CIS 5639

Survey on Federated Learning in Edge Computing

Zhengkun Ye

04/28/2022

Abstract

Federated Learning is a machine learning scheme in which a shared prediction model can be collaboratively learned by many distributed nodes using their locally stored data. It can provide better data privacy because training data are not transmitted to a central server. Federated learning is well suited for edge computing applications and can leverage the computation power of edge servers and the data collected on widely dispersed edge devices. People may need to tackle many technical challenges to build such an edge-federated learning system. This survey provides the perspective of views on the applications, development tools, communication efficiency, security & privacy, migration, and scheduling in edge federated learning.

1. Introduction

1.1 Backgrounds of federated learning

Federated learning is a machine learning technique that trains an algorithm across multiple decentralized edge devices or servers holding local data samples, without exchanging them. In other words, federated learning is capable of building machine learning models based on data sets that are distributed across multiple devices while preventing data leakage. Federated learning opens up new directions for broad research areas, including the Artificial Intelligence and edge computing fields [1]. Federated learning provides a novel training method to build personalized models yet can simultaneously protect privacy effectively. Federated learning provides a privacy protection mechanism that can effectively use the computing resources of the terminal device to train the model, which prevents private information leakage during data transmission. Moreover, since the number of edge devices and the device in other fields is countless, there is as well a considerable number of valuable dataset resources, and federated learning can fully make use of it.

Federated learning is a distributed training algorithm that considers "privacy" and "security." Its core idea is still decentralization and distribution, whereas it has developed from traditional distributed computing and distributed control to today's distributed learning. In general, it is an inheritance and an innovation.

1.2 Challenges of federated learning

Federated learning, as a trending technique, faces many challenges. Several challenges that need to be overcome include: 1. Privacy protection: Federated learning is proposed to conquer the issue of privacy leakiness. Researchers want to make sure that the training models in each individual's edge device such as mobile phone and smartwatch will never reveal user's private information. 2. Insufficient amount of dataset: In traditional machine learning or neural networks, a large amount of data is desired to train a model with decent performance, but in a distributed environment, there is not always enough data on each edge device. In addition, [2] collecting all data centralized may lead to huge costs, especially for power restraining devices. Therefore, federated learning requires each device to train a local model using local data and then upload all local models to the server to aggregate them into a global model. 3. Statistical heterogeneity: With Non-IID (Non-Independent and Identically Distributed) data, performance can be greatly reduced. Due to the rapid increase of research interest in FL, many valuable papers addressing these mentioned challenges on federated learning have been published [3].

1.3 Contributions

The contributions of this survey are as follows: 1. Review the development of federated learning. 2. Introduce the existing research works about federated learning, especially from the aspect: Edge Computing. 3. Sort out the current challenges and potential directions for continuous research of the federated learning.

2. Related Work

2.1 Federated Learning for Wireless Communications

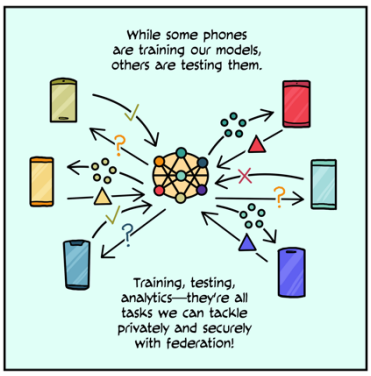
There is growing interest in the wireless communications community to complement traditional model-based design approaches with solutions based on data-driven machine learning (ML). While standard ML approaches mainly rely on the assumption of owning the data and processing head in a central entity, this is not always feasible in wireless communication applications due to the inaccessibility of private data and the significant communication overhead required to transfer the raw data to the central ML processor. Therefore, decentralized ML approaches that store the data at the generation location are more attractive. Due to its privacy-preserving nature, federated learning is particularly relevant for many wireless applications [4], especially in the context of fifth generation (5G) networks. Some up to par survey papers discussed challenges related to the security, privacy, and performance of the current federated algorithm and its essential considerations in wireless settings. These literatures also described a vivid and promising future for applying federated learning in wireless communications.

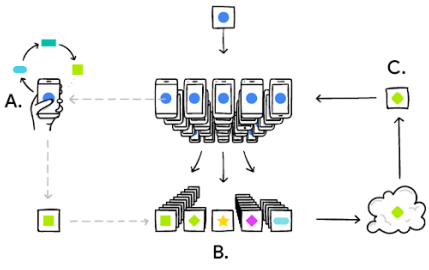
2.2 Federated Learning in Mobile Edge Networks

Mobile devices market is growing day by day. The devices have been equipped with increasingly advanced sensing and computing capabilities in recent years. Recently, the concept of federated learning has been introduced considering increasingly stringent data privacy legislation and growing privacy concerns. In federated learning, an edge device uses its local data to train the machine learning model required by the server. The end devices then send model updates instead of the original data to the server for aggregation. Federated learning can be used as an enabling technique in mobile edge networks as it enables collaborative training of machine learning models and deep learning for mobile edge network optimization. However, heterogeneous devices with different constraints are involved in large-scale and complex mobile edge networks [5]. This poses challenges in terms of communication cost, resource allocation, and privacy and security for large-scale implementations of FL.

2.3 Federated Learning for Healthcare Informatics

With the progressive development of hardware and software embedded in edge devices, more and more healthcare data can be obtained from clinical institutions, the detail information are becoming readily available for patients and insurance companies. They can check by themselves. Federal learning method has the potential to connect all isolated healthcare providers, hospitals or edge devices, allowing them to share experiences with guaranteed privacy [6]. However, most health systems suffer from data confusion and efficiency problems. Data collected from multiple sources vary in quality, and there are no uniform data standards.





3. Categorizations of federated learning

This section summarizes the categorizations of federated learning in three aspects: horizontal federated learning, vertical federated learning, and federated transfer learning.

3.1 Horizontal Federated Learning

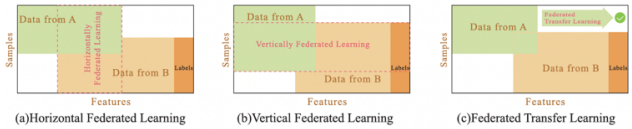
Horizontal federated learning is normally also to referred to as Homogenous Federated Learning [6], relating to the use of the same features. Horizontal Federated Learning takes place on Horizontal data. Federated learning has primarily considered supervised learning tasks where labels are naturally available on each client. With horizontal data, rows of data are available with consistent features. This is precisely the data type fed into a supervised machine learning task. In addition, each row may be implicitly or explicitly associated with a context. In the horizontal federated learning setting [7], multiple clients jointly train a model under the coordination of the central server, while the training data is kept on the client to ensure privacy.

3.2 Vertical Federated Learning

Vertical Federated Learning is also referred to as Heterogeneous Federated Learning [7], on account of differing feature sets. Vertical federated learning divides the datasets vertically (by user feature dimension), then takes out the part of data where users are the same [10]. Yet, user features are not the same for training. In other words, data in different columns have the same user (aligned by user). Therefore, vertical federated learning can increase the feature dimension of training data. An example is that Customer A (Amazon) has information about movies purchased by customers on Amazon, while Customer B (IMDB) has information about customers' movie reviews. Using these two data sets from different domains, movie reviews can be used to better serve customers.

3.3 Federated Transfer Learning

Federated transfer learning is the combination of vertical federated learning and transfer learning. Federated transfer learning is vertical federated learning, which uses pre-trained models trained on similar datasets to solve different problems [8, 14]. An example of federated transfer learning is training a personalized model, such as movie recommendations for a user's past browsing behavior. Federated transfer learning allows knowledge to be transferred across domains that do not have many overlapping features and users. In real-world scenarios, federated transfer learning can help us learn a radiology diagnosis system with other related but different tasks, such as image recognition tasks.



(a)Horizontal Federated Learning (b)Vertical Federated Learning (c)Federated Transfer Learning

4. Edge Computing

"Edge computing is a distributed computing framework that brings enterprise applications closer to data sources such as IoT devices or local edge servers." – IBM. At its inception, this proximity to data can deliver substantial business benefits, including faster insights, improved response times, and better bandwidth availability [16]. Edge computing is a computing model that provides intelligent services at the edge of the network physically close to the source of data, incorporating an open platform of the network, computing, storage, and application core capabilities. The location where edge computing occurs is called an edge node, which can be any node with computing and network resources between the data generation source and the cloud center. For example, a cell phone is the edge node between a person and a cloud center, and a gateway is the edge node between a smart home and a cloud center. Edge computing also belongs to a kind of distributed computing, in which the processing of data information, the operation of applications and even the implementation of some functional services are decentralized from the network center to the nodes at the edge of the network to process data nearby, without the need to upload a large amount of data to the remote core management platform.

4.1 Why do we need Edge Computing?

The Internet of Things (IoT) is booming in various fields, and the era of the Internet of Everything is approaching. Whereas, with the development of business and the rapid increase of IoT devices, it is gradually found that the cloud-based approach cannot meet the actual needs of many scenarios. **1**. Huge amount of data puts enormous pressure on network bandwidth. Cloud center has powerful processing performance and can handle vast amounts of data. However, with the development of the Internet of Things, almost all electronic devices can now be connected to the Internet, and these electronic devices will then generate massive amounts of data and transferring these massive amounts of data to the cloud center becomes a problem, which is a massive challenge for network bandwidth. **2**. Increased demand for low latency and collaborative work from connected devices. The system performance bottleneck of the cloud computing model lies in the limited network bandwidth. It takes a certain time to transmit massive data and a certain time to process the data in the cloud center, which increases the request-response time and makes the user experience extremely poor. In emerging IoT application scenarios, such as real-time voice translation and driverless cars, the response time requirements are incredibly high, and it is not realistic to rely on cloud computing. **3**. Connected devices involve personal privacy and security. A large amount of data in the terminal device will involve personal privacy and passing it to the cloud center will significantly increase data security risk. If edge computing can be used like octopus, massive data can be processed nearby, many devices can work together efficiently, and many problems can be solved. Thus, edge computing can theoretically meet the critical needs of many industries in terms of agility, real-time data optimization, application intelligence, and security and privacy protection.

5. Federated Learning in Edge Computing

Training deep neural network on edge device is another aroused general interest topic and a challenging issue. FL is a way of enabling DL training by leveraging many client devices (e.g., mobile devices and the IoT devices) [15] in a network. This solves the issue that a single resource-constrained device may never support training a DL model due to its limited hardware resources. All mobile devices need to devote their storage and computing resources for their data training. However, this requirement is not always satisfied. When FL deploys identical neural network models to heterogeneous devices, the ones with weak computational capacities may significantly delay the synchronous parameter aggregation [12]. Besides, a large amount of memory and power are needed for the training of advanced DL models and the storage of model parameters [13]. Thus, many aspects still need to be carefully considered, including memory limitation, energy budget, communication, synchronization, resource distribution, and privacy. On the one hand, the communication consumption of federal learning is large, and the communication capacity of edge devices is limited, so designing the network structure and communication methods (such as hierarchical communication, point-to-point communication, etc.) is a primary research direction. And due to the edge devices being closer to the data generated by users, the data collected by edge devices in different areas have non-IID characteristics so Personalization research can be considered. Federal learning also has certain requirements for privacy protection, and the research and application of common differential privacy techniques have received much attention. Edge computing is about users processing and applying data at the nearest computing facility,

protecting their own privacy; federated learning is about using other remote data and protecting the privacy of the distant data, while collaboratively modeling it. The two are still very different in concept. When federated learning is training a model, it also uses its own clients and has its own computing devices, so it will have some integration points with edge computing at this point.

With the increase in storage and computing power of mobile devices themselves, more and more mobile devices are processing data locally, such as sensors, smart wearable devices, and in-car applications. Current machine learning techniques have achieved great success in computer vision, natural language processing, pattern recognition, etc. However, current machine learning approaches are centralized, with data centers or cloud servers having access to the data. Federated learning, as a new distributed machine learning paradigm, can co-create models in machine learning without data going out of the local area by using the storage and computing capabilities of the device itself, thus protecting data privacy and thus effectively solving the problem of data silos. Furthermore, edge computing can provide computing, storage, and network resources close to the device, providing the basis for high-bandwidth, low-latency applications.

Most of the existing federated learning systems are synchronous federated learning, where the convergence speed of each training round is determined by the slowest running device. Asynchronous federated learning has been proposed in several papers to address the dependence of the training speed on the slowest device [17-18]. Asynchronous federated learning can also allow users participating in training to drop out midway or allow new users to join during the training process, which is more in line with the realistic environment of federated learning, where users participating in federated learning training may drop out or join midway due to network problems or power problems. And the proposed asynchronous federated learning can better solve the problem of federated learning scalability. Due to the need to ensure convergence, synchronous federated learning is still the most used method nowadays. Given the many advantages of asynchronous federated learning, asynchronous federated learning algorithms should also be explored and designed in the future.

5.1 Edge Cloud Computing

The construction of smart cities will have a large number of devices connected to the Internet, and data will grow explosively. Traditional cloud-centric computing can no longer afford the amount of data generated by smart cities, and new network technologies are needed to bear the pressure of data processing. Edge cloud computing is a powerful synergy and complement to cloud computing. By processing and analyzing data close to the data source side, edge cloud computing greatly improves network response speed and effectively reduces latency. At the same time, edge cloud computing relieves the processing pressure on the cloud by processing some of the data.

6. Applications

An increasing number of application domains involve a large set of heterogeneous clients, e.g., mobile phones and IoT devices [11]. As a result, federated learning can be utilized extensively with edge devices in many scenarios where privacy preservation and resource utilization are critical. This section discusses some techniques of edge federated learning and some recent work applied in these scenarios.

6.1 Healthcare System

In October 2019, NVIDIA introduced federated learning technology to its Clara platform dedicated to medical imaging and released a federated learning system for medical image analysis with privacy protection in collaboration with King's College London, UK. With the Clara platform supporting Federated Learning, researchers can significantly simplify the deployment of this system and securely and easily configure the Federated Learning central server and collaboration clients, providing everything needed to start a Federated Learning project, including application containers and initial AI models. Hospitals participating in this project use the Clara AI-assisted annotation tool, which collaborates with hospital imaging devices to tag their patient image data. In addition, Clara can use pre-trained models and migratory learning technology to help radiologists perform tagging, reducing the time for complex 3D studies from hours to minutes. Each hospital will use this data to train the models on a local EGX server. The regional training results are shared back to the Federal Learning Center server via a secure link, and the central server updates the global model. The updated models are then synchronized with each hospital server so that each hospital can further train the new models. In addition to medical institutions, the edge federated transfer learning method shall be applied to personal health measurement devices as well. Moreover, several classical federated learning baisi algorithms have been proposed to build up accurate and personalized healthcare model such as, FedAvg [19] (standard FL method, where all users share one global model), FedPer [20] (all the users share their lower K layers and their upper layers user-specific)

and pFedMe [21] (an algorithm using the distance between the global model and the user's local model as the user's regularized loss functions, the global model is an average aggregation of all the local models at the server). They can collaboratively learn a model at the network edge yet capturing personalization simultaneously.

Despite this, there are still only a handful of specific healthcare scenarios where federal learning has been implemented. This advanced architecture still faces specific problems, including generally poor data quality in healthcare organizations, lack of physician involvement in model training that makes it difficult to convince physicians to use it, lack of sufficient incentives to engage the data side, difficulty in training models with personalization, and lack of model accuracy for complex scenarios.

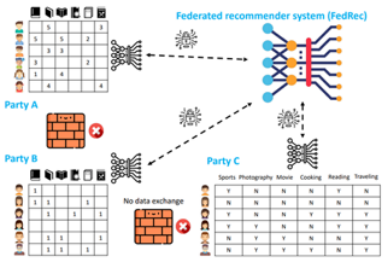
6.2 Vehicular Network

There are also research works target federated learning in in-vehicle edge computing. Regarding the implementation of model selection, the central server does not know the image quality and computational power of the on-board client, whose privacy is protected in this federated learning framework. To overcome this information asymmetry, we use two-dimensional contract theory as a distributed framework to facilitate the interaction between the central server and the on-board client. The formulaic problem is then transformed into a tractable problem by sequentially relaxing and simplifying the constraints, and finally solved by a greedy algorithm.

The vehicle data generated by the device, such as GPS-detected position and orientation, images taken by on-board cameras, and pressure data from oil pressure sensors, is a valuable resource for vehicle manufacturers to provide intelligent navigation services and alerts. The onboard computer collects the locally generated sensing data and then uploads it to the Vehicle Edge Computing (VEC) system to train a local learning model. Edge federated learning in VEC can meet user needs for intelligent vehicle decision making. For example, a cluster-based federated energy demand learning approach was implemented by *Saputra et al*. In [22], the EV network performs energy demand prediction in the considered region. Moreover, image classification is a typical task in in-vehicle networks. The performance and efficiency of edge federated learning are highly influenced by the quality of the training data and the computational power of the edge nodes, respectively. Ye et al. proposed a selective model aggregation method [23], where models are selected if their training images are of high quality and the edge nodes have sufficient computational power. To further improve the learning accuracy and encourage devices with high quality data to join the model training process, Kang et al. used contract theory to design an incentive mechanism [24]. In addition, self-driving cars are equipped with more sensors than normal vehicles, such as LIDAR and ultrasonic sensors, to sense the surroundings without human intervention. Joint edge learning is an ideal solution for learning privacy-preserving machine learning models from non-IID vehicle data in VEC systems [25].

6.3 Intelligent Recommendation

When you open online shopping Apps or web pages, why are most of the products displayed interest you? It seems that these APPs know you better than you do! This is where the recommendation system comes into play. The recommendation system usually collects your information (for example, the information you fill in when you register on the app, such as gender, age, and geographic location) and carves a user profile by your behavior on the app (for example, the content you browse on the app, etc.), and then recommends some items that may be of interest to you through the recommendation algorithm. However, the premise of enjoying a personalized recommendation service is that you need to deliver your personal information and history to the APP, which is often called "privacy for service." With the promulgation of the General Data Protection Regulation and other privacy and data protection laws and regulations and the increase in people's awareness of privacy protection, user privacy security has become more and more critical. So, while enjoying personalized recommendation service, you can't help but think, can we protect users' privacy? Federated learning allows users in a recommendation system to regain control of their data. During the training process of federated learning, the user's original data is always kept locally by the user (client). The model training and parameter updating are performed by sharing encrypted or intermediate parameters that do not contain private information between the server and the user, thus building an effective machine learning model while protecting the user's privacy.



Therefore, the combination of federal learning and recommender systems aims to provide accurate and personalized services to users while protecting user privacy and trade secrets. Communication loss is one of the main reasons that affect the performance of federated learning. Due to the high dimensional nature of the features and real-time requirements of the recommender system, the communication cost problem in the federal recommender system will be very serious. With the increasing number of participants, it will be a challenge to design better model parallelism and model update scheduling schemes to guarantee the convergence of federated recommendation models. The synchronous client-server architecture used in many federated learning systems is not conducive to flexible scaling. In a recommendation system, millions of users use the recommendation service. Too many participants accessing at the same time can congest the network on the central server, and it is difficult to guarantee that all participants can participate in the whole process of federated training. As a result, the performance of the federated model can be seriously affected. The problem of non-independent and homogeneous distribution of data still exists and causes damage to the intelligent recommender system. The performance of the federal recommendation model will be severely degraded due to the highly skewed non-independent homogeneously distributed data. As the distance between the data distributions of each participant becomes larger, the accuracy of the model will decrease accordingly.

Furthermore, in reality, there is a high probability that the participants in a recommender system are not trustworthy. The participants do not follow the assumptions often used (both the participant and the central server are semi-honest). They may misbehave in gradient collection or parameter updates, and the server may be malicious. Thus, “honest” participants may be at risk of privacy breaches in these cases.

7. Discussion

As one of the hottest data security class technologies nowadays, federal learning has gone into various application scenarios such as banking, securities, insurance, healthcare, government, and city management. This has a profound background of the times. Since society has entered the Internet era, the digitization of enterprise production, management, and operation processes has been gradually realized in many industries. As a result, the accumulation of data and data value mining have become the focus of attention. In addition to applying their own accumulated data resources, the use of valuable data from other enterprises and other industries has naturally become one of the ways to solve the problem. Correspondingly, how to ensure the security of data in the process of use, protect personal privacy from leakage, and prohibit unauthorized data from being improperly disseminated, stored, and used has become a problem. Federated learning is seen as an excellent technique to solve this problem. Federated Learning is essentially a distributed machine learning technique that trains algorithms on multiple distributed edge devices or servers without exchanging data samples. This is to achieve the role of secure federated modeling based on "available but not visible" data. Google AI first introduced the concept of "federated learning" in 2017 in a blog post, "Federated Learning: Collaborative Machine Learning without Centralized Training Data." 2019 saw the implementation of Google's first product-level federated learning system, with a focus on federated average algorithms and analytics running on cell phones for the C-suite.

The federated learning architecture consists of three main workflows: the master device (central server) sends the current global model to the working node devices (individual clients); the working node devices update the local model using the local training dataset and the global model and send the local model to the master device; the master device computes a new global model by aggregating the local model according to certain aggregation rules. For example, the Mean Aggregation Rule, which uses the average of the local model parameters as the global model, is widely used in a non-adversarial setting. However, the mean value of the global model can be arbitrarily manipulated even if only one worker device is corrupted. Therefore, the machine learning community has recently proposed several aggregation rules (e.g., Krum, Bulyan, trimmed mean and median) with the aim of providing robustness against Byzantine failures of some specific worker node devices. There are also many attack problems can be discussed for both edge computing and federated learning.

There are promising directions for the potential future work about edge federated learning. First, the current scheduling and resource allocation algorithms always try to minimize the training time. However, in this case, the central server may not select nodes with limited computational resources or unstable networks because of the long waiting time. Therefore, the data on these nodes will not be used for model training, which leads to bias in model training. We can group nodes with similar training times and consolidate their weights before sending them to the server

in batches to alleviate this problem. In this way, all the data can be used for training, and the training time can be reduced. Second, most previous work has focused on using numerical experiments to show performance in asynchronous training. However, theoretical studies on asynchronous training are still lacking. In edge federation learning, mathematical analysis and comparison of asynchronous and synchronous training are needed. Third, incentives in edge federation learning seem to be a less studied topic. It is promising to explore how to motivate the participation of nodes with high-quality data and abundant computational resources.

8. Conclusion

In this survey, we intently investigate edge federated learning, which is a paradigm to implement federated learning in edge computing environments. Federated learning is still a relatively new field with many research opportunities to improve privacy-preserving techniques. This includes system heterogeneity, statistical heterogeneity, privacy concerns, trade-off between efficiency and privacy, etc. This brings forth many complex problems in federated learning that need to be addressed before federated learning can be widely adopted by the industry. With the rapid development of edge computing and federated learning, more and more edge federated learning co-training methods are being developed for better user experience and privacy protection. However, researchers will need more efforts to solve those open problems in edge federated learning.

REFERENCES

[1] Yang Q, Liu Y, Cheng Y, et al. Federated learning[J]. Synthesis Lectures on Artificial Intelligence and Machine Learning, 2019, 13(3): 1-207.

[2] Xie M, Long G, Shen T, et al. Multi-center federated learning[J]. arXiv preprint arXiv:2108.08647, 2021.

[3] Li Q, He B, Song D. Model-contrastive federated learning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 10713-10722.

[4] Niknam S, Dhillon H S, Reed J H. Federated learning for wireless communications: Motivation, opportunities, and challenges[J]. IEEE Communications Magazine, 2020, 58(6): 46-51.

[5] Lim W Y B, Luong N C, Hoang D T, et al. Federated learning in mobile edge networks: A comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2020, 22(3): 2031-2063.

[6] J. Xu, B.S. Glicksberg, C. Su, P. Walker, J. Bian, F. Wang Federated learning for healthcare informatics J. Healthc. Inform. Res. (2020), pp. 1-19

[7] Huang, Wei, et al. "Fairness and accuracy in horizontal federated learning." Information Sciences 589 (2022): 170-185.

[8] Gao D, Liu Y, Huang A, et al. Privacy-preserving heterogeneous federated transfer learning[C]//2019 IEEE International Conference on Big Data (Big Data). IEEE, 2019: 2552-2559.

[9] Y. Chen, X. Qin, J. Wang, C. Yu, W. Gao Fedhealth: a federated transfer learning framework for wearable healthcare IEEE Intell. Syst. (2020)

[10] Zhang, Chen, et al. "A survey on federated learning." Knowledge-Based Systems 216 (2021): 106775.

[11] Xu Z, Yu F, Xiong J, et al. Helios: heterogeneity-aware federated learning with dynamically balanced collaboration[C]//2021 58th ACM/IEEE Design Automation Conference (DAC). IEEE, 2021: 997-1002.

[12] Z. Xu, F. Yu, J. Xiong, and X. Chen, “Helios: Heterogeneity-aware federated learning with dynamically balanced collaboration,” 2019, arXiv:1912.01684.

[13] M. Sankupellay and D. Konovalov, “Bird call recognition using deep convolutional neural network, ResNet-50,” in Proc. Acoust., vol. 7, 2018, pp. 1–8.

[14] Liu Y, Kang Y, Xing C, et al. A secure federated transfer learning framework[J]. IEEE Intelligent Systems, 2020, 35(4): 70-82.

[15] Savazzi, Stefano; Nicoli, Monica; Rampa, Vittorio (May 2020). "Federated Learning With Cooperating Devices: A Consensus Approach for Massive IoT Networks". IEEE Internet of Things Journal. 7 (5): 4641–4654.

[16] https://www.ibm.com/cloud/what-is-edge-computing

[17] Sprague M R, Jalalirad A, Scavuzzo M, et al. Asynchronous federated learning for geospatial applications[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2018: 21-28.

[18] Xie C, Koyejo S, Gupta I. Asynchronous federated optimization[J]. arXiv preprint arXiv:1903.03934, 2019.

[19] Sun T, Li D, Wang B. Decentralized federated averaging[J]. arXiv preprint arXiv:2104.11375, 2021.

[20] M.G. Arivazhagan, V. Aggarwal, A.K. Singh, S. Choudhary, Federated learning with personalization layers, arXiv:1912.00818 (2019).

[21] M.J. Sheller, B. Edwards, G.A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R.R. Colen, et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data Sci. Rep., 10 (1) (2020), pp. 1-12
[22] Y.M. Saputra, D.T. Hoang, D.N. Nguyen, E. Dutkiewicz, M.D. Mueck, S. Srikanteswara Energy demand prediction with federated learning for electric vehicle networks 2019 IEEE Global Communications Conference (GLOBECOM), IEEE (2019), pp. 1-6
[23] Ye D, Yu R, Pan M, et al. Federated learning in vehicular edge computing: A selective model aggregation approach[J]. IEEE Access, 2020, 8: 23920-23935.
[24] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y.-C. Liang, D.I. Kim Incentive design for efficient federated learning in mobile networks: a contract theory approach 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS), IEEE (2019), pp. 1-5
[25] A. Imteaj, M.H. Amini Distributed sensing using smart end-user devices: pathway to federated learning for autonomous IoT 2019 International Conference on Computational Science and Computational Intelligence (CSCI), IEEE (2019), pp. 1156-1161