IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 7, 2021, accepted August 28, 2021, date of publication September 7, 2021, date of current version September 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3111118

Challenges, Applications and Design Aspects of Federated Learning: A Survey

K. M. JAWADUR RAHMAN[1], (Graduate Student Member, IEEE), FAISAL AHMED[1], NAZMA AKHTER[1], MOHAMMAD HASAN[1], RUHUL AMIN[1], KAZI EHSAN AZIZ[1], A. K. M. MUZAHIDUL ISLAM[2], (Senior Member, IEEE), MD. SADDAM HOSSAIN MUKTA[2], (Member, IEEE), AND A. K. M. NAJMUL ISLAM[3]

[1]Graduate School of Science and Engineering, United International University, Dhaka 1212, Bangladesh
[2]Department of Computer Science and Engineering, United International University, Dhaka 1212, Bangladesh
[3]LUT School of Engineering Science, 53850 Lappeenranta, Finland

Corresponding author: A. K. M. Najmul Islam (najmul.islam@lut.fi)

ABSTRACT Federated learning (FL) is a new technology that has been a hot research topic. It enables the training of an algorithm across multiple decentralized edge devices or servers holding local data samples without exchanging them. There are many application domains in which considerable properly labeled and complete data are not available in a centralized location (e.g., doctors' diagnoses from medical image analysis). There are also growing concerns over data and user privacy, as artificial intelligence is becoming ubiquitous in new application domains. As such, much research has recently been conducted in several areas within the nascent field of FL. Various surveys on different subtopics exist in the current literature, focusing on specific challenges, design aspects, and application domains. In this paper, we review existing contemporary works in related areas to understand the challenges and topics emphasized by each type of FL survey. Furthermore, we categorize FL research in terms of challenges, design factors, and applications, conducting a holistic review of each and outlining promising research directions.

INDEX TERMS Data privacy, data security, decentralized data, distributed processing, federated learning, machine learning.

I. INTRODUCTION

Recently, machine learning (ML) and deep learning (DL)-based methods have seen tremendous growth, which is attributable to the availability of considerable data. However, not all application domains have considerable properly labeled and complete data available in a centralized location (e.g., doctors' diagnoses from medical image analysis). Curating such large, high-quality datasets can be time-consuming and tedious and often requires domain experts. Efforts from individual organizations result in data silos, with each containing high-quality but small datasets. In these application domains, very few organizations manage to gather high-quality, complete, fully labeled, and sufficiently large datasets, which are required for these DL applications to be effective. Traditionally, data were gathered in a centralized location to build ML models. However, due to concerns related to data ownership and confidentiality, user privacy, and new laws over data management and data usage, such as the General Data Protection Regulation, private, secure, efficient, and fair distributed model training is required.

Thus, instead of training on centralized data, separate models can be trained locally where the data reside in a distributed manner. Then, the respective local model updates can be communicated to obtain a global model. This is the concept behind federated learning (FL), in which the communication process is carefully designed such that the data of an individual organization or device remain private. FL was first introduced by researchers at Google to update language models [1], [2] in Google's keyboard system for word autocompletion. FL builds a joint model using the data located at different sites, where each party contributes some data to train the model. Note that the data belonging to each party do not leave their premises. The model is then encrypted and shared among the participants so that no participant

The associate editor coordinating the review of this manuscript and approving it for publication was Hiu Yung Wong.

can reverse-engineer others' data. This resulting joint model performance is an approximation of the ideal model trained with centralized data. In practice, this added security and privacy results in certain accuracy loss, but it is often worth for specific application domains. In addition to the privacy and security benefits, collaborative training in FL can yield better models than those trained by individual organizations or devices.

The FL architecture follows the client–server model (Fig. 1) or peer-to-peer model (Fig. 2) at the fundamental level. In the client–server model, a coordinator is responsible for centrally aggregating the model parameters using federated averaging (FedAvg).

First, the coordinator sends an initial model to each participating client. Each client then locally trains individual learning models using their own local datasets and sends the model updates back to the coordinator for aggregation. After aggregation, the combined model updates are sent back to the local participating client. This process is repeated until the model converges or a preset number of iterations is reached. The client–server architecture incurs less communication overhead. The peer-to-peer architecture is even more secure, as the participating clients communicate directly without a third-party coordinator. The trade-off, however, is that the peer-to-peer architecture requires more computation for message encryption and decryption.

Based on data partitioning among participants in feature and sample spaces, there are three fundamental categories of FL: horizontal FL (HFL) (Fig. 3), vertical FL (VFL) (Fig. 4), and federated transfer learning (FTL) (Fig. 5). For HFL, there is alignment in data features across participants, not in data samples. In contrast, for vertical FL, there is alignment in data samples, not in data features. Both HFL and VFL can be ineffective when the data are highly heterogeneous. In such cases, FTL is an effective approach that transfers the learned knowledge from the source domain to the target domain. FTL is inspired by transfer learning, where ML models that are trained on a dataset belonging to one domain are re-used and fine-tuned to solve a problem in a related domain.

The aforementioned architecture and FL categories only form the tip of the iceberg in the field of FL. There are numerous research thrusts, such as novel architectures, data partitioning schemes, and aggregation techniques. Moreover, the current research efforts aim to mitigate the core challenges in FL, such as privacy and security, communication costs,

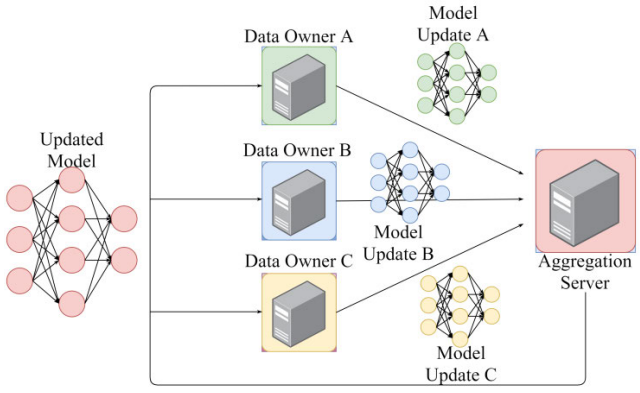

FIGURE 1. Client-server FL architecture.

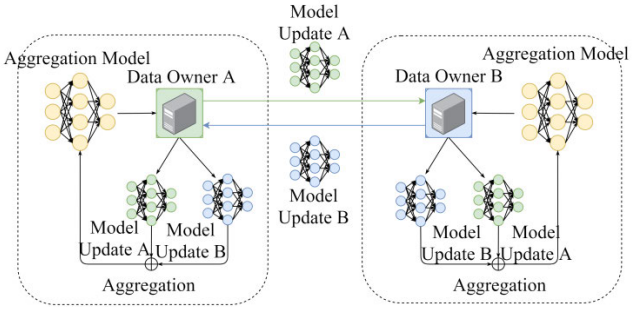

FIGURE 2. Peer-to-peer FL architecture.

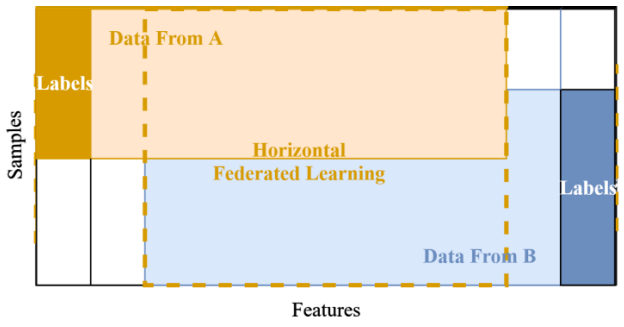

FIGURE 3. Horizontal FL architecture.

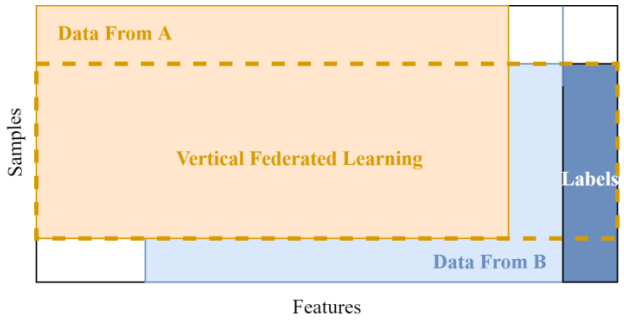

FIGURE 4. Vertical FL architecture.

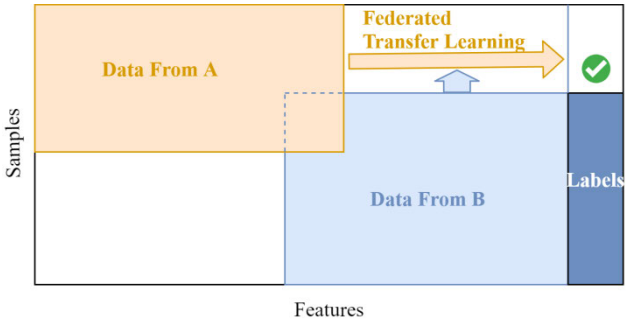

FIGURE 5. Federated Transfer Learning (FTL) architecture.

system and statistical heterogeneity, and personalization techniques. Depending on the application area in which the FL method is applied, unique application and domain-specific challenges and considerations arise.

Much research has already been conducted in the field of FL in recent years. Consequently, numerous survey papers have summarized different focus areas. In this study, we first reviewed existing surveys, which cover various domains and focus areas in FL research.

Several core challenges, such as privacy, security, communication cost, system and statistical heterogeneity, architecture, and aggregation algorithm designs, vary by domain and specific use cases. The motivation for this paper lies in reviewing the current body of literature and summarizing the state-of-the-art approaches that have recently been developed to deal with these challenges. In addition, we identify the gaps in the reviewed FL surveys and fill them by surveying the latest developments in all aforementioned FL areas of research. We conduct a holistic review of the challenges, applications, and design factors and outline promising future research directions.

We study papers in related areas and review, in depth, most of the contemporary survey papers in these areas. We classify the topics in the FL survey papers according to the following categories: communication cost, statistical heterogeneity, systems heterogeneity, and privacy/security as the core challenges; data partitioning, FL architectures, algorithms/aggregation techniques, and personalization techniques as the implementation details; and FL applications in different industries and domains.

This paper makes the following contributions to the literature:

1) It thoroughly investigates and analyzes contemporary FL survey papers.
2) It classifies FL research into broad categories of design aspects, challenges, and application areas.
3) It conducts a holistic survey of the design aspects—data partitioning, FL architectures, aggregation techniques, and personalization techniques; the core challenges—communication cost, systems heterogeneity, statistical heterogeneity, and privacy/security; and different application areas.
4) It discusses open issues and challenges in FL research.

The remainder of this paper is organized as shown in Fig. 6. In Section II, we discuss the related studies. Section III illustrates the taxonomy of the survey papers and discusses them in detail. A discussion and analysis of all topics under each category are covered in Section IV. Section V discusses the open issues and challenges in FL. Section VI concludes the paper.

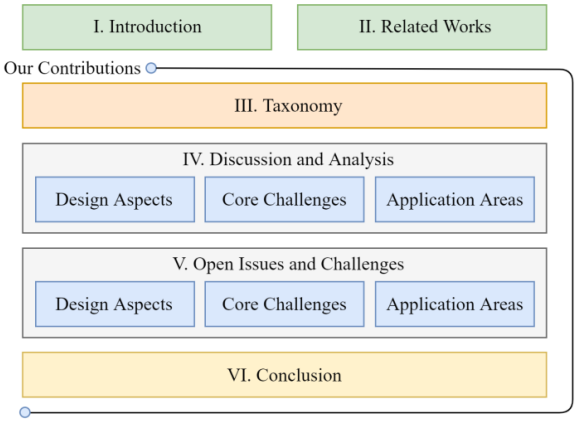

FIGURE 6. Organization of the paper.

II. RELATED WORKS

In this section, we investigate and analyze the most contemporary survey papers. The reviewed papers, along with their summaries and focuses, are listed in Table 1.

Li, Sahu *et al.* [3] discussed how FL differs from standard distributed ML. In addition, they discussed FL's unique characteristics and challenges, along with its current methods and future scope. However, the paper did not focus on any specific domain and discussed approaches that dealt with four core challenges: expensive communication, systems heterogeneity, statistical heterogeneity, and privacy/security. Local updating [1], [4] is an approach for reducing the number of communication rounds. Compression schemes [5], in contrast, reduce the message size in each round of communication. In addition, decentralized training [6], [7] decreases the burden on the central server in terms of communication. For systems heterogeneity challenges, asynchronous communication [8]–[10] reduces stragglers and active sampling selects or influences the participating devices based on system resources and overheads incurred, and fault tolerance [11]–[16] ignores failed devices using algorithmic redundancy. Statistical heterogeneity issues are dealt with by modeling heterogeneous data using methods such as meta-learning and multitask learning, adapting selection between global and device-specific models, and transferring learning for personalization. Some studies have also focused on convergence guarantees for non-independent and identically distributed (non-IID) data [4], [10], [17], [18]. Finally, this survey covers secure multiparty computation (SMC) [19], [20] and differential privacy (DP) [21]–[24] approaches.

The authors in [25] focused on mobile edge networks. The core challenges in their survey included expensive communication, systems heterogeneity, and privacy/security. Under communication cost challenges, the discussed approaches include compression schemes, such as model compression [26], [27], importance-based updating for selective gradients [28] or local model updates [29], and local updating [1], [30]–[32] focused on edge and end computation. The works mitigating systems heterogeneity include active sampling based on computation capabilities [33], data characteristics [34], and resource consumption [35] and allocation [36], [37]; joint radio and computation resource management by using superposition property of multiple-access channel [38]–[40]; asynchronous communication [41] for

TABLE 1. Summary table of survey papers and main focus.

Survey Paper	Summary	Main Focus
Li, Sahu, *et al.* [3]	Discusses the unique characteristics and challenges of FL, provides details of current approaches, outlines directions of future work.	Challenges
Lim *et al.* [25]	Highlights challenges of FL implementation and existing solutions and presents applications of FL for mobile edge network optimization.	Mobile edge networks
Briggs *et al.* [57]	Focusing on IoT, covers works related to FL challenges and privacy preserving methods, identify the strengths and weaknesses of different methods applied to FL, and outlines future directions.	IoT, privacy/security
Li, Wen, *et al.* [58]	Categorizes FL systems according to six different aspects to facilitate and guide the design of FL systems, provides case studies and future research opportunities.	FL systems
Li, Fan, *et al.* [59]	Illustrates the evolution of FL and reviews existing applications of FL in industrial engineering, mobile devices and healthcare.	Applications
Kurupathi, Maass [60]	Highlights existing privacy techniques and proposes applications of FL in industries.	Privacy/security, applications
Yang *et al.* [61]	Introduces a secure FL framework, which includes horizontal FL, vertical FL and federated transfer learning, and proposes building data networks among organizations based on federated mechanisms.	Architecture, applications
Xu *et al.* [62]	Provides a review for FL technologies mainly for biomedicine, and discusses the challenges, issues and potential of FL in healthcare.	Healthcare
Kulkarni *et al.* [63]	Highlights the need for personalization in FL and surveys research on the topic.	Personalization
Lyu *et al.* [64]	Introduces taxonomy of threat models and major attacks on FL, highlighting intuitions, techniques and assumptions adopted by different attacks and discusses future research directions.	Threat models and attack types
Aledhari *et al.* [65]	Provides a thorough summary of relevant protocols, platforms, challenges and real-life uses cases of FL.	Platforms, protocols, applications
Mothukuri *et al.* [66]	Provides a detailed study of security and privacy, and presents current approaches, challenges and future directions in FL.	Privacy/security

model aggregation; adaptive aggregation based on resource constraints [42]; incentive mechanisms such as Stackelberg game [43]–[46]; contract theoretic approach [47], [48]; reputation mechanism [49] to encourage source contribution; and effective worker selection. For privacy/security challenges, information-exploiting attacks are countered by DP [23], [50], selective participants [50], selective parameter sharing [51], secret sharing schemes [52], and GAN model training [53]. Data poisoning attacks are countered by distinguishing honest participants based on their gradient updates [54], model poisoning attacks are countered by comparing updated models [55], and free-riding attacks are countered by verifying local model updates [56].

The primary focus of the authors in [57] was privacy/security for Internet-of-Things (IoT). The approaches discussed in their survey limited the effects of individual client updates [57], [67], distinguishing honest participants [54], DP [23], [50], [51], SMC [19], [54], and homomorphic encryption (HE) [68].

Privacy/security was the focus area in [58]; in particular, approaches such as HE [69], [70], SMC [19], [71], and DP [72], [73] were covered in their survey. Data partitioning

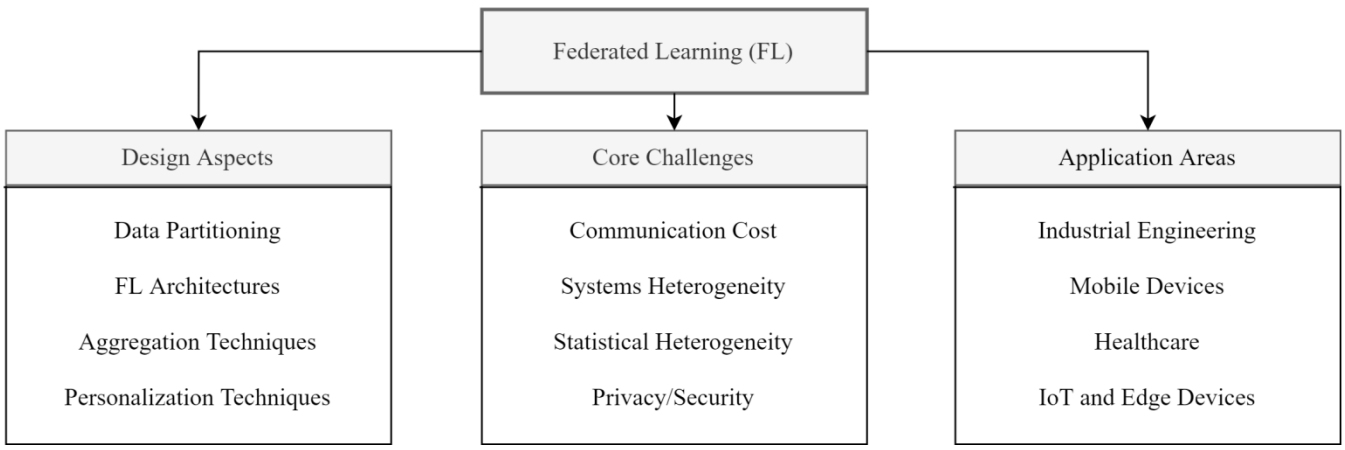

FIGURE 7. Classification of the reviewed survey papers.

schemes, namely HFL [74], VFL [75]–[78], and hybrid FL [79], [80], as well as centralized [11], [81] and decentralized [82] designs for communication architecture and cross-silo [83] versus cross-device [84], [85] FL for the federation scale, are the other challenges and approaches discussed here.

Li *et al.* [59] focused on applications in the domains of mobile devices, industrial engineering, and healthcare. Applications of FL in mobile devices included predicting user input [74], [86], [87], emoji [88], human trajectory [89], and human behavior [90]; reducing network congestion [91]; detecting physical hazards (smart-home IoT) [92]; performing industrial engineering—environmental monitoring [93]; performing visual inspection [94]; detecting malicious attacks (unmanned aerial vehicles); preventing energy congestion (charging stations); detecting credit card frauds; and predict future hospitalizations, hospital stay time, mortality over drug utilization data, and similar patient matching.

Another privacy/security-focused survey [60] elaborated on current approaches, such as SMC, DP, HE, and private information retrieval. The authors in [61] covered privacy/security approaches, namely SMC [95], DP [23], and HE [96]. Moreover, the paper discussed data partitioning approaches—HFL [97], VFL, and FTL.

The challenges and approaches discussed in [62] were centered around the healthcare domain. Consensus [37], [98] and pluralistic [99] solutions were mentioned to tackle statistical heterogeneity; client selection [33], compression schemes, update reduction, and peer-to-peer learning for expensive communication challenges; and SMC and DP for privacy/security challenges. The focus of [63] was on personalization techniques, which included adding user context [100], transfer learning [101], multitask learning [102], meta-learning [103], knowledge distillation, base + personalization layers, and combination of global and local models. Article [64] is also based on privacy/security challenges. In particular, it includes studies on threat models, various poisoning attacks, and inference attacks.

Aledhari *et al.* [65] focused on architecture options for FL-based models—HFL [61], VFL [61], multi-participant multi-class VFL (MMVFL) [104], FTL [79], FEDF: a distributed DL framework for parallel training while preserving privacy [105], PerFit: a cloud-edge framework for personalized FL [106], FedHealth: an FTL framework for wearable healthcare [107], federated-autonomous DL (FADL) [108], and Blockchain-FL [109], and the authors in [66] focused on aggregation techniques—FedAvg [1], SMC-avg [19], FedProx [4], Federated Matched Averaging (FedMA) [110], Scaffold: stochastic controlled averaging for FL [67], tensor factorization [111], federated stochastic block coordinate descent (FedBCD) [31], federated distillation (FD), federated augmentation (FAug) [18], loss-based AdaBoost (LoAdaBoost) [17], HybridFL [34], FL with client selection (FedCS) [112], PrivFL [113], and VerifyNet [114].

The reviewed survey papers did not cover all subtopics, as highlighted in Table 2. In particular, less than half of the surveys thoroughly reviewed FL architectures and personalization techniques. We classify the topics as design aspects, core challenges, and application areas, as shown in Fig. 7, and provide an in-depth discussion and analysis on all the subtopics.

TABLE 2. Comparison of topics covered by survey papers.

Survey Paper	Data Partitioning	FL Architectures	Aggregation Techniques	Personalization Techniques	Communication Cost	Systems Heterogeneity	Statistical Heterogeneity	Privacy /Security	Application Areas
Li, Sahu, *et al.* [3]	✗	✗	✓	✓	✓	✓	✓	✓	✓
Lim *et al.* [25]	✗	✓	✓	✓	✓	✓	✓	✓	✓
Briggs *et al.* [57]	✗	✗	✓	✗	✓	✓	✓	✓	✓
Li, Wen, *et al.* [58]	✓	✓	✓	✗	✓	✓	✓	✓	✓
Li, Fan, *et al.* [59]	✓	✓	✓	✗	✓	✓	✓	✓	✓
Kurupathi, Maass [60]	✓	✗	✗	✗	✗	✗	✗	✓	✓
Yang *et al.* [61]	✓	✗	✗	✓	✓	✗	✓	✓	✓
Xu *et al.* [62]	✗	✗	✓	✓	✓	✗	✓	✓	✓
Kulkarni *et al.* [63]	✗	✗	✓	✓	✗	✗	✓	✗	✗
Lyu *et al.* [64]	✓	✗	✗	✗	✗	✗	✗	✓	✗
Aledhari *et al.* [65]	✓	✓	✓	✗	✓	✓	✓	✗	✓
Mothukuri *et al.* [66]	✓	✗	✓	✓	✓	✓	✗	✓	✓
This work	✓	✓	✓	✓	✓	✓	✓	✓	✓

III. TAXONOMY

The taxonomy of FL research, in terms of design aspects, core challenges, and application areas, is presented in Fig. 7. The design aspects include data partitioning, FL architectures, aggregation techniques, and personalization techniques. Communication cost, systems heterogeneity, statistical heterogeneity, and privacy/security are among the core challenges. In addition, the reviewed survey papers focused on the application areas of industrial engineering, mobile devices, healthcare, and IoT and edge devices. Table 2 compares the topics covered by the survey papers.

Data partitioning classifies FL as HFL, VFL, or FTL, as explained in the Introduction. Beyond these variants, several specialized FL architectures have been developed to improve features such as accuracy, training speed, efficiency, generalization, and applicability for different areas, such as IoT, healthcare, electronic health records (EHRs), and privacy/security. Depending on the FL architecture used, aggregation techniques/algorithms are employed to integrate the local model updates obtained from all participating clients during training to obtain the global model. Different aggregation techniques/algorithms have different priorities, such as increased privacy, optimal communication bandwidth, and support of asynchronous updates. Personalization is another design aspect that needs to be considered for certain scenarios, namely device heterogeneity (storage, computation, and communication), data heterogeneity (i.e., non-IID data), and model heterogeneity (customized models depending on the client's environment).

Expensive communication is a major challenge in FL systems. A federated network can comprise many devices, which means that network communication is much slower than local computation. Therefore, several studies have addressed communication efficiency. Moreover, there can be varying communication capabilities of devices in federated networks due to systems heterogeneity. The different devices may also exhibit varying computing and storage capacities. Due to system and network constraints under numerous settings, only a few selected devices can participate in a training iteration, and some devices may even drop out during an iteration due to connectivity or power issues. Thus, FL techniques need to overcome such systems heterogeneity challenges. In contrast, statistical heterogeneity issues arise due to the violation of IID assumptions in distributed optimization. The violation occurs because different devices across the network often comprise non-identically distributed data. The number of data points across the devices also varies. Therefore, FL approaches must handle the statistical heterogeneity of data. Finally, privacy/security issues are at the core of FL applications. Increased privacy/security achieved using novel methods often comes at the cost of decreased system efficiency or model performance.

All these trade-offs among the various application-specific challenges and design aspects need to be carefully considered and well-balanced to obtain effective privacy-preserving FL systems. These topics are detailed in the following section.

IV. DISCUSSION AND ANALYSIS

In this section, we review and discuss the design aspects, core challenges, and application areas to provide a comprehensive summary of the subtopics—data partitioning, FL architectures, aggregation techniques, personalization techniques, communication cost, systems heterogeneity, statistical heterogeneity, privacy/security, and application areas.

A. DESIGN ASPECTS

1) DATA PARTITIONING

The data used for training FL are non-identical, as they are available on various devices. The sample space of a dataset comprises all dataset instances, whereas the feature space comprises different dataset attributes. For instance, two hospitals may have records of different sets of patients (sample space) and different types of information stored about each patient in their EHR (feature space). Based on how the data are allocated over the sample and feature spaces across multiple participating devices in the FL process, FLSs can typically be categorized as HFL, VFL, and hybrid FL [61].

1) *HFL* is used in scenarios in which the feature space of the datasets is the same but the sample space differs. In HFL, the datasets belonging to different organizations have the same featured space, but the sample space is not related. Such data partitioning is suitable for the cross-device mode, where individual users use FL to enhance their model's performance on a task. In FL, horizontal partitioning is more common. As the local data overlap the feature space, each user can train their local models using the duplicate model architecture. For example, two regional branches of an organization have different groups of users but the same feature spaces as the business. At present, the focus of FLSs is on smart and IoT devices. Work by McMahan *et al.* [1] falls into the horizontal partitioning paradigm. In this framework, an individual user on the Android platform changes the model parameters locally and sends the updated parameters to the cloud server. This enables the training of the centralized model along with other users. Furthermore, to address the issue of finite labeled entities, a hierarchical heterogeneous HFL framework was proposed in [115], which can address the shortage of labels by adapting each user multiple times as the target domain. The authors in [51] suggested a collaborative deep-learning framework in which each user trains independently and only shares a subset of parameters for updating, and classified FL research into broad categories of design aspects, challenges, and application areas.
2) In *VFL*, the datasets across institutions share the same or similar sample spaces, but their feature spaces do not have much in common. In this setting, all participants have homogeneous data, which implies that they differ in feature space but have a partial match with the sample space. For example, two organizations in

a certain area want to train an ML model in collaboration. They have identical clients, but the data of each organization are of distinct types. Due to privacy and security concerns, they cannot interchange their data. In such a scenario, VFL is suitable to train the model. VFL models aggregate these distinct features and calculate the model parameters in a privacy-preserving manner. Finally, it constructs a model by combining data from both parties. An approach using linear regression was proposed by the authors in [116], [117] for data having vertical partitioning. For such data, several secure models, including k-means [78], association rule mining [75], decision tree [77], and naive Bayesian classifier [76], were proposed by Vaidya *et al.* Usually, VFL systems perform entity alignment [118], [119] to combine common samples of different institutions. Then, employing encryption, the combined data are used for training the model. Cheng *et al.* [120] proposed a lossless VFL system to enable the joint training of gradient-boosting decision trees. To recognize common users between two distinct parties, they used privacy-preserving entity alignment. Finally, the selected samples were used to train the decision trees collaboratively.

3) *FTL* is used in situations where two datasets differ in terms of sample as well as feature space. FTL was first proposed in [79]. It enhances existing FL systems and can deal beyond the scope of existing FL algorithms. FTL has gained enormous attention in various industries, especially in the healthcare sector [121]. Using FTL, different types of information related to treatment and diagnosis can be shared between hospitals to diagnose different diseases. In general, transfer learning comprehends a common representation of the features of two different parties. Both parties still need to calculate the prediction results at the time of prediction. Hence, transfer learning [80] techniques can be adopted for the entire feature and sample space in a federated environment. To avoid the possibility of exposing the client data, FTL takes advantage of encryption and approximation to ensure that privacy is safeguarded. Hence, both the actual sensitive data and the models are preserved locally [122]. Sharma *et al.* work on improving FTL by integrating secret sharing technology [123]. The authors in [124], [125] build a FedHealth model that collects data from different institutions through FL and provides customized services for healthcare by using transfer learning.

Each of the aforementioned data partitioning paradigms has its own advantages and disadvantages. For example, two different clinics or hospitals can benefit from securely sharing data with each other based on the number of instances or features they need. One clinic can own millions of patient records, but it might only have very specific information about these patients based on their specialty (e.g., oncology). In contrast, another clinic can be relatively new, possessing much lesser patient records. However, if this is a general clinic without a specialty, then it is likely to have different types of patient information. The first clinic can benefit from VFL, whereas the second one can benefit from HFL. Finally, through FTL, healthcare providers can provide more personalized care if they are given access to data from users' wearable devices for personal fitness.

FL architectures represent how different components are integrated to form an FL environment. Two common architectures of FL are client–server and peer-to-peer architectures.

1) In *client-server architecture*, as illustrated previously in Fig. 1, a central server initiates a global model that it shares with clients to train on their local dataset. After local training, the trained models from the clients involved in the FL environment are collected by the server. The server then aggregates the models' parameters to build a global model and shares it with all clients. The client–server architecture is also known as a centralized architecture for FL. Here, the server coordinates the learning process, which is continuous. In the conventional client–server architecture, the server hosts a model and trains it on shared data. However, the server in the FL setting operates only on local models received from clients synchronously or asynchronously. The main advantage of this architecture is that it incurs less communication overhead. Google used this architecture to develop a virtual keyboard called Gboard for Android. Currently, almost all implementations of FL use client–server architecture.
2) As illustrated in Fig. 2, there is no concept of a central server in *peer-to-peer architecture*, as in the client–server architecture for model aggregations. The role of the central server is replaced with algorithms to ensure security and reliability. Each participant in the FL environment has its own model. A participant improves its model by using information obtained from its neighbors [126]. In the adopted peer-to-peer topology, a protocol is established using a central authority. During training rounds, the network follows this protocol. Such architecture is more secure, as the participating clients communicate directly without a third-party coordinator [127]. However, it requires more computation for message encryption and decryption.

The **aggregation algorithm** describes how the global model is formed by combining local model updates from all clients who participate in the training round. It plays a significant role in HFL, based on a centralized architecture. The most popular aggregation algorithms are compared in Table 3 and summarized below.

1) The FedAvg algorithm [1] proposed by Google is based on a stochastic gradient descent SGD) optimization algorithm, which is the best fit for HFL with a client–server architecture. In this algorithm, the server starts the training process by sharing the global model parameters with a group of clients selected randomly from a pool of clients. The clients then perform

multiple epochs of SGD on their local dataset to train the global model and share the locally trained model with the server. Next, the server next computes the weighted average of all local models to generate a new global model. This process is repeated for several rounds and is robust to unbalanced and non-IID data distribution. Although FedAvg has achieved great success, it has some convergence issues in some settings due to factors such as client drifting [67] and lack of an adaptive learning rate [128].

2) Scaffold [67] solves the problem of client drifting by using the variance reduction technique in its local update. It estimates the update direction of the server model and that of each client. Using the difference, it measures client drifting, which is then used for the local update. This strategy helps overcome the problem of client heterogeneity and reduces the communication round in model convergence.
3) Adaptive federated optimization [128], proposed by Google's research team, introduces adaptability in server optimization. Server optimization is more informed, as the adaptive learning rates allow knowledge to be incorporated from previous iterations. In this optimization framework, a client optimizer minimizes loss using local data over multiple training epochs. Then, to update the global model, the server performs gradient-based optimization on the average of the model updates of clients. FedAvg is a special case in which SGD is used as both a client and a server optimizer with a server learning rate of 1. Although it incorporates adaptive learning rates in server optimization, it does not increase client storage or communication costs. Moreover, it is compatible with cross-device FL. However, it does not completely remove the effect of client heterogeneity. However, for moderate, naturally arising heterogeneity, the adaptive optimizer is quite effective, especially in cross-device settings.
4) FedBoost [129] is a communication-efficient algorithm for FL based on ensemble learning technique. In this approach, an ensemble of pretrained base predictors is trained through FL. It reduces the cost of both server–client and client–server communications without gradient compression and the model compression approach. In addition to communication efficiency, other advantages of this method include computational speedups, convergence guarantees, privacy, and the optimality of the solution for density estimation, for which language modeling is a special case.
5) FedProx [4] addresses the two inherent challenges of FL. First one is system heterogeneity, which refers to the significant variable characteristics of the system or device participating in FL. Second one is statistical heterogeneity, which implies non-IID data across the network. It is a reparametrized and generalized version of FedAvg. Specifically, FedProx can be modified in two ways. First, it enables partial work to be tolerated. Based on the availability of resources, a device can perform variable amounts of work locally; for example, each device can run a varied number of local epochs. The partial solutions from resource-constrained devices are accepted for aggregation. Second, a proximal term is introduced in a device's local solver objective to control for the impact of the variable amounts of local updates.
6) The FedMA [110] algorithm is proposed for introducing FL in modern network architectures for DL. Matching and averaging, based on similarity of features, is performed layer-wise across the channels of convolutional layers, hidden states of long short-term memory networks, and fully connected layer neurons to construct the shared global model at the server. FedMA can also handle client heterogeneity. Within a few rounds of training, it performs better than FedProx and FedAvg.

TABLE 3. Comparison of aggregation algorithms.

Aggregation algorithm	Overcome client drifting?	Adaptive learning rate?	Cross-device compatible?	Communication-efficient?	Address client heterogeneity?	Ensure privacy?
FedAvg [1]	✗	✗	✓	✓	✓	✗
Scaffold [67]	✓	✗	✗	✓	✓	✗
Adaptive Federated Optimization [128]	✗	✓	✓	✓	✗	✗
FedBoost [129]	✓	✗	✗	✓	✓	✓
FedProx [4]	✓	✗	✗	✓	✓	✗
FedMA [110]	✗	✗	✗	✓	✓	✗
Secure Aggregation [19]	✗	✗	✓	✓	✗	✓

7) The secure aggregation [19] algorithm is developed based on the principle of the SMC algorithm. It does not share private information of the mutually distrustful parties, except for the learnable parameters derived from aggregation and thus defends the privacy of each client model. It is fault-tolerant up to 1/3rd of users; that is, it works well even if 1/3rd of the clients fails to engage in the aggregation.

2) PERSONALIZATION TECHNIQUES

In FL, the goal is to train models with a central repository without changing their data samples. Personalization needs to adapt a global model for individual clients and permit users to acquire a richer model so that users' models are trained over a larger set of data samples. Wu *et al.* [106] mentioned three major challenges handled by the FL process during personalization: 1) device heterogeneity for communication capabilities, storage, and computation; 2) data heterogeneity because of non-IID; and 3) model heterogeneity for different models in personalized situations.

Adding contextual features to datasets in a privacy-preserving manner can lead to more personalized predictions. Moreover, based on the similarity of client data, different groups can be formed, and a different model can be trained for each similar cluster [100]. Transfer learning can also be used in a federated setting for model personalization [130]. In transfer learning, knowledge from a global model is transferred to local models, and then the local model parameters are fine-tuned using local data. Other approaches such as multitask learning and meta-learning are used to solve multiple tasks simultaneously. The joint learning in multi-task learning enables the model to use the differences and similarities across the tasks. Meta-learning produces models that are quite adaptive and can solve new tasks with much less training data. Both meta-learning [102] and multitask learning [103], [131], [132] algorithms have been proposed in a federated setting to achieve greater personalization. Knowledge distillation is another method in which a student network mimics a larger teacher network. Using transfer learning and knowledge distillation, Li *et al.* [133] proposed an FL framework that allows clients to design their own networks independently. Arivazhagan *et al.* [134] proposed a neural network architecture in which global data are used to train only the base layers, whereas the personalization layers are trained on local data. A new gradient descent variant, developed by Hanzely *et al.* [135], called loopless gradient descent, allows each device to learn a mixture of its own local model and the global model. The different personalization techniques are summarized in Table 4.

TABLE 4. Summary of personalization techniques.

Research article	Personalization technique	Algorithm
Mansour *et al.* [100]	Adding user context	Clustering
Wang *et al.* [130]	Transfer learning	
Smith *et al.* [102]	Multi-task learning	MOCHA
Finn *et al* [131]	Meta-learning	MAML
Fallah *et al.* [103]		Per-FedAvg
Khodak *et al.* [132]		ARUBA
Li *et al.* [133]	Knowledge distillation	FedMD
Arivazhagan *et al.* [134]	Base and personalization layers	FedPer
Hanzely *et al.* [135]	Mixture of global and local models	LLGD

B. CORE CHALLENGES

Communication is a basic bottleneck in federated networks, which, coupled with security concerns over sending crude information, requires that the information produced on each device stay local. To overcome this issue, researchers have proposed several strategies, some of which involve local updating, compression schemes, decentralized training, and importance-based updating.

Local updating schemes address communication costs by performing additional work on the client that generates and consumes the ML model. As an extension of classical stochastic methods, mini-batch optimization methods have proven to be successful in many cases [142]. For both convex and non-convex objectives, distributed local-updating primal methods have also been successfully applied [143]. As the pivotal FedAvg algorithm proposed in [1], many directions have been taken, including quantizing uploads from edge devices [140].

Sketched and structured updates are among the compression schemes that enable the reduction of the model update size communicated to the FL server from the participating clients during each round [26], [141]. In addition, subsampling, probabilistic quantization, and sparsification were considered in [144]. The authors in [27] further extended the work of [26] to reduce the communication cost from the server to participant, employing approaches such as federated dropout and lossy compression. The accumulation of error and momentum is handled by the central aggregator instead of the clients [139].

Recent studies, such as [6], have carried out decentralized training over heterogeneous data. Hierarchical communication patterns [145] is another approach that reduces dependency on the central server. First, updates from edge devices are aggregated on the edge servers. Then, from the edge servers, the updates are aggregated on the cloud servers.

Importance-based updating is based on the fact that most parameter values of a deep neural network model are sparsely distributed. The edge stochastic gradient descent algorithm was proposed in [28], in which only selected important gradients are sent to the server for updating parameters in each round of communication. The authors in [29] proposed a communication-mitigated federated learning algorithm,

TABLE 5. Strategies and approaches to reduce communication costs.

Research article	Strategies	Contributions	Future concerns
Rothchild *et al.* [139]	Compression	A Count Sketch is used to compress the updates of model. Then, leverages sketch mergeability.	Explore effective ways to combine efficiency within a round and efficiency in number of rounds.
Reisizadeh *et al.* [140]	Local Updating	Method with models periodically averaged at the server and quantized uploads from edge devices.	Can experiment further with the trade-offs made between communication and computation.
Konecny *et al.* [26]	Compression	Structured updates from a restricted space; sketched updates using multiple techniques together like random rotations, subsampling and quantization.	Experimentation with selection of variables used to parametrize space.
McMahan *et al.* [1]	Local Updating	Locally computed SGD updates on each client is sent to a server, which then performs model averaging.	Mitigating the straggler problem.
Han *et al.* [141]	Compression	Pruned the network, quantized the weights and applied Huffman coding.	The quantized network with weight sharing needs to benchmarked on various hardware.

which reduces the communication cost by uploading only the relevant updates of the local model. However, global convergence is still guaranteed. A comparison is first made between the local update of a participant and the global update during each iteration to assign a relevant score to the update. Strategies and approaches for reducing communication costs are summarized in Table 5.

1) SYSTEMS HETEROGENEITY

Due to differences in factors such as network connectivity, memory, CPU, and battery power level, the participants in a federated network often exhibit varying capacities in terms of communication, computation, and storage. Straggler mitigation and other challenges are further compounded due to these system-level characteristics. Popular approaches include asynchronous communication, client participation, and fault tolerance.

Straggler mitigation in heterogeneous environments using asynchronous communication schemes [10] is a promising approach. When there is device variability, synchronous approaches are more susceptible to stragglers. However, asynchronous communication also suffers from bounded-delay assumptions made to control the measure of staleness.

Client participation schemes involve actively selecting participating devices based on system resources such as FedCS [33] and data quality [47] in each round. The FedCS protocol was extended by the authors in [34]. Their hybrid-FL protocol addresses the differences that exist in the data distributions of participating clients. Deep Q-learning [35] is also used to optimize the allocation of resources required for training models. Client participation is controlled by the number of clients in [136] and the amount of data contributed or consumed by clients in [27], [137], [138].

TABLE 6. Strategies and approaches for managing systems heterogeneity.

Research article	Strategies	Contributions	Future concerns
Yang *et al.* [136]	Client Participation, concerned with number of clients	FLASH, an FL simulation platform for developers and researchers.	Experiments were conducted using geo-specific data, yet to try with more diverse data.
Nishio *et al.* [33]	Client Participation, concerned with number of clients	FedCS, Federated Learning with client selection.	Yet to train a more complex model with several million parameters.
Anelli *et al.* [137]	Client Participation, concerned with amount of data interaction by clients	Improved aggregation by measuring contribution of each device based on multiple criteria.	Identification of other local criteria, both general purpose and domain-specific.
Xu *et al.* [138]	Client Participation, concerned with amount of data interaction by clients	ELFISH, a "soft training" method for straggler acceleration, with corresponding aggregation scheme.	Needs further exploration with non-IID datasets.
Caldas *et al.* [27]	Client Participation, concerned with amount of data interaction by clients	Federated dropout, facilitates efficient local training by allowing users to train on subsets of the global model.	Studying the effect of adaptively using these strategies to prevent unfairly biased models.

Fault tolerance [102] is used because learning over remote devices becomes more critical, as some devices in the network often drop out, even before an iteration is completed. Introducing algorithmic redundancy to tolerate device failures is another option known as coded computation. The authors in [15] explored the use of codes to increase the speed of distributed training. The strategies and approaches for managing system heterogeneity are summarized in Table 6.

Statistical heterogeneity refers to the existence of non-IID data across the network. The data generated and collected by network devices are usually non-identically distributed. This generates complexity in terms of analysis, modeling, and evaluation. The usage patterns of different users are distinct. For some clients, the globally shared model does not perform as well as models that are trained locally. Thus, they are disincentivized to participate in the federated network. Moreover, there can be significant variance in terms of the amount of data per device. Also, the possible presence of underlying structures can capture the relationship between the devices and their distributions.

In general, an FL system focuses on learning a single global model. There also exist other approaches, such as learning distinct local parameters simultaneously through multitask learning frameworks [102]. The authors of [155] developed tools to measure statistical heterogeneity using metrics such as local dissimilarity. However, calculating these metrics is quite difficult for a federated network before the training begins. These metrics influence future directions for the development of efficient algorithms to quickly quantify the heterogeneity in an FL system.

To tackle statistical heterogeneity, the authors in [134] utilized the concept of multitask learning. In the FEDPER approach, the participants use a set of base layers pretrained with the FedAvg [1] algorithm. Then, each participant individually trains another set of layers using their local data. The authors empirically showed that the FEDPER approach outperforms a pure FedAvg approach using the Flickr-AES dataset [134], considering that the personalization layers can represent the personal predilection of an FL user.

2) FL THREAT MODELS

FL offers an emerging paradigm for facilitating multiple organization data collaborations without revealing their private data to each other. However, recent research has demonstrated that FL may not always provide sufficient privacy guarantees during model update; it may face several vulnerabilities from both the server and participants. As summarized in Table 7, according to the threat models, the following are two prominent forms of attacks that occur:

1) *Poisoning Attacks*, which can be executed either in the training phase of the model or on the data. Two types of poisoning occur:
   a) *Data poisoning* occurs during local data collection. Data poisoning attacks can occur in two forms: clean-label attacks (adversaries can poison the correct class of data samples) and dirty-label

TABLE 7. Summary of FL threat models.

Research article	Attack type	Threat model	Attack strategies	Attack target
Shafahi *et al.* [146]	Poisoning Attack	Data Poisoning	"Watermarking" strategy (frog image)	Data
Gu *et al.* [147]			Backdoor attack	
Bhagoji *et al.* [55]		Model Poisoning	Stealth metrics, boosting of malicious agent's updates, parameter estimation for the benign agents' updates	Model
Fang *et al.* [148]			Manipulate global model via local model parameter manipulation on compromised devices	
Bagdasaryan *et al.* [149]			Backdoored image-classification model, backdoored word-prediction model	
Melis *et al.* [150]	Inference Attack	Membership inference, inferring properties	Gradient exchange	Data
Pyrgelis *et al.* [151]		Membership inference	ML classifier	
Zhu *et al.* [152]		Inferring training inputs (and labels; inconsistently)	DLG	
Zhao *et al.* [153]		Inferring training inputs and labels	iDLG	
Hitaj *et al.* [154]		Inferring class representative	GAN attack	

attacks (adversaries try to misclassify the target label of the FL training dataset) [146], [147].

b) *Model poisoning* occurs during model training. According to Bhagoji *et al.* [55], model poisoning is accomplished by an adversary controlling a few malicious representatives with the aim of misclassifying specific inputs with high confidence. Bagdasaryan *et al.* [149] introduced a new scope of FL vulnerability by inserting the backdoor into the joint model. FL models are more vulnerable to model poisoning attacks than data poisoning attacks. This form of attack can be used to create misclassification in image and next-word prediction problems.

2) *Inference Attacks*: Serious privacy leakage may occur in FL during updates of the model. When exchanging gradients, the private information of participants can be exposed to the adversary [70], [150], [152], [156]. Pyrgelis *et al.* [151] conducted membership inference attacks to identify vulnerability at the aggregate location. According to the threat model surveyed by Lyu *et al.* [64], the inference attack falls into two categories—white-box attack and black-box attack. Deep leakage from gradients (DLG) [152] obtains private training data in the inference phase. Another algorithm, iDLG, also exposes the labels of training inputs [153]. Hitaj *et al.* [154] applied a GAN attack, which allows the adversarial party in the training process to fabricate an inferring class representative.

Privacy is one of the most critical parts of FL. This section briefly reviews various privacy and security techniques for FL:

1) *Secure Multiparty Computation* (SMC) is a privacy mechanism used in FL. An SMC model comprises multiple parties and provides proper security. This model ensures that each party knows only its inputs and outputs and nothing about the other parties. Bonawitz *et al.* designed a communication-efficient SMC protocol for high-dimensional data to protect the privacy of users' model gradients [19].

2) *DP* is a privacy-preserving mechanism that protects individual privacy by adding noise in the data. There are various types of DP:

a) *Local DP*: Each data point is distorted with noise.

b) *Global DP*: To protect individuals' privacy, the output of the dataset query is distorted with noise.

c) *Hybrid DP*: Multiple trust models are combined by partitioning users according to their trust model preferences.

Geyer *et al.* [23] developed a method for obtaining DP at the client level for FL. Wei *et al.* [157] proposed an aggregation algorithm called NbAFL, in which noise was added to client-side parameters before aggregation. The authors in [158] used both SMC and DP mechanisms to avoid differential attacks.

3) *HE* is another security mechanism in FL that protects user data by changing parameters under the encryption method. HE is a cryptographic technique that performs mathematical operations on data as if they were unencrypted. Many researchers have worked with homomorphic encryption to preserve privacy [159], [160]. To guarantee the privacy of users' local gradients during FL, Xu *et al.* [114] proposed a double-masking protocol.

3) APPLICATIONS

Although FL faces some limitations and severe challenges, it has been successfully implemented in several real-life applications:

1) *Applications in NLP:* FL has become a hot research topic since the concept was first introduced by Google to predict the next word in a virtual keyboard for smartphones [161]. Further improvements in predicting the next word using pretrained word embeddings were achieved by other researchers [87]. Wake word detection was another contribution made by Leroy *et al.* [74]. Emoji prediction from text typed on a mobile keyboard was introduced by Ramaswamy *et al.* [88]. In addition, some researchers have worked on learning out of vocabulary words on virtual keyboards for smartphones [86], and some have tried to improve the virtual keyboard's search suggestion quality [162].

2) *Applications in healthcare:* Huang *et al.* [17] predicted the mortality rate of patients suffering from heart disease by using electronic medical records from multiple hospitals. Brisimi *et al.* [82] used an EHR to determine whether a heart disease patient needs to be hospitalized. Li *et al.* [163] also studied mortality and hospital stay time. Using health records, Lee *et al.* [164] proposed a method to determine similar patients from different hospitals while preserving the patients' privacy. They used a federated patient hashing framework.

3) *Applications in computer vision*: Another important application area of FL is computer vision. Shao *et al.* [165] proposed a federated face presentation attack detection method. Liu *et al.* [166] worked on smart city safety monitoring solutions based on computer vision.

4) *Applications in transportation*: The development of intelligent transportation systems using FL was explored by Elbir *et al.* [167]. Lim *et al.* [168] proposed an FL-based approach in UAV-enabled Internet of Vehicles for developing applications such as the management of car parking occupancy and traffic prediction.

V. OPEN ISSUES AND CHALLENGES

There are several open issues and challenges in FL [169]. Trade-offs among accuracy, privacy, communication cost, and personalization level must be carefully considered when

designing an FL system. Such considerations often depend on the specific use case or application area. In this section, we discuss some open issues related to design aspects, core challenges, and application areas.

A. DESIGN ASPECTS

1) DATA PARTITIONING AND FL ARCHITECTURES

In addition to the primary forms of data partitioning schemes and FL architectures discussed in this study, other variations in FL architectures have recently been developed. For instance, PerFit [106] is cloud-based and enables personalized FL approaches to be selected flexibly, thus making it suitable for IoT applications. Another architecture is FedHealth [107], which uses the FTL framework for wearable healthcare to build personalized models, thus enabling personalized healthcare services. Future studies can focus on developing FL architecture schemes that facilitate the specific requirements of different industries and application areas to be met.

2) AGGREGATION TECHNIQUES

Developers who wish to implement FL solutions can benefit from toolkits that offer standardized and preconfigured aggregation algorithms that are suitable for their specific application areas and use cases. Similar to AutoML solutions, such a toolkit for FL can lower the barrier of entry for nonspecialist developers.

3) PERSONALIZATION TECHNIQUES

Adding suitable user and context features to the shared global model is a possible alternative to having device-specific personalization. For example, the filter order in applications such as Snapchat can be arranged according to certain user features, such as browsing history, age, sex, likes and dislikes, and usage patterns. Thus, developing architectures that can accommodate such user and context features effectively for different tasks is another open problem.

Moreover, as observed in [170], a gap exists between the accuracy of personalized and global models, making the case of personalization techniques an important research area in FL. Nevertheless, no clear metrics have yet been formulated to evaluate the performance of personalization techniques. Wang *et al.* [130] evaluated the conditions under which personalization yields desirable models. Further research is required to develop comprehensive metrics to assess the effectiveness of personalized approaches.

B. CORE CHALLENGES

1) COMMUNICATION

There is a trade-off between communication costs and accuracy in FL. The benchmarks in ML do not usually set any restriction criterion. It is worth considering setting the communication budget as a restriction criterion in communication-focused FL benchmarks. For example, the authors in [171], [172] explored one-shot or few-shot communication schemes in FL, and those in [172] attempted to maximize the performance for fixed rounds of communication (i.e., single or few rounds). Additionally, these methods need to be thoroughly evaluated and analyzed in the FL setting, where the networks can be highly heterogeneous.

In cross-device FL, only a few devices are often active during an iteration. There is scope for an in-depth analysis of the consequences of this asynchronous communication scheme where the devices become active based on certain events.

2) SYSTEMS HETEROGENEITY

Various algorithms [33], [35] have been proposed to address systems heterogeneity. However, wireless connectivity might not be available consistently. Many participating devices may drop from the FL system during training. Future studies can design new FL algorithms that are more robust, even when a larger number of devices drop out of the network due to connectivity issues.

Li *et al.* [4] recently introduced a proximal term in the optimization objective to allow partial solutions obtained from stragglers to be carefully incorporated and aggregated instead of totally dropping them. The authors in [173] took a different approach and implemented an FL system that addressed device heterogeneity by selecting different levels of quantized models following a device-specific analysis conducted by the FL server.

3) STATISTICAL HETEROGENEITY

Eichner *et al.* [99] developed a pluralistic solution to alleviate a form of data heterogeneity in which devices exhibited different characteristics during the day versus those at night. Further research can be conducted to explore similar methods to address diurnal variations at more granular times of day (instead of only day versus night) or at different times of the week. For example, let us consider a federated network over a commercial neighborhood. The data characteristics obtained from devices available over the weekdays would likely be very different from those available over the weekends. The effectiveness of a pluralistic solution in such a scenario can be investigated.

As noted by the authors in [99], where they only worked with convex objectives and sequential SGD, further analysis can be conducted to explore block-cyclic data in a nonconvex setting and employ methods such as parallel SGD.

4) PRIVACY/SECURITY

While device-specific local or global level privacy has been well-studied and understood, finer privacy requirements at the sample level form a promising, ongoing research topic. The sample-specific privacy guarantee technique developed by Li *et al.* [174] trades off privacy for higher accuracy.

Hybrid methods deal with both sample- and device-level privacy requirements. One approach can be to use sample-specific privacy for a subset of data based on specific

levels of a category or date range while using device-specific privacy for the remaining data.

5) ABLATION ANALYSIS

The evaluation performed by an FL system is often more complex than that performed by traditional ML and DL systems. While different research efforts deal with specific focus areas, a holistic industrial system would need to consider several aspects while building FL solutions, such as privacy, accuracy/loss, communication rounds, and heterogeneity. A standard platform needs to be developed to facilitate a holistic ablation analysis of the different parts of an FL system.

C. APPLICATION AREAS

FL has mainly been applied to supervised learning problems. Future research can attempt to tackle the challenges that may arise when using FL in applications that call for data exploration, unsupervised, semi-supervised, and reinforcement learning.

The challenges faced in implementing FL solutions for different application areas have not yet been thoroughly studied, with the current studies primarily focusing on training FL models. In addition to the core challenges discussed in this paper, issues that are specific to the industry domain or application area also need to be considered. For instance, there are application areas such as mobile edge networks that require energy-efficient communication to be greatly emphasized.

VI. CONCLUSION

FL allows participating organizations to collaboratively train prediction models without having to share their data. Recently, there has been growing interest in FL research in both industry and academia. FL enables certain industries, such as healthcare, to overcome challenges related to data collection and privacy.

This growing interest in FL has motivated us to review most of the contemporary survey papers on FL and to classify FL into several topics under the design aspects, core challenges, and application domains. In this study, we thoroughly investigated and analyzed the FL survey papers and conducted a holistic review of each FL topic. Finally, we outline promising future research directions. This study is expected to help future researchers in FL and related areas to scope their work.

REFERENCES

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016, *arXiv:1602.05629*. [Online]. Available: http://arxiv.org/abs/1602.05629

[2] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Federated learning of deep networks using model averaging," 2016, *arXiv:1602.05629*. [Online]. Available: https://arxiv.org/abs/1602.05629

[3] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020, doi: 10.1109/MSP.2020.2975749.

[4] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," 2018, *arXiv:1812.06127*. [Online]. Available: http://arxiv.org/abs/1812.06127

[5] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, "Communication compression for decentralized training," 2018, *arXiv:1803.06443*. [Online]. Available: http://arxiv.org/abs/1803.06443

[6] L. He, A. Bian, and M. Jaggi, "COLA: Decentralized linear learning," 2018, *arXiv:1808.04883*. [Online]. Available: http://arxiv.org/abs/1808.04883

[7] A. Lalitha, X. Wang, O. Kilinc, Y. Lu, T. Javidi, and F. Koushanfar, "Decentralized Bayesian learning over graphs," 2019, *arXiv:1905.10466*. [Online]. Available: http://arxiv.org/abs/1905.10466

[8] W. Dai, A. Kumar, J. Wei, Q. Ho, G. Gibson, and E. P. Xing, "High-performance distributed ML at scale through parameter server consistency models," 2014, *arXiv:1410.8043*. [Online]. Available: http://arxiv.org/abs/1410.8043

[9] Q. Ho, J. Cipar, H. Cui, S. Lee, J. K. Kim, P. B. Gibbons, G. A. Gibson, G. Ganger, and E. P. Xing, "More effective distributed ML via a stale synchronous parallel parameter server," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 1223–1231.

[10] M. A. Zinkevich, M. Weimer, A. Smola, and L. Li, "Parallelized stochastic gradient descent," in *Proc. NIPS*, 2010, p. 4.

[11] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, H. B. McMahan, T. Van Overveldt, D. Petrou, D. Ramage, and J. Roselander, "Towards federated learning at scale: System design," 2019, *arXiv:1902.01046*. [Online]. Available: http://arxiv.org/abs/1902.01046

[12] Z. Charles and D. Papailiopoulos, "Gradient coding using the stochastic block model," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018, pp. 1998–2002, doi: 10.1109/ISIT.2018.8437887.

[13] Z. Charles, D. Papailiopoulos, and J. Ellenberg, "Approximate gradient coding via sparse random graphs," 2017, *arXiv:1711.06771*. [Online]. Available: http://arxiv.org/abs/1711.06771

[14] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," *IEEE Trans. Inf. Theory*, vol. 64, no. 3, pp. 1514–1529, Mar. 2018, doi: 10.1109/TIT.2017.2736066.

[15] A. Reisizadeh, S. Prakash, R. Pedarsani, and A. S. Avestimehr, "Coded computation over heterogeneous clusters," *IEEE Trans. Inf. Theory*, vol. 65, no. 7, pp. 4227–4242, Jul. 2019, doi: 10.1109/TIT.2019.2904055.

[16] R. Tandon, Q. Lei, A. G. Dimakis, and N. Karampatziakis, "Gradient coding: Avoiding stragglers in distributed learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3368–3376.

[17] L. Huang, Y. Yin, Z. Fu, S. Zhang, H. Deng, and D. Liu, "LoAdaBoost: Loss-based AdaBoost federated machine learning on medical data," 2018, *arXiv:1811.12629*. [Online]. Available: https://arxiv.org/abs/1811.12629

[18] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data," 2018, *arXiv:1811.11479*. [Online]. Available: http://arxiv.org/abs/1811.11479

[19] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. Mcmahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2017, pp. 1175–1191, doi: 10.1145/3133956.3133982.

[20] B. Ghazi, R. Pagh, and A. Velingker, "Scalable and differentially private distributed aggregation in the shuffled model," 2019, *arXiv:1906.08320*. [Online]. Available: http://arxiv.org/abs/1906.08320

[21] N. Agarwal, A. T. Suresh, F. Yu, S. Kumar, and H. B. Mcmahan, "CpSGD: Communication-efficient and differentially-private distributed SGD," 2018, *arXiv:1805.10559*. [Online]. Available: http://arxiv.org/abs/1805.10559

[22] A. Bhowmick, J. Duchi, J. Freudiger, G. Kapoor, and R. Rogers, "Protection against reconstruction and its applications in private federated learning," 2018, *arXiv:1812.00984*. [Online]. Available: http://arxiv.org/abs/1812.00984

[23] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," 2017, *arXiv:1712.07557*. [Online]. Available: http://arxiv.org/abs/1712.07557

[24] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," *arXiv:1710.06963*, 2018. [Online]. Available: https://arxiv.org/abs/1710.06963

[25] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020, doi: 10.1109/COMST.2020.2986024.

[26] J. Konecný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *Proc. ICLR*, 2018, pp. 1–10.
[27] S. Caldas, J. Konečny, H. B. McMahan, and A. Talwalkar, "Expanding the reach of federated learning by reducing client resource requirements," 2018, *arXiv:1812.07210*. [Online]. Available: http://arxiv.org/abs/1812.07210
[28] Z. Tao and Q. Li, "ESGD: Communication efficient distributed deep learning on the edge," in *Proc. USENIX Workshop Hot Topics Edge Comput. (HotEdge)*, 2018.
[29] L. Wang, W. Wang, and B. Li, "CMFL: Mitigating communication overhead for federated learning," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 954–964, doi: 10.1109/ICDCS.2019.00099.
[30] L. Liu, J. Zhang, S. H. Song, and K. B. Letaief, "Edge-assisted hierarchical federated learning with non-IID data," 2019, *arXiv:1905.06641*. [Online]. Available: https://arxiv.org/abs/1905.06641
[31] Y. Liu, Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong, and Q. Yang, "A communication efficient collaborative learning framework for distributed features," 2019, *arXiv:1912.11187*. [Online]. Available: http://arxiv.org/abs/1912.11187
[32] X. Yao, C. Huang, and L. Sun, "Two-stream federated learning: Reduce the communication costs," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2018, pp. 1–4, doi: 10.1109/VCIP.2018.8698609.
[33] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7, doi: 10.1109/ICC.2019.8761315.
[34] N. Yoshida, T. Nishio, M. Morikura, K. Yamamoto, and R. Yonetani, "Hybrid-FL: Cooperative learning mechanism using non-IID data in wireless networks," 2019, *arXiv:1905.07210*. [Online]. Available: https://arxiv.org/abs/1905.07210
[35] T. T. Anh, N. C. Luong, D. Niyato, D. I. Kim, and L.-C. Wang, "Efficient training management for mobile crowd-machine learning: A deep reinforcement learning approach," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1345–1348, Oct. 2019, doi: 10.1109/LWC.2019.2917133.
[36] H. T. Nguyen, N. Cong Luong, J. Zhao, C. Yuen, and D. Niyato, "Resource allocation in mobility-aware federated learning networks: A deep reinforcement learning approach," in *Proc. IEEE 6th World Forum Internet Things (WF-IoT)*, Jun. 2020, pp. 1–6, doi: 10.1109/WF-IoT48130.2020.9221089.
[37] T. Li, M. Sanjabi, A. Beirami, and V. Smith, "Fair resource allocation in federated learning," 2019, *arXiv:1905.10497*. [Online]. Available: http://arxiv.org/abs/1905.10497
[38] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020, doi: 10.1109/TWC.2019.2946245.
[39] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020, doi: 10.1109/TWC.2020.2974748.
[40] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020, doi: 10.1109/TWC.2019.2961673.
[41] M. R. Sprague, A. Jalalirad, M. Scavuzzo, C. Capota, M. Neun, L. Do, and M. Kopp, "Asynchronous federated learning for geospatial applications," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2019, pp. 21–28, doi: 10.1007/978-3-030-14880-5_2.
[42] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019, doi: 10.1109/JSAC.2019.2904348.
[43] S. Feng, D. Niyato, P. Wang, D. I. Kim, and Y.-C. Liang, "Joint service pricing and cooperative relay communication for federated learning," in *Proc. Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom), IEEE Cyber, Phys. Social Comput. (CPSCom), IEEE Smart Data (SmartData)*, Jul. 2019, pp. 815–820, doi: 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00148.
[44] Y. Sarikaya and O. Ercetin, "Motivating workers in federated learning: A Stackelberg game perspective," *IEEE Netw. Lett.*, vol. 2, no. 1, pp. 23–27, Mar. 2020, doi: 10.1109/lnet.2019.2947144.
[45] L. U. Khan, L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. H. Nguyen, and C. S. Hong, "Federated learning for edge networks: Resource optimization and incentive mechanism," *IEEE Commun. Mag.*, vol. 58, no. 10, pp. 88–93, Oct. 2020, doi: 10.1109/MCOM.001.1900649.
[46] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A learning-based incentive mechanism for federated learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6360–6368, Jul. 2020, doi: 10.1109/JIOT.2020.2967772.
[47] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y.-C. Liang, and D. I. Kim, "Incentive design for efficient federated learning in mobile networks: A contract theory approach," in *Proc. IEEE VTS Asia Pacific Wireless Commun. Symp. (APWCS)*, Aug. 2019, pp. 1–5, doi: 10.1109/VTS-APWCS.2019.8851649.
[48] D. Ye, R. Yu, M. Pan, and Z. Han, "Federated learning in vehicular edge computing: A selective model aggregation approach," *IEEE Access*, vol. 8, pp. 23920–23935, 2020, doi: 10.1109/ACCESS.2020.2968399.
[49] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019, doi: 10.1109/JIOT.2019.2940820.
[50] M. Abadi, A. Chu, I. Goodfellow, H. B. Mcmahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318, doi: 10.1145/2976749.2978318.
[51] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2015, pp. 1310–1321, doi: 10.1145/2810103.2813687.
[52] Y. Liu, Z. Ma, X. Liu, S. Ma, R. H. Deng, and K. Ren, "Boosting privately: Federated extreme gradient boosting for mobile crowdsensing," in *Proc. IEEE 40th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Nov./Dec. 2020, pp. 1–11.
[53] A. Triastcyn and B. Faltings, "Federated generative privacy," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 50–57, Jul. 2020, doi: 10.1109/MIS.2020.2993966.
[54] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," 2018, *arXiv:1808.04866*. [Online]. Available: http://arxiv.org/abs/1808.04866
[55] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," 2018, *arXiv:1811.12470*. [Online]. Available: http://arxiv.org/abs/1811.12470
[56] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "On-device federated learning via blockchain and its latency analysis," 2018, *arXiv:1808.03949*. [Online]. Available: https://arxiv.org/abs/1808.03949
[57] C. Briggs, Z. Fan, and P. Andras, "A review of privacy-preserving federated learning for the Internet-of-Things," in *Federated Learning Systems* (Studies in Computational Intelligence). Cham, Switzerland: Springer, 2020.
[58] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," 2019, pp. 1–41, *arXiv:1907.09693*. [Online]. Available: http://arxiv.org/abs/1907.09693
[59] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," *Comput. Ind. Eng.*, vol. 149, Nov. 2020, Art. no. 106854, doi: 10.1016/j.cie.2020.106854.
[60] S. R. Kurupathi and W. Maass, "Survey on federated learning towards privacy preserving AI," in *Proc. Comput. Sci. Inf. Technol. (CSIT)*, Sep. 2020, pp. 1–19, doi: 10.5121/csit.2020.101120.
[61] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, Feb. 2019, doi: 10.1145/3298981.
[62] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," *J. Healthcare Informat. Res.*, vol. 5, no. 1, pp. 1–19, Mar. 2021, doi: 10.1007/s41666-020-00082-4.
[63] V. Kulkarni, M. Kulkarni, and A. Pant, "Survey of personalization techniques for federated learning," in *Proc. 4th World Conf. Smart Trends Syst., Secur. Sustainability (WorldS)*, 2020, pp. 794–797, doi: 10.1109/WorldS450073.2020.9210355.
[64] L. Lyu, H. Yu, J. Zhao, and Q. Yang, "Threats to federated learning," in *Federated Learning*. Cham, Switzerland: Springer, 2020, pp. 3–16.
[65] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140699–140725, 2020, doi: 10.1109/ACCESS.2020.3013541.
[66] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, pp. 619–640, Feb. 2021, doi: 10.1016/j.future.2020.10.007.

[67] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," 2019, *arXiv:1910.06378*. [Online]. Available: https://arxiv.org/abs/1910.06378

[68] X. Zhang, S. Ji, H. Wang, and T. Wang, "Private, yet practical, multiparty deep learning," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 1442–1452, doi: 10.1109/ICDCS.2017.215.

[69] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, and B. Thorne, "Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption," 2017, *arXiv:1711.10677*. [Online]. Available: http://arxiv.org/abs/1711.10677

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

[71] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979, doi: 10.1145/359168.359176.

[72] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," *J. Privacy Confidentiality*, vol. 7, no. 3, pp. 17–51, May 2017, doi: 10.29012/jpc.v7i3.405.

[73] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2013, doi: 10.1561/0400000042.

[74] D. Leroy, A. Coucke, T. Lavril, T. Gisselbrecht, and J. Dureau, "Federated learning for keyword spotting," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6341–6345, doi: 10.1109/ICASSP.2019.8683546.

[75] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2002, pp. 639–644, doi: 10.1145/775047.775142.

[76] J. Vaidya and C. Clifton, "Privacy preserving Naïve Bayes classifier for vertically partitioned data," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2004, pp. 522–526, doi: 10.1137/1.9781611972740.59.

[77] J. Vaidya, C. Clifton, M. Kantarcioglu, and A. S. Patterson, "Privacy-preserving decision trees over vertically partitioned data," *ACM Trans. Knowl. Discovery From Data*, vol. 2, no. 3, pp. 1–27, Oct. 2008, doi: 10.1145/1409620.1409624.

[78] J. Vaidya and C. Clifton, "Privacy-preserving k-means clustering over vertically partitioned data," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2003, pp. 206–215, doi: 10.1145/956750.956776.

[79] Y. Liu, Y. Kang, C. Xing, T. Chen, and Q. Yang, "A secure federated transfer learning framework," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 70–82, Jul./Aug. 2020.

[80] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[81] L. Zhao, L. Ni, S. Hu, Y. Chen, P. Zhou, F. Xiao, and L. Wu, "InPrivate digging: Enabling tree-based distributed data mining with differential privacy," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2018, pp. 2087–2095, doi: 10.1109/INFOCOM.2018.8486352.

[82] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *Int. J. Med. Informat.*, vol. 112, pp. 59–67, Apr. 2018, doi: 10.1016/j.ijmedinf.2018.01.007.

[83] A. C. Zhou, Y. Xiao, Y. Gong, B. He, J. Zhai, and R. Mao, "Privacy regulation aware process mapping in geo-distributed cloud data centers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 8, pp. 1872–1888, Aug. 2019, doi: 10.1109/TPDS.2019.2896894.

[84] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, and F. Beaufays, "Applied federated learning: Improving Google keyboard query suggestions," 2018, *arXiv:1812.02903*. [Online]. Available: http://arxiv.org/abs/1812.02903

[85] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2018, pp. 63–71, doi: 10.1109/INFOCOM.2018.8486403.

[86] M. Chen, R. Mathews, T. Ouyang, and F. Beaufays, "Federated learning of out-of-vocabulary words," 2019, *arXiv:1903.10635*. [Online]. Available: http://arxiv.org/abs/1903.10635

[87] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," 2018, *arXiv:1811.03604*. [Online]. Available: http://arxiv.org/abs/1811.03604

[88] S. Ramaswamy, R. Mathews, K. Rao, and F. Beaufays, "Federated learning for emoji prediction in a mobile keyboard," 2019, *arXiv:1906.04329*. [Online]. Available: http://arxiv.org/abs/1906.04329

[89] J. Feng, C. Rong, F. Sun, D. Guo, and Y. Li, "PMF: A privacy-preserving human mobility prediction framework via federated learning," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 1, pp. 1–21, Mar. 2020, doi: 10.1145/3381006.

[90] K. Sozinov, V. Vlassov, and S. Girdzijauskas, "Human activity recognition using federated learning," in *Proc. IEEE Int. Conf. Parallel Distrib. Process. With Appl., Ubiquitous Comput. Commun., Big Data Cloud Comput., Social Comput. Netw., Sustain. Comput. Commun. (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, Dec. 2018, pp. 1103–1111, doi: 10.1109/BDCloud.2018.00164.

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

[92] T. Yu, T. Li, Y. Sun, S. Nanda, V. Smith, V. Sekar, and S. Seshan, "Learning context-aware policies from multiple smart homes via federated multi-task learning," in *Proc. IEEE/ACM 5th Int. Conf. Internet-Things Design Implement. (IoTDI)*, Apr. 2020, pp. 104–115, doi: 10.1109/IoTDI49375.2020.00017.

[93] B. Hu, Y. Gao, L. Liu, and H. Ma, "Federated region-learning: An edge computing based framework for urban environment sensing," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7, doi: 10.1109/GLOCOM.2018.8647649.

[94] X. Han, H. Yu, and H. Gu, "Visual inspection with federated learning," in *Proc. Int. Conf. Image Anal. Recognit.*, 2019, pp. 52–64, doi: 10.1007/978-3-030-27272-2_5.

[95] D. Bogdanov, S. Laur, and J. Willemson, "Sharemind: A framework for fast privacy-preserving computations," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2008, pp. 192–206, doi: 10.1007/978-3-540-88313-5_13.

[96] R. L. Rivest, M. L. Dertouzos, and L. Adleman, "On data banks and privacy homomorphisms," *Found. Secur. Comput.*, vol. 4, no. 11, pp. 169–180, 1978.

[97] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," 2017, *arXiv:1712.01887*. [Online]. Available: http://arxiv.org/abs/1712.01887

[98] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," 2019, *arXiv:1902.00146*. [Online]. Available: http://arxiv.org/abs/1902.00146

[99] H. Eichner, T. Koren, H. B. McMahan, N. Srebro, and K. Talwar, "Semi-cyclic stochastic gradient descent," 2019, *arXiv:1904.10120*. [Online]. Available: http://arxiv.org/abs/1904.10120

[100] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, "Three approaches for personalization with applications to federated learning," 2020, *arXiv:2002.10619*. [Online]. Available: http://arxiv.org/abs/2002.10619

[101] J. Schneider and M. Vlachos, "Personalization of deep learning," 2019, *arXiv:1909.02803*. [Online]. Available: https://arxiv.org/abs/1909.02803

[102] V. Smith, C.-K. Chiang, M. Sanjabi, and A. Talwalkar, "Federated multi-task learning," 2017, *arXiv:1705.10467*. [Online]. Available: http://arxiv.org/abs/1705.10467

[103] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning: A meta-learning approach," 2020, *arXiv:2002.07948*. [Online]. Available: http://arxiv.org/abs/2002.07948

[104] S. Feng and H. Yu, "Multi-participant multi-class vertical federated learning," 2020, *arXiv:2001.11154*. [Online]. Available: http://arxiv.org/abs/2001.11154

[105] T.-D. Cao, T. Truong-Huu, H. Tran, and K. Tran, "A federated learning framework for privacy-preserving and parallel training," 2020, *arXiv:2001.09782*. [Online]. Available: http://arxiv.org/abs/2001.09782

[106] Q. Wu, K. He, and X. Chen, "Personalized federated learning for intelligent IoT applications: A cloud-edge based framework," *IEEE Open J. Comput. Soc.*, vol. 1, pp. 35–44, 2020, doi: 10.1109/OJCS.2020.2993259.

[107] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "FedHealth: A federated transfer learning framework for wearable healthcare," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 83–93, Jul. 2020, doi: 10.1109/MIS.2020.2988604.

[108] D. Liu, T. Miller, R. Sayeed, and K. D. Mandl, "FADL: Federated-autonomous deep learning for distributed electronic health record," 2018, *arXiv:1811.11400*. [Online]. Available: http://arxiv.org/abs/1811.11400
[109] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020, doi: 10.1109/TII.2019.2942190.
[110] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," 2020, *arXiv:2002.06440*. [Online]. Available: http://arxiv.org/abs/2002.06440
[111] J. Ma, Q. Zhang, J. Lou, J. C. Ho, L. Xiong, and X. Jiang, "Privacy-preserving tensor factorization for collaborative health data analysis," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1291–1300, doi: 10.1145/3357384.3357878.
[112] R. Xu, N. Baracaldo, Y. Zhou, A. Anwar, and H. Ludwig, "HybridAlpha: An efficient approach for privacy-preserving federated learning," in *Proc. 12th ACM Workshop Artif. Intell. Secur. (AISec)*, 2019, pp. 13–23, doi: 10.1145/3338501.3357371.
[113] K. Mandal and G. Gong, "PrivFL: Practical privacy-preserving federated regressions on high-dimensional data over mobile networks," in *Proc. ACM SIGSAC Conf. Cloud Comput. Secur. Workshop (CCSW)*, 2019, pp. 57–68, doi: 10.1145/3338466.3358926.
[114] G. Xu, H. Li, S. Liu, K. Yang, and X. Lin, "VerifyNet: Secure and verifiable federated learning," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 911–926, 2020, doi: 10.1109/TIFS.2019.2929409.
[115] D. Gao, C. Ju, X. Wei, Y. Liu, T. Chen, and Q. Yang, "HHHFL: Hierarchical heterogeneous horizontal federated learning for electroencephalography," 2019, *arXiv:1909.05784*. [Online]. Available: http://arxiv.org/abs/1909.05784
[116] A. Gascón *et al.*, "Privacy preserving distributed linear regression on high-dimensional data," in *Proc. Privacy Enhancing Technol.*, vol. 2017, no. 4, 2017, pp. 345–364.
[117] I. Giacomelli, S. Jha, M. Joye, C. D. Page, and K. Yoon, "Privacy-preserving ridge regression with only linearly-homomorphic encryption," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Secur.*, 2018, pp. 243–261, doi: 10.1007/978-3-319-93387-0_13.
[118] Y. Zhuang, G. Li, and J. Feng, "A survey on entity alignment of knowledge base," *Jisuanji Yanjiu yu Fazhan/Comput. Res. Develop.*, vol. 53, no. 1, p. 165, 2016, doi: 10.7544/issn1000-1239.2016.20150661.
[119] P. Christen, *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer, 2012.
[120] K. Cheng, T. Fan, Y. Jin, Y. Liu, T. Chen, D. Papadopoulos, and Q. Yang, "SecureBoost: A lossless federated learning framework," 2019, *arXiv:1901.08755*. [Online]. Available: http://arxiv.org/abs/1901.08755
[121] Q. Jing, W. Wang, J. Zhang, H. Tian, and K. Chen, "Quantifying the performance of federated transfer learning," 2019, *arXiv:1912.12795*. [Online]. Available: http://arxiv.org/abs/1912.12795
[122] S. Caldas, V. Smith, and A. Talwalkar, "Federated kernelized multi-task learning," in *Proc. SysML Conf.*, 2018, pp. 1–3.
[123] S. Sharma, C. Xing, Y. Liu, and Y. Kang, "Secure and efficient federated transfer learning," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 2569–2576, doi: 10.1109/BigData47090.2019.9006280.
[124] Y. Chen, Y. Ning, and H. Rangwala, "Asynchronous online federated learning for edge devices with non-IID data," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 15–24.
[125] Y. Chen, X. Sun, and Y. Jin, "Communication-efficient federated deep learning with layerwise asynchronous model update and temporally weighted aggregation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 4229–4238, Oct. 2020, doi: 10.1109/TNNLS.2019.2953131.
[126] P. Vanhaesebrouck, A. Bellet, and M. Tommasi, "Decentralized collaborative learning of personalized models over networks," 2016, *arXiv:1610.05202*. [Online]. Available: http://arxiv.org/abs/1610.05202
[127] Z. Jiang, A. Balu, C. Hegde, and S. Sarkar, "Collaborative deep learning in fixed topology networks," 2017, *arXiv:1706.07880*. [Online]. Available: http://arxiv.org/abs/1706.07880
[128] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," 2020, *arXiv:2003.00295*. [Online]. Available: http://arxiv.org/abs/2003.00295
[129] J. Hamer, M. Mohri, and A. T. Suresh, "FedBoost: Communication-efficient algorithms for federated learning," in *Proc. ICML*, 2020, pp. 3973–3983.
[130] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, and D. Ramage, "Federated evaluation of on-device personalization," 2019, *arXiv:1910.10252*. [Online]. Available: http://arxiv.org/abs/1910.10252
[131] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," 2017, *arXiv:1703.03400*. [Online]. Available: http://arxiv.org/abs/1703.03400
[132] M. Khodak, M. F. Balcan, and A. Talwalkar, "Adaptive gradient-based meta-learning methods," 2019, *arXiv:1906.02717*. [Online]. Available: https://arxiv.org/abs/1906.02717
[133] D. Li and J. Wang, "FedMD: Heterogenous federated learning via model distillation," 2019, *arXiv:1910.03581*. [Online]. Available: http://arxiv.org/abs/1910.03581
[134] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," 2019, *arXiv:1912.00818*. [Online]. Available: http://arxiv.org/abs/1912.00818
[135] F. Hanzely and P. Richtárik, "Federated learning of a mixture of global and local models," 2020, *arXiv:2002.05516*. [Online]. Available: http://arxiv.org/abs/2002.05516
[136] C. Yang, S. Wang, Q. Wang, K. Bian, M. Xu, and X. Liu, "Heterogeneity-aware federated learning," 2020, *arXiv:2006.06983v1*. [Online]. Available: http://arxiv.org/abs/2006.06983v1
[137] V. W. Anelli, Y. Deldjoo, T. Di Noia, and A. Ferrara, "Towards effective device-aware federated learning," in *Proc. Int. Conf. Italian Assoc. Artif. Intell.*, 2019, pp. 477–491, doi: 10.1007/978-3-030-35166-3_34.
[138] Z. Xu, Z. Yang, J. Xiong, J. Yang, and X. Chen, "ELFISH: Resource-aware federated learning on heterogeneous edge devices," 2019, *arXiv:1912.01684v1*. [Online]. Available: http://arxiv.org/abs/1912.01684v1
[139] D. Rothchild, A. Panda, E. Ullah, N. Ivkin, I. Stoica, V. Braverman, J. Gonzalez, and R. Arora, "FetchSGD: Communication-efficient federated learning with sketching," 2020, *arXiv:2007.07682*. [Online]. Available: http://arxiv.org/abs/2007.07682
[140] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization," 2019, *arXiv:1909.13014*. [Online]. Available: http://arxiv.org/abs/1909.13014
[141] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," 2015, *arXiv:1510.00149*. [Online]. Available: http://arxiv.org/abs/1510.00149
[142] P. Richtárik and M. Takáč, "Distributed coordinate descent method for learning with big data," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2657–2681, 2016.
[143] S. Zhang, A. Choromanska, and Y. Lecun, "Deep learning with elastic averaging SGD," 2015, *arXiv:1412.6651*. [Online]. Available: https://arxiv.org/abs/1412.6651
[144] B. S. Kašin, "Diameters of some finite-dimensional sets and classes of smooth functions," *Math. USSR-Izvestiya*, vol. 11, no. 2, pp. 317–333, Apr. 1977, doi: 10.1070/IM1977v011n02ABEH001719.
[145] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi, "Don't use large mini-batches, use local SGD," 2018, *arXiv:1808.07217*. [Online]. Available: http://arxiv.org/abs/1808.07217
[146] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein, "Poison frogs! Targeted clean-label poisoning attacks on neural networks," 2018, *arXiv:1804.00792*. [Online]. Available: https://arxiv.org/abs/1804.00792
[147] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying vulnerabilities in the machine learning model supply chain," 2017, *arXiv:1708.06733*. [Online]. Available: http://arxiv.org/abs/1708.06733
[148] M. Fang, X. Cao, J. Jia, and N. Z. Gong, "Local model poisoning attacks to byzantine-robust federated learning," 2019, *arXiv:1911.11815*. [Online]. Available: http://arxiv.org/abs/1911.11815
[149] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," 2018, *arXiv:1807.00459*. [Online]. Available: http://arxiv.org/abs/1807.00459
[150] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 691–706, doi: 10.1109/SP.2019.00029.
[151] A. Pyrgelis, C. Troncoso, and E. de Cristofaro, "Knock knock, who's there? Membership inference on aggregate location data" 2017, *arXiv:1708.06145*. [Online]. Available: https://arxiv.org/abs/1708.06145, doi: 10.14722/ndss.2018.23183.
[152] L. Zhu and S. Han, "Deep leakage from gradients," in *Federated Learning*. Cham, Switzerland: Springer, 2020, pp. 17–31.

[153] B. Zhao, K. R. Mopuri, and H. Bilen, "IDLG: Improved deep leakage from gradients," 2020, *arXiv:2001.02610*. [Online]. Available: http://arxiv.org/abs/2001.02610
[154] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the GAN: Information leakage from collaborative deep learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2017, pp. 603–618, doi: 10.1145/3133956.3134012.
[155] I. I. Eliazar and I. M. Sokolov, "Measuring statistical heterogeneity: The Pietra index," *Phys. A, Stat. Mech. Appl.*, vol. 389, no. 1, pp. 117–125, Jan. 2010, doi: 10.1016/j.physa.2009.08.006.
[156] L. Su and J. Xu, "Securing distributed gradient descent in high dimensional statistical learning," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 47, no. 1, pp. 83–84, Dec. 2019, doi: 10.1145/3376930.3376983.
[157] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, 2020, doi: 10.1109/TIFS.2020.2988575.
[158] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, "A hybrid approach to privacy-preserving federated learning," in *Proc. 12th ACM Workshop Artif. Intell. Secur. (AISec)*, 2019, pp. 1–11, doi: 10.1145/3338501.3357370.
[159] H. Takabi, E. Hesamifard, and M. Ghasemi, "Privacy preserving multiparty machine learning with homomorphic encryption," in *Proc. 29th Annu. Conf. Neural Inf. Process. Syst.*, 2016.
[160] B. D. Rouhani, M. S. Riazi, and F. Koushanfar, "DeepSecure: Scalable provably-secure deep learning," 2017, *arXiv:1705.08963*. [Online]. Available: http://arxiv.org/abs/1705.08963
[161] J. Stremmel and A. Singh, "Pretraining federated text models for next word prediction," 2020, *arXiv:2005.04828*. [Online]. Available: http://arxiv.org/abs/2005.04828
[162] F. Hartmann, S. Suh, A. Komarzewski, T. D. Smith, and I. Segall, "Federated learning for ranking browser history suggestions," 2019, *arXiv:1911.11807*. [Online]. Available: http://arxiv.org/abs/1911.11807
[163] S. Li, Y. Cheng, Y. Liu, W. Wang, and T. Chen, "Abnormal client behavior detection in federated learning," 2019, *arXiv:1910.09933*. [Online]. Available: http://arxiv.org/abs/1910.09933
[164] J. Lee, J. Sun, F. Wang, S. Wang, C. H. Jun, and X. Jiang, "Privacy-preserving patient similarity learning in a federated environment: Development and analysis," *J. Med. Internet Res.*, vol. 6, no. 2, p. e7744, 2018, doi: 10.2196/medinform.7744.
[165] R. Shao, P. Perera, P. C. Yuen, and V. M. Patel, "Federated face anti-spoofing," 2020, *arXiv:2005.14638v1*. [Online]. Available: http://arxiv.org/abs/2005.14638v1
[166] Y. Liu, A. Huang, Y. Luo, H. Huang, Y. Liu, Y. Chen, L. Feng, T. Chen, H. Yu, and Q. Yang, "FedVision: An online visual object detection platform powered by federated learning," 2020, *arXiv:2001.06202*. [Online]. Available: http://arxiv.org/abs/2001.06202
[167] A. M. Elbir and S. Coleri, "Federated learning for vehicular networks," 2020, *arXiv:2006.01412v1*. [Online]. Available: http://arxiv.org/abs/2006.01412v1
[168] W. Y. Bryan Lim, J. Huang, Z. Xiong, J. Kang, D. Niyato, X.-S. Hua, C. Leung, and C. Miao, "Towards federated learning in UAV-enabled internet of vehicles: A multi-dimensional contract-matching approach," 2020, *arXiv:2004.03877*. [Online]. Available: http://arxiv.org/abs/2004.03877
[169] P. Kairouz *et al.*, "Advances and open problems in federated learning," 2019, *arXiv:1912.04977*. [Online]. Available: https://arxiv.org/abs/1912.04977
[170] Y. Jiang, J. Konečný, K. Rush, and S. Kannan, "Improving federated learning personalization via model agnostic meta learning," 2019, *arXiv:1909.12488*. [Online]. Available: http://arxiv.org/abs/1909.12488
[171] N. Guha, A. Talwalkar, and V. Smith, "One-shot federated learning," 2019, *arXiv:1902.11175*. [Online]. Available: http://arxiv.org/abs/1902.11175
[172] N. Guha and V. Smith, "Model aggregation via good-enough model spaces," 2018, pp. 1–21, *arXiv:1805.07782*. [Online]. Available: http://arxiv.org/abs/1805.07782
[173] A. M. Abdelmoniem and M. Canini, "Towards mitigating device heterogeneity in federated learning via adaptive model quantization," in *Proc. 1st Workshop Mach. Learn. Syst. EuroMLSys*, 2021, pp. 96–103, doi: 10.1145/3437984.3458839.
[174] J. Li, M. Khodak, S. Caldas, and A. Talwalkar, "Differentially private meta-learning," 2019, pp. 1–18, *arXiv:1909.05830*. [Online]. Available: http://arxiv.org/abs/1909.05830

K. M. JAWADUR RAHMAN (Graduate Student Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from North South University. He is currently pursuing the M.Sc. degree in computer science and engineering with United International University. His research interests include privacy-preserving machine learning, federated learning, document layout analysis, and handwriting recognition. He has four years of experience as a Data Science Practitioner in the software industry. His honors thesis was on short-term electrical load forecasting using artificial neural networks and self-organizing feature maps.

FAISAL AHMED received the B.Sc. degree in CSE from the University of Chittagong. He is currently pursuing the M.S. degree in CSE with United International University (UIU), Dhaka, Bangladesh. He is also a Lecturer with the CSE Department, Premier University, Chittagong, Bangladesh. He has some good publications in both international conferences and journals. His research interests include machine learning, deep learning, expert systems, natural language processing, computer vision, and bioinformatics.

NAZMA AKHTER is currently pursuing the degree with the Department of Computer Science and Engineering (CSE), United International University (UIU), Bangladesh. She also works as a Lecturer with the Department of Computer Science and Engineering, Premier University Chittagong (PUC), Bangladesh. Her research interests include machine learning, reinforcement learning, networking, and the Internet of Things.

MOHAMMAD HASAN received the B.Sc. degree in computer science and engineering from Premier University, Chattogram, Bangladesh. He is currently pursuing the M.Sc. degree in CSE with United International University (UIU), Dhaka, Bangladesh. He worked as a Teaching Assistant (TA) for about nine months with the CSE Department, Cox's Bazar International University, Cox's Bazar, Bangladesh. He is also working as a Lecturer with the CSE Department, Premier University. His research interests include medical image processing, blockchain intelligence, healthcare systems, natural language processing, and data mining. His undergraduate thesis was in wireless sensor networks.

RUHUL AMIN received the B.Sc. degree in computer science and engineering from United International University, Dhaka, Bangladesh, in 2020, where he is currently pursuing the M.Sc. degree with the Computer Science and Engineering Department. He has publications in several international conferences and journals. His research articles are published in *Bioinformatics* and *Scientific Reports* journals. His research interests include deep learning, bioinformatics, natural language processing, data mining, and digital forensics.

KAZI EHSAN AZIZ received the B.Sc. degree in computer science and engineering from Shahjalal University of Science and Technology (SUST), Bangladesh, in 2016. He is currently pursuing the M.Sc. degree in computer science and engineering with United International University (UIU), Bangladesh. He is also working as a Software Engineer in the industry. His research interests include intelligent transportation systems, computer vision, and audio signal processing.

A. K. M. MUZAHIDUL ISLAM (Senior Member, IEEE) received the M.Sc. degree in computer science and engineering from Kharkiv National University of Radio Electronics, Ukraine, and the D.Eng. degree in computer science and engineering from Nagoya Institute of Technology, Japan. From January 2011 to January 2017, he worked as a Senior Lecturer with Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM), Malaysia. He worked as an Associate Professor and the Head of the Department of Computer Science and Engineering (CSE), University of Liberal Arts Bangladesh (ULAB). He is currently a Professor with the CSE Department, United International University (UIU), Bangladesh. He has published 90 international research publications (including two book chapters, 26 peer-reviewed indexed journals, and 62 conference papers). He has secured several National and International Research Grants and supervised many Ph.D., master's, and B.Eng. students through their graduation. He is actively involved with BAETE's accreditation process. His research interests include network architecture, communication protocol, cognitive radio networks, wireless sensor networks, the IoT, and cloud computing, healthcare, and smart farming. He is a fellow of IEB (FIEB). He has also served as the Program Chair for ICAICT 2016 and 2020, ETCCE 2020, and 2021 International Conferences. He has also served as the Secretariat for ICaTAS 2016 International Conference, Malaysia, and the 7th AUN/SEED-Net 2014 International Conference on EEE. He is a Chartered Engineer (C.Eng.).

MD. SADDAM HOSSAIN MUKTA (Member, IEEE) received the Ph.D. degree from the Data Science and Engineering Research Laboratory (DataLab), BUET, in 2018. He is currently an Assistant Professor and an Undergraduate Program Coordinator with the Department of Computer Science and Engineering (CSE), United International University (UIU), Bangladesh. He also works as the Lead NLP Researcher in a project funded by the Information and Communication Technology Division (ICTD), Ministry of Posts, Telecommunications and Information Technology, Bangladesh. He has a number of quality publications in both national and international conferences and journals. His research articles have been published in top ranking journals and conferences such as ICWSM, ASONAM, SocInfo, SNAM, *JASIST*, IEEE Access, and *ACM TiiS*. His research interests include social network analysis and mining, social computing, data mining, and machine learning.

A. K. M. NAJMUL ISLAM received the M.Sc. degree in engineering from Tampere University of Technology, Finland, and the Ph.D. degree in information systems from the University of Turku, Finland. He is currently an Adjunct Professor with Tampere University, Finland. He is also a Scientist with the LUT School of Engineering Science, LUT University, Finland. He also works as a University Research Fellow with the Department of Future Technologies, University of Turku, Finland. He has more than 80 publications. His research has been published in top outlets, such as IEEE Access, *Information Systems Journal*, *Journal of Strategic Information Systems*, *European Journal of Information Systems*, *Technological Forecasting and Social Change*, *Computers in Human Behavior*, *Journal of Medical Internet Research*, *Internet Research*, *Computers & Education*, *Information Technology & People*, *Telematics & Informatics*, *Journal of Retailing and Consumer Research*, *Communications of the AIS*, *Journal of Information Systems Education*, *AIS Transactions on Human-Computer Interaction*, and *Behaviour & Information Technology*, among others. His research interest includes human centered computing.