Blockchain-based Federated Learning for Bearing Fault Detection in the Industrial Internet of Things

Made Adi Paramartha Putra[*][†], Ahmad Zainudin[‡], Revin Naufal Alief[*], Gabriel Avelino Sampedro[§]
Dong-Seong Kim[*], *Senior Member, IEEE*, and Jae-Min Lee[*], *Member, IEEE*,
[*]Department of IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea.
[†]Informatic Enginering, Primakara University, Denpasar, Indonesia.
[‡]Department of Electronic Engineering, Kumoh National Institute of Technology, Gumi, South Korea.
[§]Faculty of Information and Communication Studies, University of the Philippines Open University, Laguna, Philippines
{mdparamartha95, zai, revinnaufal, garsampedro, dskim, ljmpaul}@kumoh.ac.kr

***Abstract*—The reliability of Industrial Internet of Things (IIoT) systems requires robust fault detection, which can be achieved with AI. However, the current centralized learning approach is inefficient. Federated Learning (FL) solves this problem by enabling distributed training without exposing individual information. This article proposes a parameter aggregation technique for bearing fault detection in IIoT using a lightweight smart contract with a PoA-based blockchain consensus. The findings indicate that the proposed system provides a secure aggregation process with an accuracy of 94.00% and a processing time of 1.54 s, which is suitable for the IIoT environment.**

***Index Terms*—Bearing fault detection; blockchain; federated learning; industrial internet of things;**

I. Introduction

The Industrial Internet of Things (IIoT) applies connected technology to monitor industrial processes and manufacturing, and fault detection is a critical component of IIoT [1]. Sensors and monitoring devices continuously monitor equipment, systems, or processes, transmitting data to a central control system [2]. Fault detection algorithms rely on advanced technologies such as Artificial Intelligence (AI) and predictive analytics to analyze data from multiple sources, identify potential faults, and take corrective action [3]. AI-based fault detection in IIoT can enhance industrial systems' efficiency, reliability, and reduce maintenance costs. The IIoT environment typically uses centralized learning, which involves collecting data from multiple sensors and transmitting it to a single server for analysis. However, this approach has limitations, such as increased network latency and traffic overhead, data security and privacy concerns, and the risk of single points of failure. The large volume of data generated in IIoT systems can also make it impractical to transmit all data to a central location for AI model updating [4].

Federated Learning (FL) is a decentralized training technique that addresses the limitations of centralized learning in the IIoT environment. FL allows distributed training to be performed on the client side, preserving privacy and creating robust models without sharing client data [5]. However, FL must consider factors such as device heterogeneity, autonomy, security, and data distribution [6]. Techniques such as client selection and implementing security measures such as data privacy, access control, and encryption can optimize FL in the IIoT. In our previous work proposed an accuracy-based client selection method that ranks clients based on the centralized evaluation [7]. The security of FL is crucial to prevent white-box attacks that could lead to model corruption. Techniques such as differential privacy, homomorphic encryption, and blockchain can mitigate these attacks. Blockchain offers decentralized and secure parameter aggregation by using various consensus mechanisms such as Proof of Work (PoW), Proof of Stake (PoS), and Proof of Authority (PoA), making it a suitable option for FL in IIoT environments [8], [9].

A blockchain network can be used to store the updated parameters, and an incentive mechanism can be applied to reward clients for their contributions [10]. However, the efficiency of these techniques in the IIoT environment is still not fully explored, and a secure and efficient approach is needed to accommodate various types of machinery. The main contribution of this article is detailed as follows:

1) We propose a blockchain-based secure parameter aggregation architecture for FL in IIoT. In the proposed design, every procedure related to the model parameter is executed on the blockchain network for security purposes.
2) We propose an efficient Deep Learning (DL) called DC-MLP, a multilayer perception model. It is designed to address fault detection challenges in IIoT FL settings. We prioritize detection accuracy without sacrificing inference time.
3) To evaluate the performance and reliability of the proposed blockchain-based secure aggregation mechanism using the DC-MLP model, we conduct an extensive evaluation over 5-fold cross-validation. The evaluation is performed on both public and private architectures.

The structure of this article is as follows: Section II covers related literature and any research gaps. Section III describes the proposed blockchain-based parameter aggregation methods, the DL model, the dataset, and the simulation setup. Section IV presents the results and discussions of the study, while Section V concludes the paper and outlines future research opportunities.

979-8-3503-1327-7/23/$31.00 ©2023 IEEE

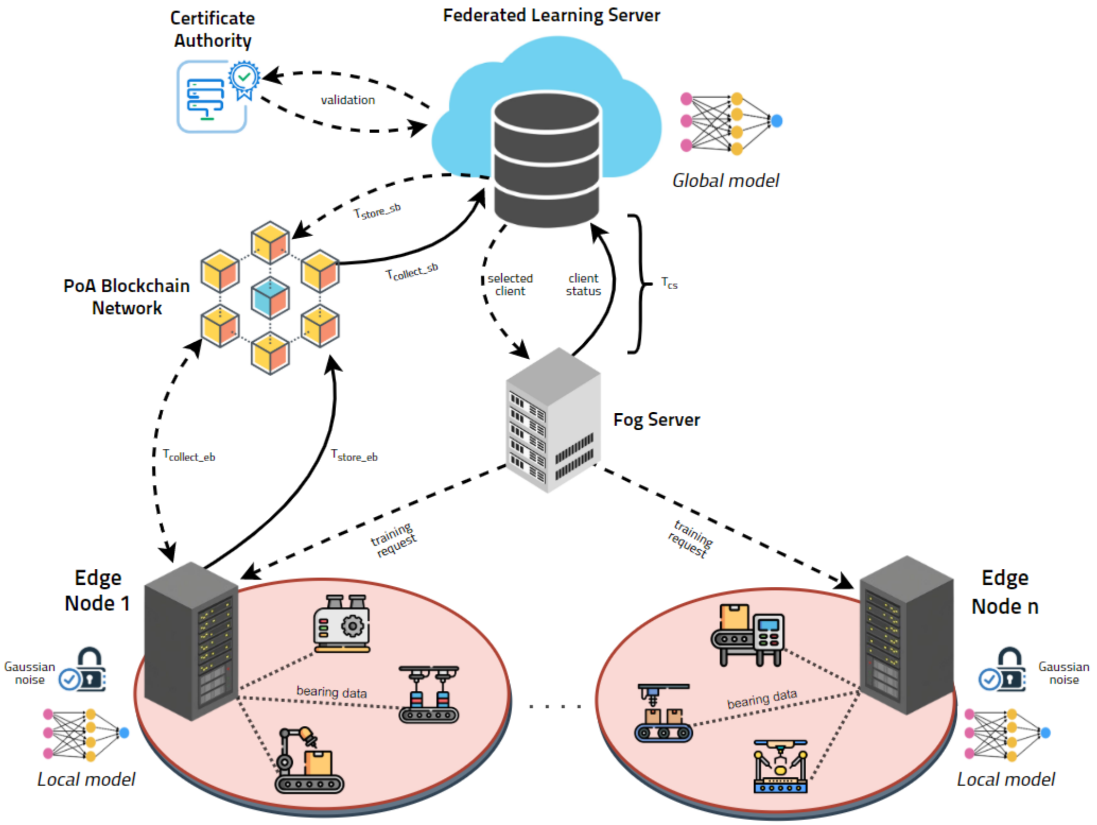


Fig. 1. The overall design of the secure aggregation procedure for FL using a PoA consensus.

II. Related Works

A. Secure Aggregation in Federated Learning

The use of encryption is a well-known technique to prevent data leakage and protect information. In [11], the authors propose using Advanced Encryption Standard (AES), a symmetric-key encryption method, to encrypt and decrypt updated local parameters in FL. However, AES may not be as secure with increasing clients and communication rounds as the same key is used. Another approach, using Functional Encryption (FE), was studied in [12], where AONT was utilized to secure partial information of the updated local parameters through matrix transformation. The authors showed that the FE scheme is more efficient than the Homomorphic Encryption (HE) scheme. Despite the benefits of encryption, vulnerabilities still exist, such as an attacker intercepting encrypted packets or collecting downloaded global parameters.

B. Blockchain-based Secure Aggregation

Blockchain provides decentralized access to global parameters and update functions through smart contract functions, making it a suitable tool for federated learning. However, the implementation of blockchain in IIoT is still a new area and poses several challenges, such as ensuring network efficiency, security, and privacy. Previous studies have mainly focused on general IoT implementation without considering the time constraints of IIoT. Some studies have proposed blockchain-based solutions, such as ChainFL [13] and Shapley value [14], but their impact on FL efficiency has not been fully evaluated. Additionally, some studies have proposed secure aggregation in trusted execution environments, but these approaches may not be suitable for IIoT networks due to their high processing time [15]. This paper proposes a blockchain-based aggregation mechanism that ensures security and efficiency in FL for IIoT environments, with an average processing time of less than one second.

III. Proposed System

A. System Model

The system model proposed in this work is shown in Fig. 1. It comprises four main components: edge nodes, fog server, FL server, and blockchain networks. The proposed architecture is specifically designed to address TSP issues in the IIoT environment by performing distributed learning for privacy. It also includes a certificate authority (CA) for public key infrastructure (PKI) and a blockchain network for trust and security problems. The edge nodes are grouped into n groups denoted by $E = 1, 2, 3, ..., n$, and they are connected to a fog server, which is connected to an FL server. The CA is used to generate private and public keys for each participant in the FL process. The FL technique is implemented with a PoA-based consensus to provide secure information transmission

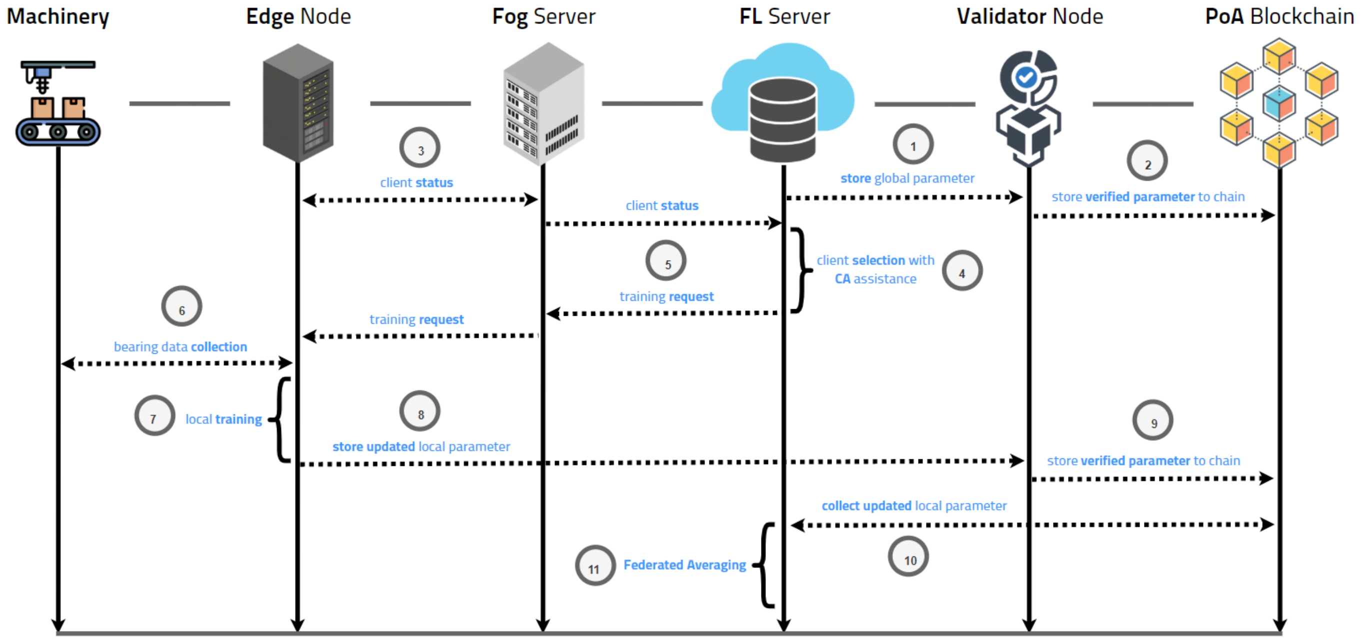


Fig. 2. Communication flow of the proposed PoA-based FL for intelligent fault classification.

on the network. The model parameter is fully stored on the blockchain network, supervised by numerous validators, using the PoA approach.

$$T_r = T_{store_{sb}} + T_{cs} + T_{train} + T_{collect_{eb}} + T_{store_{eb}} . \quad (1)$$

The equation (1) calculates the total delay for a single communication round in the proposed architecture. It consists of several components, including the time needed to store global parameters on the blockchain network ($T_{store_{sb}}$), the time required by the FL server to select participants based on their status (T_{cs}), and the time needed to train the local model (T_{train}), which consists of three sub-processes: parameter download from the blockchain ($T_{collect_{eb}}$), local training, and parameter update to the blockchain after the local training process is completed ($T_{store_{eb}}$). Finally, the FL server collects all updated parameters from the blockchain network within the $T_{collect_{sb}}$ interval before the aggregation process.

B. Proposed Blockchain-based Secure Aggregation

The communication process for the proposed blockchain-based FL for intelligent bearing fault detection is shown in Fig. 2. The proposed system is designed to maintain distributed learning while preserving privacy for each FL participant. In this study, we assume that the local training for FL is carried out on the edge node connected to the fog server and FL server. Instead of sharing parameter information directly through a traditional network, we use a PoA-based network as the blockchain consensus. There are several factors that support the use of a PoA-based blockchain in this study, including:

- PoA consensus is exceptionally lightweight compared to the other consensus algorithm (e.g., PoW). A fast transaction time is crucial for the FL process because the updated parameter needs to be aggregated within a short period of time. The adoption of PoW is not sufficient to cover these requirements.
- Compared to the PoS ‘algorithm, the PoA consensus algorithm still provides faster transaction times while also providing sufficient security features.
- Parameter aggregation in FL requires fast data transmission to shorten the federation process. Thus, the PoA-based blockchain is suitable for the use case scenario investigated in this work.

The algorithm for the proposed PoA-based secure parameter aggregation in the FL process is described in Algorithm 1. Initially, some parameters and functions are initialized along with the smart contract. Then, the FL server performs the federation by iterating through communication rounds t_{th}. In each round, the FL server chooses a particular client who is validated by the CA and stores the global parameter with a round number on the blockchain. Next, the selected participants collect the global parameter from the blockchain and perform local training to get updated local parameters using the following equation:

$$\omega_l^k \leftarrow \omega_g - \eta \nabla \ell \left(\omega_g ; b \right) . \quad (2)$$

where the updated local model ω_l^k from client k is calculated based on the global model ω_g and training results from a batch b of local data β with a learning rate η.

Every client k in the set of clients K is assigned a timeout interval of $T_i = 600$ seconds. If a client fails to transmit

Algorithm 1: PoA-based Parameter Aggregation

```
Initialize global DL parameter: $\omega_g$
Initialize other parameters: $C, K,\ t,\ b,\ \eta,\ e$
Initialize the client's private and public keys
Initialize the client's blockchain address

**FL Server** executes:
**for** each communication round $t$ **do**
    $K_{valid} \leftarrow$ validated client from CA
    $S_k \leftarrow K_{valid} \,.\, C$
    Store initial global parameter $\omega_g,\ t$ to PoA blockchain
    **for** each $k \in S_k$ in parallel **do**
        $LocalTraining(\omega_g,\ t)$
    **end for**
    $\omega_k \leftarrow$ Collect parameter from PoA blockchain
    $\omega_{g_{t+1}} \leftarrow$ aggregate new global parameter using
    equation (3)
    $Acc, F_1 \leftarrow globalmodel(\omega_{g_{t+1}}).evaluate()$
**end for**

**Function** $LocalTraining(\omega_c,\ t)$
$\omega_l \leftarrow$ Collect $\omega_g$ from PoA blockchain
**for** local epoch 1 to $e$ **do**
    **for** batch $b \in B$ **do**
        $\omega_l \leftarrow$ update local parameters using equation (2)
    **end for**
**end for**
Store updated local parameter $\omega_l,\ t$ to PoA blockchain
```

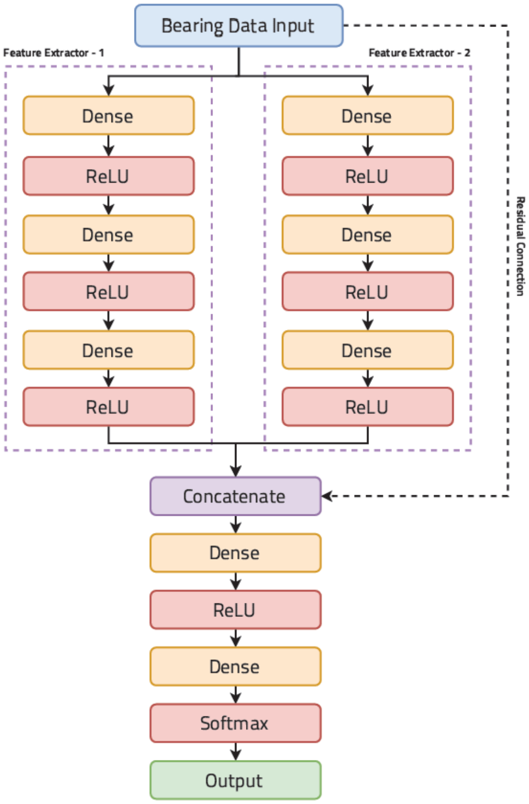


Fig. 3. The overall architecture of the proposed DC-MLP model.

their updated local model parameters ω_l within this time, they will not be considered for the parameter aggregation process. Then, the FL server retrieves the updated local parameter from the blockchain network. After collecting all the updated local parameters with a specific round number from the blockchain network, the new global model parameter aggregation process is calculated using the following equation:

$$\omega_{g_{t+1}} \leftarrow \sum_{k=1}^{K} \frac{\eta\, k}{\eta} \omega_{l_{t+1}}^{k} \,. \tag{3}$$

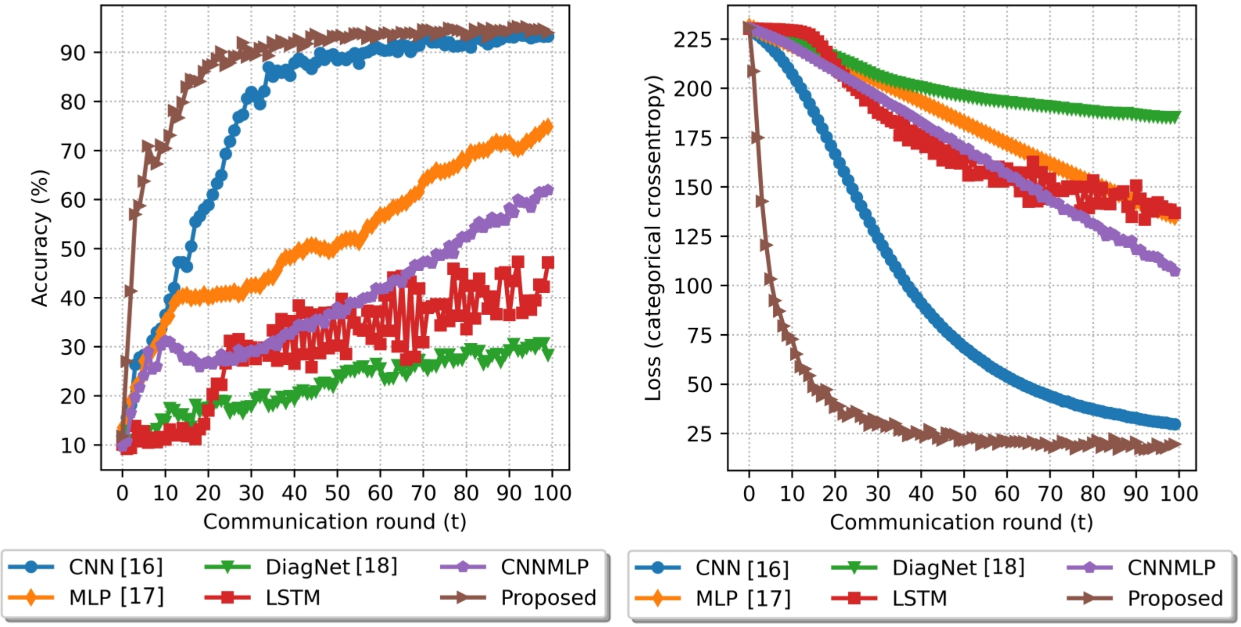
C. Proposed DC-MLP Model

In order to facilitate the intelligent fault detection system proposed for the IIoT environment, a robust and efficient DL model has been designed to extract information from the bearing sensor data and determine the actual condition of the machinery. The overall architecture of this DL model, called the deep concatenated multilayer perceptron (DC-MLP), is shown in Fig. 3. The model is constructed entirely from dense layers, with two extraction flows and a residual connection for better feature extraction. The first extractor uses a larger number of neurons in each dense layer, while the second extractor has a smaller number of neurons. A residual connection is added to allow gradients to flow directly through the network without being affected by activation functions. Additionally, a concatenation layer is used to merge three different inputs into a single output, which is then passed through Fully Connected (FC) and classification layers. In addition, two different activation functions were used. The first activation function is the rectified linear unit (ReLU), which operates based on the threshold value and eliminates the vanishing gradient problem. The second activation function, SoftMax, is applied at the end of the proposed DL model. SoftMax is selected due to its ability to generate an output with the sum of the probabilities equal to one.

D. Dataset and Simulation Details

The proposed system's performance was evaluated using a dataset from the Case Western Reserve University (CWRU) bearing data center, and the effectiveness of the DC-MLP was compared to state-of-the-art studies. The dataset consists of data collected from a machinery motor equipped with a torque transducer, dynamometer, and control electronics. The data was captured at two different sampling rates: 12,000 and 48,000 samples per second. MATLAB was used to process the collected data, which was saved in the .mat file format. The study focused on a shaft rotation speed of 1,772 rpm, and the accelerometers had a sampling frequency of 48 kHz.

The PoA-based FL for intelligent fault detection proposed in this paper was implemented using the Flower framework [19]. Flower is a framework designed for conducting large-scale FL experiments with heterogeneous data distribution among FL clients. It is implemented in the Python programming language



(a) Accuracy

(b) Loss

Fig. 4. FL performance comparison among different DL models investigated in this work with $K = 20$ clients and $C = 0.75$ fraction size.

TABLE I
PERFORMANCE EVALUATION OF NUMEROUS DL MODELS BASED ON 5-FOLD CROSS-VALIDATION TESTED WITH $K = 20$ AND $C = 0.75$.

DL Model	Accuracy (%)	Loss (cross-entropy)	Precision (%)	Recall (%)	F1-Score (%)
CNN [16]	93.26 ± 1.38	29.71 ± 5.89	93.07 ± 1.32	93.38 ± 1.15	93.06 ± 1.32
MLP [17]	74.86 ± 7.52	134.79 ± 6.10	77.82 ± 7.70	74.05 ± 7.76	71.00 ± 8.62
DiagNet [18]	28.30 ± 25.96	185.33 ± 61.63	20.89 ± 27.36	28.78 ± 25.79	20.99 ± 26.47
LSTM	47.17 ± 12.95	136.80 ± 16.85	42.63 ± 12.95	47.01 ± 12.74	41.36 ± 13.44
CNNMLP	61.95 ± 9.10	107.21 ± 13.99	58.86 ± 7.63	62.44 ± 8.05	56.17 ± 9.03
Proposed	**94.00 ± 1.30**	**19.54 ± 5.62**	**94.43 ± 1.05**	**93.98 ± 1.27**	**94.00 ± 1.29**

and can be modified to fit the proposed architecture in this work.

Moreover, in the blockchain-based on PoA, multiple tools are used, including Web3 libraries, which allow interaction with the blockchain network through a Web3 provider, such as Infura or Alchemy. These libraries are available in Python, making it easy to synchronize with the Flower framework. The Ganache library is utilized to manage account creation in local deployment, while five distinct accounts are used in the public blockchain infrastructure. Furthermore, the smart contract for the suggested architecture is developed, deployed, and tested using the Remix Integrated Development Environment (IDE).

IV. PERFORMANCE EVALUATION

The evaluation process involves a comparison of the proposed DC-MLP model with state-of-the-art DL models, including CNN [16], MLP [17], and DiagNet [18]. Furthermore, the performance of vanilla models of LSTM and CNNMLP architectures is also examined. The evaluation setup involves 20 clients with 0.75 fraction sizes and 100 communication rounds. The performance evaluation results in terms of accuracy are presented in Fig. 4(a). The proposed DC-MLP model demonstrates exceptional performance, outperforming other DL models with a significant average accuracy of $94.00 \pm 1.30\%$. It is also noted that the proposed model can achieve higher performance with fewer communication rounds, indicating a faster model convergence rate. Additionally, Fig. 4(b) demonstrates that the proposed DC-MLP model has the best categorical cross-entropy loss performance compared to other DL models, with an average loss value of 19.54 ± 5.62.

Additionally, a comprehensive performance evaluation is carried out using precision, recall, and F1-score metrics. The results are presented in Table I, which shows the overall performance of five DL models, including the proposed DC-MLP, based on the average value over 5-fold cross-validation. The proposed DC-MLP achieved the highest F1-score performance of $94.00 \pm 1.29\%$, followed by CNN and MLP with an average F1-score of $93.06 \pm 1.32\%$ and $71.00 \pm 8.62\%$, respectively.

V. Conclusion and Future Work

The article outlines a technique for secure and efficient parameter aggregation in the FL process using a PoA-based consensus and investigates public and private blockchain network implementations. A DL model named DC-MLP is developed to achieve fast processing times for classifying bearing status. The system is evaluated on a real-world bearing fault dataset using 5-fold cross-validation and achieves 94.00% accuracy compared to state-of-the-art DL models for bearing fault detection using the FL approach. The secure and efficient parameter aggregation proposed in this work is evaluated with an average transaction time of 1.54 s and 24.39 s for private and public blockchain networks, respectively, under various numbers of FL participants. The findings suggest that the proposed system is well-suited for FL implementation in the IIoT environment using a private PoA-based blockchain network. Future work should explore optimization techniques that are tailored to the FL process in the IIoT environment, such as client selection and local training, to improve efficiency.

Acknowledgment

This work was supported by Priority Research Centers Program through the NRF funded by the MEST(2018R1A6A1A03024003) and by the MSIT, Korea, under the Innovative Human Resource Development for Local Intellectualization support program (IITP-2023-2020-0-01612) supervised by the IITP and by NRF-2022R1I1A3071844.

References

[1] W. Zhang, Q. Lu, Q. Yu, Z. Li, Y. Liu, S. K. Lo, S. Chen, X. Xu, and L. Zhu, "Blockchain-based federated learning for device failure detection in industrial iot," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5926–5937, 2021.

[2] M. Banerjee, C. Borges, K.-K. R. Choo, J. Lee, and C. Nicopoulos, "A hardware-assisted heartbeat mechanism for fault identification in large-scale iot systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 1254–1265, 2022.

[3] G. Kim, J. G. Choi, M. Ku, H. Cho, and S. Lim, "A multimodal deep learning-based fault detection model for a plastic injection molding process," *IEEE Access*, vol. 9, pp. 132 455–132 467, 2021.

[4] S. R. Saufi, Z. A. B. Ahmad, M. S. Leong, and M. H. Lim, "Challenges and opportunities of deep learning models for machinery fault detection and diagnosis: A review," *IEEE Access*, vol. 7, pp. 122 644–122 662, 2019.

[5] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140 699–140 725, 2020.

[6] L. Li, Y. Fan, and K.-Y. Lin, "A survey on federated learning," in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, 2020, pp. 791–796.

[7] M. A. P. Putra, A. R. Putri, A. Zainudin, D.-S. Kim, and J.-M. Lee, "Acs: Accuracy-based client selection mechanism for federated industrial iot," *Internet of Things*, vol. 21, p. 100657, 2023.

[8] J. Heiss, E. Grunewald, S. Tai, N. Haimerl, and S. Schulte, "Advancing blockchain-based federated learning through verifiable off-chain computations," in *2022 IEEE International Conference on Blockchain (Blockchain)*. Los Alamitos, CA, USA: IEEE Computer Society, aug 2022, pp. 194–201.

[9] D. Hou, J. Zhang, K. L. Man, J. Ma, and Z. Peng, "A systematic literature review of blockchain-based federated learning: Architectures, applications and issues," in *2021 2nd Information Communication Technologies Conference (ICTC)*, 2021, pp. 302–307.

[10] J. Qi, F. Lin, Z. Chen, C. Tang, R. Jia, and M. Li, "High-quality model aggregation for blockchain-based federated learning via reputation-motivated task participation," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18 378–18 391, 2022.

[11] H. Priya.N, A. H. S M, S. G. S, and P. Rathika, "Improving security with federated learning," in *2021 International Conference on Computational Performance Evaluation (ComPE)*, 2021, pp. 234–239.

[12] D. Wu, M. Pan, Z. Xu, Y. Zhang, and Z. Han, "Towards efficient secure aggregation for model update in federated learning," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[13] C. Korkmaz, H. E. Kocas, A. Uysal, A. Masry, O. Ozkasap, and B. Akgun, "Chain fl: Decentralized federated machine learning via blockchain," in *2020 Second International Conference on Blockchain Computing and Applications (BCCA)*, 2020, pp. 140–146.

[14] S. Ma, Y. Cao, and L. Xiong, "Transparent contribution evaluation for secure federated learning on blockchain," in *2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW)*. Los Alamitos, CA, USA: IEEE Computer Society, 2021, pp. 88–91.

[15] A. P. Kalapaaking, I. Khalil, M. S. Rahman, M. Atiquzzaman, X. Yi, and M. Almashor, "Blockchain-based federated learning with secure aggregation in trusted execution environment for internet-of-things," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1703–1714, 2023.

[16] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, "Federated learning for machinery fault diagnosis with dynamic validation and self-supervision," *Knowledge-Based Systems*, vol. 213, p. 106679, 2021.

[17] G.-Y. Huang and C.-H. Lee, "Federated learning architecture for bearing fault diagnosis," in *2021 International Conference on System Science and Engineering (ICSSE)*, 2021, pp. 408–411.

[18] Z. Yao, C. Jin, M. Ragab, K. M. M. Aung, and X. Li, "Diagnet: Machine fault diagnosis using federated transfer learning in low data regimes," in *2022 FL Workshop of Association for the Advancement of Artificial Intelligenb26ce (AAAI)*, 2022.

[19] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, T. Parcollet, and N. D. Lane, "Flower: A friendly federated learning research framework," *arXiv preprint arXiv:2007.14390*, 2020.