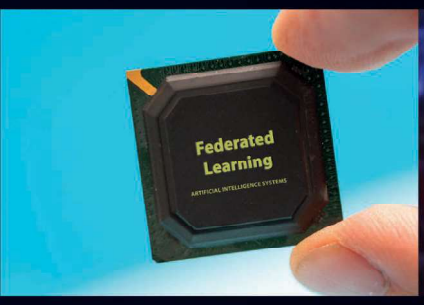
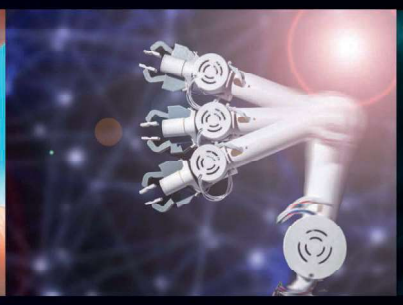
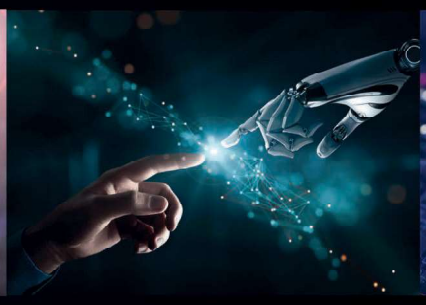
Federated Learning

Foundations and Applications




Edited by
Rajkumar Buyya
Anwesha Mukherjee
Sajal K. Das

MORGAN KAUFMANN

Federated Learning

This page intentionally left blank

Federated Learning

Foundations and Applications

Edited by

Rajkumar Buyya
Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory
School of Computing and Information Systems
The University of Melbourne
Melbourne, VIC, Australia

Anwesha Mukherjee
Department of Computer Science
Mahishadal Raj College (Vidyasagar University)
Mahishadal, India

Sajal K. Das
Department of Computer Science
Missouri University of Science and Technology
Rolla, MO, United States

ELSEVIER
MORGAN KAUFMANN PUBLISHERS
AN IMPRINT OF ELSEVIER

Morgan Kaufmann is an imprint of Elsevier
50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States

ISBN: 978-0-443-44433-3

Publisher: Mara Conner
Acquisitions Editor: Chris Katsaropoulos
Editorial Project Manager: Ruchi Bhargava
Production Project Manager: Vijayaraj Purushothaman
Cover Designer: Raman Kumar

Typeset by VTeX

Contents

List of contributors

Olasupo Ajayi, Department of Computer Science, University of the Western Cape, Cape Town, South Africa

Turki Alhazmi, Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Irina Arévalo, Technical University of Madrid, Madrid, Spain

Andrea Augello, University of Palermo, Palermo, Italy

Farag Azzedin, Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Antoine Bagula, Department of Computer Science, University of the Western Cape, Cape Town, South Africa

Rajkumar Buyya, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Rodrigo N. Calheiros, Western Sydney University, Sydney, NSW, Australia

Timothy Castiglia, Rensselaer Polytechnic Institute, Troy, NY, United States

M.A.P. Chamikara, CSIRO's Data61, Clayton, VIC, Australia

Mohan Baruwal Chhetri, CSIRO's Data61, Clayton, VIC, Australia

Federico Corò, University of Padua, Padua, Italy

Sajal K. Das, Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, United States

Ashish Gupta, BITS Pilani Dubai Campus, Dubai, United Arab Emirates

Zhu Han, Department of Electrical and Computer Engineering, University of Houston, Houston, TX, United States

Chanuka A.S. Hewa Kaluannakkage, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Wenjing Hou, University of Electronic Science and Technology of China, Chengdu, China

Bahman Javadi, Western Sydney University, Sydney, NSW, Australia

Ferdinand Kahenga, Department of Computer Science, University of the Western Cape, Cape Town, South Africa ESISALAMA, Lubumbashi, Democratic Republic of the Congo

Wenxin Lei, University of Electronic Science and Technology of China, Chengdu, China

Haojie Lin, University of Electronic Science and Technology of China, Chengdu, China

Qiang Liu, University of Electronic Science and Technology of China, Chengdu, China

Giuseppe Lo Re, University of Palermo, Palermo, Italy

Supriya M, Department of Computer Science and Engineering, Amrita School of Computing, Bengaluru, Amrita Vishwa Vidyapeetham, Bengaluru, India

Jovita Mateus, Department of Computer Science, University of the Western Cape, Cape Town, South Africa Department of Computer Science, Namibia University of Science and Technology, Windhoek, Namibia

Ana Milanova, Rensselaer Polytechnic Institute, Troy, NY, United States

Fatemeh Mirhakimi, Western Sydney University, Sydney, NSW, Australia

Anwesha Mukherjee, Department of Computer Science, Mahishadal Raj College, Mahishadal, West Bengal, India Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Stacy Patterson, Rensselaer Polytechnic Institute, Troy, NY, United States

Pravija Raj P V, BITS Pilani Dubai Campus, Dubai, United Arab Emirates

Shinu M. Rajagopal, Department of Computer Science and Engineering, Amrita School of Computing, Bengaluru, Amrita Vishwa Vidyapeetham, Bengaluru, India

Murtaza Rangwala, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Priyesh Ranjan, Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, United States

Satyaki Roy, Department of Mathematical Sciences, University of Alabama in Huntsville, Huntsville, AL, United States

Siva Sai, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Jose L. Salmeron, CUNEF University, Madrid, Spain

Abhishek Sawaika, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Navid Seidi, Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, United States

Prabhjot Singh, Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia

Dipanwita Thakur, Department of Computer Engineering (DIMES), University of Calabria, Rende, Italy

Wenhong Tian, University of Electronic Science and Technology of China, Chengdu, China

Linh Tran, Rensselaer Polytechnic Institute, Troy, NY, United States

K.R. Venugopal, Department of Computer Science and Engineering, University of Visvesvaraya College of Engineering, Bangalore University, Bangalore, India

Radical Rakhman Wahid, Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Hong Wen, University of Electronic Science and Technology of China, Chengdu, China

Feng Yan, University of Houston, Houston, TX, United States

Nan Yang, Western Sydney University, Sydney, NSW, Australia

Ning Zhang, Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, Canada

Guy-Alain Lusilao Zodi, Department of Computer Science, Namibia University of Science and Technology, Windhoek, Namibia

Chapter 1

Federated learning at a glance

Anwesha Mukherjee[a,c], **Sajal K. Das**[b], **and Rajkumar Buyya**[c]

[a]*Department of Computer Science, Mahishadal Raj College, Mahishadal, West Bengal, India,* [b]*Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, United States,* [c]*Quantum Cloud Computing and Distributed Systems (qCLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC, Australia*

1.1 Introduction

Society 5.0 [1] has focused on the integration of technology with different aspects of societal applications including healthcare, agriculture, retail, and transportation. Internet of Things [2], cloud computing [3], edge computing [4], and data analytics are integrated to provide smart solutions for the society. The machine learning (ML) [5] and deep learning (DL) [6] algorithms are used for data analysis. The ML algorithms can be classified into supervised, unsupervised, semi-supervised, and reinforcement learning, which are briefly discussed as follows [7]:

- **Supervised learning:** The supervised ML algorithm learns from a labeled dataset. In a labeled dataset, there exists a pair between each input and the corresponding output (target). The algorithm learns patterns from the labeled data and aims to learn a mapping function to predict the target or output for unseen input values. Logistic regression, decision tree, support vector machine, etc., are examples of supervised ML algorithms.
- **Unsupervised learning:** In this case, the dataset is unlabeled, and the algorithm finds hidden patterns, structures, and relationships within the data. K-means clustering, hierarchical clustering, etc., are examples of unsupervised ML algorithms.
- **Semi-supervised learning:** Semi-supervised learning falls between the categories of supervised and unsupervised learning algorithms. In case of semi-supervised learning, dataset contains a small amount of labeled data and a large amount of unlabeled data, and the algorithm needs to learn from the dataset. If there is a scarcity of labeled dataset, semi-supervised ML can be used.
- **Reinforcement learning:** In reinforcement learning, the algorithm learns through the interaction with the environment and receiving rewards or penalties. The algorithm learns decision-making based on maximizing the collected rewards over time. An automated robot is an example of reinforcement learning, where a robot learns to navigate an environment.

Apart from ML, the DL algorithms are also popular for data analysis, especially, in case of large number of samples. For big data analysis, DL plays a significant role. DL is a subset of ML, where multi-layered neural networks are used for simulating complex decision-making ability of the human brain. Traditional ML algorithms use simple neural networks with one of the two layers of computation, whereas DL models use three or more layers of computation. DL models use artificial neural networks (ANNs) [8] with multiple layers for data analysis and complex pattern learning. DL algorithms are based on the structure and function of the human brain, where interconnected neurons are used for information processing. DL algorithms are suitable for image recognition, natural language processing, speech recognition, etc. There are various DL models such as gated recurrent unit (GRU), multilayer perceptron (MLP), long short-term memory (LSTM), etc.

1.1.1 Why is federated learning needed?

As DL models for large-scale data analysis require high computational resources, cloud servers are mainly used to execute DL models. However, entire data transmission and storage inside the cloud may raise several issues such as high network traffic, high latency, and security concerns. Further, collaborative learning is significant for accurate prediction. *Federated learning* has come with the solutions towards these challenges. The use of federated learning with edge/fog computing [9,10] can result in efficient data analysis in terms of accuracy, latency, etc. Federated learning [11–13] is a collaborative model training approach where the local models are used for analyzing local data, and the clients and the server exchange model parameters between them for updating the model.

Federated Learning. https://doi.org/10.1016/B978-0-44-344433-3.00006-X

Definition of federated learning: Federated Learning is defined as a machine learning technique, where a model is trained across multiple decentralized devices or servers, containing local data, without exchanging the raw data. Each device trains a local model using its own dataset and sends only model updates (gradients or weights) to a central server or among themselves, to build an improved global model by aggregating the local model updates. Federated learning improves data privacy protection and security by keeping sensitive data localized.

The basic diagram of federated learning (FL) is presented in Fig. 1.1. In the figure, four nodes are considered which have their local datasets. Each node k receives global model parameters from the server, and trains its local model (M_k) using own dataset D_k. After training, each node sends the local model update (w_k) to the server. The server aggregates the received model updates to generate the aggregated global model update (w_g). If there are K nodes and federated averaging is used for aggregation, then the aggregated global model update is given as $\frac{1}{K}\sum_{k=1}^{K} w_k$. In the figure, the number of nodes is four, thus, $K = 4$.

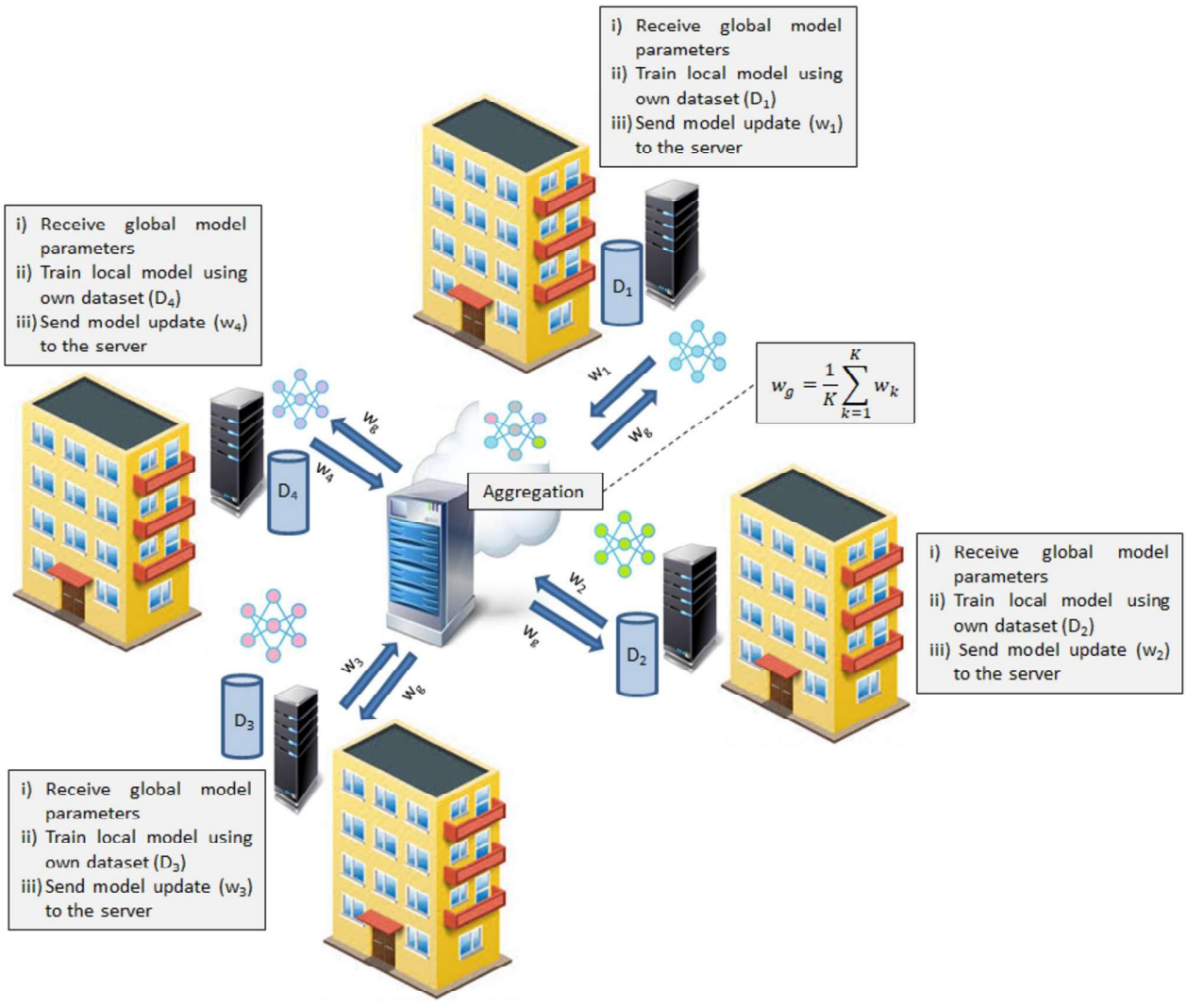


FIGURE 1.1 Federated learning framework with four nodes.

1.1.2 Characteristics of FL

The characteristics of FL are briefly stated as follows:

- **Data localization, decentralization, and privacy protection:** In FL, the raw data is stored inside the clients, and not shared to protect data privacy. Only, the local model updates are transmitted to the server.
- **Collaborative training and aggregation:** The clients share their local model updates with the server that aggregates the received updates to build the global model. Hence, multiple clients collaboratively train a shared global model in FL.
- **Heterogeneity:** FL deals with heterogeneous devices with heterogeneous data.
- **Efficient communication:** As FL is based on the exchange of model updates between the clients and the server, communication should be efficient, and network connectivity should be good enough for smooth execution of the FL process.
- **Distributed learning:** As each client locally trains its model and only updates are shared, FL is a distributed learning model. However, FL can handle both independent and identically distributed (IID) and non-independent and non-identically distributed (non-IID) datasets, stored inside the clients.
- **Scalability:** As training is enabled on vast and distributed datasets, FL can scale a large number of clients.

In FL, local data analysis is performed that enhances data privacy protection. Further, the collaborative learning improves prediction accuracy. In Section 1.2, we discuss on different types of FL. After that, we discuss on various applications of FL in Section 1.3. The research challenges of FL are stated in Section 1.4. The chapter concludes in Section 1.5.

1.2 Types of FL architectures

The FL architectures can be classified according to the following two dimensions [2]:

- Networking structure
- Data partitioning

1.2.1 Networking structure-based FL classification

Based on the networking structure, FL is classified into the following two types:

- Centralized federated learning (CFL)
- Decentralized federated learning (DFL)

In CFL [14], the server shares the global model parameters with the clients. Based on the global model parameters, clients train their local models using their local datasets. The locally trained model updates are shared with the server. The server, after receiving all updates to the model, aggregates them to build the global model. The global model update is then shared with the clients. This process is continued for a number of rounds until a global model with minimal loss is generated.

In decentralized federated learning (DFL) [14], clients form a network among themselves using a ring or mesh topology and share model updates with their neighbors to build the global model.

1.2.2 Data partitioning-based FL classification

Based on the data partitioning process, FL can be classified into three types:

- Horizontal FL (HFL)
- Vertical FL (VFL)
- Federated Transfer Learning (FTL)

In HFL [2], local datasets stored within the clients have the same feature space but different sample spaces. The clients use the same ML/DL model for local training. After training, local model updates are masked using encryption or differential privacy to enhance security, if required, before sharing with the server. The server aggregates the received updates and builds the global model. The global model update is shared with the clients. This process is repeated until a global model with minimal loss is developed.

In VFL [2], the client data sets have the same sample space but different feature spaces. An entity alignment method is used to collect the overlapped data samples of the clients. The data samples are combined to train a global model using encryption.

In FTL [2], the features from different feature spaces are transferred to the same representation to be used for training data aggregated from multiple clients. FTL is suitable where the sample size as well as feature space both differ. For data privacy protection and enhanced security, masking is used for gradient encryption. Depending on the combined updates from the clients, the server performs model learning through aggregation to find a global model update with minimal loss.

1.3 Applications of FL

FL has several application areas, including healthcare, agriculture, mobile applications, and autonomous vehicles [12,14–16]—some of them are discussed below.

1.3.1 Healthcare

FL provides collaborative model training with data privacy protection. As healthcare data is highly sensitive and confidential, FL plays a vital role in healthcare applications such as medical image analysis, activity recognition, disease detection, and electronic health records analysis [17–19]. The Internet of Medical Things (IoMT) devices collect patients' health-related data. The patients' datasets are used for local model training, and the model updates are shared with the central server for aggregation. The aggregated global model update is then shared with the clients. Fig. 1.2 shows an FL-based healthcare system, where three hospitals have their own datasets containing patients' health records. The three hospitals serve as three clients and train their local models using local datasets and share the local model updates with the server. The central server aggregates the received model updates and builds the global model. The global model update is shared with the hospitals. In [20], FL was used for the diagnosis of COVID-19. In [21–23], the use of FL for medical image analysis was illustrated. FL also demonstrated its efficacy in human activity recognition [19,24]. Recently, privacy-aware FL, incentive-based FL, and personalized FL, have become popular in FL-based smart healthcare systems [17,25]. For privacy-enhanced FL, differential privacy [26] has become popular. The artificial noise is added to the dataset for protecting data privacy. The noise can also be added to the model gradients or weights during transmission of model updates. In conventional FL, all IoMT devices participate in the FL process. This may not be possible in real-life scenarios due to the lack of willingness of the participants, because of limited computing resources, bandwidth requirements, etc. As a solution, incentive-based FL [17] method has come to incentivize the participants in the FL process based on the contribution, reputation, and resource allocation. As FL produces a global model after aggregation, personalized healthcare [17] is a challenge. To solve this issue, the devices can download the global model and train the personalized local models. It is also important to deal with label heterogeneity in FL for personalized healthcare.

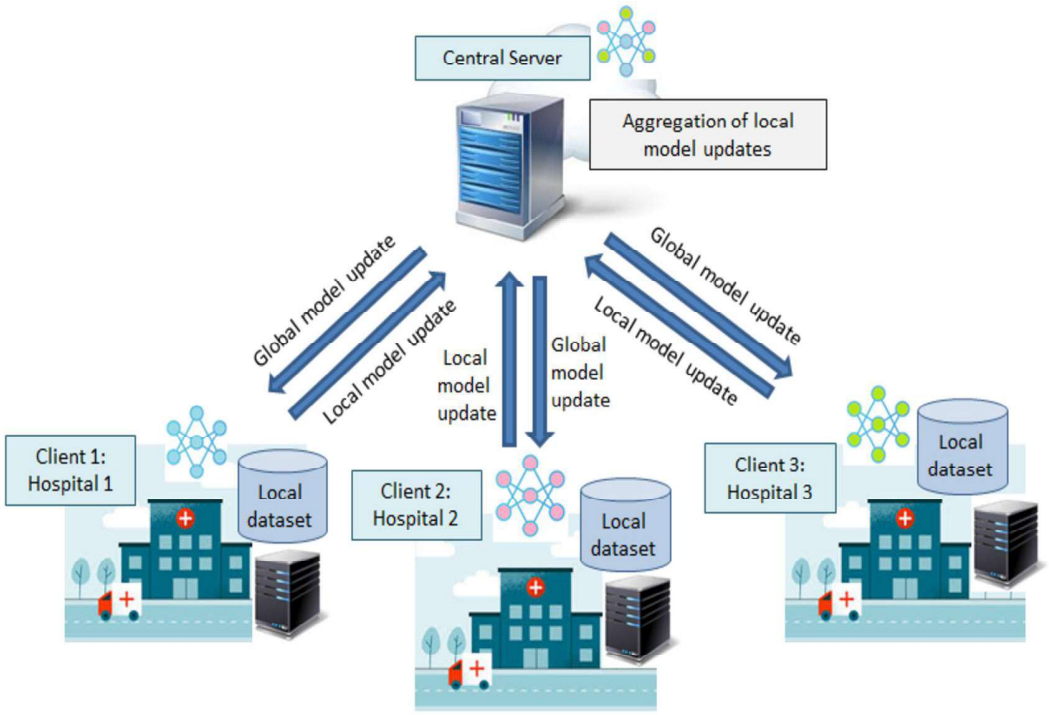


FIGURE 1.2 Federated learning-based healthcare system.

1.3.2 Agriculture

Smart agriculture is an emerging domain, especially in countries where agriculture plays a vital role in the economy. Soil properties, environmental properties of different regions vary; further, the unique soil features, farmers' information, and land-related information are sensitive and confidential. Hence, FL also has a vital application in smart agriculture [27]. Soil health monitoring, moisture prediction, irrigation management, etc., are various applications of smart agriculture, in which FL can play a vital role to provide accurate decision-making through collaborative learning with privacy protection [9,10]. FL plays a significant role in soil health monitoring, crop yield prediction, irrigation management, etc. The farms have their local datasets, which are used for local model training. The local model weights are aggregated by the central server to build the global model [10,14,28]. The global model update is shared with the farms. Fig. 1.3 shows an FL-based agricultural system. In the figure, three agricultural farms are shown as three clients, which have own datasets containing the data regrading the agricultural land, soil parameters, and environmental parameters. The three farms train their local models using own datasets and share the local model updates with the server. The central server aggregates the received model updates and builds the global model. The global model update is shared with the farms. The farms can also collaborate using DFL to build a global model [14]. For enhancing security, blockchain is integrated with FL for smart agriculture in [28]. In [10,28], FL was used for soil moisture monitoring and irrigation decision-making. In [9,14], FL was used to predict crop yields. In [29], FL played a significant role in image-based crop disease detection. In [30], FL was used in soil spectroscopy. For soil fertility prediction, FL was used in [31]. Agriculture is one of the pillars of socio-economic development of most of the countries. FL can play a crucial role in smart sustainable agricultural practices to improve the overall economic growth of the countries.

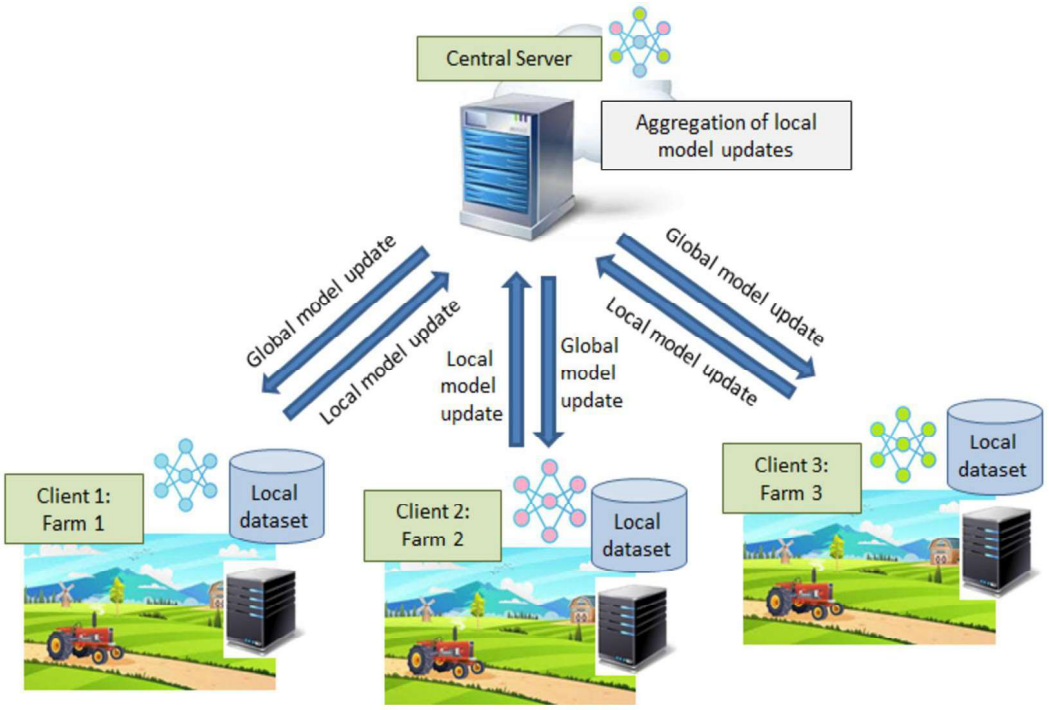


FIGURE 1.3 Federated learning-based agricultural system.

1.3.3 Mobile applications

The smartphones have several applications such as image recognition, speech recognition, recommendation applications, predictive text, etc. Smartphone users' data is highly confidential; hence, FL can play a crucial role in mobile applications [15]. Accurate prediction and recommendation can be performed without sharing raw data if FL is used in various mobile applications. The mobile devices have application-specific personal datasets of the users, which are used for local model training. The local model updates are aggregated by the edge server or cloud server to build the global model. The mobile

devices download the global model parameters. The mobile devices can build their personalized models also by training their local models using the downloaded global model parameters. In [32], a system is proposed to provide federated learning as a service (FLaaS) by enabling third party applications to build decentralized, privacy-aware, and collaborative ML models. The use of FL for latency-sensitive mobility applications is highlighted in [33]. In [34], an FL-based advertising platform was proposed to recommend mobile applications based on ecosystems. Mobile applications for health monitoring and activity recognition have also become popular nowadays for smart and healthy lifestyle recommendations. In [35], mobile-cloud FL framework was discussed for mobile applications with a case study on activity recognition. In [36], the use of FL in mobile health applications was demonstrated. Like physical health, mental health also plays a significant role in the general health and fitness of a person. In [37], FL was used for mental health monitoring. FL can also play significant roles for music and movie recommendation applications [38,39].

1.3.4 Autonomous vehicles

Road navigation and decision-making are vital for autonomous vehicles. FL can be used for autonomous vehicles to build an accurate decision-making model without sharing raw driving data with the central server. FL-based vehicular network [40] can perform road traffic prediction and select the optimal route to the destination. For connected and autonomous vehicles, CFL and DFL both have gained popularity [41]. In [42], FL was used for object detection in autonomous vehicles. In [43], FL was used in autonomous driving vehicles to provide end-to-end on-device ML. In [44], a blockchain-based decentralized FL framework was proposed for privacy-enhanced vehicular communication networking. Privacy is a critical issue of vehicular edge computing (VEC). In [45], FL was used for autonomous driving by protecting data privacy through collaborative model training using edge computing server without raw data sharing. In [46], deep FL was used for autonomous driving. In this work, the authors considered decentralized peer-to-peer model and proposed a federated autonomous driving network. In [47], Byzantine-Fault-Tolerant privacy-aware DFL was proposed for autonomous vehicles. This work also used peer-to-peer network to form a DFL framework. In [40], FL with genetic algorithm was used for optimal route selection in edge computing-based vehicular network.

1.3.5 Intrusion detection

To build a robust intrusion detection model [48], FL can play an important role. The network traffic data from different sources can be used separately for the local models' training, and the updates can be aggregated to build a robust intrusion detection model, without sharing raw data. In [49], the use of FL for intrusion detection was studied in detail with a discussion on the existing approaches. In [50,51], the authors illustrated the use of FL for intrusion detection. There are several challenges still remain such as communication overhead, non-IID data, resource management in low-power IoT devices, and federated poisoning attacks [50]. In [52], the use of FL for intrusion detection in IoT were discussed. In [53], FL-based intrusion detection was proposed for smart agriculture. Future research directions of FL-based intrusion detection systems include the use of edge computing, secure channels, encryption standards, model heterogeneity and interpretability, and efficient handling of non-IID data [50].

Beyond the various types of FL applications discussed above, there are some other areas also where FL can play an important role, such as smart grids, robotics, financial sectors, etc. For energy optimization and predicting demands in smart grids without affecting users' privacy, FL can be used [54]. For swarm robotics, FL can be used [55,56]. It will permit the robots to learn from each other's experiences and improve their performance. For personalized financial services without sharing raw data and prevention of frauds, an FL-based model can play an important role. As no raw data is shared, users' data is protected. Further, an accurate decision-making model can be built through collaboration among various financial organizations [16]. FL can be also used with edge computing [9] to provide an accurate decision-making framework for various real-time applications, where response time needs to be minimal. The FL and edge computing-based frameworks for various applications are studied in several research works [9,10,57].

1.4 Challenges of FL

Though FL provides privacy-aware and accurate prediction and recommendation models, there are several challenges of FL [13,14,29,58]. In this section, we highlight some of the significant issues of FL, which need to be taken care of.

1.4.1 Communication overhead

In FL, the model updates are transmitted between the clients and the server in case of CFL, and among the clients in DFL. Hence, a high communication overhead is involved in both CFL and DFL [58]. Therefore, a subset of client selection at each round along with a periodic updates exchange process can be used to reduce the communication overhead. The number of communication rounds in FL can also be reduced to diminish the communication overhead. However, there should be a trade-off between the communication overhead and the prediction accuracy in that case, so that the model's performance does not degrade. Hence, developing a communication-efficient FL framework is a significant challenge of FL.

1.4.2 Heterogeneity

System heterogeneity and data heterogeneity are two crucial issues of FL [58]. The clients participating in the FL process have different configurations in terms of memory, processing ability, storage, network bandwidth, battery life, etc. Further, the local data contained by the clients also differ in terms of sample space and feature space. The non-identical distribution of the datasets across the participating clients creates a major impact on the model's performance. Further, class imbalance issues in the local datasets, the limited number of samples in the local datasets, etc., also create bottlenecks in the performance of the model.

1.4.3 Resource limitation of clients

The clients can have limited resources in terms of processing ability, memory, storage, battery power, etc. As FL is a collaborative process, the resource limitation of the clients may cause performance degradation. This problem is known as the Straggler effect [59]. Clients with limited resources may slow down during the execution of DL models. As a result, the entire process becomes slow, and the performance of the model is degraded.

1.4.4 Privacy and security

Though, FL avoids raw data sharing, and protects data privacy, there is still a probability of gradient information leakage during the exchange of model updates. Further, a malicious client can affect the entire process. The attacks can be introduced by one or more compromised clients or a compromised server. Various types of attacks are possible in FL, such as model poisoning attacks, backdoor attacks, data poisoning attacks, and membership inference attacks [29]. By identifying malicious participants, data poisoning attacks can be prevented [29]. On the other hand, the rejection of local models based on error rate and loss function, can prevent model poisoning attacks [29]. The backdoor attacks can be prevented using weak differential privacy [29]. Finally, homomorphic encryption, differential privacy, and a trusted execution environment can prevent membership inference attacks [29]. In [60], federated graph anomaly detection can be performed for training high-quality graph anomaly detection models through collaboration on distributed graph data.

Though the challenges are critical, various research is going on to address them. One-shot FL [29] can be used to reduce communication overhead. In one-shot FL, the global model is learned in a single round of communication. Federated offloading, lightweight local models, etc., can be used to deal with the issue of resource limitations of the clients. Hessian-weighted aggregation [61] can address the issue of statistically heterogeneous data. Resource-aware clustering can address the issue of heterogeneous participants in FL [62]. Incentive mechanisms can also help to overcome various issues by providing rewards to the participating nodes based on their performance. Integration of quantum computing with FL is also an emerging future research direction [63].

1.5 Conclusion

This chapter provides a brief overview of federated learning. At first, we have provided an overview of machine learning and deep learning in this chapter. Then, the requirements of federated learning are briefly demonstrated. The characteristics, types, and applications of federated learning are discussed in detail. Finally, the research challenges of federated learning are discussed, and the future research solutions to address the challenges are also briefly mentioned in the chapter.

References

[1] Lavanya Sharma, Pradeep Kumar Garg, Technological Prospects and Social Applications of Society 5.0, CRC Press, 2023.

[2] Dinh C. Nguyen, Ming Ding, Pubudu N. Pathirana, Aruna Seneviratne, Jun Li, H. Vincent Poor, Federated learning for Internet of Things: a comprehensive survey, IEEE Communications Surveys and Tutorials 23 (3) (2021) 1622–1658.

[3] Anwesha Mukherjee, Debashis De, Rajkumar Buyya, Cloud computing resource management, in: Resource Management in Distributed Systems, Springer, 2024, pp. 17–37.

[4] Megha Sharma, Abhinav Tomar, Abhishek Hazra, Edge computing for Industry 5.0: fundamental, applications, and research challenges, IEEE Internet of Things Journal 11 (11) (2024) 19070–19093.

[5] Zhi-Hua Zhou, Machine Learning, Springer Nature, 2021.

[6] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[7] Taiwo Oladipupo Ayodele, Types of machine learning algorithms, in: New Advances in Machine Learning, 2010, Chapter 3, pp. 19–48.

[8] Osval Antonio Montesinos López, Abelardo Montesinos López, Jose Crossa, Fundamentals of artificial neural networks and deep learning, in: Multivariate Statistical Machine Learning Methods for Genomic Prediction, Springer, 2022, pp. 379–425.

[9] Tanushree Dey, Somnath Bera, Anwesha Mukherjee, Debashis De, Rajkumar Buyya, FLyer: federated learning-based crop yield prediction for Agriculture 5.0, IEEE Transactions on Artificial Intelligence 6 (7) (2025) 1943–1952.

[10] Somnath Bera, Tanushree Dey, Anwesha Mukherjee, Debashis De, FLAG: federated learning for sustainable irrigation in Agriculture 5.0, IEEE Transactions on Consumer Electronics 70 (1) (2024) 2303–2310.

[11] Sawsan AbdulRahman, Hanine Tout, Hakima Ould-Slimane, Azzam Mourad, Chamseddine Talhi, Mohsen Guizani, A survey on federated learning: the journey from centralized to distributed on-site learning and beyond, IEEE Internet of Things Journal 8 (7) (2020) 5476–5497.

[12] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, Yuan Gao, A survey on federated learning, Knowledge-Based Systems 216 (2021) 106775.

[13] Jie Wen, Zhixia Zhang, Yang Lan, Zhihua Cui, Jianghui Cai, Wensheng Zhang, A survey on federated learning: challenges and applications, International Journal of Machine Learning and Cybernetics 14 (2) (2023) 513–535.

[14] Anwesha Mukherjee, Rajkumar Buyya, Federated learning architectures: a performance evaluation with crop yield prediction application, Software, Practice & Experience 55 (2025) 1165–1184.

[15] Li Li, Yuxi Fan, Mike Tse, Kuo-Yi Lin, A review of applications in federated learning, Computers & Industrial Engineering 149 (2020) 106854.

[16] Walaa Hassan, Habiba Mohamed, Applications of federated learning in AI, IoT, healthcare, finance, banking, and cross-domain learning, in: Artificial Intelligence Using Federated Learning, CRC Press, 2024, pp. 175–195.

[17] Dinh C. Nguyen, Quoc-Viet Pham, Pubudu N. Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia Dobre, Won-Joo Hwang, Federated learning for smart healthcare: a survey, ACM Computing Surveys 55 (3) (2022) 1–37.

[18] Hao Guan, Pew-Thian Yap, Andrea Bozoki, Mingxia Liu, Federated learning for medical image analysis: a survey, Pattern Recognition 151 (2024) 110424.

[19] Dongzhou Cheng, Lei Zhang, Can Bu, Xing Wang, Hao Wu, Aiguo Song, ProtoHAR: prototype guided personalized federated learning for human activity recognition, IEEE Journal of Biomedical and Health Informatics 27 (8) (2023) 3900–3911.

[20] Adnan Qayyum, Kashif Ahmad, Muhammad Ahtazaz Ahsan, Ala Al-Fuqaha, Junaid Qadir, Collaborative federated learning for healthcare: multi-modal COVID-19 diagnosis at the edge, IEEE Open Journal of the Computer Society 3 (2022) 172–184.

[21] Md Fahimuzzman Sohan, Anas Basalamah, A systematic review on federated learning in medical image analysis, IEEE Access 11 (2023) 28628–28644.

[22] Mohammed Adnan, Shivam Kalra, Jesse C. Cresswell, Graham W. Taylor, Hamid R. Tizhoosh, Federated learning and differential privacy for medical image analysis, Scientific Reports 12 (1) (2022) 1953.

[23] Sajid Nazir, Mohammad Kaleem, Federated learning for medical image analysis with deep neural networks, Diagnostics 13 (9) (2023) 1532.

[24] Zhiwen Xiao, Xin Xu, Huanlai Xing, Fuhong Song, Xinhan Wang, Bowen Zhao, A federated learning system with enhanced feature extraction for human activity recognition, Knowledge-Based Systems 229 (2021) 107338.

[25] Tao Tang, Zhuoyang Han, Zhen Cai, Shuo Yu, Xiaokang Zhou, Taiwo Oseni, Sajal K. Das, Personalized federated graph learning on non-IID electronic health records, IEEE Transactions on Neural Networks and Learning Systems 35 (9) (2024) 11843–11856.

[26] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q.S. Quek, H. Vincent Poor, Federated learning with differential privacy: algorithms and performance analysis, IEEE Transactions on Information Forensics and Security 15 (2020) 3454–3469.

[27] Krista Rizman Žalik, Mitja Žalik, A review of federated learning in agriculture, Sensors 23 (23) (2023) 9566.

[28] Somnath Bera, Tanushree Dey, Anwesha Mukherjee, Pronaya Bhattacharya, Debashis De, Fedchain: decentralized federated learning and blockchain-assisted system for sustainable irrigation, IEEE Transactions on Consumer Electronics (2024).

[29] Priyanka Mary Mammen, Federated learning: opportunities and challenges, arXiv preprint, arXiv:2101.05428, 2021.

[30] Giannis Gallios, Nikolaos Tsakiridis, Nikolaos Tziolas, Federated learning applications in soil spectroscopy, Geoderma 456 (2025) 117259.

[31] Murali Krishna Senapaty, Abhishek Ray, Neelamadhab Padhy, Enhancing soil fertility prediction through federated learning on IoT-generated datasets with a feature selection perspective, Engineering Proceedings 82 (1) (2024) 39.

[32] Jiahui Geng, Neel Kanwal, Martin Gilje Jaatun, Chunming Rong, DID-eFed: facilitating federated learning as a service with decentralized identities, in: Proceedings of the 25th International Conference on Evaluation and Assessment in Software Engineering, 2021, pp. 329–335.

[33] Melike Gecer, Benoit Garbinato, Federated learning for mobility applications, ACM Computing Surveys 56 (5) (2024) 1–28.

[34] Jiang Bian, Jizhou Huang, Shilei Ji, Yuan Liao, Xuhong Li, Qingzhong Wang, Jingbo Zhou, Dejing Dou, Yaqing Wang, Haoyi Xiong, Feynman: federated learning-based advertising for ecosystems-oriented mobile apps recommendation, IEEE Transactions on Services Computing 16 (5) (2023) 3361–3372.

[35] Xiaopeng Jiang, Han Hu, Thinh On, Phung Lai, Vijaya Datta Mayyuri, An Chen, Devu M. Shila, Adriaan Larmuseau, Ruoming Jin, Cristian Borcea, et al., FLSys: toward an open ecosystem for federated learning mobile apps, IEEE Transactions on Mobile Computing 23 (1) (2022) 501–519.

[36] Tongnian Wang, Yan Du, Yanmin Gong, Kim-Kwang Raymond Choo, Yuanxiong Guo, Applications of federated learning in mobile health: scoping review, Journal of Medical Internet Research 25 (2023) e43006.
[37] Banuchitra Suruliraj, Rita Orji, Federated learning framework for mobile sensing apps in mental health, in: 2022 IEEE 10th International Conference on Serious Games and Applications for Health (SeGAH), IEEE, 2022, pp. 1–7.
[38] Jasper Sha, Nathaniel Basara, Joseph Freedman, Hailu Xu, Flor: a federated learning-based music recommendation engine, in: 2022 International Conference on Computer Communications and Networks (ICCCN), IEEE, 2022, pp. 1–2.
[39] David Neumann, Andreas Lutz, Karsten Müller, Wojciech Samek, A privacy preserving system for movie recommendations using federated learning, ACM Transactions on Recommender Systems 3 (2) (2024) 1–51.
[40] Sushovan Khatua, Anwesha Mukherjee, Debashis De, FedGen: federated learning-based green edge computing for optimal route selection using genetic algorithm in Internet of Vehicular Things, Vehicular Communications 49 (2024) 100812.
[41] Vishnu Pandi Chellapandi, Liangqi Yuan, Christopher G. Brinton, Stanislaw H. Żak, Ziran Wang, Federated learning for connected and automated vehicles: a survey of existing approaches and challenges, IEEE Transactions on Intelligent Vehicles 9 (1) (2023) 119–137.
[42] Deepthi Jallepalli, Navya Chennagiri Ravikumar, Poojitha Vurtur Badarinath, Shravya Uchil, Mahima Agumbe Suresh, Federated learning for object detection in autonomous vehicles, in: 2021 IEEE Seventh International Conference on Big Data Computing Service and Applications (BigDataService), IEEE, 2021, pp. 107–114.
[43] Hongyi Zhang, Jan Bosch, Helena Holmström Olsson, End-to-end federated learning for autonomous driving vehicles, in: 2021 International Joint Conference on Neural Networks (IJCNN), IEEE, 2021, pp. 1–8.
[44] Shiva Raj Pokhrel, Jinho Choi, A decentralized federated learning approach for connected autonomous vehicles, in: 2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), IEEE, 2020, pp. 1–6.
[45] Yijing Li, Xiaofeng Tao, Xuefei Zhang, Junjie Liu, Jin Xu, Privacy-preserved federated learning for autonomous driving, IEEE Transactions on Intelligent Transportation Systems 23 (7) (2021) 8423–8434.
[46] Anh Nguyen, Tuong Do, Minh Tran, Binh X. Nguyen, Chien Duong, Tu Phan, Erman Tjiputra, Quang D. Tran, Deep federated learning for autonomous driving, in: 2022 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2022, pp. 1824–1830.
[47] Jin-Hua Chen, Min-Rong Chen, Guo-Qiang Zeng, Jia-Si Weng, BDFL: a Byzantine-fault-tolerance decentralized federated learning method for autonomous vehicle, IEEE Transactions on Vehicular Technology 70 (9) (2021) 8639–8652.
[48] Elena Fedorchenko, Evgenia Novikova, Anton Shulepov, Comparative review of the intrusion detection systems based on federated learning: advantages and open challenges, Algorithms 15 (7) (2022) 247.
[49] Ansam Khraisat, Ammar Alazab, Sarabjot Singh, Tony Jan, Alfredo Jr.Gomez, Survey on federated learning for intrusion detection system: concept, architectures, aggregation strategies, challenges, and future directions, ACM Computing Surveys 57 (1) (October 2024).
[50] Shaashwat Agrawal, Sagnik Sarkar, Ons Aouedi, Gokul Yenduri, Kandaraj Piamrat, Mamoun Alazab, Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, Thippa Reddy Gadekallu, Federated learning for intrusion detection system: concepts, challenges and future directions, Computer Communications 195 (2022) 346–361.
[51] Jose Luis Hernandez-Ramos, Georgios Karopoulos, Efstratios Chatzoglou, Vasileios Kouliaridis, Enrique Marmol, Aurora Gonzalez-Vidal, Georgios Kambourakis, Intrusion detection based on federated learning: a systematic review, ACM Computing Surveys 57 (12) (July 2025).
[52] Enrique Mármol Campos, Pablo Fernández Saura, Aurora González-Vidal, José L. Hernández-Ramos, Jorge Bernal Bernabe, Gianmarco Baldini, Antonio Skarmeta, Evaluating federated learning for intrusion detection in Internet of Things: review and challenges, Computer Networks 203 (2022) 108661.
[53] Othmane Friha, Mohamed Amine Ferrag, Lei Shu, Leandros Maglaras, Kim-Kwang Raymond Choo, Mehdi Nafaa, FELIDS: federated learning-based intrusion detection system for agricultural Internet of Things, Journal of Parallel and Distributed Computing 165 (2022) 17–31.
[54] Zikai Zhang, Suman Rath, Jiaohao Xu, Tingsong Xiao, Federated learning for smart grid: a survey on applications and potential vulnerabilities, ACM Transactions on Cyber-Physical Systems (2025).
[55] Seongin Na, Tomáš Rouček, Jiří Ulrich, Jan Pikman, Tomáš Krajník, Barry Lennox, Farshad Arvin, Federated reinforcement learning for collective navigation of robotic swarms, IEEE Transactions on Cognitive and Developmental Systems 15 (4) (2023) 2122–2131.
[56] Yu Xianjia, Jorge Peña Queralta, Jukka Heikkonen, Tomi Westerlund, Federated learning in robotic and autonomous systems, Procedia Computer Science 191 (2021) 135–142.
[57] Ruijin Wang, Jinshan Lai, Zhiyang Zhang, Xiong Li, Pandi Vijayakumar, Marimuthu Karuppiah, Privacy-preserving federated learning for Internet of Medical Things under edge computing, IEEE Journal of Biomedical and Health Informatics 27 (2) (2022) 854–865.
[58] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, Virginia Smith, Federated learning: challenges, methods, and future directions, IEEE Signal Processing Magazine 37 (3) (2020) 50–60.
[59] Anwesha Mukherjee, Rajkumar Buyya, A joint time and energy-efficient federated learning-based computation offloading method for mobile edge computing, arXiv preprint, arXiv:2409.02548, 2024.
[60] Xiangjie Kong, Wenyi Zhang, Hui Wang, Mingliang Hou, Xin Chen, Xiaoran Yan, Sajal K. Das, Federated graph anomaly detection via contrastive self-supervised learning, IEEE Transactions on Neural Networks and Learning Systems 36 (5) (2024) 7931–7944.
[61] Adnan Ahmad, Wei Luo, Antonio Robles-Kelly, Robust federated learning under statistical heterogeneity via Hessian-weighted aggregation, Machine Learning 112 (2) (2023) 633–654.
[62] Rahul Mishra, Hari Prabhat Gupta, Garvit Banga, Sajal K. Das, Fed-RAC: resource-aware clustering for tackling heterogeneity of participants in federated learning, IEEE Transactions on Parallel and Distributed Systems 35 (7) (2024) 1207–1220.
[63] A. Mathur, A. Gupta, S.K. Das, When federated learning meets quantum computing: survey and research opportunities, IEEE Communications Surveys and Tutorials (2026).