanishing gradient problem [40].

Like classical RNN top-level design, LSTM still follows the chain-like structure of repeating modules, called LSTM cells. The information flowing in the chain are controlled by four network passes, namely forget gate (f_t), input gate (i_t), cell state candidate ($\hat{C}_t$), and output gate (o_t). Each passes results from an element-wise sigmoid function (σ) and are combined through point-wise multiplication operation, as shown in Fig. 4. The inference (h_t) of the model is obtained by the following

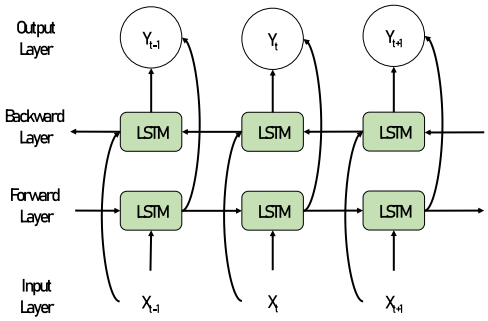

Fig. 5. Illustration of Bidirectional LSTM.

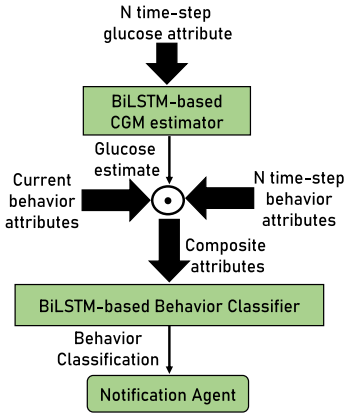

Fig. 6. Two tier design of collaboratively-trained BiLSTM-based MDS.

composite function:

$$
\begin{aligned}
f_t &= \sigma(W_f \times [h_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i \times [h_{t-1}, x_t] + b_i) \\
\hat{C}_t &= \tanh\left(W_C \times [h_{t-1}, x_t] + b_{\hat{C}_t}\right) \\
C_t &= f_t \times C_{t-1} + i_t \times \hat{C}_t \\
o_t &= \sigma(W_o \times [h_{t-1}, x_t] + b_o) \\
h_t &= o_t \times \tanh(C_t) \\
&\quad where\ W_* \ and\ b_* \ are\ weight\ matrix\ and \\
&\quad bias\ of\ the\ gate\ layers.
\end{aligned} \tag{8}
$$

The processing flow of a typical LSTM-based prediction model is limited to forward direction, making use of only the previous events [41]. To overcome this limitation, LSTM is extended to a bidirectional flow structure with the goal of enhancing the model performance in sequence-related problems. The output of the model is obtained wherein the future context is included in the analysis. Fig. 5 portrays the logistic flow of BiLSTM. Accordingly, BiLSTM is composed of two hidden layers, respectively processing input sequence data in forward and reverse timestep directions.

2) Misbehavior Detection System (MDS) Architecture Overview: The proposed MDS contains two-tier of the FL-based model, as presented in Fig. 6. The first tier implements a BiLSTM-based estimator, which periodically forecasts the blood glucose level (BGL) of a patient based on the previous n timesteps. The estimated value is utilized in computing insulin dosage and insulin amount in the pump vial. Furthermore, the estimate and its derivatives are operationally combined with other attributes, generating input features for behavior prediction. The second level also implements the BiLSTM structure in classifying the system as well-behaved or malicious. In the the latter case, the logical agent notifies the authorized person to quickly address the anomaly. Note that this paper leverages the input features introduced in [42]. For detail, expressions 9 and 10 defines the input features of the CGM estimator and classifier, respectively. We refer the reader to [42] for clarity.

$$
\begin{aligned}
&Estimated\_CGM_t \leftarrow [G_{t-n}, \ldots, G_{t-2}, G_{t-1}] \\
&where G \in \{glucosereading, carbsintake, insulinintake\}
\end{aligned} \tag{9}
$$

$$
\begin{aligned}
&Behavior \leftarrow \ [B_{t-n}, \ldots, B_{t-2}, B_{t-1}] \\
&where\ B\ \in \{glucose\ estimation\ error, \\
&message\ 1\ arrival\ error, \\
&insulin\ dosage\ estimation\ error, \\
&message\ 2\ arrival\ error \\
&in-vial\ estimation\ error, \\
&message\ 3\ arrival\ error
\end{aligned} \tag{10}
$$

To this end, the application of this deep learning solution generally demands model-carrier with high computation capability and memory due to increasing complexity. Hence, the conversion of trained models to lightweight version is desirable to the case study, i.e., memory requirement of the model is reduced and the latency of arriving to a decision is significantly low, but still achieving considerably high accuracy. Accordingly, this work applies the simplest model compression method called post-training quantization technique, wherein the 32-bit floating-point precision of the based model parameters are transformed into an 8-bit precision. In turn, the inference latency and memory consumption is ideally reduced by 16 and 4 times, respectively [43], [44].

C. Decentralized Privacy Preserving

Our incentive scheme is designed by diversifying the CryptoNote protocol [45], where this protocol is a pioneer in delivering user's privacy in a decentralized incentive mechanism. This technique is implemented in one of the blockchain cryptocurrencies known as Monero (XMR). It is an open-source, private, decentralized cryptocurrency that keeps transactions confidential and secure.

1) Secure Distributed Ledger Management: We construct a secure distributed ledger management protocol empowered by Ethereum smart contract by referring to the CryptoNote layer protocol to solve specific issues identified in Bitcoin transactions. The designed protocols leverage a group of ring signatures. It is a multi-signer digital signature scheme where a group of ring signature schemes possesses N signers forming a ring. Whenever the signers in the group receive a transaction to be endorsed, they can produce a ring signature with a corresponding private key. Once the ring is successfully created, every group member can use the signature on behalf of the group to keep their identity secret. A ring signature can be defined as a type of digital

signature algorithm that any member of the group can calculate. Whilst, the ring confidential transactions perform a list of prior transactions to obscure the original value of current transactions. The user US_x enables to choose the number of signatures RG_{sx} from the ring RNG_{tot} to be used in a transaction Tx_n. The selected signature is part of the ring that has been generated in advance, where $RG_{sx} \in RG_{tot} \geqslant 1$.

1) Healthcare provider $GVR_n \rightarrow hash(Pub_{GVR}, Sec_{GVR}) \rightarrow Pub_{GVR} \rightarrow P_{GVR} = trap_{GVR}(q_{GVR})$
2) New Entity n $New_n \rightarrow hash(Pub_{New_n}, Sec_{New_n}) \rightarrow Pub_{New_n} \rightarrow P_{New_n} = trap_{New_n}(q_{New_n})$

The primary key values for each lightweight IoT device within MEC are obtained from the trapdoor permutation function as a set of one-way function $fn_{kr} : M_{kr} \rightarrow N_{kr}(kr \in KR)$, where for all KR, M_{kr}, N_{kr} is a subset of binary strings value $\{0,1\}^*$, fulfilling a certain number of requirements, such as there exists a sampling of probabilistic polynomial time $Create(1^n) = (kr, x_{kr})$ with $kr \in KR \cap \{0,1\}^n$; where $x_{kr} \in \{0,1\}^*$ meets $|x_{kr}| < pol(n)$ with pol is defined as some polynomial values. Thus, every x_{kr} is known as trapdoor corresponding to kr value. Suppose any $kr \in KR$ with pol algorithm for every $x \in M_{kr}$, let $z = \alpha(kr, fn_{kr}(X), x_{kr})$. Accordingly, the function possess $fn_{kr}(z) = fn_{kr}(x)$. Each entity holds a pair of parent keys (Pub_n, Sec_n) generated beforehand. The encryption $P_n = trap_n(q_n)$ generates the public key; $trap_n$ is a trapdoor permutation function, while $trap_n(q_n)$ specifies $f_i(q_i) = q^2 mod n_i$ over $\{0,1\}^b$. Eventually, the signature keys of entities can be defined as follows:

$$
\begin{aligned}
&RNG_{sgn} \rightarrow GVR_n \oplus PET_n \oplus INC_n, \oplus, \ldots, \oplus New_n;\\
&\{trap_{GVR}(q_{GVR}) \oplus trap_{PET_n}(q_{PET_n}) \oplus\\
&\ldots, \oplus trap_{New_n}(q_{New_n})\};\\
&\left\{ \sum_{i=1}^{RNG_{tot}} trap_i(q_i) = trap_{GVR}(q_{GVR}) \right.\\
&\left. \oplus trap_{PET_1}(q_{PET_1}) \oplus trap_{New_n}(q_{New_n}) \right\}; \qquad (11)
\end{aligned}
$$

The observers have no knowledge about the sender's information due to the transaction is being signed on behalf of the group. The sender is free to choose the number of signatures as stated in the formula (11). Members can use the signature within the ring for any lightweight IoT transactions without requiring approval from each group member. Each addition of a new entity can be executed regularly as long as the public key is known ($Update\_RNG_{sgn}$). Likewise, excluding members from the ring can be managed undeviatingly by the manager ($Exclude\_RNG_{sgn}$).

The constructed protocols are based on the elliptic curve cryptography established with regard to multiplicative cyclic groups. The secret key Sec_{ENT_n} defines $Q\alpha \in [1, l-1]$; with l represents the prime order of a base point in the elliptic curve cryptography. Meanwhile, the public key Pub_{ENT_n} is understood as a point of $Pub\alpha = Sec\alpha \cdot \mathbb{G}$ (with $\mathbb{G}$ is a generator for $Pub\alpha$). There exists a pair of tracking keys $track\_keys(Q\alpha, Pub\beta)$ obtained from $Secret$ and $Publiceckeys(Pub\beta = Sec\beta \cdot \mathbb{G}$ with condition $Sec\alpha \neq Sec\beta$) [46]. Finally, the description of protocols, which is also a part of ring confidential transactions, can be interpreted as follows:

$$
\begin{aligned}
&RNG_{sgn} := \left\{ \{(AD_1^1, CRSP_1^1), ..., (AD_1^n, CRSP_1^n), \right.\\
&\left. \left(\sum_j AD_1^j + \sum_{j=1}^{n} CRSP_1^j - \sum_i CRSP_{i,out} \right) \right\}.\\
&\{(AD_{p+1}^1, CRST_{p+1}^1), ..., (AD_{p+1}^m, CRST_{p+1}^m),\\
&\left. \left. \left(\sum_j AD_{p+1}^j + \sum_{j=1}^{n} CRSP_{p+1}^j - \sum_i CRSP_{i,out} \right) \right\} \right\}. \qquad (12)
\end{aligned}
$$

1) Let $\{(AD_\pi^1, CRSP_\pi^1), (AD_\pi^2, CRSP_\pi^2), ..., (AD_\pi^n, CRSP_\pi^n)\}$ be a set of addresses/commitments including identical secret keys $Sec_j, j = 1, ..., n$.
2) Search $p+1$ sets $\{(AD_i^1, CRSP_i^1), ..., (AD_i^n, CRSP_i^n)\}, i =, ..., p+1$ (never being used beforehand).
3) Select on a collection of output addresses $P_i, CRSP_{i,out} \rightarrow \sum_{j=1}^{n} CRSP_\pi^j - \sum_i CRSP_{i,out}$ supposed to be zero.
4) Let formula in (12) be the generalized ring that the sender expects to sign.

$$
\begin{aligned}
&{}^{RNG_{sgn}}\sum_{i=1} trap_{INC_n}(q_{INC_n}) \oplus trap_{PET_{n+1}}(q_{PET_{n+1}})\\
&\oplus trap_{GVR}(q_{GVR}) \in RNG_{tot} \geqslant 1);\\
&(WherePET_1(RNG_{sgn}) must > 1)\\
&Combining\ function\ can\ be\ defined\ as\ follow:\\
&CF_{k,v}(Pub_{GVR}, Pub_{PET_1}, Pub_{PET_n}, Pub_{INC_n}),\\
&E_k(P_{INC_n}, E_k(P_n - 1 \oplus E_k(P_{PET_1} \oplus v))) \cong v \qquad (13)
\end{aligned}
$$

To disguise the sender's transaction, the senders conceive the ring group by selecting a certain number of signatures to be included within the smart contracts. For instance, patient 1 PET_1 selects 5 public keys out of the ring including himself $(P_{GVR}, P_{PET_1}, P_{PET_2}, P_{PET_3}, P_{INC_n})$; 50 Pub_{ENT_n} in total available. In this sense, the group of ring can be defined as follows: $PET_1(RNG_{sgn}) \rightarrow trap_{GVR}(q_{GVR}) \oplus trap_{PET_1}(q_{PET_1}) \oplus trap_{PET_2}(q_{PET_2})$ $\oplus, \ldots, \oplus trap_{INC_n}(q_{INC_n})$; signed with PET_1's private key. The final construction of the ring generated by PET_1 is formed in (13). The sender's contract is executed with a combining function $CF_{k,v}$; where k is defined as a key for E_k. Finally, by adopting these protocols, the sender's privacy is preserved since the actual signers are disguised.

2) Deploying Transactions With Untraceable Incentive Schemes: For ease of understanding, we consider patient 1 PET_1 to be a sender in the lightweight IoT devices system. The ET_1 possesses all requirements needed to commit a transaction via smart contract. We also note the PET_1's transaction as TX_PET_1. This transaction consists of the type of function used by the sender, such as the main diagnosis function ($''Main\_Diagnos.''$) along with the description ($\lambda\_Descript.$) concatenated with the actual data/patient's diagnosis (ψ_IPFS_{CID}) and the details ($\Delta\_Details$'). The (ψ_IPFS_{CID}) is the unique content identifier (CID) associated with the data stored in a distributed file system known as the InterPlanetary File System (IPFS). CID is short, regardless of

TABLE II
SUMMARY OF NOTATIONS USED (BLOCKCHAIN TRANSACTIONS)

Notations	Definition
RNG_{sgn}	The number of ring signatures; Total RG_{sx}
RNG_{tot}	Total number of RNG_{sgn} available
GVR_n; PET_n	Healthcare providers (local hospitals); Patients
New_n; INC_n	New entity (n numbers); Incentive manager
Pub_n, Sec_n	Public key n and secret key n
$hash$; $trap(f)$	256-bit hash; trapdoor permutation function
fn_{kr}	A set of one-way function
$Update\_RNG_{sgn}$	Updated version of constructed ring signature
$Exclude\_RNG_{sgn}$	Removing a member of RNG_{sgn}
$Current\_RNG_{sgn}$	Current version of RNG_{sgn}
$Take\_New\_Pub$	Collecting a public key of potential member
$\mathbb{G}$	A generator in elliptic curve cryptography
$track\_keys$	Tracking keys assigned in Blockchain network
$(AD^n_\pi, CRSP^n_\pi)$	A set of addresses/commitments
$CF_{k,v}$	Combining functions
TX_PET_n	A secure Blockchain & FL transactions
$\lambda\_Descript.$	Description of stored data
ψ_IPFS_{CID}	A unique content identifier of IPFS
GVR_OTU_k	One-time used key
OPK_{cx1}	One-time private key to claim a reward

the size of the underlying content based on a cryptographic hash. The type of data stored can be adjusted accordingly, including aggregation values of FL, the information of continuous glucose monitors (GMC), and any other behavioural aspects.

$$\frac{\left[\frac{''Main\_Diagnos.''||\lambda\_Descript.||\psi\_IPFS_{CID}||\Delta\_Details}{PET_1'sPubKey \rightarrow PET\_Pub\alpha_1, PET\_Pub\beta_1}\right]}{\{signedwithPET_1(RNG_{sgn}) \in RNG_{tot} \geqslant 1 || PET\_Sec\alpha_1\}}$$
$$condition \rightarrow PET\_Sec\alpha_1 \neq PET\_Sec\beta_1\ ''\&\&''\mathbb{G}_\alpha \neq \mathbb{G}\beta;$$
$$(applied\ to\ all\ senders\ with\ respective''\mathbb{G}'') \tag{14}$$

The patient TX_PET_1 attaches his/her public keys $PET\_Pub\alpha_1, PET\_Pub\beta_1$ into transaction TX_PET_1 (unique feature). The first public key $PET\_Pub\alpha_1$ is created based on PET's private key $PET\_Sec\alpha_1$ from a certain base point/generator $\mathbb{G}_\alpha$ as defines: $PET\_Pub\alpha_1 \rightarrow PET\_Sec\alpha_1 \cdot \mathbb{G}_\alpha$. This key is being used together with the recipient's random data $rand$; where $R = rand \cdot \mathbb{G}$. Concurrently, the other PET's public key $PET\_Pub\beta_1$ is generated from another PET's secret key $PET\_Sec\beta_1$ corresponded to its generator: $PET\_Pub\beta_1 \rightarrow PET\_Sec\beta_1 \cdot \mathbb{G}_\beta$; with condition $\mathbb{G}_\alpha \neq \mathbb{G}_\beta$. Hence, $PET\_Sec\alpha_1 \neq PET\_Sec\beta_1$ as depicted in formula (14), inspired by [47], [48]. The second public key $PET\_Pub\beta_1$ is assigned as a tracking key within blockchain network. The sender TX_PET_1 can recognize the reward which belongs to him by checking the tracking key attached into TX_PET_1 as defined in (14).

Once TX_PET_1 is completed, the data owner receives a reward in the form of *Ether* cryptocurrency. The amount of *Ether* obtained can be adjusted by the provider. In the first place, the provider confirms the transactions claimed by the sender. If all requirements are satisfied, then the provider unpacks the public keys attached in TX_PET_1, and straightforwardly executes a random base point $rand \in [1, l-1]$ while also computes a provider's $one-time\ used\ GVR\_OTU_k$ for the sender PET_1 as shown in (15).

$$GVR\_OTU_k = Hash(rand\_PET\_Sec\alpha_1) \cdot \mathbb{G} + PET\_Pub\beta_1$$
$$Let\ rand = GVR's\ random\ data \rightarrow R = rand \cdot \mathbb{G}, \tag{15}$$

$$OPK_{cx1} = H_s(Priv\alpha_1 \cdot R) + Priv\beta_1 \tag{16}$$

The reward sent by provider GVR is only available for patient 1 PET_1 because only PET_1 knows the secret information of the transaction. Before a certain amount of *Ether* is transferred through the Ethereum network, the GVR must confirm that all states in TX_PET_1 are correct (labelled as "*True*"). Patient 1 examines every passing transaction using PET's private key $PET\_Sec\alpha_1, PET\_Sec\beta_1$. PET_1 is authorized to recover the corresponding GVR_OTU_k key since PET_1 is the owner of $PET\_Sec\alpha_1, PET\_Sec\beta_1$. Eventually, the PET_1's one time private key is signified in (16). This key is being used to spend the *Ether* transferred by GVR. In the original state of Monero (XMR) technology, the combination of these keys is being utilized as a part of a ring signature to obscure the sender's information. A fresh *key-image* is also attached to prevent the double-spending attack.

IV. PERFORMANCE RESULTS AND COMPARISON

A. Misbehavior Detection Experiment and Performance

In the experiment, we train and measure the performance of the proposed MDS. The accuracy of the estimation model is quantified by the root mean square error, while the classification model performance is measured based on the prediction accuracy, precision, recall, and F1-score.

The root mean square error (RMSE) calculates the total deviation of the forecasted glucose values (G_{est_i}) with the actual measurements (G_{meas_i}), as expressed in equation (17). A smaller total deviation from a number of samples indicates better accuracy. Meanwhile, classification performance indicators are measured by collecting the T_n, T_p, F_n, and F_p of the given test samples. These key parameters denote true negative, true positive, false negative, and false-positive counts, respectively. The metric accuracy measures how well the models classify both the malignant and benign events, whereas the other three metrics measure how well the models are in classifying malignant events. Precision is the level of reliability with which the trained model correctly identifies the malignant events. As shown in expression (19), this indicator is calculated as the ratio between of correctly identified malignant events (T_p) and all events identified as malignant, where F_p is the number of benign events misidentified as malignant. Meanwhile, recall indicates how good the models are at classifying malignant events. It is the ratio between the number of correctly identified malignant events and the total of actual malignant events. Regarding misbehavior detection performance, the models with relatively low recall rates are regarded as ineffective due to the high incorrect classification of malignant events. Thereby, an effective method requires a recall rate as high as possible since undetected attacks can place patients in unfavorable health conditions. Finally, the F1-score is a performance metric that incorporates precision and recalls into a single value. A high F1-score indicates F_p and F_n are both low.The system is implemented using Python with environment details listed in Table III. MEC server is the aggregator of submodels and Raspberry-Pis are end devices collaboratively building the estimation and classification model.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(G_{est_i} - G_{meas_i})^2} \tag{17}$$

TABLE III
MACHINE ENVIRONMENT AND LIBRARIES USED IN THE EXPERIMENT

Description	Platform	Configuration/Version
MEC Server	Laptop	Windows 10 64-bit AMD Ryzen 5 4600H 20 GB RAM
5 End devices (Raspberry-Pi 3 Model B+)	Microcomputer	Raspbian OS 64-bit Arm Cortex-A530 1.4GHz 1.0 GB RAM
Compiler Environment	*****	Python 3.9
Extension Library	*****	Numpy 1.19.5
Extension Library	*****	Pandas 1.3.3
Machine Learning Library	*****	Scikit-learn 1.0
Machine learning library	*****	TensorFlow 2.6.0
Machine learning library	*****	Keras 2.6.0
Glucose-insulin Simulator	*****	UVa/Padova Simulator (Python version)

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{18}$$

$$Precision = \frac{T_p}{T_p + T_n} \tag{19}$$

$$Recall = \frac{T_p}{T_p + F_n} \tag{20}$$

$$F_1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{21}$$

1) Dataset Preparation: The dataset utilized for the training and evaluation was generated from the glucose-insulin simulator of [19]. In this experiment, 2400 of 6-tuple time-series samples was collected from a virtual diabetes patient. The collected data contains six features, including glucose level, the message transmission time of glucose level to the controller, insulin dosage, message transmission time of injection command to the insulin pump, insulin-on-vial (IoV), and message transmission time of IoV back to the controller. In this case, five virtual patients, each having distinct glucose-insulin behavior, were simulated. Forty percent of the samples were utilized in training and evaluation of the estimation model, and the remainder was allocated for the evaluation of the classification model.

It should be noted that the data being collected from the simulator are technically based on the regular operation, wherein the samples are assigned as benign. Furthermore, we augment the data to include malignant samples in the training and evaluation of the classification model. This is done by arbitrarily changing a feature value using equation (22). In addition, the dataset of each patient is augmented at different degrees to differentiate training and validation data size. Finally, the input features for the training and validation are arranged in blocks containing n timesteps sequence. In the end, the generated training and validation datasets have sizes summarised in Table IV.

$$malignant\_feature = benign\_feature \pm benign\_feature \times rand([10\%\ 50\%]) \tag{22}$$

2) Performance Results: Finally, the performance results of the proposed method are collated in different settings. Firstly, we investigate the performance of the forecasting model at varying numbers of participating devices and in the timesteps sequence of 5, 7, and 9. Accordingly, we collate the RMSE from the collaborating devices and compute the average value at every setting. For comparison purposes, we also gathered RMSE from each while building model individually. Fig. 7 displays the average RMSE value at every communication round or epoch. It can be observed in the figure that when devices individually build the model, the average RMSE decreases slowly. On the other hand, as the number of model-building devices increases, the average RMSE between collaborating devices decreases at shorter communication rounds. It should be noted that the figure only presents the results when the timestep sequence is set to 5, and the same trend was observed in the other settings. To this end, the input training sample with five timesteps sequence produces the lowest RMSE from 5 collaborating devices as shown in Table V.

Subsequently, we investigate the accuracy of the classification model at varying collaborating devices while adapting the timesteps sequence with the lowest average RMSE from the first experiment. As shown in Fig. 8, when more devices participate in the model building, the learning rate increases in shorter communication rounds. Finally, Table VI the comparison of the average classification performance of different deep learning approaches. On the one hand, the size of the BiLSTM model has decreased by 43% after applying post-quantization method, and the inference latency has reduced by almost 94%. However, both indicators are relatively higher than the other neural network structures due to the significantly more complex structure. On the other hand, the proposed BiLSTM-based MDS dominates the other approaches, achieving a high recall of 99.93%. This

TABLE IV
DATASET DETAILS USED IN THE EXPERIMENT

Purpose	Total Size	Remarks
Device 1 Training	3532 blocks	2668 benign and 864 malicious
Device 2 Training	3140 blocks	2387 benign and 753 malicious
Device 3 Training	2748 blocks	2080 benign and 668 malicious
Device 4 Training	2356 blocks	1771 benign and 585 malicious
Device 5 Training	1963 blocks	1498 benign and 465 malicious
Validation Set	19631 blocks	14846 benign and 4785 malicious

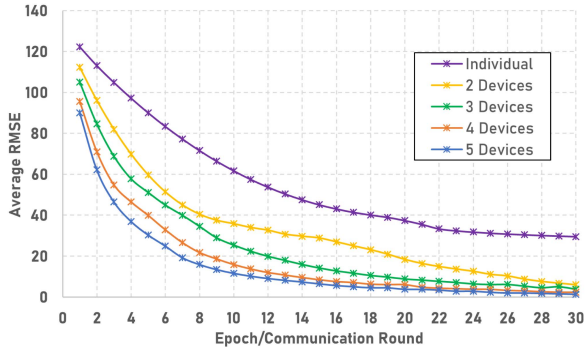

Fig. 7. Average RMSE at every epoch/communication round in varying collaborating devices using 5 timesteps sequence.

TABLE V
AVERAGE RMSE OF 5 COLLABORATING DEVICES AT VARYING TIME-STEP SEQUENCE

Timestep Sequence	Average RMSE
5	1.3482
7	1.9959
9	2.6099

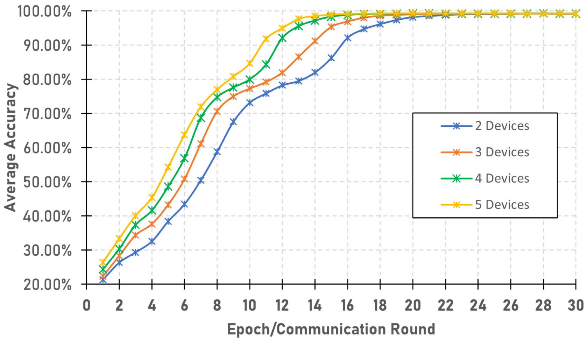

Fig. 8. Average classification accuracy at every communication round in varying collaborating devices using 5 timesteps sequence.

TABLE VI
AVERAGE PERFORMANCE COMPARISON OF DIFFERENT NEURAL NETWORK STRUCTURES

Performance Metrics	Algorithm MLP [42]	CNN [42]	BiLSTM
Accuracy	98.71%	99.17%	99.74%
Precision	98.79%	99.07%	99.72%
Recall	98.69%	99.24%	99.93%
F1-Score	98.73%	99.16%	99.83%
Latency (ms)	0.1930	0.7127	1.1110 (**17.2256)
Mode size (bytes)	1553	1749	25161 (**43677)

** Performance of the base model.

indicates that the proposed model can capture almost all potential malicious events in the target use case.

B. Blockchain Environment Setup

In order to connect with the Ethereum network, we utilize Truffle - Suite framework for testing and deploying the instruction within the smart contract. The functions of smart contracts can be adjusted over time through the migrations feature provided by the framework. The smart contract is executed by a computation engine that acts as a decentralized computer network known as the Ethereum virtual machine (EVM). Every blockchain node operates on the EVM to maintain consensus across the network [49]. The EVM can be understood as a mathematical function that takes any given input to produce a deterministic output with a state transition function. Ethereum-enabled applications using Truffle Suite execute and inspect the state transition function with all essential dependencies installed. This framework is an *automining* mode and running on the remote call server http://127.0.0.1:7545 (network ID: 5777) with the currency symbol is '*Ether*'. The gas price and gas limit are set to be 20,000,000,000 Wei, and 6,721,975 respectively. Prefix values of hexadecimal number and the unique identification of the entities are obtained from Truffle - Suite framework.

C. Compatible Privacy Preservation

The Blockchain-based privacy preservation framework is formed by deploying transactions (e.g., TX_PET_n patient's data) into the Ethereum network. The patients state their relevant information in the smart contract, such as the primary diagnosis function, description, actual data, and details. The patient also attaches a pair of public keys to be used mutually with the recipient's random data (in line with Diffie-Hellman key exchange), while the other key acts as a tracking key. The fundamental objectives of smart contract adoption in this research are the immutable transactions record and commensurate incentive mechanism. Tamper-proof property is achieved by design, while the cost of incentive given to the data owner is proportional to the data amount. We deploy our contracts to the Ethereum network by running the migration file. Over time, the contract functions can be changed by re-running the migrations file (responsible for staging the deployment tasks). The network id and block gas limit is set to be 5777 and 6721975 (0x6691b7), respectively. When the contract is successfully deployed to the Ethereum network, then the contract will have an address. Eventually, the migration process details are displayed together, such as the number of blocks, current balance (*Ether*), block timestamp, account, gas used in deploying the contract, gas price, the value sent, and total cost. Roughly, the contract migration process consumes 225237 gas units, with a total cost is 0.00450474 ETH.

Fig. 9. The detail of information that the data owner has conducted. This transaction is mined in *Block 5* with 111367 gas usage.

In deploying the contracts, the genesis block is created automatically with zero units of gas used, and it is recorded in the *Block 0* on the chain. The genesis block is perpetually hardcoded into the software of the applications that use the Ethereum smart contract. The first transaction TX_PET_1 of patient 1 PET_1 is recorded in the *Block 5* with transaction index is 0, gas used is 111367 units (6721975 gas limits), and the ID of transaction is 'log_125444eb'. Intuitively, a timestamp for every block can be understood as a block generation after getting confirmation from the miners in the network. When the node receives a new block from another node in the network, the recipient confirms that the timestamp value is correct and does not outpace the Universal Time Coordinated (UTC) by more than 100 milliseconds. Otherwise, the block is rejected. The information of this transaction can be seen in Fig. 9. The number of the block is begun with the genesis block (*Block 0*), contract creation block, contract deployment block, and the block of the patient's transaction (*Block 5*). By the framework default, every transaction conducted by entities is recorded into one block. For instance, when the five patients upload their diagnosis data into the blockchain network, the smart contract records the data into five consecutive blocks.

To estimate the gas consumption and *Ether* spent, we captured 20 consecutive transactions TX_PET_1 deployed by two patients TX_PET_1 and TX_PET_2 (40 transactions in total). The arbitrary values of transactions stated in the smart contract are varied. The input of transactions is distinct by the type of diagnosis data, the description, IPFS content identifier,

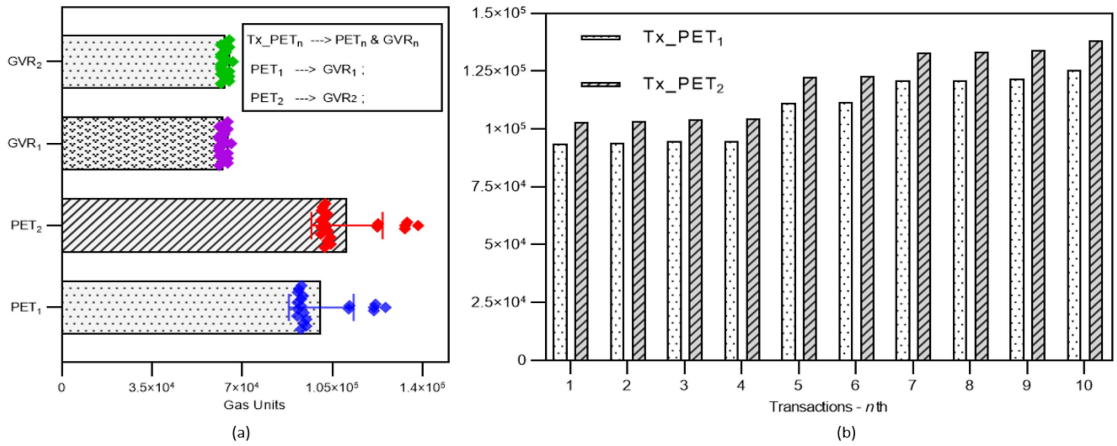

Fig. 10. (a) Comparison of Ethereum gas usage between patients and healthcare providers in conducting transactions. Two patients with their respective data conducted TX_PET_1 and TX_PET_2, respectively; (b) Comparison for the last ten transactions of patients.

TABLE VII
SUMMARY OF THE CUMULATIVE GAS CONSUMPTION AND *ETHER* SPENT BY PET_1 AND GVR

No.	Benchmark	PET_1	PET_2	GVR_1	GVR_2
1	*Ether* amount (init.)	100 ETH	100 ETH	100 ETH	100 ETH
2	Gas usage (min)	91468	100615	61141	61752
3	Gas usage (max)	125721	138293	65754	66411
4	Gas usage (avg)	100717	110789	62846	63474
5	*Ether* spent (min)	5.56 x 10^{-3}	1.82 x 10^{-2}	6.42 x 10^{-3}	1.94 x 10^{-2}
6	*Ether* spent (max)	9.87 x 10^{-3}	7.94 x 10^{-2}	9.57 x 10^{-3}	3.44 x 10^{-2}

TABLE VIII
THE INTERPLANETARY FILE SYSTEM NETWORK BENCHMARK

Benchmark	Details
Sender_1 ID	*12D3KooWNqta5mdmLYYE2GWa9Z ....xxx*
Public key	*CAESIMGL4w62ZkS3clCVUQtlCMry ....xxx*
AgentVersion & UI ver.	go-ipfs v0.11.0 & v2.13.0
API	/ip4/127.0.0.1/tcp/5001
Data size & Peers avb.	966 KB; 125 Peers (dynamically changing)
Content identifier (CID)	*QmdeFRzC3VhQyecQtoWiamyyZ ....xxx*
String structure	*Object type: "file", data: undefined, blockSizes: Array[4]*
Links	4 links (Path: Links/0, Links/1 - Links/3)
Multihash function	*0x1220E3617...xxx (0x12 = sha2-256; 0x20 = 256 bits)*
Avg. Network traffic	76 KiB/s incoming; 32 KiB/s outgoing (3 mins pre - post transactions)

the details, etc. The sequence of patient's activities are linked with the government healthcare provider transactions GVR_1 for transaction PET_1, and GVR_2 for PET_2. GVR provides the reward for the data owner in a secure manner, whereby the information of corresponding entities is confidential. The lowest gas consumption of the PET_n's transaction is recorded to be 91468 units (PET_1). The maximum use of gas is around 100717 units, with an average of 100717 units for the 20 transactions that have been carried out. On the other hand, PET_2's lowest gas used is 100615 units; the maximum usage is 138293 units with 110789 average units.

The visual output of the transaction is shown in Fig. 10. The knowledge about *Ether* spent in Table VII describe an accumulated calculation of gas usage, gas limits, and the gas price calculated automatically by the Truffle - Suite framework. The IPFS network benchmark used in our framework can be seen in Table VIII. Every entity possesses a UNIX ID (ipfs-unixfs) and public key connected through a specific gateway, i.e. http://127.0.0.1:8080. The agent and UI versions are go-ipfs v0.11.0 and v2.13.0. As highlighted in Table VIII, the CID and multihash functions are derived once the data are successfully stored. Eventually, our proposed scheme can be a plausible solution to address the privacy and transparency issues in the blockchain. Nevertheless, the hard fork in the blockchain is needed since the core of protocols is updated that can be a significant obstacle in embracing the proposed model in the real world.

V. CONCLUSION

We have presented a blockchain-based privacy preservation framework for secure misbehavior detection employed in lightweight IoT devices. The privacy concerns are tackled by diversifying several cryptography protocols that cut off the linkage between private data and the corresponding owner. We also applied the FL strategy that enhances patient privacy by keeping training data within the owner's digital realm and building the global model out of sub-models trained locally by participating devices. Furthermore, the entities' information is expressly disguised by an Ethereum smart contract. We have completed the main requirements stated in this research, such as privacy preservation and a commensurate incentive mechanism that legitimate entities can only recognize and the results of the variant neural network, which prove the effectiveness of the proposed model.

REFERENCES

[1] P. P. Ray, D. Dash, and N. Kumar, "Sensors for internet of medical things: State-of-the-art, security and privacy issues, challenges and future directions," *Comput. Commun.*, vol. 160, pp. 111–131, 2020.

[2] A. Ghubaish, T. Salman, M. Zolanvari, D. Unal, A. K. Al-Ali, and R. Jain, "Recent advances in the internet of medical things (IoMT) systems security," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 8707–8718, Jun. 2021.
[3] M. Papaioannou et al., "A survey on security threats and countermeasures in internet of medical things (IoMT)," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 6, 2020, Art. no. e4049.
[4] S. M. Karunarathne, N. Saxena, and M. K. Khan, "Security and privacy in IoT smart healthcare," *IEEE Internet Comput.*, vol. 25, no. 4, pp. 37–48, Jul./Aug. 2021.
[5] R. M. Aileni, G. Suciu, C. Valderrama, and S. Pasca, "Iot performability for medical wearable device by data privacy and fault tolerance," in *Smart Systems for E-Health.* Cham, Switzerland: Springer, 2021, pp. 113–133.
[6] S. S. Gopalan, A. Raza, and W. Almobaideen, "IoT security in healthcare using AI: A survey," in *2020 Int. Conf. Commun., Signal Process., Appl. (ICCSPA)*, 2021, pp. 1–6.
[7] P. V. Astillo, G. Choudhary, D. G. Duguma, J. Kim, and I. You, "TrMAps: Trust management in specification-based misbehavior detection system for IMD-enabled artificial pancreas system," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 10, pp. 3763–3775, Oct. 2021.
[8] H. Sedjelmaci, S. M. Senouci, and N. Ansari, "A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1594–1606, Sep. 2018.
[9] H. Sedjelmaci, S. M. Senouci, and N. Ansari, "Intrusion detection and ejection framework against lethal attacks in UAV-aided networks: A Bayesian game-theoretic methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1143–1153, May 2017.
[10] G. Choudhary, P. V. Astillo, I. You, K. Yim, R. Chen, and J.-H. Cho, "Lightweight misbehavior detection management of embedded iot devices in medical cyber physical systems," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 4, pp. 2496–2510, Dec. 2020.
[11] M. T. Khan, D. Serpanos, and H. Shrobe, "ARMET: Behavior-based secure and resilient industrial control systems," *Proc. IEEE*, vol. 106, no. 1, pp. 129–143, Jan. 2018.
[12] V. Sharma, I. You, K. Yim, R. Chen, and J.-H. Cho, "BRIoT: Behavior rule specification-based misbehavior detection for IoT-embedded cyber-physical systems," *IEEE Access*, vol. 7, pp. 118556–118580, 2019.
[13] P. Jokar and V. C. Leung, "Intrusion detection and prevention for ZigBee-based home area networks in smart grids," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1800–1811, May 2018.
[14] O. Salem, Y. Liu, A. Mehaoua, and R. Boutaba, "Online anomaly detection in wireless body area networks for reliable healthcare monitoring," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 5, pp. 1541–1551, Sep. 2014.
[15] A. D. Dwivedi, G Srivastava, S. Dhar, and R. Singh, "A decentralized privacy-preserving healthcare blockchain for IoT," *Sensors*, vol. 19, no. 2, 2019, Art. no. 326.
[16] A. Saeed, A. Ahmadinia, A. Javed, and H. Larijani, "Intelligent intrusion detection in low-power IoTs," *ACM Trans. Internet Technol. (TOIT)*, vol. 16, no. 4, pp. 1-25, 2016.
[17] X. Zhu, H. Li, and Y. Yu, "Blockchain-based privacy preserving deep learning," in *Proc. Int. Conf. Inf. Secur. Cryptol.*, 2018, pp. 370–383.
[18] F. Loukil, C. Ghedira-Guegan, K. Boukadi, A.-N. Benharkat, and E. Benkhelifa, "Data privacy based on IoT device behavior control using blockchain," *ACM Trans. Internet Technol. (TOIT)*, vol. 2 1, no. 1, pp. 1-20, 2021.
[19] P. V. Astillo, J. Jeong, W.-C. Chien, B. Kim, J. Jang, and I. You, "SMDAps: A specification-based misbehavior detection system for implantable devices in artificial pancreas system," *J. Internet Technol.*, vol. 22, no. 1, pp. 1–11, 2021.
[20] W. Wang, H. Xu, M. Alazab, T. R. Gadekallu, Z. Han, and C. Su, "Blockchain-based reliable and efficient certificateless signature for IIoT devices," *IEEE Trans. Ind. Inform.*, vol. 18, no. 10, pp. 7059–7067, Oct. 2022.
[21] A. Uprety, D. B. Rawat, and J. Li, "Privacy preserving misbehavior detection in IoV using federated machine learning," in *Proc. IEEE 18th Annu. Consum. Commun. Netw. Conf.*, 2021, pp. 1–6.
[22] I. Makhdoom, "Defense against integrity and privacy attacks in the Internet of Things," Sch. Elect. Data Eng., Univ. Technol. Sydney, Sydney, NSW, Australia, Ph.D. dissertation, 2020.
[23] T. Sultana, A. Almogren, M. Akbar, M. Zuair, I. Ullah, and N. Javaid, "Data sharing system integrating access control mechanism using blockchain-based smart contracts for IoT devices," *Appl. Sci.*, vol. 10, no. 2, 2020, Art. no. 488.
[24] H. Zhang, L. Tong, J. Yu, and J. Lin, "Blockchain aided privacy-preserving outsourcing algorithms of bilinear pairings for Internet of Things devices," *IEEE Internet Things J.*, vol. 8, no. 20, pp. 15596–15607, Oct. 2021.
[25] B. Houtan, A. S. Hafid, and D. Makrakis, "A survey on blockchain-based self-sovereign patient identity in healthcare," *IEEE Access*, vol. 8, pp. 90478–90494, 2020.
[26] W. Meng, W. Li, Y. Wang, and M. H. Au, "Detecting insider attacks in medical cyber-physical networks based on behavioral profiling," *Future Gener. Comput. Syst.*, vol. 108, pp. 1258–1266, 2020.
[27] A. H. Celdrán, M. G. Pérez, F. J. G. Clemente, and G. M. Pérez, "Sustainable securing of medical cyber-physical systems for the healthcare of the future," *Sustain. Comput.: Inform. Syst.*, vol. 19, pp. 138–146, 2018.
[28] S. Nithya, M. Sangeetha, and K. Apinaya Prethi, "Role of cyber physical systems in health care and survey on security of medical data," *Int. J. Pharma Res. Health Sci.*, vol. 6, no. 1, pp. 2075–2080, 2018.
[29] I. Lee and O. Sokolsky, "Medical cyber physical systems," in *Proc. Des. Automat. Conf.*, 2010, pp. 743–748.
[30] R. Mitchell and R. Chen, "Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 1, pp. 16–30, 2014.
[31] W. Schneble and G. Thamilarasu, "Attack detection using federated learning in medical cyber-physical systems," in *Proc. 28th Int. Conf. Comput. Commun. Netw. (ICCCN)*, 2019, pp. 1–8.
[32] T.-T. Kuo, J. Kim, and R. A. Gabriel, "Privacy-preserving model learning on a blockchain network-of-networks," *J. Amer. Med. Inform. Assoc.*, vol. 27, no. 3, pp. 343–354, 2020.
[33] W. Wang et al., "Secure-enhanced federated learning for AI-empowered electric vehicle energy prediction," *IEEE Consum. Electron. Mag.*, early access, Sep. 30, 2021, doi: 10.1109/MCE.2021.3116917.
[34] Z. Lian, W. Wang, H. Huang, and C. Su, "Layer-based communication-efficient federated learning with privacy preservation," *IEICE Trans. Inf. Syst.*, vol. 105, no. 2, pp. 256–263, 2022.
[35] Z. Lian, W. Wang, and C. Su, "COFEL: Communication-efficient and optimized federated learning with local differential privacy," in *Proc. ICC 2021-IEEE Int. Conf. Commun.*, 2021, pp. 1–6.
[36] S. Rahmadika and K.-H. Rhee, "Enhancing data privacy through a decentralised predictive model with blockchain-based revenue," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 37, no. 1, pp. 1–15, 2021.
[37] S. S. Vedaei et al., "COVID-SAFE: An IoT-based system for automated health monitoring and surveillance in post-pandemic life," *IEEE Access*, vol. 8, pp. 188538–188551, 2020.
[38] S. Gyawali, Y. Qian, and R. Q. Hu, "Machine learning and reputation based misbehavior detection in vehicular communication networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8871–8885, Aug. 2020.
[39] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "A stackelberg game for distributed formation of business-driven services communities," *Expert Syst. Appl.*, vol. 45, pp. 359–372, 2016.
[40] K. A. Althelaya, E.-S. M. El-Alfy, and S. Mohammed, "Evaluation of bidirectional LSTM for short-and long-term stock market prediction," in *Proc. 9th Int. Conf. Inf. Commun. Syst. (ICICS)*, 2018, pp. 151–156.
[41] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proc. IEEE Workshop Autom. Speech Recognit. Understanding*, 2013 pp. 273–278.
[42] P. V. Astillo, D. G. Duguma, H. Park, J. Kim, B. Kim, and I. You, "Federated intelligence of anomaly detection agent in IoTMD-enabled diabetes management control system," *Future Gener. Comput. Syst.*, vol. 128, pp. 395–405, 2022.
[43] H. Wu, P. Judd, X. Zhang, M. Isaev, and P. Micikevicius, "Integer quantization for deep learning inference: Principles and empirical evaluation," *arXiv:2004.09602*, 2020.
[44] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep learning with limited numerical precision," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1737–1746.
[45] S. Noether, "Ring signature confidential transactions for monero," *IACR Cryptol. ePrint Arch.*, vol. 2015, pp. 1–34, 2015.
[46] S. Noether et al., "Ring Confidential Transactions," Ledger, vol. 1, pp. 1–18, 2016.
[47] S. Rahmadika and K.-H. Rhee, "Unlinkable collaborative learning transactions: Privacy-awareness in decentralized approaches," *IEEE Access*, vol. 9, pp. 65293–65307, 2021.
[48] S. Rahmadika, M. Firdaus, S. Jang, and K.-H. Rhee, "Blockchain-enabled 5g Edge Networks and Beyond: An Intelligent Cross-Silo Federated Learning Approach," *Secur. Commun. Netw.*, vol. 2021, pp. 1–14, 2021.
[49] P. Sajana, M. Sindhu, and M. Sethumadhavan, "On blockchain applications: Hyperledger fabric and ethereum," *Int. J. Pure Appl. Math.*, vol. 118, no. 18, pp. 2965–2970, 2018.