Privacy aware-based federated learning framework for data sharing protection of internet of things devices

Yuris Mulya Saputra, Ganjar Alfian
Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada, Yogyakarta, Indonesia

Article Info

Article history:

Received Oct 17, 2022
Revised Mar 31, 2023
Accepted Apr 15, 2023

Keywords:

Artificial intelligence
Data privacy
Encryption
Federated learning
Internet of things

ABSTRACT

Federated learning (FL) has emerged as one of the most effective solutions to deal with the rapid utilization of internet of things (IoT) in big data markets. Through FL, local data at each IoT device can be trained locally without sharing the local data to the cloud server. However, this conventional FL may still suffer from privacy leakage when the local data are trained, and the trained model is shared to the cloud server to update the global prediction model. This paper proposes a FL framework with privacy awareness to protect data including the trained model for IoT devices. First, a data/model encryption method using fully homomorphic encryption is introduced, aiming at protecting the data/model privacy. Then, the FL framework for the IoT with the encryption method leveraging logistic regression approach is discussed. Experimental results using random datasets show that the proposed framework can obtain higher global model accuracy (up to 4.84%) and lower global model loss (up to 66.4%) compared with other baseline methods.

Corresponding Author:

Ganjar Alfian
Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada
Bulaksumur, Sleman, Yogyakarta, Indonesia
Email: ganjar.alfian@ugm.ac.id

1. INTRODUCTION

The rapid utilization of internet of things (IoT) in big data market has widely attracted both academia and industry [1]. A centralized learning, where all data from IoT devices are trained at the cloud server, is one of the most popular big data learning approaches to date [2]. This is because the cloud server has unlimited computing capability to deal with a huge amount of data from various IoT devices. However, such an approach has high probability from information privacy leakage when data from IoT devices are shared to and processed at the cloud server. To minimize the privacy leakage, there exists a local learning where data at each IoT device are trained locally [3]. Nevertheless, this approach may suffer from inherent limited computation of the IoT devices, which then leads to the data training quality.

Recently, federated learning (FL) has emerged as one of the most suitable solutions that can address problems of the aforementioned approaches [4]. Specifically, each participating IoT device can first collect local data and then train them individually to generate a local trained model. Next, all participating IoT devices can share the local trained models to the cloud server, aiming at updating the global model repetitively. Here, the IoT devices do not need to share the real local data to the cloud server to preserve the data privacy [5].

For example, the works in [6]–[18] utilize the FL approach to obtain high-accurate prediction model accuracy in various applications such as geospatial applications, electric vehicle networks, internet of vehicles, intelligent transportation system, and edge computing/caching system. However, such conventional FL can still reveal the sensitive information privacy especially when the local data of IoT devices are trained locally and the generated trained models are shared to the cloud server. To this end, previous works has investigated the possibility of such privacy concern. In particular, the authors in [19] show that the local trained model of a malicious training participant can be utilized to train a duplicate model to disclose the data protection of other participants. Additionally, [20], [21] reveal that attackers can obtain personal information of FL learners by exploiting the local trained models shared to the cloud server.

To address the above problem, privacy-aware based learning approaches can be exploited. For example, prior to sharing the local data to the cloud server, an encryption method can be performed to the local data such that only the encrypted data are sent to the cloud server. To perform this encryption, the authors in [22], [23] utilize a homomorphic encryption which enables operations to be directly performed to the encrypted data without decrypting the data. However, both works do not consider the FL approach. Instead, the whole encrypted data are trained at the cloud server.

Meanwhile, the works in [21], [24] use the FL approach when all the unencrypted local data are trained locally and only the trained local models are encrypted. In this case, the encrypted trained models from the FL learners can be then shared to the cloud server for the encrypted global model update. To produce global model aggregation with privacy awareness, the authors in [25]–[27] leverage a differential privacy approach by adding noised model updates to the aggregated model. Nonetheless, the above works only take the conventional FL into account without any privacy for the local data to be trained.

In this paper, a privacy aware-based FL framework to protect local data training and trained model sharing for IoT devices is proposed. First, a data/model encryption method using fully homomorphic encryption (FHE) is introduced, aiming at protecting the data/model privacy from any malicious IoT devices and/or attackers. Then, the FL framework for the IoT devices with the FHE method leveraging logistic regression ap- proach is developed. To this end, the participating IoT devices not only encrypt the local data, but also the local trained models for the cloud server. Additionally, the global model update at the cloud server also uses the FHE method. In other words, the proposed FL approach fully use the FHE for the entire processes except for the prediction step. This is the first work that combines both data and trained model encryption for the FL process. Based on the experimental results using random datasets, the proposed FL framework can provide comparable global prediction accuracy and less global prediction loss (up to 64%) compared with other baseline methods. In the following sections, the encryption and FL training processes for data and trained models are explained in details. Then, comprehensive comparisons in terms of accuracy, loss, and learning performances are evaluated.

2. METHOD

Generally, the research method is shown in Figure 1. Suppose that there are a cloud server and a set of participating IoT devices, i.e., $\mathcal{K} = \{1, \dots, k, \dots, K\}$. Specifically, the cloud server is connected to K participating IoT devices via wireless connections, e.g., Wi-Fi or cellular networks, in a considered IoT network at a specified period. Both cloud server and all participating IoT devices have computing resources to conduct the training processes for the FL. There are two key procedures that can be executed to comply with the proposed privacy aware-based FL framework in the following: i) pre-training process and ii) training process.

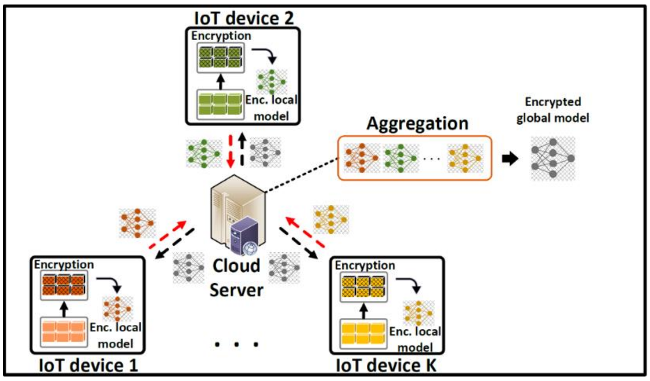


Figure 1. The privacy aware-based FL framework for IoT devices

2.1. Pre-training process

This pre-training process is executed once before the FL process. In this step, each IoT device can first generate IoT data from sensing devices, e.g., sensors and camera. Upon collecting the IoT data from the sensing process, the IoT device can store them at the limited local storage of the IoT device. These data may contain meaningful information such as features and label. Then, the IoT device can implement the data pre-processing to remove some unused data, e.g., incomplete features of the data. At the same time, the cloud server can determine the number of rounds and pre-defined training time at each round for the expected FL process.

Using the data in the local storage, each IoT device performs data encryption using FHE approach to protect the sensitive information privacy and conduct the data operation without a need of data decryption in non-trustworthy areas, e.g., with the existence of third parties [28]. In this ecryption, Cheon-Kim-Kim-Song (CKKS)-based FHE is adopted to provide simultaneous arbitrary operations on encrypted data in real numbers, e.g., addition, substraction, and multiplication. This method works well with polynomials since it has a better efficiency-and-security tradeoff than standard computations on data vectors/matrices [29]. Here, the CKKS-based FHE contains the following polynomial-time algorithms for the encryption processes.

- GenSK(k) to generate the secret key G_k^{sk} for IoT device-*k*.
- GenPK$\left(G_k^{sk}\right)$ to generate the public key as the function of G_k^{sk} IoT device-*k*.
- Encrypt$\left(G_k^{pk}, \delta\right)$ to encrypt data δ using G_k^{pk} with the output of encrypted data δ^e.
- Add(δ_1^e, δ_2^e), Sub(δ_1^e, δ_2^e), and Mul(δ_1^e, δ_2^e) to add, substract, and multiply encrypted data δ_1^e and δ_2^e with the output of δ_+^e, δ_-^e, and δ_*^e, respectively.

Based on the above definitions, $\delta_k = (F_k, L_k)$ can be defined as the total local data at IoT device-*k*, where F_k and L_k are the training feature and training label matrices of the data at IoT device-*k*. Here, each IoT device-*k* can first obtain the secret key G_k^{sk} using GenSK (k) (which is hidden from other IoT devices and the cloud server) and public key G_k^{pk} using GenPK $\left(G_k^{sk}\right)$. Each IoT device-*k* can then encrypt δ_k using G_k^{pk}, i.e., Encrypt $\left(G_k^{pk}, \delta_k\right)$, to obtain the encrypted data $\delta_k^e = (F_k^e, L_k^e)$, where F_k^e and L_k^e are the encrypted training feature and encrypted training label matrices of the data at IoT device-*k*. Upon completing the data encryption, each IoT device can execute corresponding arbitrary operations, e.g., Add (.), Sub (.), and Mul (.) for the FL process later.

2.2. Training process

Upon concluding the pre-training process, the training process using FL between the cloud server and all participating IoT devices in $\mathcal{K}$ can be observed in Figure 1. All participating IoT devices first train their local encrypted data and then share the encrypted local trained models to the cloud server (to preserve the privacy of the local trained models) at each learning round. Upon completing the training process within a pre-defined training time at each round, the encrypted local models from the participating IoT devices can be collected and aggregated by the cloud server with the aim to produce the aggregated encrypted local model. This aggregated encrypted local model is then processed to obtain the current encrypted global model. The updated encrypted global model can be used for the next iteration of the FL process at the participating IoT devices and the cloud server. To this end, this training process are repeated until the encrypted global model converges, or the training time reaches the pre-defined FL time threshold.

To implement the FL process, a logistic regression approach [30] is applied for a binary classification prediction model. Let $f_{k,j}^e$ and $l_{k,j}^e$ denote the *j*-th encrypted feature and *j*-th encrypted label from F_k^e and L_k^e, respectively. Considering J number of rows from F_k^e and L_k^e, and the *j*-th expected encrypted label $\hat{l}_{k,j}^e$, the binary cross entropy loss function with regularization of each participating IoT device-*k*, k ∈ $\mathcal{K}$, at learning round σ can be computed as:

$$l(\xi_k) = -\frac{1}{J}\sum_{j=1}^{J}\left(l_{k,j}^e \, log \, \hat{l}_{k,j}^e + \left(1 - l_{k,j}^e\right) log \left(1 - \hat{l}_{k,j}^e\right)\right) + \frac{\lambda}{2J}\sum_{j=1}^{j^*}\xi_{k,\sigma}^2 \tag{1}$$

where ξ is the j^*-sized weight vector.

Next, to update the encrypted weight vector parameter $\xi_{k,\sigma}$ for each IoT device-*k*, the following equation can be defined:

$$\xi_{k,\sigma} = \xi_{k,\sigma} - \gamma\left(\frac{1}{J}\sum_{j=1}^{J}\left(\hat{l}_{k,j}^e - l_{k,j}^e\right)f_{k,j}^e + \frac{\lambda}{J}\xi_{k,\sigma}\right) \tag{2}$$

where γ is a weight update coefficient. Here, $\gamma = 1$ and $\frac{\lambda}{J} = 0.05$ are set to comply with the FHE constraint and reduce the multiplication such that.

$$\xi_{k,\sigma} = \xi_{k,\sigma} - \left(\frac{1}{J}\sum_{j=1}^{J}\left(\hat{l}_{k,j}^e - l_{k,j}^e\right)f_{k,j}^e + 0.05\xi_{k,\sigma}\right) \tag{3}$$

To approximate the computation of sigmoid activation function on the encrypted local data at each IoT device, a low degree polynomial can be utilized to minimize the use of weight parameters and then optimize the encrypted data computation. Cheon and kim [29], a 3-degree polynomial can approximate the sigmoid activation function within range [-5,5]. For that, the polynomial value for the encrypted data $f^e_{k,j}$ can be expressed by:

$$Pol(f^e_{k,j}) = 0.5 + 0.197 f^e_{k,j} + (f^e_{k,j})^3 \tag{4}$$

at each learning round, the IoT device-k can then send its $\xi_{k,\sigma}$ to the cloud server for the encrypted global model update by aggregating all $\xi_{k,\sigma}, \forall k \in \mathcal{K}$, i.e.,

$$\xi_{\sigma+1} = \frac{1}{K}\sum_{k\in\mathcal{K}} \xi_{k,\sigma} \tag{5}$$

This $\xi_{\sigma+1}$ is then sent back to each participating IoT device for the next FL round. The above process completes when the global model converges, or the learning rounds achieve a given threshold. In this case, the final encrypted global model ξ^* is produced and then can be used to predict the test data for accuracy calculation.

3. RESULTS AND DISCUSSION

To evaluate the performance comparison between the proposed FL with privacy-awareness (i.e., FL with Enc) and other baseline approaches, random datasets with various number of samples and features are used. Specifically, the dataset contains 1,000, 5,000, and 10,000 number of samples. Since the logistic regression is used, binary labels, i.e., 0 and 1, are applied. For the number of features, 2, 3, and 5 features are utilized. In this case, the baseline approaches include centralized learning without encryption (i.e., CL), centralized learning with encryption (i.e., CL with Enc), and conventional FL (i.e., FL). To implement the training processes using the logistic regression, PyTorch CPU 1.10.1 [31] is used. Meanwhile, TenSEAL 0.3.12 library [32] is utilized to execute the encryption method, i.e., FHE. For the FL, 10 active IoT devices with batch size 50 samples are considered.

3.1. Accuracy performance

The accuracy performance comparison is first analyzed in this section. From Table 1, when 1,000 samples are used, the FL with Enc can outperform the CL with Enc in terms of global model accuracy up to 4.09% when 3 features are applied. Additionally, it is observed that the FL with Enc slightly has a lower accuracy compared with the conventional FL. This is due to the approximation activation function of sigmoid when it is used in the encryption process.

When 5,000 samples are executed as can be seen in Table 2, the FL with Enc can still maintain its performance compared with the CL with Enc. Here, the accuracy difference between them is up to 3.09% especially when 5 features are used. The accuracy performance of the proposed FL with Enc gets better when 10,000 samples are used as shown in Table 3. To this end, the proposed FL can improve the accuracy up to 99.7% when 2 and 5 features are utilized. In addition, it can also be summarized that when the number of features gets higher, most of the accuracy performances for all methods get improved to the higher accuracy.

Table 1. The accuracy performance with 1,000 samples

Method	2 features	3 features	5 features
CL	99%	94.4%	99.2%
CL with Enc	96.4%	93.8%	95.4%
FL	97.8%	99.2%	97.4%
FL with Enc	**96.4%**	**97.8%**	**97%**

Table 2. The accuracy performance with 5,000 samples

Method	2 features	3 features	5 features
CL	98.32%	98.2%	98.74%
CL with Enc	96.06%	93.74%	97.52%
FL	99.7%	99.86%	99.74%
FL with Enc	**99.48%**	**95.5%**	**99.7%**

Table 3. The accuracy performance with 10,000 samples

Method	2 features	3 features	5 features
CL	98.32%	98.2%	98.74%
CL with Enc	96.06%	93.74%	97.52%
FL	99.7%	99.86%	99.74%
FL with Enc	**99.48%**	**95.5%**	**99.7%**

3.2. Loss performance

Next, the loss performances of CL with Enc and proposed FL with Enc can be evaluated. Here, a lower loss implies a better learning quality. When 1,000 samples are considered as shown in Table 4, both methods have loss performance between 0.27 and 0.34, where the FL with Enc has lower loss by 21.9% than that of the CL with Enc when 3 features are taken into account. The gap gets higher when 5,000 and 10,000 samples are used (in Table 5 and Table 6, respectively). While the CL with Enc has almost the same loss as that of 1,000 samples scenario, the FL with Enc can even reduce the loss performance to 0.16 and 0.12 when 5,000 samples and 10,000 samples are considered, respectively. Here, the FL with Enc can obtain 54.1% lower loss than that of the CL with Enc when 5,000 samples with 5 features are used. Meanwhile, the FL with Enc can obtain 66.4% lower loss than that of the CL when 10,000 samples with 3 features are used. This indicates that the use of FHE for the FL can improve the loss performance which aligns with the learning quality to produce better global model accuracy.

Table 4. The loss performance with 1,000 samples

Method	2 features	3 features	5 features
CL with Enc	0.3376	0.3508	0.3442
FL with Enc	**0.3082**	**0.2739**	**0.3061**

Table 5. The loss performance with 5,000 samples

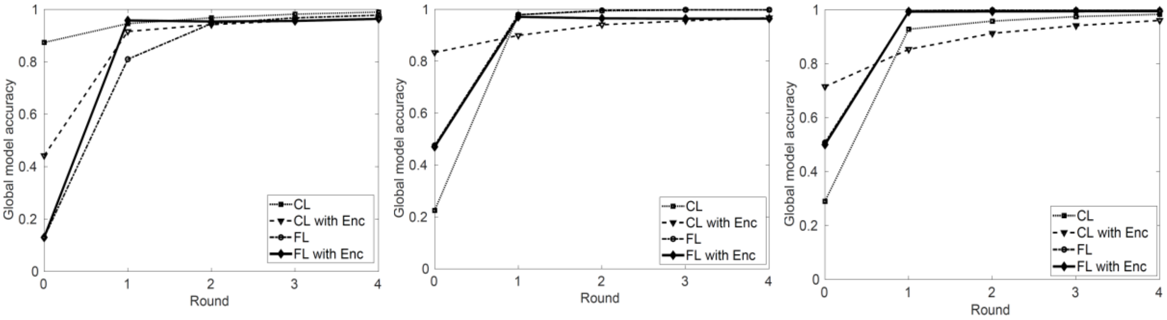
Method	2 features	3 features	5 features
CL with Enc	0.3329	0.3329	0.3635
FL with Enc	**0.1658**	**0.1686**	**0.1668**

Table 6. The loss performance with 10,000 samples

Method	2 features	3 features	5 features
CL with Enc	0.3441	0.377	0.3354
FL with Enc	**0.1238**	**0.1267**	**0.1263**

3.3. Learning performance

To show the accuracy and loss performances in more detail, the learning performance for various scenarios with 5 training rounds is investigated. From Figure 2 when 1,000, 5,000, and 10,000 samples are provided, i.e., Figures 2(a), 2(b), and 2(c), it can be seen that the FL and FL with Enc can outperform the CL and CL with Enc in terms of the dynamic accuracy performance. This can be observed clearly when the number of samples gets higher. In particular, the FL with Enc can follow the accuracy trend of the conventional FL and obtain the accuracy convergence after 5 training rounds at level 99.7%.



(a) (b) (c)

Figure 2. The accuracy performance for different learning approaches with 2 features when (a) 1,000 samples; (b) 5,000 samples; and (c) 10,000 samples are used

For the dynamic performance of loss, the trend can be observed in Figure 3. Despite the FL with Enc have higher loss compared with the CL with Enc at the beginning of training round, the FL with Enc can improve the loss performance significantly for the rest of training rounds. This provides an insight that the use of trained model aggregation and model averaging in the FL process can minimize the error loss more when all the local trained models are combined together prior to sending back to the IoT devices.

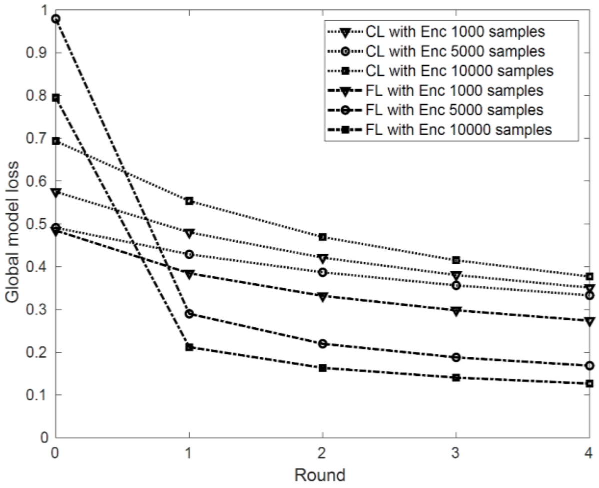


Figure 3. The loss performance for different learning approaches with encryption using 3 features

4. CONCLUSION

In this paper, a FL framework with privacy awareness has been proposed to protect data including the trained model for IoT devices. Particularly, a data/model encryption method using fully homomorphic encryption has been introduced to preserve the data/model privacy. Then, the federated learning framework for the IoT with the encryption method leveraging logistic regression approach has been discussed. Through experimental results using random datasets with various number of samples and features, it has been shown that the proposed FL framework can obtain higher global model ac- curacy up to 4.84% and lower global model loss up to 64.4% compared with other baseline methods, i.e., CL with and without encryption. This implies that the FL with encryption can be the effective solution to replace the CL with and without encryption through producing a higher learning quality.

REFERENCES

[1] B. Pramod, K. Shadaab, and K. Vineet, “Big data and business analytics market statistics,” Alliedmarketresearch, Sep 2021, Accessed: Aug. 01, 2022. [Online.] Available: https://www.alliedmarketresearch.com/big-data-and-business-analytics-market.

[2] C. Zhang, P. Patras, and H. Haddadi, “Deep learning in mobile and wireless networking: a survey,” *IEEE Communications Surveys and Tutorials*, vol. 21, no. 3, pp. 2224–2287, 2019, doi: 10.1109/COMST.2019.2904897.

[3] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, “Application of machine learning in wireless networks: key techniques and open issues,” *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 302–3108, 2019, doi: 10.1109/COMST.2019.2924243.

[4] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, and H. Yu, “Federated Learning,” in *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2020. doi: 10.1007/978-3-031-01585-4.

[5] W. Y. B. Lim *et al.*, “Federated learning in mobile edge networks: a comprehensive survey,” *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020, doi: 10.1109/COMST.2020.2986024.

[6] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, “Distributed federated learning for ultra-reliable low-latency vehicular communications,” *IEEE Transactions on Communications*, vol. 68, no. 2, pp. 1146–1159, Feb. 2020, doi: 10.1109/TCOMM.2019.2956472.

[7] S. R. Pokhrel and J. Choi, “Improving TCP performance over WiFi for internet of vehicles: a federated learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6798–6802, Jun. 2020, doi: 10.1109/TVT.2020.2984369.

[8] Y. M. Saputra, D. N. Nguyen, D. T. Hoang, T. X. Vu, E. Dutkiewicz, and S. Chatzinotas, “Federated learning meets contract theory: economic-efficiency framework for electric vehicle networks,” *IEEE Transactions on Mobile Computing*, vol. 21, no. 8, pp. 2803–2817, Aug. 2022, doi: 10.1109/TMC.2020.3045987.

[9] Y. M. Saputra, H. T. Dinh, D. Nguyen, L. N. Tran, S. Gong, and E. Dutkiewicz, “Dynamic federated learning-based economic framework for internet-of-vehicles,” *IEEE Transactions on Mobile Computing*, vol. 22, no. 4, pp. 2100–2115, Apr. 2021, doi: 10.1109/TMC.2021.3122436.

[10] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” *arXiv:1812.06127*, 2018.

[11] J. Xu and H. Wang, “Client selection and bandwidth allocation in wireless federated learning networks: a long-term perspective,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1188–1200, Feb. 2021, doi: 10.1109/TWC.2020.3031503.

[12] C. Xie, S. Koyejo, and I. Gupta, “Asynchronous federated optimization,” *arXiv:1903.03934*, 2019.

[13] M. R. Sprague *et al.*, “Asynchronous federated learning for geospatial applications,” in *Communications in Computer and Information Science*, vol. 967, 2019, pp. 21–28, doi: 10.1007/978-3-030-14880-5_2.

[14] S. Wang *et al.*, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019, doi: 10.1109/JSAC.2019.2904348.

[15] Y. Sarikaya and O. Ercetin, "Motivating workers in federated learning: a stackelberg game perspective," *IEEE Networking Letters*, vol. 2, no. 1, pp. 23–27, Mar. 2019, doi: 10.1109/lnet.2019.2947144.

[16] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, Sep. 2019, doi: 10.1109/MNET.2019.1800286.

[17] J. Ren, H. Wang, T. Hou, S. Zheng, and C. Tang, "Federated learning-based computation offloading optimization in edge computing-supported internet of Things," *IEEE Access*, vol. 7, pp. 69194–69201, 2019, doi: 10.1109/ACCESS.2019.2919736.

[18] Y. M. Saputra, D. T. Hoang, D. N. Nguyen, E. Dutkiewicz, D. Niyato, and D. I. Kim, "Distributed deep learning at the edge: a novel proactive and cooperative caching framework for mobile edge networks," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1220–1223, Aug. 2019, doi: 10.1109/LWC.2019.2912365.

[19] S. Truex, L. Liu, M. E. Gursoy, L. Yu, and W. Wei, "Demystifying membership inference attacks in machine learning as a service," *IEEE Transactions on Services Computing*, vol. 14, no. 6, pp. 2073–2089, Nov. 2021, doi: 10.1109/TSC.2019.2897554.

[20] C. Xu, J. Ren, D. Zhang, Y. Zhang, Z. Qin, and K. Ren, "GANobfuscator: mitigating information leakage under GAN via differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 9, pp. 2358–2371, Sep. 2019, doi: 10.1109/TIFS.2019.2897874.

[21] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, May 2018, doi: 10.1109/TIFS.2017.2787987.

[22] J. W. Lee *et al.*, "Privacy-preserving machine learning with fully homomorphic encryption for deep neural network," *IEEE Access*, vol. 10, pp. 30039–30054, 2022, doi: 10.1109/ACCESS.2022.3159694.

[23] Z. Yue *et al.*, "Privacy-preserving time-series medical images analysis using a hybrid deep learning framework," *ACM Transactions on Internet Technology*, vol. 21, no. 3, pp. 1–21, Aug. 2021, doi: 10.1145/3383779.

[24] H. Fang and Q. Qian, "Privacy preserving machine learning with homomorphic encryption and federated learning," *Future Internet*, vol. 13, no. 4, p. 94, Apr. 2021, doi: 10.3390/fi13040094.

[25] P. Kairouz, Z. Liu, and T. Steinke, "The distributed discrete gaussian mechanism for federated learning with secure aggregation," in *Proceedings of the 38th International Conference on Machine Learning*, 2021, pp. 5201–5212.

[26] W. N. Chen, C. A. Choquette-Choo, and P. Kairouz, "Communication efficient federated learning with secure aggregation and differential privacy," *NeurIPS 2021 Workshop PRIML Paper50 Decision*, Nov. 2021.

[27] K. Bonawitz, P. Kairouz, B. Mcmahan, and D. Ramage, "Federated learning and privacy," *Communications of the ACM*, vol. 65, no. 4, pp. 90–97, Apr. 2022, doi: 10.1145/3500240.

[28] W. Fu, R. Lin, and D. Inge, "Fully homomorphic imageprocessing," *arXiv:1810.03249*, 2018.

[29] J. H. Cheon and A. Kim, "Homomorphic encryption for approximate matrix arithmetic," *Cryptology ePrint Archive*, 2018.

[30] Wright, R. E. "Logistic regression, In L. G. Grimm and P. R. Yarnold (Eds.), *Reading and understanding multivariate statistics*, American Psychological Association, pp. 217–244, 1995.

[31] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "PyTorch: From research to production," Circulation, 2016. Accessed: Oct. 01, 2022. [Online.] Available: https://pytorch.org/.

[32] OpenMined Org., "OpenMined/TenSEAL: A library for doing homomorphic encryption operations on tensors," Github, Jul 6, 2020. Accessed: Oct. 25, 2022. [Online.] Available: https://github.com/OpenMined/TenSEAL.

BIOGRAPHIES OF AUTHORS

Yuris Mulya Saputra is a full-time lecturer at Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada, Indonesia since 2016 and an Adjunct Fellow at School of Electrical and Data Engineering, University of Technology Sydney since 2022. He obtained Ph.D. in Electrical and Data Engineering from the University of Technology Sydney (Australia) in 2022. His research interests include mobile computing, energy and economic efficiency, machine learning, and optimization problems for wireless communication networks. He is affiliated with IEEE as a member. He is currently an active reviewer for various Q1 journals including IEEE TMC, IEEE JSAC, IEEE WCM, IEEE TWC, IEEE IoT Journal, IEEE WCL, and Elsevier JCLEPRO. He can be contacted at email: ym.saputra@ugm.ac.id.

Ganjar Alfian currently works at Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada, Indonesia since 2022. Previously, he worked as Assistant Professor at Dongguk University-Seoul, Republic of Korea for five years. In July 2017, he was a short-term visiting researcher with the VSB-Technical University of Ostrava, Czech Republic. He received Dr.Eng. degrees from the Department of Industrial and Systems Engineering, Dongguk University, Seoul, South Korea, in 2016. He has published numerous research articles in several international peer-reviewed journals, including Computers and Industrial Engineering, Journal of Food Engineering, Journal of Public Transportation, IEEE access, Food Control, Sensors, Applied Sciences, Asia Pacific Journal of Marketing and Logistics, and Sustainability. His research interests include artificial intelligence, RFID, IoT, machine learning, deep learning, carsharing, simulation, and health informatics. He was a recipient of the International Conference on Science and Technology (ICST) best paper award, in 2019. He can be contacted at email: ganjar.alfian@ugm.ac.id.