Preserving Data Privacy via Federated Learning: Challenges and Solutions

Zengpeng Li
Qingdao University

Vishal Sharma
Soonchunhyang University

Saraju P. Mohanty
University of North Texas

***Abstract*—Data have always been a major priority for businesses of all sizes. Businesses tend to enhance their ability in contextualizing data and draw new insights from it as the data itself proliferates with the advancement of technologies. Federated learning acts as a special form of privacy-preserving machine learning technique and can contextualize the data. It is a decentralized training approach for privately collecting and training the data provided by mobile devices which are located at different geographical locations. Furthermore, users can benefit from obtaining a well-trained machine learning model without sending their privacy-sensitive personal data to the cloud. This paper focuses on the most significant challenges associated with the preservation of data privacy via Federated Learning. Valuable attack mechanisms are discussed, and associated solutions are highlighted to the corresponding attack. Several research aspects along with promising future directions and applications via federated learning are additionally discussed.**

I. Introduction

As the driving force of human life, data lead to a new wave of a scientific and technological revolution. With the analysis of consumer data, companies predict customers' needs, formulate the consumption circle, ensure the greater interests of marketing to make the marketing smoother. Unconsciously, Artificial Intelligence (AI) has penetrated every aspect of our lives, which prompts data analytics to become a powerful field for helping enterprises identify opportunities and avoid risks.

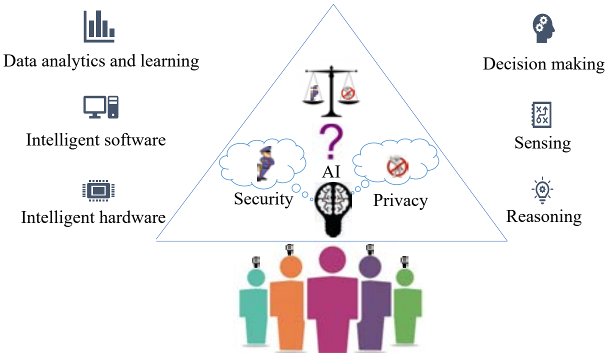

Figure 1: An exemplary illustration of data and privacy dilemma in AI.

Some companies even build a professional data analyst platform, providing efficient data solutions for other institutions. However, data seem like a "double-edged sword", which also brings a variety of personal information leakage risks if customer, industrial or public data are not properly used and maintained [1] [2]. Therefore, many organizations have emphasized data security and deployed new strategies to handle the variation in the types of data. For example, the European Union's General Data Protection Requirements (GDPR) and the United States' California Consumer Privacy Act (CCPA) successively lay out the rules to strengthen the protection of personal data and privacy by standardizing the behavior of enterprises [3]. Notably, CCPA claims consumers have the right to instruct companies to stop selling their personal information to third parties, which is a stricter restriction on the sharing of individual information for business purposes.

II. Data and Privacy Dilemma in AI

The "data islands" dilemma and the emphasis on data privacy and security, as shown in Figure 1, are

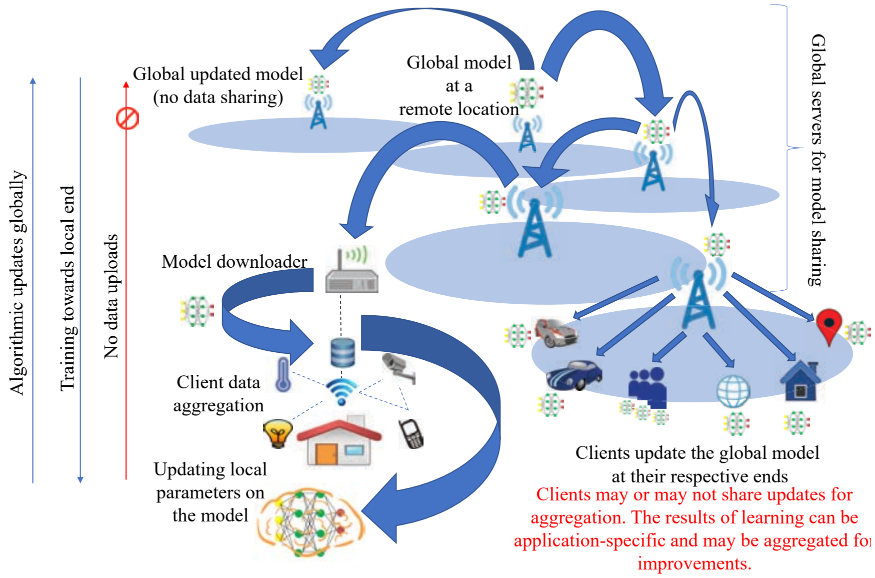
Algorithmic updates globally
Training towards local end
No data uploads
Global updated model (no data sharing)
Global model at a remote location
Global servers for model sharing
Model downloader
Client data aggregation
Updating local parameters on the model
Clients update the global model at their respective ends
Clients may or may not share updates for aggregation. The results of learning can be application-specific and may be aggregated foı improvements.

Figure 2: An exemplary illustration of the practical usage of federated learning in modern networks.

the two new challenges that AI is facing [3]. These two challenges followed by the reason why federated learning, which is the decentralized training approach with features and applications shown in Figures 2 and 3, is suited for their resolution are discussed below:

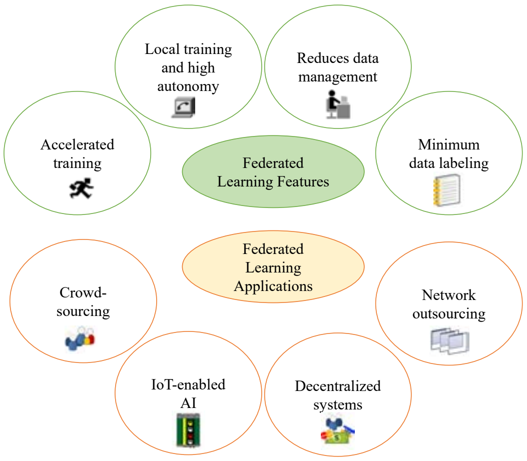

Figure 3: An illustration of features and applications of federated learning.

A. Data Islands Dilemma

AI has experienced certain low points in its development process which are resultant because of the lack of excellent algorithms and computing power. Driven by the wave of big data, AI has reached the next evolving peak, which is the big data-driven AI instances that were supposed to appear in various industries. While things go athwart, “data isolated islands” means data is stored, maintained, and isolated from each other in different departments. In most cases, “data isolated islands” is a big challenge to integrate the data scattered in various organizations, and probably at a huge cost.

B. Privacy-preserving Dilemma

With the development of big data, it has become a global consensus to focus on data privacy and security. Once the data is leaked, it may not only endanger individuals’ privacy but also cause social panic. However, driven by economic advantages, companies usually capture customer data from many sources, such as asking customers directly, tracking customers, and appending other sources of customer data to their own. Then the data are analyzed and turned into knowledge. In the era of big data, the behavior of individuals on the Internet is precipitated into data, and the collection of these data may eventually lead to the disclosure of personal privacy. In terms of the frequent incidents of personal data leakage, personal data rights and institutional data rights are not equal, where consumers

Table I: A comparison of private distributed deep learning methods (R1: Hyperparameters revealed, R2: Bandwidth, R3: Synchronization updates).

Distributed Method	Unique Characteristics	Hybrid approaches	Privacy-Efficiency Trade-Offs		
			R1	R2	R3
Federated Learning	Huge number of clients Non-identical distributions Unbalanced number of samples Slow and unstable communication	Differential Privacy Homomorphic Encryption Oblivious Transfer Garbled Circuits	Yes	Medium	Client-server
Split Learning	No sharing raw data Training deep networks Large batch synchronous stochastic gradient descent	Partial/Full Leakage	No	Low	Client-server
Large Batch Synchronous SGD	large-scale training data Utilizing the full power of the hardware Reducing the communication Mitigating the effect of stragglers	Partial/Full Leakage	Yes	High	Backup workers to compensate slow machines

are passive while enterprises are active. Such issues can be resolved through strict data privacy regulations. As traditional machine learning exposes more and more of its drawbacks, finding new and secure effective ways to collect data becomes crucial. Various privacy-preserving enhancement techniques and privacy-preserving machine learning solutions should be proposed in succession.

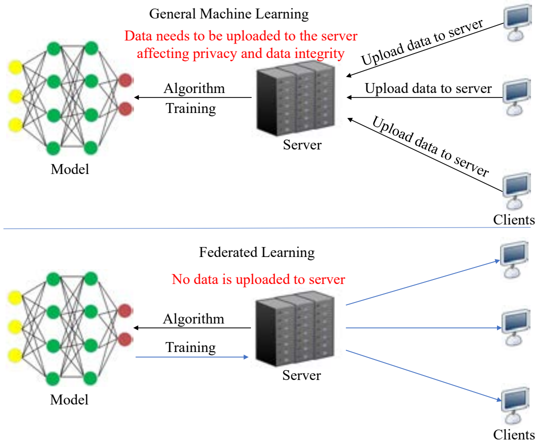

Figure 4: Difference between Machine Learning and Federated Learning.

III. Preliminaries of Federated Learning

Compared with traditional machine learning using centralized approaches, federated learning is a decentralized training approach (e.g., split learning and large-batch synchronous Stochastic Gradient Descent (SGD), etc.) which enables smartphones located at different geographical locations to collaboratively learn a machine learning model while keeping all the personal data that may contain private information on the device. The existing federated learning can be classified into three types, namely, horizontal (or sample-based) federated learning, vertical (or feature-based) federated learning and federated transfer learning [3]. Vertical federated learning and federated transfer learning have similar types of protocols- both involve at least two participants and can be used for privacy-preserving machine learning algorithms.

Table I explains the differences between the decentralized training approaches. Meanwhile, Figure 4 is introduced to depict the differences between general machine learning and federated learning. In a nutshell, federated learning inherits most of the features of the general machine learning with a difference of decentralized training. Another difference is that federated learning maintains the users' privacy by not uploading sensitive data to a centralized server, which is only used for sharing global updates. This feature also increases efficiency by decentralizing the training process to many devices. The requirements and architecture of federated learning are briefly introduced in the next subsection.

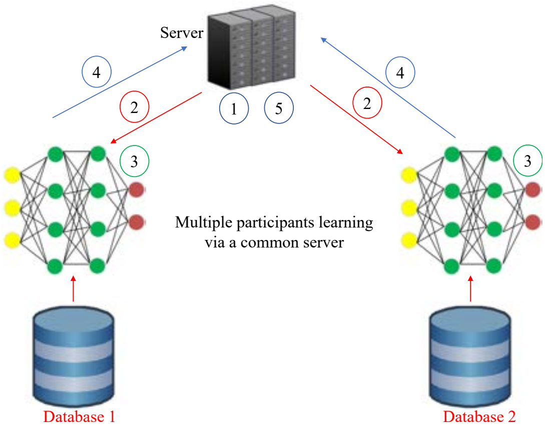

Figure 5: Federated Learning System (Horizontal) [3].

A. *Requirements of Federated Learning*

Federated learning allows designing machine learning systems without direct access to the training data. Similar to the evolution of computing, from mainframes to client-server setups, federated learning decentralizes the machine learning with privacy by default. The key features of federated learning are, **1)** Performance improves with more data. **2)** Models can be meaningfully combined. **3)** Edge devices can train models locally.

B. *System Architecture of Federal Learning*

In federal learning, each edge device trains the model with its data locally and sends the small update to the central server. A horizontal federated learning technique [3] is taken as an example, shown in Figure 5, with details as follows:

1) Train global model in the server.
2) Deploy global model to edge devices.
3) Optimize model from each edge device.
4) Upload locally trained model update.
5) Average the update values and apply the average to the global model.
6) Repeat step 2 to step 5.

The updates in the model contain the parameters and corresponding weights, and all these updates from various users are then averaged to improve the shared global model.

C. *Two Approaches of Sending Updates*

Sending the update to the server is a stepping-stone of federated learning to success. Currently, there are mainly two ways of attaining this: Federated Stochastic Gradient Descent (FedSGD) and Federated Averaging (FedAvg).

FedSGD. FedSGD is inspired by SGD, which is a well-established approach in the field of statistical optimization. FedSGD is an extended SGD that assumes there are k participants P_j $(j \in [1, k])$ of the training data, and n elements in the input data while forming the global objective function. When FedSGD is to be used, each edge device needs to send gradients or parameters to the server which averages gradient or parameters and applies to new parameters. Note that the FedSGD is naive than FedAvg but needs frequent communication between devices and servers.

FedAvg. In FedSGD, each client performs gradient descent on the deployed model by using the local data, then the server calculates the average of the resulting models. The FedAvg is designed by adding more computation to each client. Specifically, FedAvg iterates the local update multiple times before the averaging step. Different from FedSGD, FedAvg enables each edge device to train and update parameters by using gradient descent iteratively. Therefore, even though FedAvg has a higher requirement for the edge devices, it results in better performance than FedSGD.

IV. Challenges and Solutions

Federated learning plugs the most obvious and gaping security issues in distributed machine learning by leaving the training data at its source. It protects the privacy of user-data in different ways for various situations, such as by using differential privacy and homomorphic encryption. Many researchers [3] have contributed to a better understanding of the challenges to be considered with a primary focus on efficiency and accuracy.

A. *Challenge: How to hide updates?*

In federated learning, only global updates are sent to the the central server. However, the cloud is not trusted and still allows to steal sensitive information from the data owners. For example, Phong et. al. [4] demonstrated that even a small portion of the gradients obtained by the malicious-cloud, useful information leaked by these portions are still enough to be exploited by malicious-cloud. The attack usually increases neurons and the noise in the model.

Solutions: Fully Homomorphic Encryption (FHE) is an elegant solution for this challenge, and it aims to preserve the structure of ciphers such as that addition and multiplicative operations can be performed after the encryption. All operations in a neural network except for activation functions are sum and product operations which can be encoded using FHE. Activation functions are approximated with either higher degree polynomials, Taylor series, standard or modified Chebyshev polynomials that are then implemented as part of homomorphic encryption schemes. In practice, FHE seems theoretical, and additively homomorphic encryption [5] are widely used to evaluate non-linear functions in machine learning algorithms that require balancing the trade-offs between data privacy and prediction accuracy. Recently, Phong et al. [4] built an enhanced system to guarantee that no information is leaked to the server. Inspired by [4], all asynchronous stochastic gradients can be encrypted using the somewhat homomorphic encryption and stored on the cloud server. Then, the encrypted gradients can be applied to neural networks, where homomorphic properties (addition and multiplication) enable the computation across the gradients.

B. *Challenge: How to optimize communication and computation complexity?*

In federated learning, to predict the next word for a smartphone user when a he/she is composing a message is one of the classical scenarios.

The main reason is that mobile devices have only sporadic access to power and network connectivity. Additionally, it is difficult to establish direct and stable communication channels among mobile devices, and authenticate locally other devices that are in-charge by the service provider. Thus, how to reduce communication and computational overheads decide whether federated learning can be employed in practice while settling the trade-offs between power consumption and local training.

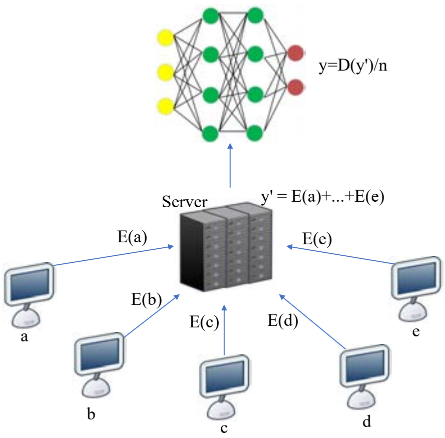

Figure 6: Aggregation of Vertical Federated Learning via Homomorphic Encryption [4].

Solutions: Bonawitz et al. [6] discussed the problem of computing a multiparty sum in federated learning by leveraging the spirit of *secure aggregation* protocol. Inspired by the work of [6], it is concluded that multi-party computing (MPC) and FHE are two important approaches to federated learning, and the above-mentioned challenge in federated learning can be solved via FHE-based MPC. Specifically, compared with Garbled circuit-based MPC, FHE-based MPC can be executed in limited rounds. Therefore, to reduce the communication and computation overhead, a constant (at most 3) rounds threshold FHE-based MPC protocol can be designed under the common reference string (CRS) model against a semi-honest adversary by combining light-weight cryptographic primitives, e.g., secret sharing, authenticated encryption, and somewhat FHE. Additionally, FHE can guarantee the privacy and confidentiality of the updates, and threshold-FHE guarantees that the approach can tolerate users dropping out of the protocol in the recovery phase (see Figure 6).

C. *Challenge: How to defend inference attacks?*

Irrespective of the promising collaboration via federated learning, some attacks [7] have demonstrated that machine learning models remembered too much that the privacy of the user cannot be protected. An inference attack is one of those attacks. It implies that an attacker can infer sensitive information to which it has no granted access, by using prevailing common knowledge and authorized query results. The overview of inference attacks against collaborative learning is shown in Figure 7. To the most recent, new inference attacks [8], [9] emerge endlessly and show that information about individual training data can also be inferred from the model itself, and the most indirect way requires only the ability to query the model several times.

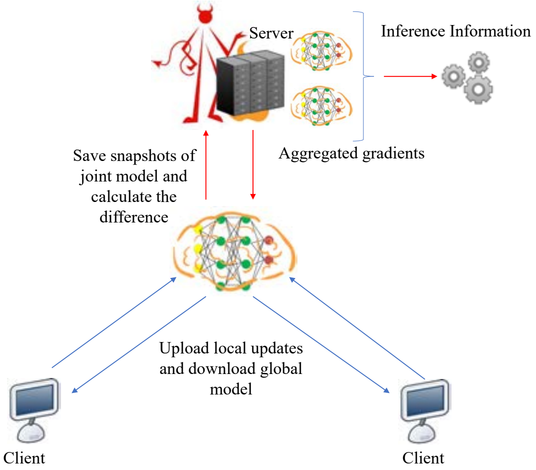
Server
Inference Information
Aggregated gradients
Save snapshots of joint model and calculate the difference
Upload local updates and download global model
Client
Client

Figure 7: An illustration of the Inference Attacks against Collaboration Learning [9].

Notably, Orekondy et al. [8] proposed two linkability attacks against decentralized learning to learning generalizable user-specific patterns in the model updates. This is an identification attack to associate a user profile with a model update and a matching attack to associate two model updates with each other. In addition, Melis et al. [9] designed and evaluated several inference attacks against collaborative learning. The authors showed that an adversary can infer the presence of exact data points leading to the exposure of sensitive information, however, for a certain subset of training data.

Solutions: To address the above-mentioned inference challenge, the most often used method is differential privacy [10] that provides efficient and statistical guarantees against learning for an adversary. The common practice to utilize differential privacy is adding noise to the data to obscure sensitive items such that the other party cannot distinguish the individual's information. Therefore, it is impossible to restore the original data, which means inference attacks become ineffective. Notably, application-specific trade-off between the privacy of the training data and accuracy of the resulting model is an open question, thus, how to choose the parameters (e.g., ε) to control this trade-off is a central issue, but the discussions on this is out of the scope of this paper. As discussed in [9], record-level ε-differential privacy is an elegant approach to constitute an obstacle to the success of membership inference whereas it cannot prevent property inference. To mitigate the risks of linkability attacks, according to various strategies of Orekondy et al. [8], it is required to reduce the distinctiveness in model updates by using calibrated domain-specific data augmentation. Such a technique can provide promising results in achieving privacy with minimal impact to the utility.

D. Challenge: How to prevent model poisoning attacks?

According to this study, a formidable challenge is the possibility of the existence of misbehaving clients introducing backdoor functionality [11], mounting Sybil attacks [12], or label flipping attacks [13] to poison the global model, often named as the poisoning attacks. It is difficult to assert which kind of poisoning attack is the most threatening attack because they happen in different scenarios. Contrary to inference attacks, poisoning attacks happen when the adversary can inject bad data into the model's training pool, and has a chance to learn something it shouldn't. The most common result of a poisoning attack is that the model's boundary shifts in some way (see Figure 8). In fact, Bagdasaryan et al. [11] showed that stealthy backdoor functionality can be introduced into the global model in the federated learning, and designed a new approach based on the model replacement. The idea of this attack is depicted in Figure 9. Specifically, the attacker compromises one or several participants; trains a model on the backdoor data using their new constrain-and-scale technique; submits the resulting model. After federated averaging, the global model is replaced by the attacker's backdoored model.

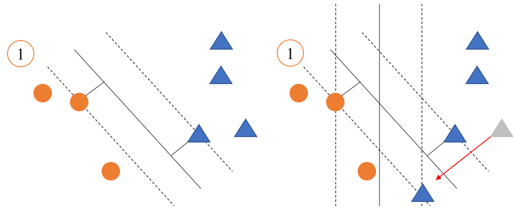

Figure 8: An exemplary illustration of Poisoning Attack [13].

Solutions: There are distinct solutions to prevent model poisoning attacks. Especially, to prevent the backdoor attack, Bagdasaryan et al. [11] is a competitive one, who analyzed and evaluated several defenses to suggest their approach for federated learning by specifically combining anomaly detection, Byzantine-tolerant gradient descent, and participant-level differential privacy. Alongside, to resist Sybil attacks, Fung et al. [14] proposed a new defense approach to federated learning and named it *FoolsGold*. Additionally, to defend model poisoning, Bonawitz et al. [6] suggested using secure aggregation because the updates from each participant are invisible to the aggregator. However, to mitigate known risks, the mentioned solutions just target one particular type of attack that happened at a different place. Thus, it is hard to convince which kind of solution is best. Furthermore, integrating these solutions into an automatic predictable model to prevent poisoning attacks depending on the actual conditions is an open question. Detecting the types of attacks, and determining an accurate solution accordingly, can be a good strategy.

To resist Sybil-based poisoning attacks, one of the known defenses is suggested to assume that the training data can be explicitly observed or clients can be controlled. But how to apply to federated learning for these assumptions is another problem, because the server only touches the updates from each participants' interaction. To prevent backdoor attacks, it seems to be a candidate solution that can keep their backdoor attack into limits but at the expense of sacrificing the model's performance. To prevent data poisoning attacks, the approach of participant-level differential privacy is highly recommended. Specifically, participant-level differential privacy for federated learning relies heavily on two prior works: the FedAvg algorithm which trains deep networks on user-partitioned data, and the moments' accountant of Abadi et al. [15] which provides tight composition guarantees for the repeated application of the Gaussian mechanism combined with amplification-via-sampling. Another feature of participant-level differential privacy is providing a required level of privacy to each participant.

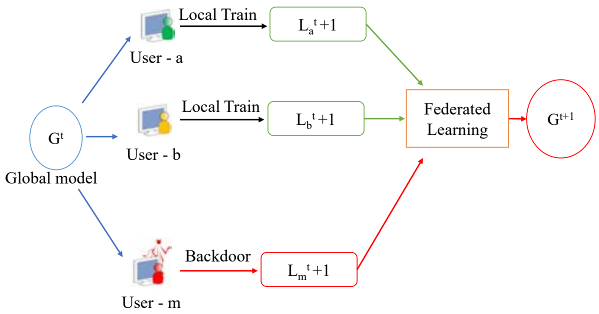

Figure 9: An exemplary illustration of the attacker's backdoored model [11].

V. Promising Research Directions

Federated Learning can be a great fit for the resource-constrained mobile devices, Internet-of-things (IoT), industrial sensor applications, and other privacy-sensitive use cases. Some of the promising open issues for data integrity and privacy through federated learning along with basic research directions are shown in Figure 10.

Applications for protected data including on-device item ranking, next-word prediction, and content suggestion based on federated learning are the major research aspects. Recently, Google released its first production-level federated learning platform to operate sensitive data in the privacy-preserving ways that covers many federated learning-based applications. However, many trade-offs between performance and security are waiting for us to explore. How to train the data without counting on the computational resources while users need not trade their privacy for better services is a prompt problem. Once solved, an immediate and meaningful application is the computationally inexpensive privacy-preserving for Genome-Wide Association

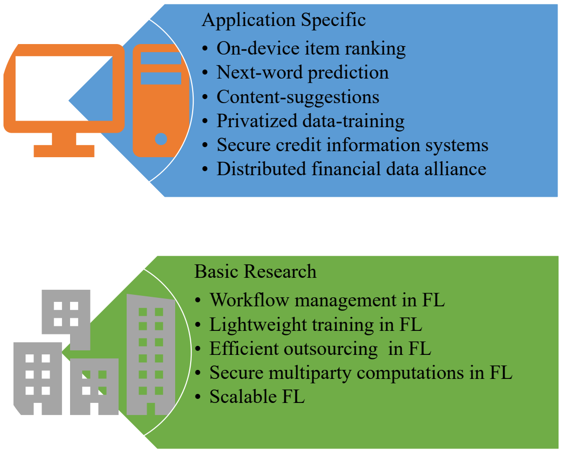
Application Specific
• On-device item ranking
• Next-word prediction
• Content-suggestions
• Privatized data-training
• Secure credit information systems
• Distributed financial data alliance
Basic Research
• Workflow management in FL
• Lightweight training in FL
• Efficient outsourcing in FL
• Secure multiparty computations in FL
• Scalable FL

Figure 10: An illustration of the research directions for application-specific as well as general FL.

Study (GWAS) and smart healthcare armed with FHE under decentralized settings.

Federated learning can be used in the construction of smart cities. For example, various government agencies have established different information systems or data platforms, and large enterprises, especially state-owned enterprises, have accumulated a variety of massive data. To construct a smart city, establishing a credit information system is an important milestone in the process, but it requires to be jointly completed by the data of the joint government and large enterprises. In particular, taking into account the confidentiality of government and business data, the use of federal learning in the joint modeling between government and enterprises can establish a complete credit system.

Furthermore, federated learning for finance applications via FHE-based MPC techniques is another research direction [3]. In particular, the financial industry can form a financial data alliance that needs the collaborative effort of all financial institutions. However, one of the important obstacles is that no one wants to share his/her data in an unrequited way while he/she also would like to collaborate with other financial institutes. Hence, how to collaborate while keeping personal sensitive information by using the FHE-based MPC protocol can be an important direction to follow.

Besides, with modern networking of 5G and beyond, edge-cloud integration can certainly help the easier deployment of federated learning mechanisms. However, with the availability of different functions from the 5G or beyond, it is required to decide the location of the servers as well as plan for a function that will accommodate the global updates. Moreover, aspects of authentication also need to consider when the initial model is exchanged for localized operations.

VI. Conclusions

Federated learning is revolutionizing the way machine learning models are trained. In this paper, the existing challenges in federated learning are investigated and details of the corresponding solutions are additionally provided for each problem. Several solutions for the associated challenges in federated learning are discussed, such as how to hide updates, how to optimize communication and computation complexity, how to defend inference attacks, and how to prevent model poisoning attacks. The discussions in terms of generalized methods can be followed to build fully-fledged solutions for resolving the privacy-protection of data via federated learning. Some applications related to IoT ecosystem, genome-studies, smart city, and finance application via federated learning can be the candidate areas for future works.

Acknowledgment

This research was supported by the National Natural Science Foundation of China under Grant No. 61802214, the National Natural Science Foundation of Shandong Province, China under Grant No. ZR2019BF009, and the Applied Basic Research Project of Qingdao under Grant No.19-6-2-6-cg. At the time of writing, the first author was affiliated with Lancaster University, UK.

References

[1] D. A. Hahn, A. Munir, and S. P. Mohanty, “Security and privacy issues in contemporary consumer electronics [energy and security],” *IEEE Consumer Electronics Magazine*, vol. 8, no. 1, pp. 95–99, December 2018.

[2] W. Z. Khan, M. Y. Aalsalem, M. K. Khan, and Q. Arshad, “Data and privacy: Getting consumers to trust products enabled by the internet of things,” *IEEE Consumer Electronics Magazine*, vol. 8, no. 2, pp. 35–38, March 2019.

[3] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 12:1–12:19, January 2019.

[4] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Trans. Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, May 2018.

[5] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 79:1–79:35, January 2018.

[6] 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 *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, 2017, pp. 1175–1191.

[7] A. Pyrgelis, C. Troncoso, and E. D. Cristofaro, "Knock knock, who's there? membership inference on aggregate location data," in *Proceedings of the 25th Annual Network and Distributed System Security Symposium, NDSS 2018*, 2018.

[8] T. Orekondy, S. J. Oh, B. Schiele, and M. Fritz, "Understanding and controlling user linkability in decentralized learning," *arXiv Computing Research Repository*, vol. abs/1805.05838, 2018, https://arxiv.org/abs/1805.05838.

[9] L. Melis, C. Song, E. D. Cristofaro, and V. Shmatikov, "Inference attacks against collaborative learning," *arXiv Computing Research Repository*, vol. abs/1805.04049, 2018, https://arxiv.org/abs/1805.04049.

[10] I. Wagner and D. Eckhoff, "Technical privacy metrics: a systematic survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 3, p. 57, July 2018.

[11] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," *arXiv Computing Research Repository*, vol. abs/1807.00459, 2018, https://arxiv.org/pdf/1807.00459.pdf.

[12] A. K. Mishra, A. K. Tripathy, D. Puthal, and L. T. Yang, "Analytical model for sybil attack phases in internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 379–387, February 2018.

[13] R. Tourani, S. Misra, T. Mick, and G. Panwar, "Security, privacy, and access control in information-centric networking: A survey," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 1, pp. 566–600, Firstquarter 2018.

[14] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *arXiv Computing Research Repository*, vol. abs/1808.04866, 2018, https://arxiv.org/abs/1808.04866.

[15] M. Abadi, A. Chu, I. J. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28,*, 2016, pp. 308–318.

About the Authors

Zengpeng Li is with the Information Security Department, Qingdao University, Qingdao, China. He obtained his Ph.D. from Harbin Engineering University, China in 2018. Contact him at: zengpengli@hotmail.com

Vishal Sharma is with the Department of Information Security Engineering, Soonchunhyang University, Asan-si, South Korea. He obtained his Ph.D. from Thapar University, India in 2016. Contact him at: vishal_sharma2012@hotmail.com

Saraju P. Mohanty is the Editor in Chief of the IEEE Consumer Electronics Magazine and Professor in the Department of Computer Science and Engineering (CSE), University of North Texas (UNT), Denton, TX, USA. Contact him at Saraju.Mohanty@unt.edu.