Knowledge-Based Systems 216 (2021) 106775

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A survey on federated learning

Chen Zhang [a], Yu Xie [b,*], Hang Bai [a], Bin Yu [a], Weihong Li [a], Yuan Gao [c]

[a] School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi Province 710071, China
[b] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan 030006, China
[c] School of Electronic Engineering, Xidian University, Xi'an, Shaanxi Province 710071, China

ARTICLE INFO

Article history:
Received 3 September 2020
Received in revised form 12 December 2020
Accepted 11 January 2021
Available online 21 January 2021

Keywords:
Federated learning
Privacy protection
Machine learning

ABSTRACT

Federated learning is a set-up in which multiple clients collaborate to solve machine learning problems, which is under the coordination of a central aggregator. This setting also allows the training data decentralized to ensure the data privacy of each device. Federated learning adheres to two major ideas: local computing and model transmission, which reduces some systematic privacy risks and costs brought by traditional centralized machine learning methods. The original data of the client is stored locally and cannot be exchanged or migrated. With the application of federated learning, each device uses local data for local training, then uploads the model to the server for aggregation, and finally the server sends the model update to the participants to achieve the learning goal. To provide a comprehensive survey and facilitate the potential research of this area, we systematically introduce the existing works of federated learning from five aspects: data partitioning, privacy mechanism, machine learning model, communication architecture and systems heterogeneity. Then, we sort out the current challenges and future research directions of federated learning. Finally, we summarize the characteristics of existing federated learning, and analyze the current practical application of federated learning.

1. Introduction

1.1. Backgrounds of federated learning

With the development of big data, the amount of data is no longer the focus of our attention. The urgent problem that needs to be solved is the privacy and security of data. The leakage of data is never a small problem, and recently the public pay growing attention to data security [1–3]. Not only individuals, collectives and society are also strengthening the protection of data security and privacy. Taking the General data Protection Regulations implemented by the European Union on May 25th, 2018 as an example, GDPR [4] aims to protect users' personal privacy and data security. It requires operators to clearly express the user agreements and cannot deceive or induce users to give up the privacy requirements. In addition, operators are prohibited from training model without the user's permission. At the same time, it allows users to delete their private data. Similarly, China's Cyber Security Law of the People's Republic of China [5] and the General principles of the Civil Law of the People's Republic of China [6], which have been implemented since 2017, also point out that network operators shall not disclose, tamper with or destroy the personal information they collect. When conducting data transactions with the third, it is necessary to ensure that the proposed contract clearly specifies the scope of the data to be traded and the obligations of data protection. The establishment of these laws and regulations poses new challenges to the traditional data processing mode of artificial intelligence to varying degrees.

In the field of artificial intelligence, data is the foundation, therefore model training cannot be performed without data. However, data often exists in the form of data islands. The direct solution to data islands is to process the data in a centralized manner. The popular data processing method is through centralized collection, unified processing, cleaning and modeling. In most cases, data is leaked during collection and processing. With the improvement of regulations, user's private information is well protected, but it is getting harder to collect data to train models. How to legally solve the problem of data islands has attracted a lot of attention and thinking of artificial intelligence. To solve the dilemma of data silos, traditional data statistics methods are already stretched in the face of various regulations. Federated learning shifts the focus of research to the problem of data islands. The traditional machine learning mostly uses the centralized method to train the machine learning model, which requires the training data to be concentrated in the same server.

* Corresponding author.
E-mail address: sxlljcxy@gmail.com (Y. Xie).

https://doi.org/10.1016/j.knosys.2021.106775
0950-7051/

In fact, due to the laws and regulations of data privacy protection, the centralized training method which may leak the data and invade the privacy of the data owner is getting harder to be implemented. In the centralized training settings, if mobile phone users want to train machine learning models with their own data, it is obvious that the amount of their data is not enough. Therefore, before the federated learning, mobile phone users have to send their personal phone data to the central server, which can train the machine learning models with the data integrated from users. Compared with the centralized training method, federated learning which belongs to distributed training method, enables individual users in different spatial locations to collaborate with other users to learn machine learning models, and all personal data that may contain sensitive personal information can be retained on the device. With the help of federated learning, individual users can benefit from obtaining a well-trained machine learning model without having to send their privacy-sensitive personal data to a central server [7].

Federated learning opens up new research directions for artificial intelligence. Federated learning provides a novel training method to build personalized models without violating user privacy. With the advent of artificial intelligence chipsets, the computing resources of client devices have become more powerful. Artificial intelligence model training is also gradually shifting from the central server to the terminal equipment. Federated learning provides a privacy protection mechanism that can effectively use the computing resources of the terminal device to train the model, which prevent private information from being leaked during data transmission. Since the number of mobile devices and the devices in other fields is countless, there are a large amount of valuable dataset resources, and federated learning can make full use of it.

The main feature of federated learning is to ensure the privacy of users, but it is very different from traditional privacy protection algorithms applied in the field of big data, such as differential privacy and k-order anonymity. Federated learning mainly protects user privacy by exchanging encrypted processed parameters, while the attackers cannot get source data. All these guarantee that federated learning will not leak user privacy at the data level, and there is no violation of GDPR and other bills. Federated learning can be divided into horizontal federated learning, vertical federated learning and federated transfer learning according to the distribution of data. Horizontal federated learning is suitable in the case that the user features of the two datasets overlap a lot, but the users overlap little. Vertical federated learning is available in the case that the user features of the two datasets overlap little, but the users overlap a lot. In the case that the users and user features of the two datasets both rarely overlap, we can use transfer learning to overcome the lack of data or tags. Federated learning is similar to multi-party computing and distributed machine learning. There are many types of distributed machine learning, including distributed publishing model results, distributed storage training data, and distributed computing tasks. The parameter server in distributed machine learning is one of the tools to accelerate the training speed of machine learning models. It stores data on different working nodes in a distributed manner and allocates resources through a trusted central server in order to efficiently obtain the final training model. In federated learning, compared with distributed machine learning, each worker node is the sole owner of its own data and a training participant of the model.

The main embodiment of federated learning to ensure privacy is that users have complete autonomy over local data, which emphasizes the privacy protection of data owners. There are two main types of privacy protection mechanisms in a federated learning environment. A common method is to use encryption algorithms such as homomorphic encryption and secure aggregation. Another popular method is to add the noise of differential privacy to the model parameters. The federated learning [8] proposed by Google adopts a combination of secure convergence and differential privacy to ensure privacy. There are also other studies [9] that use only homomorphic encryption protection parameters to achieve privacy protection.

1.2. Challenges to federated learning

In order to effectively protect the privacy of enterprises and users, some challenges need to be solved in the federated learning. (1) Privacy protection: Since federated learning is proposed to solve the problem of privacy data protection in machine learning, we must ensure that the training model in federated learning does not reveal users' private information. (2) Insufficient amount of data: A large amount of data is required to train a model with excellent performance in traditional machine learning, but in a distributed environment, the amount of data on each mobile device is insufficient. On the other hand, collecting all data in a centralized manner can result in huge expenses. Therefore, federated learning requires each device to use local data to train the local model, and then all the local models are uploaded to the server to be aggregated into a global model. (3) Statistical heterogeneity: There are a large number of edge devices in the federal environment, and the data held by these devices may be Non-IID (Non-Independent and Identically Distributed). For example, in a smart medical system, the electronic medical record data structure of different types of diseases is different, and it is a big challenge to train these Non-IID data sets.

1.3. Main contributions

The main contributions of this paper are as follows: (1) Review the development of federated learning. (2) Introduce the existing work of federated learning from five aspects: Data Partitioning, Privacy Mechanism, Machine Learning Model, Communication Architecture and Systems Heterogeneity. (3) Sort out the current challenges and future research directions of federated learning. (4) Summarize the characteristics of existing federated learning, and analyzes the current practical application of federated learning.

2. Related works

Federated learning is actually a kind of encrypted distributed machine learning technology, in which participants can build a model without disclosing the underlying data, so that the self-owned data of each enterprise does not leave the local. Through the parameter exchange under the encryption mechanism, a virtual common model is established. Under such a mechanism, all parties involved can successfully link up the data island and move towards common development.

2.1. Definition of federated learning

In the practical application scenario [8], it is assumed that N users $\{\mathcal{U}_1, \ldots \mathcal{U}_n\}$ own their own database $\{\mathcal{D}_1, \ldots \mathcal{D}_n\}$, and each of them cannot directly access to other people's data to expand their own data. As shown in Fig. 1, federated learning is to learn a model by collecting training information from distributed devices. It contains three basic steps [10]: (1) Server sends the initial model to each device. (2) The device $\mathcal{U}_i$ does not need to share its own source data, but can federally train its own model $\mathcal{W}_i$ with the local data $\mathcal{D}_i$. (3) Server aggregates the collected local models $\{\mathcal{W}_1, \ldots \mathcal{W}_n\}$ to the global model $\mathcal{W}'$, and then update the

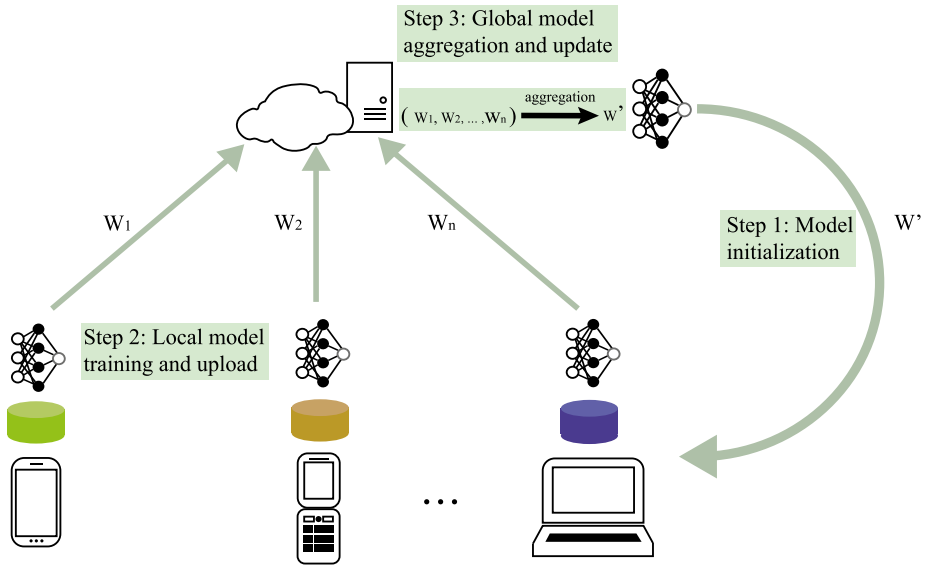

Fig. 1. A schematic diagram of federated learning. In order to guarantee the privacy of the data, federated learning only permits all the remote devices exchange the model gradient with central server. During this process, each distributed devices train their own model with the local data, then they upload the local model to the central server. After aggregating all the gathered models, the server returns the new global model to each devices.

global model to replace each user's local model. With the rapid development of federated learning, the efficiency and accuracy of federated training models are getting closer and closer to centralized training models [11]. It is playing an important role in many areas that need to take into account privacy.

2.2. The development of federated learning

As a new privacy protection framework, federated learning is not well understood by the public. The following examples describe the working process of federated learning. Suppose that there are a host of different enterprises want to collaborate on training a machine model [12]. According to the GDPR criterion, the data of each sides cannot be roughly merged without the consent of their respective users [4]. On the other hand, an enterprise can train a machine learning model according to its local data. It is assumed that all parties establish a task model, but it is difficult to train an ideal machine learning model because of the limited and incomplete data of their own enterprises. The purpose of federated learning is to solve these problems. Federated learning ensures that the local data of their respective enterprises do not go out. Under the principle of not violating the law of privacy protection, parameters are exchanged between the clients and the server through an encryption mechanism to establish a global model.

3. Categorizations of federated learning

This section summarizes the categorizations of federated learning in five aspects: data partition, privacy mechanisms, applicable machine learning models, communication architecture, and methods for solving heterogeneity. For easy understanding, we list the advantages and applications of these categorizations in Table 1.

3.1. Data partition

According to the different distribution patterns of sample space and feature space of data, as shown in Fig. 2, federated learning can be divided into three categories: horizontal federated learning, vertical federated learning, and federated transfer learning [12].

3.1.1. Horizontal federated learning

Horizontal federated learning is suitable in the case that the user features of the two datasets overlap a lot, but the users overlap little. Horizontal federated learning is to split the datasets horizontally (by the user dimension), then take out the part of the data that user features are the same but users are not exactly the same for training. In other words, data in different rows have the same data features (aligned by user features). Therefore, horizontal federated learning can increase the user sample size. For example, there are two providers of the same service in different regions whose user groups come from their respective regions and have little overlap with each other. However, their businesses are very similar, so the user features of the records are the same. In this respect, we can use horizontal federated learning to train a model, which can not only increase the total number of training samples, but improve the accuracy of the model. In horizontal federated learning, it is common for all parties to calculate and upload local gradients so that the central server can aggregate them into a global model. The processing and communication of gradients in horizontal federated learning may leak users' private information. The common solutions for this problem are homomorphic encryption [13], differential privacy [14] and secure aggregation [15], which can ensure the security of switching gradients in horizontal federated learning.

Google proposed a data federated modeling scheme for Android phone model updates in 2016 [8,10]: when a single user uses an Android phone, the user constantly updates the model parameters locally and uploads the parameters to the Android cloud, so that all data owners with the same feature dimension can establish a federated model. The system is a typical application of horizontal federated learning, which adopts the methods of differential privacy [14] and secure aggregation. Kim et al. [16] proposed a horizontal federated learning framework called BlockFL, in which each mobile device uses the block chain network to update the local learning model. Smith et al. [17] proposed a federated learning method called MOCHA to solve security problems in multitasking, which allows many sites to work together to complete tasks and ensure privacy and security. The multi-task federated learning also improves the communication cost of the original distributed multi-task learning and improves the fault tolerance of the original mechanism. In [11,18], the data

Table 1
Categorizations of federated learning.

Categorization	Methods	Advantage	Applications
Data partitioning	Horizontal federated learning	Increase user sample size	Android phone model update; logistic regression
	Vertical federated learning	Increase feature dimension	Decision tree; neural network
	Federated transfer learning	Increase user sample size and feature dimension	Transfer learning;
Privacy mechanism	Model aggregation	Avoid transmitting the original data	Deep network federation learning; PATE method
	Homomorphic encryption	Users can calculate and process the encrypted data	Ridge regression; federated learning
	Differential privacy	Can successfully protect user privacy by adding noise	Traditional machine learning; deep learning
Applicable machine learning model	Linear models	Concise form, easy to model	Linear regression; ridge regression
	Tree models	Accurate, stable, and can map non-linear relationships	Classification tree; regression tree
	Neural network models	Learning capabilities, highly robust and fault-tolerant	Pattern recognition, intelligent control
Methods for solving heterogeneity	Asynchronous communication	Solve the problem of communication delay	Device heterogeneity
	Sampling	Avoid simultaneous training with heterogeneous equipment	Pulling Reduction with Local Compensation (PRLC)
	Fault-tolerant Mechanism	Can prevent the whole system from collapsing	Redundancy algorithm
	Heterogeneous Model	Can solve the corresponding heterogeneous device	(LG-FEDAVG) algorithm

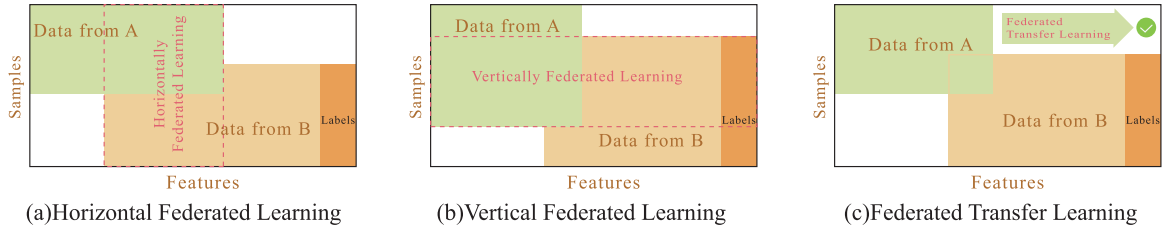

Fig. 2. The different data partition of horizontal federated learning, vertical federated learning, and federated transfer learning.

is divided by client, allowing the client to keep private data from being uploaded to the server. Instead, each client calculates the local gradient, uploads the server, and maintains the global model for gradient updates.

3.1.2. Vertical federated learning

Vertical federated learning is available in the case that the user features of the two datasets overlap little, but the users overlap a lot. Vertical federated learning is to divide the datasets vertically (by user feature dimension), then take out the part of data that users are the same but user features are not exactly 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. For example, there are two different institutions, one is a bank in one place, and the other is an e-commerce company in the same place. Their user groups are likely to include most of the residents of the area, so there is a greater intersection of users. However, because banks record users' income and expenditure behavior and credit rating, while e-commerce keeps users' browsing and purchasing history, their user features have almost no intersection. Vertical federated learning is to aggregate these different features in an encrypted state to enhance the ability of the model. At present, many machine learning models such as logical regression model, tree structure model and neural network model have been gradually proved to be based on this federated system.

There are many machine learning algorithms for vertical partition of data, such as classification [19], statistical analysis [20], gradient descent [21], safe linear regression [22,23], data mining [24]. In some vertical federated learning, there are also data based on vertical partition. In [25], a vertical federated learning system called SecureBoost is proposed, in which all parties combine user features to train together to improve the accuracy of decision-making, which is a lossless training scheme. In [26], Hardy et al. proposed a logical regression model with privacy protection based on vertical federated learning. The model uses pipelined entity analysis and distributed logic regression of Paillier additive homomorphic encryption [27], which can effectively protect privacy and also improve the accuracy of the classifier.

3.1.3. Federated transfer learning

In the case that the users and user features of the two datasets both rarely overlap, we do not segment the data, but can use transfer learning to overcome the lack of data or tags. This method is called federated transfer learning [9]. For example, there are two different institutions, one is an e-commerce in China, the other is a social application in the United States. Due to geographical restrictions, the user groups of the two institutions have little overlap. At the same time, due to the different types of institutions, the data features of the two datasets are only a small part of the overlap. In this case, in order to carry out effective federated learning, transfer learning must be introduced to solve the problems of small unilateral data size and small label samples, so as to improve the effectiveness of the model. The most suitable situation for transfer learning [28] is when you try to optimize the performance of a task but there is not enough related data for training. For example, it is difficult for hospital

radiology departments to collect many X-ray scans to build a good radiology diagnosis system. At this time, transfer learning will help us with other related but different tasks, such as image recognition task, to learn a radiology diagnosis system. Through federated transfer learning, we can not only ensure the privacy of data, but also transfer the model of auxiliary tasks to director learning, and solve the problem of small amount of data.

3.2. Privacy mechanisms

The most important feature of federated learning is that cooperative clients can keep their own data locally, and need to share model information to train the target model, but the model information will also disclose some private information [29]. The common means to protect federal privacy are model aggregation [15], homomorphic encryption [13] and differential privacy [14].

3.2.1. Model aggregation

Model aggregation is one of the most common privacy mechanisms of federation learning, which trains the global model by summarizing the model parameters from all parties, so as to avoid transmitting the original data in the training process. Shashi et al. [30] enabled multiple devices to participate in federal training based on the established incentive mechanism. In order to obtain efficient results, the communication efficiency optimization during the parameter exchange process must be considered in real time. Compared with establishing incentive mechanisms, Yu et al. [31] demonstrated local adaptation based on fine-tuning, multi-task learning and knowledge extraction to help improve the privacy of individual participants and the accuracy of robust federated models. Therefore, participants can obtain the benefits of federated learning and achieve better results than local models without compromising the privacy or integrity of the model.

McMahan et al. [18] proposed a deep network federation learning framework based on iterative model averaging, which trains the global model by summarizing the local model in each round of updates. The PATE [32] method is based on the aggregation of knowledge and is transferred from the Teacher model trained by separated data to the Student model whose attributes can be exposed. The PATE combines multiple models trained with disfederated data sets in a black box way, which provides an accurate guarantee for the privacy of the training data. Yurochkin et al. [33] developed a Bayesian nonparametric framework for federated learning of neural networks, which establishes a global model by matching neurons in the local model. The combination of federated learning and multitasking [17] allows multiple users to train models of different tasks locally, which is also a typical method of model aggregation. In [16,34], federated learning and block chain are combined to exchange and update the model data of each equipment based on the block chain. Finally, under the guarantee of the block chain protocol, the model parameters are safely aggregated.

3.2.2. Homomorphic encryption

General encryption schemes focus on data storage security. It is impossible for users without a key to get any information about the original data from the encryption results, and cannot perform any calculation operations on the encrypted data, otherwise it will lead to unsuccessful decryption. However, homomorphic encryption can solve the computing problem of general encrypted data, because it is concerned with the security of data processing. The most important feature of homomorphic encryption is that users can calculate and process the encrypted data, but no original data will be disclosed in the process. At the same time, the user with the key decrypts the processed data, which is exactly the expected result. It is common in the Ridge regression system [15,35], which combines homomorphic encryption to meet the privacy requirements. The performance of communication and computing overhead has been improved.

Homomorphic encryption is the icing on the cake for federated learning. When using federated learning, the gradient exchange between the users and the server may leak the private information of users. Homomorphic encryption can solve this problem very well, it can deal with the encrypted model without affecting the training results of the model. In [13], the additive homomorphism is used to ensure the sharing security of model parameters, so that the privacy of each client will not be leaked out by the central server. Hardy et al. [26] proposed a federated logical regression model, which uses additive homomorphism scheme to effectively resist honest and curious attackers. Liu et al. [36] proposed a federated learning framework for transfer learning, in which the privacy mechanism also uses additive homomorphic encryption to encrypt model parameters to protect data privacy. Cheng et al. [25] used entity alignment technology to obtain common data to build a decision tree model called SecureBoost, and homomorphic encryption is used to protect model parameters.

3.2.3. Differential privacy

Differential Privacy [37] is a new privacy definition proposed by Dwork in 2006 to solve the problem of privacy disclosure in statistical databases. Under this definition, the calculation results of the database are insensitive to the changes of a specific record, and a single record in the dataset or not in the dataset has little impact on the calculation results. Therefore, the risk of privacy disclosure caused by the addition of a record to the dataset is controlled in a very small and acceptable range, and the attacker cannot obtain accurate individual information by observing the calculation results. In the training process of traditional machine learning [38] and deep learning [39], it is popular to add noise to the output to apply differential privacy in the process of gradient iteration, so as to achieve the goal of protecting user privacy. In practice, Laplace mechanism and exponential mechanism are usually used to achieve differential privacy protection. A lot of research work is carried out around the two aspects of privacy protection and validity. Adding more noise will inevitably affect the validity. To achieve the balance between privacy and validity is the most popular research direction at present. For example, differential privacy can be combined with model compression technology [40] to maximize privacy benefits while improving performance.

Differential privacy is divided into global differential privacy and localized differential privacy. Both kinds of differential privacy can guarantee ε—the differential requirements of a single user, but the application scenarios are slightly different. Geyer et al. [41] proposed a federated optimization algorithm with differential privacy, which is applied to clients to ensure their global differential privacy. The trained model itself contains a large number of super-parameters to ensure communication and accuracy. Due to the addition of noise, it will cause a great loss of validity. In the follow-up work, in order to avoid blindly adding unnecessary noise, Thakkar et al. [42] designed a paradigm pruning scheme based on adaptive gradient to reduce the penetration of noise to the gradient. Under the condition that the validity of global privacy protection is limited, Bhowmick et al. [29] designed a minimax optimal privatization mechanism, which simulates the local privacy protection strategy of users, limits the power of potential attackers, and achieves better model performance than strict local privacy. Li et al. [43] proposed a new gradient-based differential private parameter transfer algorithm, which is applied to the modeling task of non-convex federal language, and achieves the performance close to that of non-private model. Qi

et al. [44] designed a recommendation model based on federated learning for news recommendation. The model with local checking privacy is trained on multi-user equipment, and users are randomly selected to upload the local model to the server and aggregate into a new global model.

3.3. Applicable machine learning models

Federated learning is gradually infiltrating into the popular machine learning model, which aims to ensure the privacy and efficiency of the model. We mainly consider three types of models supported by federation learning: linear model, decision tree and neural network.

3.3.1. Linear models

Linear models are mainly divided into three categories: linear regression, ridge regression and lasso regression. Du et al. [19] proposed the training of linear model in the federated environment, which solves the security problem of entity parsing, and finally achieves the same accuracy as the non-private solution. Nikolaenko et al. [35] designed a ridge regression system with homomorphic encryption and Yao's protocol [45], which obtained the best performance. Compared with other models, the linear model is simple and easy to implement, and it is an effective model for implementing federated learning.

3.3.2. Tree models

Federated learning can be used to train single or multiple decision trees, such as gradient boosting decision trees and random forests. Gradient Boosting Decision Tree (GBDT) algorithm is a widely mentioned algorithm in recent years, which is mainly due to its good performance in many classification and regression tasks. Zhao et al. [46] implemented the GBDT privacy protection system for the first time in regression and binary classification tasks. The system securely aggregates the regression trees trained by different data owners into a collection to prevent the disclosure of user data privacy. Cheng et al. [25] proposed a framework called SecureBoost, which trains the gradient lifting decision tree model for horizontal and vertical partition data, and enables users to establish a federated learning system.

3.3.3. Neural network models

Neural network model is a popular direction of machine learning at present, which aims to train neural network to carry out complex tasks. In the federal environment, the research on deep neural network is becoming more and more popular. Drones can play an important role in a variety of services, such as trajectory planning, target recognition and target location. In order to provide more efficient services, the UAV (Unmanned Aerial Vehicle) group usually trains the model through deep learning, but due to the lack of continuous connection between the UAV group and the ground base station, the centralized training method cannot play the real-time performance of the UAV. Zeng et al. [47] was the first to implement distributed federated learning algorithm on UAV group, joint power allocation and scheduling, and optimize the convergence speed of federated learning. The main step of this algorithm is that the leading UAV summarizes the local flight model trained by the rest of the UAV to generate the global flight model, which is forwarded to the rest of the UAV through the intra-group network. Bonawitz et al. [48] built a scalable federated learning system for mobile devices on the basis of TensorFlow, which can train a large number of distributed data models. Yang et al. [12] set up a federated deep learning framework based on data partition to achieve priority application in enterprise data. In addition to enterprise data applications, the traffic flow information in government affairs big data often contains a lot of user privacy. Liu et al. [49] combine GRU (Gated Recurrent Unit) neural network for traffic flow prediction with federated learning, and propose a clustering FedGRU algorithm, which integrates the optimal global model, and captures the spatio-temporal correlation of traffic flow data more accurately. Experiments on real data sets show that its performance is much better than that of non-federated learning methods.

At present, federated learning has been widely used in machine learning models, but with the rapid development of machine learning, it is still a challenge to propose practical and efficient federated learning tasks.

3.4. Communication architecture

The federated learning application scenario is faced with some problems, such as uneven distribution of user data, equipment computing power and so on. With the development of smart home and other devices, there is also a multitude of Non-IID data needs to be processed without leaking sensitive information. According to the actual complex situation, choosing an appropriate training method is helpful to the implementation of the model.

In the design of distributed training, all remote devices can communicate with the central server and participate in the update of the global model. In the federal setting, the flexibility of local updates and customer participation affect the training validity of the overall model. In [50], a model called FedProx is proposed, which combines edge device data for distributed training, and uses a federal average [18] model optimization method to ensure the robustness and stability of the target task. Federated Averaging (FedAvg) [18] is the most common model optimization method in federated learning. This method averages the randomly declining gradient data uploaded locally, and then updates it and distributes it back locally. In multitask learning [17], the FedAvg model optimization method was proved to have good performance. In order to solve the critical problem that the communication cost of model updating is too high in federated learning, Konecny et al. [11] compressed the model data by the methods of quantization, random rotation and secondary sampling to reduce the communication pressure between the central server and all users. Caldas et al. [51] adopted lossy compression and Federated Dropout to reduce server-to-device communication. Sattler et al. [52] proposed a Sparse Ternary Compression protocol, which converges faster than the federated average algorithm for federated training of Non-IID Data. In order to protect their data privacy and solve the imbalance of Non-IID data, Yang et al. [53] proposed a new federated average algorithm, which aggregates the global model by calculating the model weighted average of different devices.

3.5. Methods for solving heterogeneity

In the application scenario of federated learning, the difference of equipment will affect the inefficiency of the whole training process. In order to solve the problem of system heterogeneity, there are four kinds of diversion: asynchronous communication, device sampling, fault-tolerant mechanism and model heterogeneity.

3.5.1. Asynchronous communication

In the traditional data center setup, there are two common schemes based on parallel iterative optimization algorithm: synchronous communication and asynchronous communication. However, in the face of the diversity of devices, the synchronous scheme is easy to be disturbed, so in the federated learning multi-device environment, the asynchronous communication scheme can better solve the problem of scattered devices. Duchi et al. [54]

made use of the sparsity of data to study parallel and asynchronous algorithms, which can better solve the problem of heterogeneity of training equipment. In the memory sharing system [55], the asynchronous scheme solves the problem of device heterogeneity very well. Although asynchronous update has achieved good benefits in distributed systems [56–60], the problem of delay in device communication aggravates the disadvantage of device heterogeneity. In the process of federation learning, because of the necessity of real-time communication, it is the first choice to solve the heterogeneity of the system according to the scheme of asynchronous communication.

3.5.2. Sampling

In federated learning, not every equipment needs to participate in every iterative training process. In some federated learning scenarios, the equipment is selected to participate in the training, while in another part of the scene, the equipment takes the initiative to participate in the training. In the work of [17,18,48], the equipment is passively involved in the process of federated learning. Nishio et al. [61] proposed a new protocol FedCS, to solve the problem of resource-constrained client selection, which adds more clients to the training process and improves the performance of the model. Kang et al. [62] designed an incentive mechanism based on contract theory to encourage local devices with high-quality data to actively participate in the effective federated learning process and improve learning accuracy. Qi et al. [44] designed a news recommendation model based on federated learning, which also randomly selected local gradients of users to upload to the server to train the global model. Wang et al. [63] proposed a novel approach named Pulling Reduction with Local Compensation (PRLC), which is based on federated learning to achieve end-to-end communication. The main idea of PRLC is that in each iteration, only part of the devices participate in the model update, and the devices that do not participate are updated locally through the PRLC method to reduce the gap with the global model. Finally, it is proved that the convergence rate of the PRLC method is the same as that of the uncompressed method in the case of strong convexity and non-convexity, and has better scalability.

3.5.3. Fault-tolerant mechanism

In the unstable network environment, the fault-tolerant mechanism can prevent the system from collapsing, especially in the distributed environment. When multiple devices work together, once there is a device failure, it will affect other devices. Federated learning is a hot research direction at present, with the assistance of multiple devices to protect the privacy of multiple users. Similarly, we also need to consider the device admissibility in the federated learning environment. Wang et al. [64] focused on the federated learning method and proposed a control algorithm to determine the best tradeoff between local update and global parameter aggregation to adapt to the limitation of equipment resources. Yu et al. [50] improved the linear acceleration characteristics of the distributed random gradient descent algorithm by reducing communication. There are also some works [11,17] that ignore the participation of equipment directly, which does not affect the efficiency of federated learning in multi-task learning. Another option for tolerating equipment failures [65] is to introduce algorithm redundancy by coding calculation. Incorrect data on mobile devices may lead to fraud in federal learning. Kang et al. [66] proposed a federal learning scheme based on reliable staff selection by introducing reputation as a metric and block chain as a reputation management scheme, which can effectively prevent malicious attacks and tampering.

3.5.4. Model heterogeneity

Data is the cornerstone of the training model. When collecting unevenly distributed data from multi-party devices to train the federated model, it will seriously affect the final efficiency of the model. Reasonable processing of data from different devices has a vital impact on federated learning. In order to solve the problem of statistical data heterogeneity, the federated learning network is mainly divided into three modeling methods: (a) single device has its own model; (b) trains a global model suitable for all devices; (c) trains relevant learning models for tasks.

Yu et al. [67] proposed a general framework for training using only positive labels, that is Federated Averaging with Spreadout (FedAwS), in which the server adds a geometric regularizer after each iteration to promote classes to be spread out in the embedding space. However, in traditional training, users also need to use negative tags, which greatly improves the training efficiency and ensures the accuracy of classification tasks. Zhao et al. [52] built a global model by training a small part of the data between edge devices to improve the training accuracy of Non-IID data. Khodak et al. [68] designed and implemented an adaptive learning method in the setting of statistical learning, which improved the performance of small sample learning and federated learning. Eichner et al. [69] considered fast data adaptive training between the global model and specific equipment to solve the problem of data heterogeneity during federated training. Corinzia et al. [70] proposed a federated learning algorithm called VIRTUAL, which regards the federated network of the central server and the client as a Bayesian network and uses approximate variational reasoning to train on the network, showing the most advanced performance on federated learning real datasets. Different from previous methods, the center of gravity is biased towards local or global models. Liang et al. [71] proposed a Local Global Federated Averaging (LG-FEDAVG) algorithm that combines local representation learning with global model federated training. Theoretical analysis shows that the combination of local and global models reduces data variance, reduces device variance, and improves the flexibility of the model when dealing with heterogeneous data. Experiments show that LG-FEDAVG can reduce the communication cost, deal with heterogeneous data and effectively learn the fair representation of fuzzy protected attributes.

4. Applications

4.1. Service recommendation

4.1.1. Google keyboard

Google began a project in 2016 to establish federated learning among Android mobile users [8] to improve the quality of keyboard input prediction, while simultaneously ensure the security and privacy of users. The development of the language model will also promote the development of the recommendation system [72]. Combined with federated learning, it can be extended to other recommendation applications. When users make a request, the subsequent suggestion can be quickly provided by the model.

4.1.2. Intelligent medical diagnosis system

Due to the protection of patient privacy, it becomes very difficult to collect medical data scattered in various hospitals. As a result, medical data becomes a scarce resource. The development of artificial intelligence has brought revolutionary changes to the allocation of medical resources and disease diagnosis. However, there are security challenges in the collection and processing of data, such as the disclosure of patients' private data [73]. Cohen et al. [74] analyzed the existing legal and moral challenges according to the privacy needs of patients, and discussed

how to make better use of patient data without divulging privacy in the future. Too small amount of data and insufficient labels are two problems faced by medical data, and the existing federated transfer learning can solve these problems. Lee et al. [75] used the interconnected medical system to collate health outcome data and longitudinal real data, and design and implement an integrated multi-federated learning network based on APOLLO network to transform real-world data into medical diagnostic evidence to assist doctors in forward-looking diagnosis of patients.

4.2. Wireless communication

For wireless communication, the early methods based on traditional models are no longer suitable for the existing increasingly complex wireless networks, and the popularity of deep networks has also brought a new direction to the establishment of wireless network models [76].

Niknam et al. [77] applied the important functions of federated learning in the field of wireless communication, such as edge computing and 5G network, and made a detailed analysis. Then carried out simulations on standard data sets to prove the availability and security of federated learning in the field of wireless communication. Mohammad et al. [78] studied the application of federated learning in wireless network and edge computing, and established a federated model with the help of remote parameter server through its own data set of each device. Tran et al. [79] designed and implemented a federated learning model based on light wave power, which is a new method applied in the physical layer to manage the network through resource allocation to achieve the highest transmission efficiency. However, the noise problem is always difficult to solve, so Ang et al. [80] proposed a robust federated learning algorithm against wireless communication noise. They simplified the noise problem in the aggregation process and the broadcast process to a parallel optimization problem based on the expected model and the worst-case model. The corresponding optimization problem can be achieved through the SLA (Service-Level Agreement) algorithm and the sampling-based SCA (Service Component Architecture) algorithm. The experimental results show that the algorithm has achieved good results in improving the prediction accuracy and reducing loss.

We can not only obtain a good global model without sharing our own private data, through the training process of federated learning, but also can reduce the communication burden of the equipment. Nguyen et al. [81] applied federated learning to the Wireless Internet of Things system in smart home, which improves the accuracy of attack detection and increases the communication efficiency. Savazz et al. [82] proposed a serverless learning method for federated learning applications of 5G wireless networks, which shares model parameters through local gradient iterative calculation of each device and a consistency-based method. Abad et al. [83] designed a hierarchical federated learning framework for wireless heterogeneous cellular network (HCN), in which the method of gradient sparse and period average is adopted to improve the communication efficiency of the model.

5. Challenge and future work

5.1. Challenge

Federated learning is an emerging field, although federated learning has played a role in some area, it still faces several challenges in performance optimization, next is the three main challenges.

5.1.1. Privacy protection

In federated learning, privacy protection is a major concern. Federated learning protects the private data on each device by exchanging model gradients with server, instead of raw data. However, the model communication during the entire training process can also leak sensitive information to a third party, for example, the reverse deduction of models. Although there are some methods to improve the privacy of data recently, these methods all increase the complexity of calculation and increase the computational burden of the federated network. In order to further effectively protect the security of private data, we need to find new methods to prevent private data from being leaked during model transmission.

5.1.2. Communication cost

In federated learning, communication is a key bottleneck. In fact, a federated network may consist of a multitude of devices, such as millions of remote mobile devices. A training of a federated learning model may involve a large amount of communication. In addition, the communication speed in the network cannot be guaranteed, so the communication cost of federated learning is very worth considering. Therefore, in order to make federated learning practical, it is necessary to develop methods with high communication efficiency.

5.1.3. Systems heterogeneity

Due to different hardware and network connections, the computing and communication capabilities of each device in the federated network may be different. Devices which are simultaneously active in a network usually account for only a small portion. For example, a millions of devices network sometimes only has hundreds of active devices simultaneously. Each device may also be unreliable, thus the heterogeneity of these systems greatly exacerbates the challenges of fault tolerance. Therefore, federated learning methods must tolerate heterogeneous hardware and be robust to offline devices in the network.

5.1.4. Unreliable model upload

In federated learning, mobile nodes may mislead the server [66] to aggregate the global model intentionally or unintentionally. For deliberate behavior, the attacker may send malicious model parameters to affect the aggregation of the global model, thereby causing errors in model training. On the other hand, the unstable mobile network environment may cause some unexpected behaviors of mobile devices, such as uploading some low-quality models, which will adversely affect federated learning. Therefore, for federated learning, it is crucial to resist this unreliable local model upload.

5.2. Future work

In order to solve the challenges indicated above, there are some possible future work directions worth studying:

5.2.1. Privacy restrictions

In fact, due to the heterogeneity of various devices in the network, their privacy restrictions have their own different characteristics, so it is necessary to define the privacy restrictions of batch devices at a more detailed level to ensure the privacy guarantee of specific samples, which can provide strong privacy. The development of privacy protection methods based on privacy restrictions of specific devices is an interesting and continuing direction for future work.

5.2.2. Trade-off between communication cost and computational pressure

We can mainly consider two aspects to improve the efficiency of communication: iteratively send small messages, or reduce the total number of communication rounds. For example, we can use model compression technology to reduce the data size communicated in federated learning. In terms of reducing communication rounds, the models that need to be communicated can be screened according to their importance. We can also combine these two methods, which can greatly reduce the cost of communication between mobile devices and servers, but it also increases some computational pressure. Finding the trade-off between communication cost and computational pressure is the main direction of future work.

5.2.3. Multi-center federated learning

The challenge of heterogeneity hinders federated learning. Some recent studies [83–86] have shown that if the heterogeneity of the devices in the system can be obtained in advance, all mobile devices can be grouped according to the heterogeneity, and a local central server can be assigned to each group. We can first aggregate a group of similarly heterogeneous device models, and then send them to the server to aggregate into a global model. Studying multi-center federated learning to solve heterogeneous challenges is a promising direction in future work.

5.2.4. Reliable client selection

In federated learning, mobile devices may upload unreliable data, which could cause that the server fail to aggregate the global model. Therefore, it is crucial to find trustworthy and reliable clients in federated learning tasks. [66] introduced the concept of reputation as a metric to measure the reliability of the client. Therefore, we can select a highly reliable client during each round of model update to ensure the reliability of federated learning. The improvement of reliable federated learning based on this method is a far-reaching research direction in the future.

6. Conclusion

With the development of big data and artificial intelligence, the public's requirements for privacy are becoming more and more stringent. Consequently, federated learning was brought up, which is a new solution for cross-platform privacy protection. As a model that can be used in practical, federated learning has been accepted by more and more researchers and enterprises today when it emphasizes data privacy and data security. On the one hand, if users are unable to train satisfactory models because of insufficient data, federated learning can aggregate multi-party user models and update the integrated model without exposing the original data. On the other hand, when users have not enough data labels to learn, federated learning can not only provide them with a secure model sharing mechanism, but also migrate models to specific tasks to solve the problem of insufficient data labels. This paper introduces the basic definition, related technologies and specific classification of federated learning, then discusses the practical application scenarios of federated learning, and sort out the current challenges and future research directions of federated learning. It is believed that in the near future, federated learning can provide secure and shared security services for more applications and promote the stable development of artificial intelligence.

CRediT authorship contribution statement

Chen Zhang: Formal analysis, Conceptualization, Funding acquisition, Resources. **Yu Xie:** Formal analysis, Investigation, Supervision, Writing - original draft. **Hang Bai:** Formal analysis, Investigation, Supervision, Writing - original draft. **Bin Yu:** Funding acquisition, Writing - review & editing. **Weihong Li:** Formal analysis, Supervision, Writing - original draft. **Yuan Gao:** Formal analysis, Supervision, Writing - original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors wish to thank the editors and anonymous reviewers for their valuable comments and helpful suggestions which greatly improved the paper's quality. This work was supported by the Key Research and Development Program of Shaanxi Province, China (Grant no. 2019ZDLGY17-01, 2019GY-042), the Fundamental Research Funds for the Central Universities, China, and the Innovation Fund of Xidian University, China.

References

[1] C. Zhang, X. Hu, Y. Xie, M. Gong, B. Yu, A privacy-preserving multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition, Front. Neurorob. 13 (2020) 112.
[2] M. Gong, J. Feng, Y. Xie, Privacy-enhanced multi-party deep learning, Neural Netw. 121 (2020) 484–496.
[3] Y. Xie, H. Wang, B. Yu, C. Zhang, Secure collaborative few-shot learning, Knowl. Based Syst. (2020) 106157.
[4] J.P. Albrecht, How the GDPR will change the world, Eur. Data Prot. Law Rev. 2 (2016) 287.
[5] M. Parasol, The impact of China's 2016 cyber security law on foreign technology firms, and on China's big data and smart city dreams, Comput. Law Secur. Rev. 34 (1) (2018) 67–98.
[6] W. Gray, H.R. Zheng, General principles of civil law of the people's Republic of China, Am. J. Comp. Law 34 (4) (1986) 715–743.
[7] M. Gong, Y. Xie, K. Pan, K. Feng, A.K. Qin, A survey on differentially private machine learning, IEEE Comput. Intell. Mag. 15 (2) (2020) 49–64.
[8] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H.B. McMahan, S. Patel, D. Ramage, A. Segal, K. Seth, Practical secure aggregation for privacy-preserving machine learning, in: Conference on Computer and Communications Security (CCS), 2017, pp. 1175–1191.
[9] Y. Liu, Y. Kang, C. Xing, T. Chen, Q. Yang, A secure federated transfer learning framework, IEEE Intell. Syst. 35 (4) (2020) 70–82.
[10] H.B. McMahan, E. Moore, D. Ramage, B.A. y Arcas, Federated learning of deep networks using model averaging, 2016, CoRR abs/1602.05629.
[11] J. Konečný, H.B. McMahan, D. Ramage, P. Richtárik, Federated optimization: Distributed machine learning for on-device intelligence, 2016, arXiv preprint arXiv:1610.02527.
[12] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: Concept and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) 1–19.
[13] Y. Aono, T. Hayashi, L. Wang, S. Moriai, et al., Privacy-preserving deep learning via additively homomorphic encryption, IEEE Trans. Inf. Forensics Secur. 13 (5) (2017) 1333–1345.
[14] B. McMahan, D. Ramage, K. Talwar, L. Zhang, Learning differentially private recurrent language models, in: International Conference on Learning Representations (ICLR), 2018.
[15] Y.-R. Chen, A. Rezapour, W.-G. Tzeng, Privacy-preserving ridge regression on distributed data, Inform. Sci. 451 (2018) 34–49.
[16] H. Kim, J. Park, M. Bennis, S.-L. Kim, On-device federated learning via blockchain and its latency analysis, 2018, arXiv preprint arXiv:1808.03949.
[17] V. Smith, C.-K. Chiang, M. Sanjabi, A.S. Talwalkar, Federated multi-task learning, in: Advances in Neural Information Processing Systems, 2017, pp. 4424–4434.
[18] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[19] W. Du, Y.S. Han, S. Chen, Privacy-preserving multivariate statistical analysis: Linear regression and classification, in: Proceedings of the Fourth SIAM International Conference on Data Mining, SIAM, 2004, pp. 222–233.
[20] W. Du, M.J. Atallah, Privacy-preserving cooperative statistical analysis, in: Annual Computer Security Applications Conference (ACSAC, IEEE, 2001, pp. 102–110.
[21] L. Wan, W.K. Ng, S. Han, V.C. Lee, Privacy-preservation for gradient descent methods, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2007, pp. 775–783.
[22] A. Gascón, P. Schoppmann, B. Balle, M. Raykova, J. Doerner, S. Zahur, D. Evans, Secure linear regression on vertically partitioned datasets, IACR Cryptol. ePrint Arch. 2016 (2016) 892.
[23] A.F. Karr, X. Lin, A.P. Sanil, J.P. Reiter, Privacy-preserving analysis of vertically partitioned data using secure matrix products, J. Off. Stat. 25 (1) (2009) 125.
[24] J. Vaidya, C. Clifton, Privacy preserving association rule mining in vertically partitioned data, in: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002, pp. 639–644.
[25] K. Cheng, T. Fan, Y. Jin, Y. Liu, T. Chen, Q. Yang, Secureboost: A lossless federated learning framework, 2019, arXiv preprint arXiv:1901.08755.
[26] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, B. Thorne, Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption, 2017, arXiv preprint arXiv:1711.10677.
[27] B. Schoenmakers, P. Tuyls, Efficient binary conversion for paillier encrypted values, in: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2006, pp. 522–537.
[28] L. Zhang, Transfer adaptation learning: A decade survey, 2019, arXiv preprint arXiv:1903.04687.
[29] A. Bhowmick, J. Duchi, J. Freudiger, G. Kapoor, R. Rogers, Protection against reconstruction and its applications in private federated learning, 2018, arXiv preprint arXiv:1812.00984.
[30] S.R. Pandey, N.H. Tran, M. Bennis, Y.K. Tun, A. Manzoor, C.S. Hong, A crowdsourcing framework for on-device federated learning, IEEE Trans. Wirel. Commun. 19 (5) (2020) 3241–3256.
[31] T. Yu, E. Bagdasaryan, V. Shmatikov, Salvaging federated learning by local adaptation, 2020, arXiv preprint arXiv:2002.04758.
[32] N. Papernot, M. Abadi, lfar Erlingsson, I. Goodfellow, K. Talwar, Semi-supervised knowledge transfer for deep learning from private training data, in: Proceedings of the International Conference on Learning Representations, 2017.
[33] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, T.N. Hoang, Y. Khazaeni, Bayesian nonparametric federated learning of neural networks, in: International Conference on Machine Learning (ICML), Vol. 97, PMLR, 2019, pp. 7252–7261.
[34] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, H. Wang, Blockchain challenges and opportunities: A survey, Int. J. Web Grid Serv. 14 (4) (2018) 352–375.
[35] V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, N. Taft, Privacy-preserving ridge regression on hundreds of millions of records, in: IEEE Symposium on Security and Privacy, IEEE, 2013, pp. 334–348.
[36] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, F. Beaufays, Applied federated learning: Improving google keyboard query suggestions, 2018, arXiv preprint arXiv:1812.02903.
[37] C. Dwork, Differential privacy: A survey of results, in: Theory and Applications of Models of Computation (TAMC), Springer, 2008, pp. 1–19.
[38] R. Bassily, A. Smith, A. Thakurta, Private empirical risk minimization: Efficient algorithms and tight error bounds, in: IEEE 55th Annual Symposium on Foundations of Computer Science, IEEE, 2014, pp. 464–473.
[39] M. Abadi, A. Chu, I. Goodfellow, H.B. McMahan, I. Mironov, K. Talwar, L. Zhang, Deep learning with differential privacy, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016, pp. 308–318.
[40] N. Agarwal, A.T. Suresh, F.X.X. Yu, S. Kumar, B. McMahan, Cpsgd: Communication-efficient and differentially-private distributed SGD, in: Annual Conference on Neural Information Processing Systems, 2018, pp. 7564–7575.
[41] R.C. Geyer, T. Klein, M. Nabi, Differentially private federated learning: A client level perspective, 2017, arXiv preprint arXiv:1712.07557.
[42] O. Thakkar, G. Andrew, H.B. McMahan, Differentially private learning with adaptive clipping, 2019, arXiv preprint arXiv:1905.03871.
[43] Y. Jiang, J. Konečnỳ, K. Rush, S. Kannan, Improving federated learning personalization via model agnostic meta learning, 2019, arXiv preprint arXiv:1909.12488.
[44] T. Qi, F. Wu, C. Wu, Y. Huang, X. Xie, Privacy-preserving news recommendation model learning, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, 2020, pp. 1423–1432.
[45] Y. Lindell, B. Pinkas, A proof of security of yao's protocol for two-party computation, J. Cryptol. 22 (2) (2009) 161–188.
[46] L. Zhao, L. Ni, S. Hu, Y. Chen, P. Zhou, F. Xiao, L. Wu, Inprivate digging: Enabling tree-based distributed data mining with differential privacy, in: IEEE INFOCOM 2018-IEEE Conference on Computer Communications, IEEE, 2018, pp. 2087–2095.
[47] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, M. Bennis, Federated learning in the sky: Joint power allocation and scheduling with uav swarms, in: International Conference on Communications (ICC), IEEE, 2020, pp. 1–6.
[48] K.A. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C.M. Kiddon, J. Konen, S. Mazzocchi, B. McMahan, T.V. Overveldt, D. Petrou, D. Ramage, J. Roselander, Towards federated learning at scale: System design, in: Proceedings of Machine Learning and Systems, 2019.
[49] Y. Liu, J. James, J. Kang, D. Niyato, S. Zhang, Privacy-preserving traffic flow prediction: A federated learning approach, IEEE Internet Things J. 19 (5) (2020) 3241–3256.
[50] H. Yu, S. Yang, S. Zhu, Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning, in: AAAI Conference on Artificial Intelligence, 33, 2019, pp. 5693–5700.
[51] S. Caldas, J. Konečny, H.B. McMahan, A. Talwalkar, Expanding the reach of federated learning by reducing client resource requirements, 2018, arXiv preprint arXiv:1812.07210.
[52] F. Sattler, S. Wiedemann, K.-R. Müller, W. Samek, Robust and communication-efficient federated learning from non-iid data, IEEE Trans. Neural Netw. Learn. Syst. 31 (9) (2020) 3400–3413.
[53] K. Yang, T. Jiang, Y. Shi, Z. Ding, Federated learning via over-the-air computation, IEEE Trans. Wirel. Commun. 19 (3) (2020) 2022–2035.
[54] J. Duchi, M.I. Jordan, B. McMahan, Estimation, optimization, and parallelism when data is sparse, in: Advances in Neural Information Processing Systems, 2013, pp. 2832–2840.
[55] W. Dai, A. Kumar, J. Wei, Q. Ho, G. Gibson, E.P. Xing, High-performance distributed ML at scale through parameter server consistency models, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.
[56] J. Wei, W. Dai, A. Qiao, Q. Ho, H. Cui, G.R. Ganger, P.B. Gibbons, G.A. Gibson, E.P. Xing, Managed communication and consistency for fast data-parallel iterative analytics, in: Proceedings of the Sixth ACM Symposium on Cloud Computing, 2015, pp. 381–394.
[57] Q. Ho, J. Cipar, H. Cui, S. Lee, J.K. Kim, P.B. Gibbons, G.A. Gibson, G. Ganger, E.P. Xing, More effective distributed ml via a stale synchronous parallel parameter server, in: Advances in Neural Information Processing Systems, 2013, pp. 1223–1231.
[58] H. Mania, X. Pan, D. Papailiopoulos, B. Recht, K. Ramchandran, M.I. Jordan, Perturbed iterate analysis for asynchronous stochastic optimization, SIAM J. Optim. 27 (4) (2017) 2202–2229.
[59] E.P. Xing, Q. Ho, W. Dai, J.K. Kim, J. Wei, S. Lee, X. Zheng, P. Xie, A. Kumar, Y. Yu, Petuum: A new platform for distributed machine learning on big data, IEEE Trans. Big Data 1 (2) (2015) 49–67.
[60] M. Li, D.G. Andersen, J.W. Park, A.J. Smola, A. Ahmed, V. Josifovski, J. Long, E.J. Shekita, B.-Y. Su, Scaling distributed machine learning with the parameter server, in: 11th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 14), 2014, pp. 583–598.
[61] T. Nishio, R. Yonetani, Client selection for federated learning with heterogeneous resources in mobile edge, in: IEEE International Conference on Communications, IEEE, 2019, pp. 1–7.
[62] 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, in: IEEE VTS Asia Pacific Wireless Communications Symposium, IEEE, 2019, pp. 1–5.
[63] H. Wang, Z. Qu, S. Guo, X. Gao, R. Li, B. Ye, Intermittent pulling with local compensation for communication-efficient federated learning, 2020, arXiv preprint arXiv:2001.08277.
[64] S. Wang, T. Tuor, T. Salonidis, K.K. Leung, C. Makaya, T. He, K. Chan, Adaptive federated learning in resource constrained edge computing systems, IEEE J. Sel. Areas Commun. 37 (6) (2019) 1205–1221.
[65] A. Reisizadeh, S. Prakash, R. Pedarsani, A.S. Avestimehr, Coded computation over heterogeneous clusters, IEEE Trans. Inform. Theory 65 (7) (2019) 4227–4242.
[66] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, M. Guizani, Reliable federated learning for mobile networks, IEEE Wirel. Commun. 27 (2) (2020) 72–80.
[67] F.X. Yu, A.S. Rawat, A.K. Menon, S. Kumar, Federated learning with only positive labels, in: Proceedings of the 37th International Conference on Machine Learning (ICML), PMLR, 2020, pp. 10946–10956.
[68] M. Khodak, M.-F.F. Balcan, A.S. Talwalkar, Adaptive gradient-based meta-learning methods, in: Advances in Neural Information Processing Systems, 2019, pp. 5915–5926.
[69] H. Eichner, T. Koren, H.B. McMahan, N. Srebro, K. Talwar, Semi-cyclic stochastic gradient descent, in: International Conference on Machine Learning (ICML), 2019.
[70] L. Corinzia, J.M. Buhmann, Variational federated multi-task learning, 2019, arXiv preprint arXiv:1906.06268.

[71] P.P. Liang, T. Liu, L. Ziyin, R. Salakhutdinov, L.-P. Morency, Think locally, act globally: Federated learning with local and global representations, 2020, arXiv preprint arXiv:2001.01523.
[72] Y. Mansour, M. Mohri, J. Ro, A.T. Suresh, Three approaches for personalization with applications to federated learning, 2020, arXiv preprint arXiv:2002.10619.
[73] N. Rieke, J. Hancox, W. Li, F. Milletari, H. Roth, S. Albarqouni, S. Bakas, M.N. Galtier, B. Landman, K. Maier-Hein, et al., The future of digital health with federated learning, 2020, arXiv preprint arXiv:2003.08119.
[74] W.N. Price, I.G. Cohen, Privacy in the age of medical big data, Nat. Med. 25 (1) (2019) 37–43.
[75] J.S. Lee, K.M. Darcy, H. Hu, Y. Casablanca, T.P. Conrads, C.L. Dalgard, J.B. Freymann, S.E. Hanlon, G.D. Huang, L. Kvecher, et al., From discovery to practice and survivorship: Building a national real-world data learning healthcare framework for military and veteran cancer patients, Clin. Pharmacol. Ther. 106 (1) (2019) 52–57.
[76] C. Ma, J. Li, M. Ding, H.H. Yang, F. Shu, T.Q. Quek, H.V. Poor, On safeguarding privacy and security in the framework of federated learning, IEEE Netw. 34 (4) (2020) 242–248.
[77] S. Niknam, H.S. Dhillon, J.H. Reed, Federated learning for wireless communications: Motivation, opportunities, and challenges, IEEE Commun. Mag. 58 (6) (2020) 46–51.
[78] M.M. Amiri, D. Gündüz, Federated learning over wireless fading channels, IEEE Trans. Wirel. Commun. 19 (5) (2020) 3546–3557.
[79] H.-V. Tran, G. Kaddoum, H. Elgala, C. Abou-Rjeily, H. Kaushal, Lightwave power transfer for federated learning-based wireless networks, IEEE Commun. Lett. 24 (7) (2020) 1472–1476.
[80] F. Ang, L. Chen, N. Zhao, Y. Chen, W. Wang, F.R. Yu, Robust federated learning with noisy communication, IEEE Trans. Commun. 68 (6) (2020) 3452–3464.
[81] T.D. Nguyen, S. Marchal, M. Miettinen, M.H. Dang, N. Asokan, A.-R. Sadeghi, Dïot: A crowdsourced self-learning approach for detecting compromised iot devices, 2018, arXiv preprint arXiv:1804.07474.
[82] S. Savazzi, M. Nicoli, V. Rampa, Federated learning with cooperating devices: A consensus approach for massive IoT networks, IEEE Internet Things J. 7 (5) (2020) 4641–4654.
[83] M.S.H. Abad, E. Ozfatura, D. Gunduz, O. Ercetin, Hierarchical federated learning across heterogeneous cellular networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2020, pp. 8866–8870.
[84] M. Duan, D. Liu, X. Ji, R. Liu, L. Liang, X. Chen, Y. Tan, Fedgroup: Ternary cosine similarity-based clustered federated learning framework toward high accuracy in heterogeneity data, 2020, arXiv preprint arXiv:2010.06870.
[85] M. Xie, G. Long, T. Shen, T. Zhou, X. Wang, J. Jiang, Multi-center federated learning, 2020, arXiv preprint arXiv:2005.01026.
[86] H. Jiang, M. Liu, B. Yang, Q. Liu, J. Li, X. Guo, Customized federated learning for accelerated edge computing with heterogeneous task targets, Comput. Netw. 183 (2020) 107569.