

Department: Head  
Editor: V. S. Subrahmanian, vs@cs.umd.edu

# Hiding in the Crowd: Federated Data Augmentation for On-device Learning

Eunjeong Jeong<sup>†</sup>, Seungeun Oh<sup>†</sup>, Jihong Park<sup>‡</sup>, Hyesung Kim<sup>§</sup>,  
Mehdi Bennis<sup>¶</sup>, Seong-Lyun Kim<sup>†\*</sup>

<sup>†</sup>Yonsei University

<sup>‡</sup>Deakin University

<sup>§</sup>Samsung Research

<sup>¶</sup>University of Oulu

\*Corresponding author. Contact him at slkim@yonsei.ac.kr.

**Abstract**—To cope with the lack of on-device machine learning samples, this article presents a distributed data augmentation algorithm, coined *federated data augmentation (FAug)*. In FAug, devices share a tiny fraction of their local data, i.e., seed samples, and collectively train a synthetic sample generator that can augment the local datasets of devices. To further improve FAug, we introduce a multi-hop based seed sample collection method and an oversampling technique that mixes up collected seed samples. Both approaches enjoy the benefit from the crowd of devices, by hiding data privacy from preceding hops and feeding diverse seed samples. In the image classification tasks, simulations demonstrate that the proposed FAug frameworks yield stronger privacy guarantees, lower communication latency, and higher on-device ML accuracy.

**Keywords**—Machine learning, Distributed networks, Distributed artificial intelligence, Wireless communication

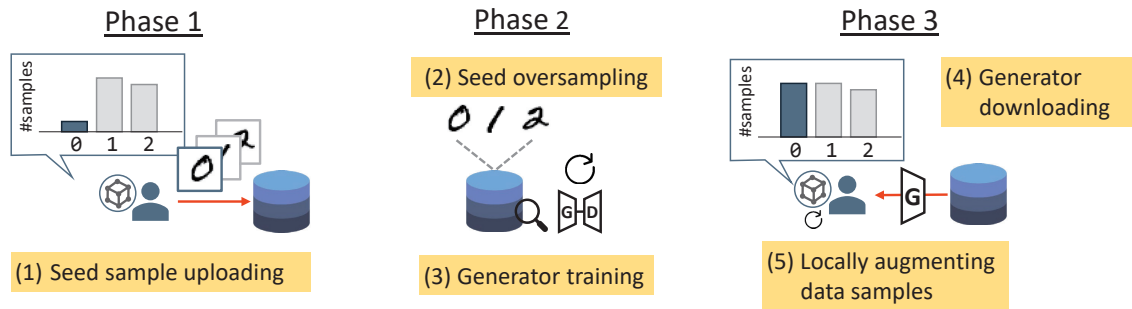
## Introduction

■ **ON-DEVICE** machine learning (ML) has come to fruition thanks to recent advances in mobile computing power [1], but one key bottleneck has remained unchanged – the lack of training samples. Since user-generated data is often scarce and biased, it impedes generalization of the local model. Federated learning (FL) resolves this problem by exchanging locally trained neural network (NN) models, without revealing raw data

samples [2]. Nevertheless, FL may still require additional data exchanges for reflecting the distribution of global data generated by devices, or otherwise their non-independent and identically distributed (non-IID) datasets may hamper the performance of FL [3], [4].

As a solution to these problems, our prior work proposed a distributed data augmentation algorithm, coined *federated data augmentation (FAug)* [5]. In FAug, the objective of all devices is

## Department Head



**Figure 1.** Three operational phases of Vanilla federated data augmentation (FAug).

to obtain a generative model from the server to locally replenish missing samples of each of them. For this purpose, every device uploads a tiny fraction of its data samples, called *seed samples*, to the server. By feeding these samples, the server trains a generative model such as the generator of a generative adversarial network (GAN) or the encoder of a variational autoencoder (VAE).

By downloading the trained generative model, each device can locally generate synthetic-yet-realistic data samples reflecting the global data distribution, thereby securing a sufficient amount of samples for training a downstream on-device NN model.

This article aims to showcase how to utilize the crowd of devices, towards developing a more communication-efficient and privacy-preserving FAug framework. To this end, inspired by *multi-hop* routing in communication systems, the seed samples are collected from a device to the server through neighboring devices in multiple hops [6]. Such a multi-hop seed sample collection has a two-fold benefit. One is to reduce the distance attenuation from faraway devices to the server by relaying the seed samples through multiple short hops, thereby reducing the sample collection latency. Another advantage is to hide the private information in the preceding hops by superpositioning new data in the current hop into the previously collected information. Consequently, the more hops are progressed, the more data privacy can be preserved.

Afterward, the collected seed samples are oversampled by the server using the *mixup* algorithm [7]. Traditionally, training of a generative NN requires a large number of samples, which is not comparable to the number of collected seed

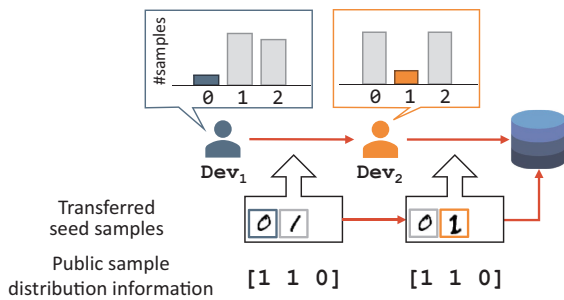
samples. To overcome this deficiency, mixup can be adopted to augment data by linearly combining any of the two seed samples. In mixup, the more seed samples are collected, the more diverse synthetic samples can be created. Therefore, the accuracy of on-device ML after applying FAug increases with the number of devices.

We introduce FAug and its advanced algorithms along with multi-hop communication and mixup technique. The ways of measuring and coping with privacy leakage are also discussed. In this paper, experiments with classification tasks are conducted under datasets including MNIST (hand-written digits with 10 classes), CIFAR10 (color images with 10 classes), and STL10 (higher-resolution color images with 10 classes). Numerical evaluations corroborate their effectiveness in improving FAug, in terms of communication latency, privacy guarantees, and on-device ML test accuracy after applying FAug.

## Related Works

Beyond machine learning that relies on the central entity's capacity, numerous researchers have been focusing on the idea of training models on edge. Federated learning (FL) [8] has arisen as an effective way of training on the edge that reduces uplink communication costs. To strengthen the security, devices can prevent data leakage against an honest-but-curious server that aggregates encrypted/masked gradients from devices. [9] It also applies to release non-IIDness across devices in topic modeling [10]. However, these FL-based algorithms have no guidelines in accordance with data distribution among the devices.

Momentum exchange relieves this bottleneck.



**Figure 2.** The first phase of FAug with multi-hop seed sample collection (MultFAug).

Nevertheless, the works do not provide the way to set the momentum based on data distribution. Still, the most powerful and basic solution for overcoming a biased dataset is exchanging local samples or providing equivalent information to each device’s samples. Direct data exchange is often refrained as it incurs a massive communication overhead and even violates data privacy. [11]

Recent works have focused on encrypting local training traits, for instance, local gradients or momentum updates, in order to privatize the distributed learning system [12], [13]. Under cross-device settings wherein a large group of mobile devices with limited transmission power, participants have to compensate computation for communication in a distributed learning system. However, the computation cost of complex cryptographic operations is significantly large. In contrast, selectively chosen public dataset alleviates the privacy violation and non-iid data distribution problems [14], [15], yet it assumes users that are insensitive to share their local samples as proxy data. As a compromise, our previous work suggested sharing a generative model trained with collected seed samples. These samples are mixed and augmented to alleviate the privacy violation problem.

## Preliminary

### Vanilla Federated Data Augmentation (FAug)

For the sake of explanation, we hereafter focus on the MNIST dataset, comprising handwritten images of 0-9 digits, i.e., 10 labels. Each device aims to classify the correct digit label by observing each image sample. Reflecting the user-generated nature of data, each device stores a fraction of the MNIST dataset, such that a

randomly selected target label lacks data samples.

With the said non-IID datasets, the operations of Vanilla FAug are described in three phases.

- 1) Each device uploads its seed samples directly to the server to report its missing target labels. From its local dataset, dummy samples are selected and sent together with target labeled seed samples.
- 2) Next, the server oversamples the uploaded seed samples using high computing power and internet accessibility, and trains a generator from a conditional Generative Adversarial Network (cGAN) model. [16]
- 3) Each device downloads the generator for local data augmentation. By feeding the target labels and random noises, the generator outputs synthesized samples, which are used for local training.

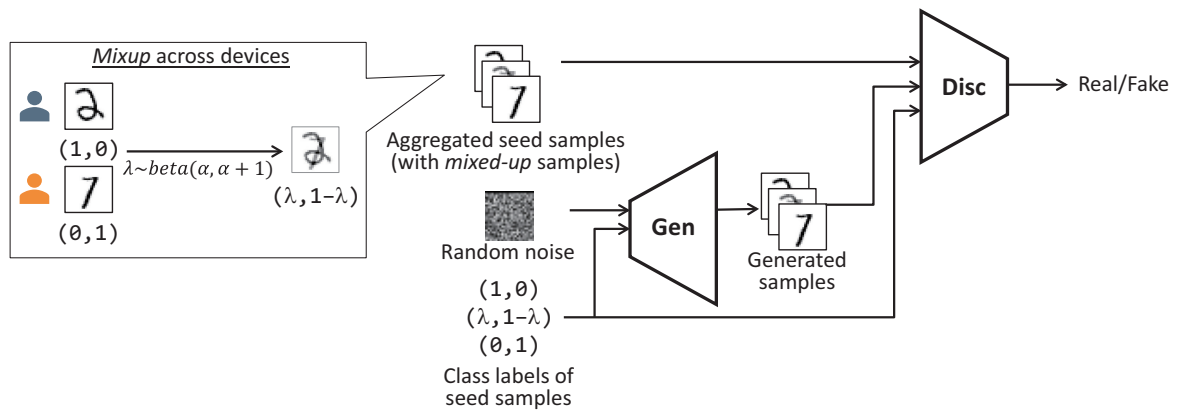
### Multi-hop Seed Sample Collection

In this section, we focus on the first phase of FAug (see Phase 1 in Fig. 1). To improve the communication efficiency and privacy guarantees of the target labels, we propose MultFAug in which the seed samples collected through multiple hops. Compared to a direct communication, it compensates distance attenuation from isolated devices to the server, and thus decreases the communication latency. At the same time, each device can hide its target label information in other devices’ information of the preceding hops. The device can therefore append less dummy labels, reducing the communication payload sizes while achieving the same level of privacy guarantees.

To realize MultFAug, for a given network topology, the server determines seed sample collecting routes, and informs each device about its next hop device. Then, the first operational phase of MultFAug is elaborated as follows. (See Fig. 2)

- 1) A device ( $Dev_1$ ) uploads seed samples while appending dummy label 1 to hide its target label 0. This results in an obfuscated private sample distribution information  $[1, 1, 0]$ .
- 2) A predetermined recipient,  $Dev_2$ , receives a group of seed samples from  $Dev_1$ . If the target label of  $Dev_2$  is already packed in the received bundle,  $Dev_2$  can skip adding any dummy label and toss a group of seed

## Department Head



**Figure 3.** The second phase of mixup oversampling aided FAug (MixFAug). The seed samples aggregated from different devices are mixed-up and fed as ground-truth input for generative models such as conditional GAN.

samples to its recipient.

- 3)  $Dev_2$  overwrites its target samples without additional dummy samples.

Fewer dummy samples are used when the public sample distribution information includes a sufficient number of labels to dilute the origin of each sample, thereby yielding a smaller size of communication payload.

## Mixup Oversampling Aided FAug

### Mixup Oversampling

Mixup oversampling aided FAug (MixFAug) enhances the oversampling process at the second phase, while the remainder follows Vanilla FAug (Fig. 1). Mixup [7] is introduced as a data augmentation method through linear interpolation of two data samples. These blended combinations have moderate features of their raw inputs and therefore smooth out the decision surface. It is proven to improve the accuracy of classification tasks by applying mixup with equal weight allocation (i.e.,  $\lambda = 0.5$  for a pair of mixing data samples). [17]

For the extension of the dataset for a training generator, two data samples from the collected seed samples are randomly selected and mingled with different mixing ratios following beta distribution as in Fig. 3. The server continues mixing-up until the server's new dataset reaches a predetermined oversampling size, e.g., 3,000 mixed-up samples. Assuming that each device uploads a fixed fraction of seed samples, a larger number of devices lead to improvement of test accuracy after implementing FAug because they

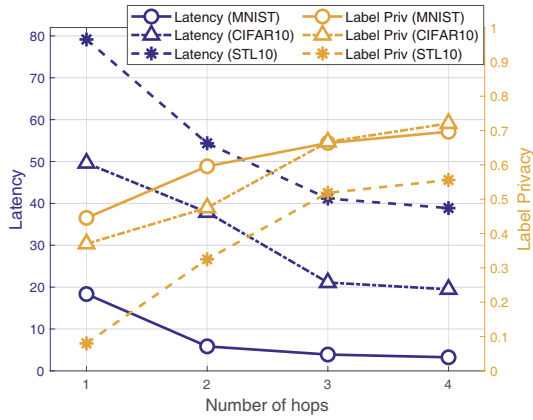
enlarge the total amount of mixed-up samples at the server.

### Privacy Guarantee

The identity of information providers are widely regarded as sensitive as well as the information itself. Any deep learning model needs to access data while conducting tasks such as medical records, web browsing history in searching engines, or facial images. It should be done without revealing who is the owner of the provided information. Hence, it is recommended to reduce the number of seed samples uploaded from a single device to minimize the privacy leakage. By contrast, uploading samples from a large crowd of devices helps strengthen the privacy guarantee by increasing the difficulty of distinguishing the source devices of those received samples.

The operation of FAug hides the target labels from the server as it cannot discriminate between the received target and dummy labels. A larger group of FAug participants eases a device's burden of the number of seed samples for transmission, due to the advantage of more of collected samples. Accordingly, the scheme relieves communication overhead and privacy leakage compared to exchanging data samples directly across devices. Combined with multi-hop communications, the scheme helps the devices with a substantial distance from the server to obtain path loss gain.

The label privacy guarantee for a specific device is measured as how much a device concealed its target labels when the whole package of



**Figure 4.** Trend of label privacy guarantee and latency of a reference user as the number of hops increases.

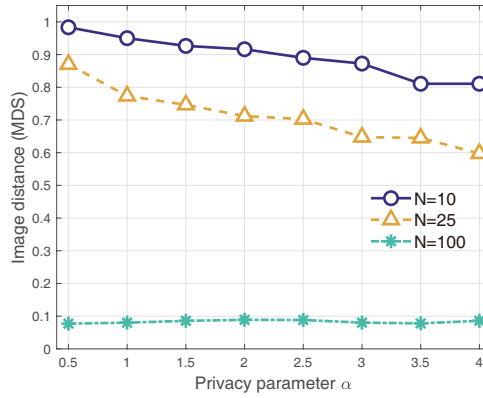
accumulated seed samples from this route arrives at the server. The numerator indicates the number of target labels that the device had at first, and the denominator indicates the number of labels that the server receives from the route that this device occupied during multi-hop transmission.

For quantifying sample privacy guarantee of mixup, we adopt a metric based on differential privacy. Since adding the other samples weighted elements makes the output follow their own distribution unlike random noises, mixup as a randomization algorithm needs to be measured with a variant type of differential privacy.

Condensed local differential privacy (CLDP) [18] is a variation of local differential privacy (LDP), which was proposed as a specialized method in scenarios with small user populations. Similar to LDP, data samples in CLDP protocol undergo local perturbation before they are uploaded to the server, but each device systematically inclines to choose an output that has higher similarity with its raw data than distant outputs. Likewise, CLDP-based data collection strengthens indistinguishability of mixup algorithm by additional distance term among input data.

A CLDP protocol perturbs a true value  $v$  in a finite and discrete dataset  $\mathcal{D}$  according to the knowledge of distance between  $v$  and any other elements in  $\mathcal{D}$ , which is formalized as follows.

**Definition 1.** Let the randomization algorithm  $\Phi$  takes  $v_1$  and  $v_2$  as input and outputs  $y$ , i.e.,  $\Phi : \mathcal{D} \rightarrow \mathcal{D}$ . For any inputs  $v_1, v_2 \in \mathcal{D}$ , perturbation



**Figure 5.** Distance between original and mixed-up dataset for  $N$  mixed-up samples with respect to privacy parameter  $\alpha$  in  $\alpha$ -CLDP.

method  $\Phi$  satisfies

$$\forall y \in \mathcal{D}_{\text{range}} : \frac{\Pr[\Phi(v_1) = y]}{\Pr[\Phi(v_2) = y]} \leq e^{\alpha \cdot d(v_1, v_2)} \quad (1)$$

where  $\mathcal{D}_{\text{range}}$  indicates the range of outputs and a random sample  $y$  is chosen from  $\mathcal{D}$  with a probability

$$\Pr[\Phi(v) = y] = \frac{e^{-\alpha \cdot d(v, y)/2}}{\sum_{z \in \mathcal{D}} e^{-\alpha \cdot d(v, z)/2}}. \quad (2)$$

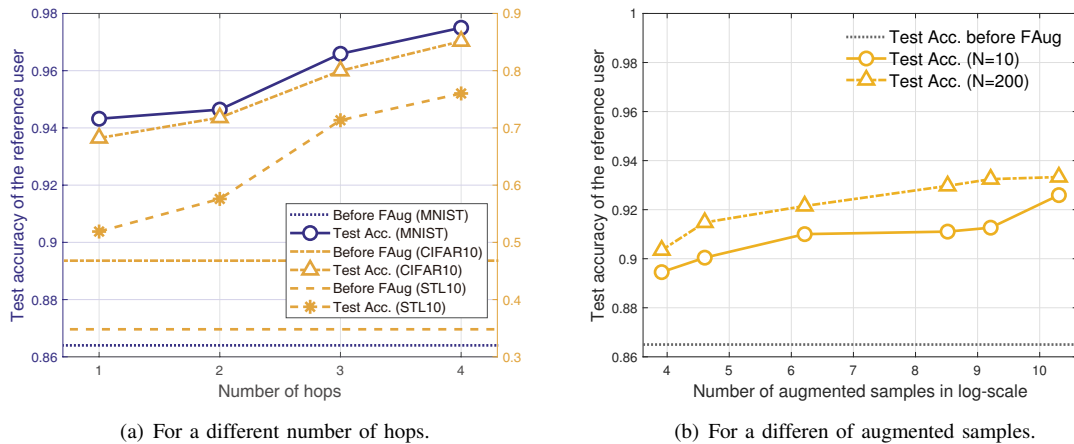
The randomization algorithm that satisfies CLDP can extend into mixup as follows. Let an augmented dataset  $\mathcal{D}_{\text{aug}}$  composed of mixed-up samples from  $\mathcal{D}$  in Definition 1.  $\mathcal{D}_{\text{aug}}$  is also discrete and finite if assumed that each user uploads equally-likely mixed-up samples to guarantee label privacy. Substituting the augmented dataset for a collected seed samples' set  $\mathcal{D}$ , mixup as a perturbation method  $\Phi_{\text{mixp}}$  satisfies CLDP for any inputs  $v_1, v_2 \in \mathcal{D}_{\text{aug}}$ .

## Numerical Evaluations

To corroborate the effectiveness of MultFAug and MixFAug, we consider the following simulation settings and figure of merits.

- Each device has one target label. The local dataset of the device consists of 5 samples for the target label, while the other labels occupy more than 200 samples per label.
- A device retransmits its packets with an unlimited number of attempts when it occupies a data rate less than the instantaneous channel capacity. The uplink transmission latency is

## Department Head



**Figure 6.** Test accuracy after FAug with respect to (a) the number of hops and (b) the number of devices that upload samples to the server with a fixed rate 5 samples per device.

defined as the number of time slots consumed until all the seed samples are transmitted. For evaluation of communication cost, we consider each pixel of a data sample (28x28 pixels for MNIST, 32x32 for CIFAR10, and 96x96 for STL10) accounts for 8 bits.

- Image similarity between data samples are measured from the distance among data samples converted onto a multidimensional space.

A crowd of devices in FAug enhance the impact on communication delay and privacy guarantee in Fig. 4 and 5. Fig. 4 shows that the label privacy protection occurs when the server receives only a mixed set of target labels and samples. Multi-hop communications can also reduce the uplink latency of the participants, especially for those with noisy channels or long distances from the server. Notably, while implementing FAug with two-hop communication, the transmission delay minimizes at around 2-3 hops. Exceeding number of hops, on the contrary, leads to increment of store and forward delay.

Fig. 5 shows the impact of dataset size on sample privacy guarantee. The image distance between mixed-up datasets and original raw datasets is used as a utility function. Larger distance implies more deviation from perfect exposure of raw data, which is ideal for transferring data information to the server but not implemented in reality because it reveals full features. Privacy protection compensates for the diversity of an augmented dataset. Notably, this utility is robust

against the level of privatization in case of a large number of seed samples involved.

In Fig. 6 the enhanced performance via the advanced approaches of FAug is described. Fig. 6(a) demonstrates the test accuracy of a reference user before and after applying MultFAug. Compared with the baseline accuracy around 0.86 before applying FAug, it reaches up to 0.95 after replenishing the missing samples. GAN on the server can recover from the scarcity of received seed samples by oversampling them through mixup. In Fig. 6(b), the test accuracy of a user shows improvement proportional to the size of oversampled products until it saturates when the server produces more than 5,000 mixed-up samples.

## Conclusions

By adding dummy information, the devices can preserve their privacy when participating in a collaborative learning system with user-generated data. Accordingly, the protection against leakage of private information necessitates longer transmission time or sacrifice of the accuracy. The numerical evaluations validate the effectiveness of FAug and its variations, emphasizing the importance of minimizing privacy leakage during transmission of samples within a tolerable transmission time.

The remaining issues in developing the performance and usability of FAug includes ensuring privacy protection without compensating for lessened dataset quality. For further analytic research,

it is also possible to analyze finding optimal conditions for distributed learning using multi-hop communication, such as an optimal number of hops.

## Acknowledgment

This research was supported by a grant to Bio-Mimetic Robot Research Center Funded by Defense Acquisition Program Administration, and by Agency for Defense Development (UD190018ID).

## REFERENCES

1. J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proceedings of the IEEE*, vol. 107, no. 11, pp. 2204–2239, 2019.
2. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, pp. 1273–1282, 2017.
3. Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.
4. F. Sattler, S. Wiedemann, K. R. Mller, and W. Samek, "Robust and communication-efficient federated learning from non-i.i.d. data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3400–3413, 2020.
5. E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Communication-efficient on-device machine learning: federated distillation and augmentation under non-iid private data," *arXiv preprint arXiv:1811.11479*, 2018.
6. E. Jeong, S. Oh, J. Park, H. Kim, M. Bennis, and S.-L. Kim, "Multi-hop federated private data augmentation with sample compression," *arXiv preprint arXiv:1907.06426*, 2019.
7. H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: beyond empirical risk minimization," *arXiv preprint arXiv:1908.05895*, 2017.
8. J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
9. Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.
10. D. Jiang, Y. Song, Y. Tong, X. Wu, W. Zhao, Q. Xu, and Q. Yang, "Federated topic modeling," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1071–1080, 2019.
11. M. F. Balcan, A. Blum, S. Fine, and Y. Mansour, "Distributed learning, communication complexity and privacy," in *Conference on Learning Theory*, pp. 26–1, 2012.
12. C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "BatchCrypt: Efficient homomorphic encryption for cross-silo federated learning," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, pp. 493–506, 2020.
13. 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, 2018.
14. N. Yoshida, T. Nishio, M. Morikura, K. Yamamoto, and R. Yonetani, "Hybrid-FL for wireless networks: Cooperative learning mechanism using non-iid data," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2020.
15. T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2019.
16. M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.
17. H. Inoue, "Data augmentation by pairing samples for images classification," *arXiv preprint arXiv:1801.02929*, 2018.
18. M. E. Gursoy, A. Tamersoy, S. Truex, W. Wei, and L. Liu, "Secure and utility-aware data collection with condensed local differential privacy," *IEEE Transactions on Dependable and Secure Computing*, 2019.

**Eunjeong Jeong** is with Yonsei University, Seoul, Korea. She received the B.S. and M.S. degrees in Electrical and Electronic Engineering from Yonsei University, in 2018 and 2020, respectively. Her research interests lie in distributed machine learning and wireless networks. She received the 2018 IEEE Seoul Section Student Paper Contest Silver Prize. Contact her at [ejjeong@ramo.yonsei.ac.kr](mailto:ejjeong@ramo.yonsei.ac.kr).

**Seungeun Oh** is currently with Yonsei University, Seoul, South Korea. He received the B.S. degree in 2017 and is currently pursuing the Ph.D. degree in Electrical and Electronic Engineering. His research interests include distributed machine learning and 5G V2X communication. He received the 2018 IEEE Seoul Section Student Paper Contest Silver Prize.

## Department Head

Contact him at [seoh@ramo.yonsei.ac.kr](mailto:seoh@ramo.yonsei.ac.kr).

**Jihong Park** is currently a lecturer with the School of Information Technology, Deakin University, Waurn Ponds, VIC, Australia. He was a postdoctoral researcher with Aalborg University, Denmark, from 2016 to 2017 and with the University of Oulu, Finland, from 2018 to 2019. He was a visiting researcher with the Hong Kong Polytechnic University, Hong Kong, in 2013; the KTH Royal Institute of Technology, Stockholm, Sweden, in 2015; Aalborg University, in 2016; and the New Jersey Institute of Technology, Newark, NJ, USA, in 2017. His current research interest includes communication-efficient distributed machine learning and distributed ledger technology also focusing on ultrareliable, ultradense, and mmWave system designs in 5G and beyond. He received the B.S. and Ph.D. degrees from Yonsei University, Seoul, South Korea, in 2009 and 2016, respectively with the School of Electrical and Electronic Engineering. He was the recipient of the 2014 IEEE GLOBECOM-Student Travel Grant, the 2014 IEEE Seoul Section Student Paper Contest Bronze Prize, and the 6th IDIS-ETNEWS (The Electronic Times) Paper Contest Award sponsored by the Ministry of Science, ICT, and Future Planning of Korea. Contact him at [jihong.park@deakin.edu.au](mailto:jihong.park@deakin.edu.au).

**Hyesung Kim** is currently a staff engineer with the Samsung Electronics Company, Ltd., Seoul, South Korea. He was a visiting researcher with the University of Oulu in 2018. His research interests include edge computing/caching, 5G communications system, distributed machine learning, and blockchain. He received the B.S. and Ph.D. degrees electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2012 and 2019, respectively. He was awarded the IEEE Seoul Section Student Paper Contest Bronze Prize in 2017, titled "Mean-field game-theoretic edge caching for ultra dense networks," as a first author. Contact him at [hye1207@gmail.com](mailto:hye1207@gmail.com).

**Mehdi Bennis** is currently an Associate Professor with the Centre for Wireless Communications, University of Oulu, Oulu, Finland, where he is also an Academy of Finland Research Fellow and the Head of the Intelligent Connectivity and Networks/Systems Group (ICON). He has coauthored one book and published more than 200 research articles in international conferences, journals, and book chapters. His current research interests include radio resource management, heterogeneous networks, game theory, and machine learning in 5G networks and beyond. Dr. Bennis was a recipient of several prestigious awards,

including the 2015 Fred W. Ellersick Prize from the IEEE Communications Society, the 2016 Best Tutorial Prize from the IEEE Communications Society, the 2017 EURASIP Best Paper Award for the Journal on Wireless Communications and Networks, the All-University of Oulu Award for research, and the 2019 IEEE ComSoc Radio Communications Committee Early Achievement Award. He is an Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS. Contact him at [mehdi.bennis@oulu.fi](mailto:mehdi.bennis@oulu.fi).

**Seong-Lyun Kim** is a Professor and Head of the School of Electrical & Electronic Engineering, Yonsei University, Seoul, Korea, heading Robotic & Mobile Networks Laboratory (RAMO) and Center for Flexible Radio (CFR+). He is co-directing H2020 EUK PriMO-5G project, and leading Smart Factory Committee of 5G Forum, Korea. He was an Assistant Professor of Radio Communication Systems at the Department of Signals, Sensors & Systems, KTH Royal Institute of Technology, Stockholm, Sweden. He was a Visiting Professor at the Control Engineering Group, Aalto University, Finland, and the KTH Center for Wireless Systems, and the Graduate School of Informatics, Kyoto University, Japan. He served as a technical committee member or a chair for various conferences, and an editorial board member of IEEE Transactions on Vehicular Technology, IEEE Communications Letters, Elsevier Control Engineering Practice, Elsevier ICT Express, and Journal of Communications and Network. He served as the leading guest editor of IEEE Wireless Communications and IEEE Network for wireless communications in networked robotics, and IEEE Journal on Selected Areas in Communications. His research interest includes radio resource management, information theory in wireless networks, collective intelligence, and robotic networks. He is the corresponding author for this article. Contact him at [slkim@yonsei.ac.kr](mailto:slkim@yonsei.ac.kr).