Digital Object Identifier 10.48165/iitmjit.2026.12.1.7

Expanding Federated Learning using Distributed Ledger Technologies for Resilience

Narinder K. Seera[1], Ms. Harsha Aggarwal[2]
[1]Associate Professor, [2]Assistant Professor
[1,2]Institute of Innovation in Technology & Management, Janakpuri, New Delhi
[1]narinderkaur.ipu@gmail.com,, [2]harsha.iitmfaculty@gmail.com

Abstract: federated Learning (FL) has emerged as a distributed platform for machine learning models that ensures users' data privacy, however trained models are vulnerable to challenges such as unreliable clients, single points of failure (server), data poisoning, and trust issues among clients. To address these issues, DLT (Distributed Ledger Technology) offers resilience by providing transparency among clients, decentralized model aggregation, and tamper-proof transaction recording. Integrating DLT with FL not only ensures secure and verifiable model updates but also enhances fault tolerance through consensus mechanisms. This research is an attempt to explore how blockchain-based DLT architecture can strengthen the resilience of trained models by providing security and reliability in heterogeneous environments. The chapter discusses the components of the DLT-based architecture and how resilience is ensured.

Keywords: Federated Learning, Distributed Ledgers, Smart Contracts, Blockchain, Model Aggregation

1. Introduction

Since the inception of machine learning domain, the trained models have laid the foundation of various predictive analysis, classification tasks, regression, image recognition and NLP based tasks. These models are trained using regression, neural networks, naïve bayes etc. These intelligent systems need highly trained and accurate models trained on loss minimization algorithms, such as SGD (Stochastic Gradient Descent Search).Training models for such systems often require large data sets which are mostly geographically or demographically distributed. For centralized data collection and model training, there are certain privacy and security issues that avoid centralized training [1]. An alternative is to get the training done in the distributed manner. However, the availability of data and data preprocessing at distributed platform raise other concerns like model update, aggregation, fault tolerance etc. Specialized ML architectures also known as Federated Learning (FL) [2], propose a feasible solution to train ML models in decentralized network. Recently, FL has emerged as a powerful learning framework where sensitive data over local machines are not shared on the server rather the model is trained locally on independent machines and only model updates are shared.

FL facilitates model learning in a distributed manner such that multiple devices of the network train a global model using their local data independently. It involves the iterative training of local models by different clients, coupled with periodic updates exchanged with a central server. At fixed intervals, the server aggregates updates to train a global model which is then iteratively shared among all clients. This is how federated learning works. The main challenges posed by this novel learning framework includes updating the global model locally by each client while preserving robustness and accuracy of the model, ensuring the security of the local sensitive data and overcoming communication failures in the network. It is important to know that while FL enhances privacy as compared to traditional centralized leaning models, it does not guarantee data leakage during aggregation at server. FL models are prone to attacks, either because of malicious nodes in the network or due to curious servers. To address these concerns in FL, various studies have proposed privacy-preserving algorithms and techniques that ensure the robustness of the models. However, the FL settings impose various constraints on these algorithms. Firstly, the algorithms must avoid unrealistic assumptions and excessive computational costs on the FL system. Secondly, they must ensure the accuracy of the global model, as the efficiency of FL model entirely depends on the quality of the training model and results achieved.

On the other hand, Distributed Ledger Technologies (DLT) [3] are nowadays prevalent since the inception of blockchain and other consensus based systems - programs that use data encryption techniques. DLT offers a secure, transparent, and immutable way to record and verify information on transactions. By amalgamate DLT with FL [4] [5], many critical issues can be handled, such as ensuring the integrity of model updates, providing security to the model against attackers, data leakage etc. Combining these two technologies also present few challenges to the model

developers. This chapter is an attempt to explore how DLT and RFL can work together to create decentralized, secure, and fault-tolerant models or AI systems along with challenges. The chapter covers the fundamentals of DLT and its types, the working of resilient FL systems and how FL fails when resilience is lacking. By the end of this chapter, readers will understand not only the technical synergy between DLT and RFL, but also why this combination holds significant promise for the future of resilient, privacy-preserving and collaborative learning frameworks. The chapter is organized as follows: use cases, evaluation criteria of resilience systems and future research directions in this domain.

2. Background

The term Federated Learning (FL) came into existence after years of research in distributed computing, privacy-preserving machine learning, and efficient algorithms. The idea was formalized because of the growing need for AI models that can learn from distributed and sensitive data.

2.1 Evolution of FL

In the early 2000s, machine learning models were trained on centralized systems which caused difficulties in transferring huge datasets. This led to the idea of distributed machine learning; it became possible to train models over distributed systems without moving datasets. The main focus was on efficiency and not privacy but soon data confidentiality concerns were raised and advancements in privacy-preserving computing laid the foundation for keeping sensitive information safe during computation, hence encryption and differential privacy algorithms were introduced with learning frameworks. In 2016, Google introduced the concept of Federated Learning where devices were allowed to train models on their local data and share only model updates with other nodes in the distributed network. However, the researchers identified the challenges of model updation and aggregation in FL and refined it using various effective techniques that make FL more communication-efficient and successful. Since 2020, researchers began focus on robust and secure FL models that have no impact on learning even when some nodes behave unexpectedly or maliciously due to inference or poisonous attacks. This effort gave birth to a novel term, called ***Resilient Federated Learning (RFL)***, [4] [6] which aims at handling malicious updates, ignoring suspicious contributions, spotting unusual patterns in model updates and ensuring transparency and accountability. RFL can be thought of as an improved version of FL that includes robustness, fault tolerance, and security. It has enhanced features to handle failures, malicious participants, and unreliable networks, which traditional FL does not fully address.
Nowadays, RFL has various use cases in healthcare, finance, and IoT. This is not the end, yet a new beginning towards a future where data remains private, yet collaborative learning in AI systems keeps growing [7]. Figure 8.1 shows the the growth in the domain of RFL since its inception in the year 2000 to the present era.

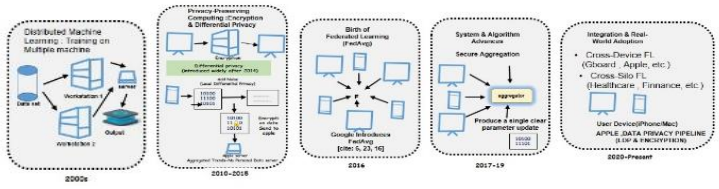

Fig. 7.1: Evolution of Federated Learning

2.2 DLT - A solution to challenges over FL

Distributed ledger technologies (DLT) offer a secure way of recording transactions without the need for a central authority. It is distributed in nature which means all participants in a network share same copies of the ledger. New transactions are added in cryptographically secured manner where all participants validate transactions and update their records which are immutable. The transactions are verified using a consensus protocol – a set of rules which allows participants of a distributed system to agree on a single consistent state of data.
DLT offer various benefits over traditional centralized ledger systems. The distributed nature of DLT makes it more reliable as there is no central point of failure. Therefore, DLT based systems are more resilient to network attacks and less vulnerable to node failures or malicious attacks [8]. Also, because DLT based systems are cryptographic secure, it is very difficult to tamper with or forge records. It encourages the trustworthiness of the data and reduces the risk of fraud.
Though FL offers privacy, it is vulnerable to:

- Malicious updates during model updates or aggregation due to model poisoning

- Single point of failure, because of dependency on a central coordinator
- Lack of trust, security, and auditability.

DLT address these challenges by replacing the central aggregator with decentralized aggregation, where model updates are shared and verified by multiple nodes using a consensus protocol. Each participating node verifies the authenticity of model updates using digital signatures; hence model updates cannot be manipulated, leaked, or falsely attributed. If any malicious node or malicious update is detected, it is possible to trace its origin. Each and every model update is recorded transparently. Nodes don't have to trust a central server, they all agree upon the consensuss [9].

It is important to note that DLT is not the only way to provide resilience to FL systems, it's one of the various available technologies used to secure FL. Resilience in FL can also be achieved using robust aggregation algorithms, anomaly detection on model updates, SMPC (Secure Multi-Party Computation) and homomorphic encryption techniques.

2.3 Challenges of Combining FL and DLT

Though, the Federated Learning has been benefitted with the integration of DLT, however this integration presented not more, a few set of challenges in front of researcher, which need to be addressed. The first challenge is related to the scalability and performance issues. As we know, DLT is not optimized for large data or high-frequency transactions. However, Federated learning requires frequent model updates (e.g., weights or gradients), which can devastate the beauty of traditional blockchain systems (like Ethereum or Bitcoin). The recurring model update scause limited transaction throughput, high latency in consensus mechanisms and costly operations. Second, challenge is concerned with privacy and data Leakage. While FL keeps raw data local, model updates can still leak sensitive information (e.g., via model inversion or gradient leakage attacks).DLTs are transparent by design where nodes can see the transactions, this makes it challenging to hide the shared model updates [10]. This requires additional encryption or differential privacy techniques to be employed to protect the updates. Lastly, FL requires complex consensus and coordination for training the models. Traditional consensus mechanisms like Proof of Work (PoW) or Proof of Stake (PoS) are computationally intensive and do not perform well in real-time FL coordination. Researchers are innovating with some advanced consensus models like PBFT and DAG-based, which tends to be well-suited to FL.

3. Related work

The lack of resilience in Federated Learning has attracted several researchers in past few years. There is a continuous work in this domain done by researchers to improve the features of FL such as scalability, complex consensus and resilience, especially when it is combined with DLT. To achieve resilience in resource-constrained environments, studies by [6] [11-13].have achieved excellent results. But these studies make use of central fusion center to train their intelligent systems. In FL, central fusion node is replaced by local nodes in the network that cause another challenge of heterogeneity of the network. The stragglers cause disruptions in training models and hence few studies [14] focused on how to overcome the problems of stragglers. One solution is to drop straggler node from the network, but it can have important data which may cause baised-model and hence [10] developed FedProx algorithm which allows collecting partial works from stragglers. Contrary, [15] proposed FedMax for local training of model by maximizing the locl activation vectors of all devices in the network, as these vectors directly contribute to the model's accuracy.

For privacy preserving, there exist many studies in the literature that propose solutions based on Differential Privacy (DP) [16] and Homomorphic Encryption (HE) [17-20]. DP employs a cost-effective and lightweight approach of introducing noise to each uploaded model gradient [21- 22]. This degrades model performance in terms of accuracy and does not address the problem of data reconstruction attacks on the model gradients [23]. Homomorphic Encryption allows the server to do aggregations using cipher text, which is another effort to preserve privacy in FL systems. This involves encrypting the updates at the node before it is sent to the server. The server performs model aggregation on cipher text without decrypting it. Once the server shares the encrypted final updated model back to the clients, they can decrypt it to get the finally trained model. This entire process ensures that the server never accesses the raw data, preserving the privacy of individual client contributions throughout the FL process. There are few studies that marks HE-based FL [24 – 25] as a limitation that should be further addressed. Though HE plays an important role in preparing privacy-preserving FL systems, adding a Trusted Third Party (TTP) which creates and distribute cryptographic keys can present another challenge that may affect the use of FL systems in terms of efficiency and security.

4. System components and architecture of resilient fl

As discussed in the previous section, DLT enabled resilient FL offers several benefits in model training, this section introduces its backbone architecture. The architecture constructs a collaborative learning framework that prioritizes security and privacy. There are four main pillars of in this architecture – clients, aggregate servers, DLT layer and smart contract, as shown in the figure 8.2. Accordingly, the goal of this design is to specifically address the main challenges of traditional FL including data privacy, sharing model updates in the absence of centralized trusted third parties, and protection against malicious client nodes in heterogeneous or decentralized networks [26].

4.1 Clients (Edge Nodes)

The clients are the primary nodes in federated learning framework that provide data privacy to the local data. At times, they are the edge devices sometimes the servers at bigger organizations but regardless of that each one employs models on its own private data. The key tasks of client nodes are, first, they train a global model locally on purpose to avoid raw data transmission and privacy risks thereby supporting GDPR, HIPAA and helping with regulations [27]. Second, when the training is over, each client take a guess what has changed – these are models of course or gradients. Third, they use their private keys to electronically sign these updates, which cannot be forged by anyone and retraced later on. 4. When participants sign off their updates, they send them to the aggregator server ONLY at some point these details may be saved by them - as hashes, client IDs or timestamps will get into the blockchain. This is good for transparency and easier tracking is provided by this system. The network is kept perpetually on by edge nodes, tasks that allow clients to join or leave at their wills. The system operates even when connections are unstable or devices are underpowered. Some of the measures put in place includes use of digital signatures and checks which are updated from time to time to prevent such attacks as Sybil or poisoned models [28].

4.2 Aggregator Server (Coordinator)

The central server or aggregator collects the model updates by all the clients to enable privacy-preserving and decentralized model training. Initially the server accepts the connection from all the clients and shares a global model with them. The client trains the model with their local data and sends back the model updates either by sharing gradients or model parameters. The server aggregates the updates to improve the global model which is then send to all the clients for the next round of training. It runs an aggregation algorithm that actually performs the aggregation using one of the various aggregation techniques such as FedProx [10], FedAvg [39], DP-Fed [40] etc. All the model updates are logged at the server for any errors that may occur in future. Certain optimization aggregator algorithms are also capable of handling stragglers in the network that slows down the entire model training. All the logs are shared with blockchain layer which makes the system auditable and tamper-resistant.

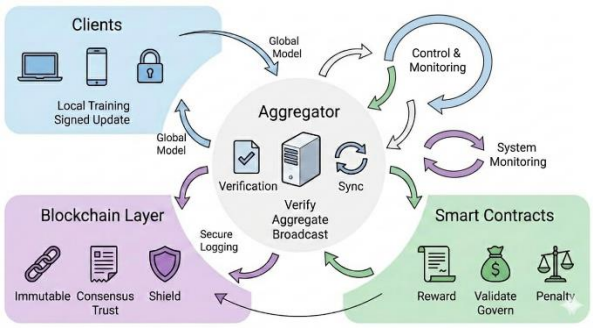

Fig. 7.2: DLT-Enabled Federated Learning

4.3 Blockchain / DLT Layer

This is the trust-building block, and it is especially important when you cannot be certain that everyone is playing by the rules. It does just a few big things. Captures the history of model updates and training cycles in an immutable and permanent way; all hashes are stored on-chain.

1. Establishes credibility: customers can earn trust ratings by how trustworthy and useful their contributions are, and if they adhere to policies.
2. Manage Decentralized Identity: Clients manage identities on their own (PKI or DIDs), therefore even without central authority, responsibilities can be discharged. It is also to make sure that people who upload the updates

can be traced, thus making those behind it to be known. In the consortium members, there is mutual trust-- this is aimed at preventing any one member in particular and therefore ensures network security.

3. There are two classes of blockchain choices for DLT layer – Public Blockchain and Permissioned Ledger. Public blockchains (Ethereum, Polygon) possess high level of transparency, are well-suited to open trustless environment but consume high computation power. Permissioned ledger (Hyperledger Fabric, Corda) can offer better scalability and governance which makes them suitable for private consortium like networks of hospitals. The layered architecture uses decentralized mechanisms for achieving a reputable, strong and verifiable federated learning system.

4.4 Smart Contracts

Smart contracts are basically programs that run themselves on a blockchain. They handle all sorts of tasks- enforcing rules, handling out rewards, running the show- without anyone needing the step in (Ferretti et al,2025). Here's what they actually do:

1. **Participation Validation:** they check if someone's allowed in, usually by looking at cryptographic proofs, past trust scores, or the quality of their data.
2. **Reward Distribution:** They automatically give out tokens, credits, or other rewards based on what you have contributed, like updated data or computing power.
3. **Penalty Enforcement:** if someone acts up or does something fishy, smart contracts can lower their reputation or block them for a while to keep things fair.
4. **Model Checkpointing:** Only models that pass verifications end up in the training ledger, so each round stays solid.

By putting all the decision- making and governance into code, smart contracts cut out any chances of one person gaming the system. Everyone knows the rules, and the system enforces them-no tricks, no hidden moves. This keeps things fair, transparent, and accountable. All these pieces come together to make federated learning more resilient, hence the name, Resilient Federated Learning (RFL). The architectural workflow phases are briefly described below.

1. Training Phase: Clients download the latest global model, train it locally on their own data, sign the update, and send it back to the aggregator.
2. Validation and Aggregation: The aggregator checks the updated and combines them, and comes up with an improved global model.
3. Blockchain Integration: the system records a cryptographic hash of the model and related metadata on the blockchain. Smart contracts update participant reputations and handle rewards.
4. Model Redistribution: the new global model goes back out to clients, starting the whole cycle again.

The benefits of RFL framework are summarized below in the table 8.1.

Table 7.1: Benefits of the RFL framework

Feature	Benefit
Modular design	Ensures compatibility with standard federated learning and blockchain frameworks.
Decentralized trust	Eliminates reliance on a central authority, fostering distributed governance.
Auditability	Enables verifiable training history through immutable blockchain records.
Byzantine fault tolerance	Enhances system resilience against faulty or malicious participants.
Incentive compatibility	Promotes honest and high-quality contributions via incentive alignment
Privacy preservation	Ensures that raw data remains localized to client devices, safeguarding privacy

5. Aggregation in Federated Learning

FL facilitates model learning in a distributed manner such that multiple devices of the network train a global model using their local data independently. It involves the iterative training of local models by different clients, coupled with periodic updates exchanged with a central server. At fixed intervals, this server aggregates all the updates to train a global model which is then iteratively shared among all clients. There are various aggregation algorithms available in federated learning, each having its own advantages and limitations. Aggregation in FL is not simply merging model updates rather it tracks model performance across devices that actually determines the success of model training. Additional statistical indicators such as loss functions or accuracy measurements can also be aggregated. Aggregation can be performed in many ways such as hierarchical aggregating where local models are shared with intermediate servers before sending them to the central server, clipped average aggregation, secure aggregation, and DP average

aggregation, momentum aggregation, weighted aggregation, Bayesian, quantization, adversarial aggregation and many more. The type of aggregation algorithm used by FL system is defined by the goal to be achieved such as data privacy, avoiding updates from malicious clients, increasing the convergence rate etc. This is how federated learning works.

Besides using these diverse aggregation methods, FL systems are prone to attacks and the main challenges posed by this novel learning framework can be noted around security and communication overheads. It includes updating the global model locally by each client while preserving fairness, robustness and accuracy of the model, ensuring the security of the local sensitive data from leakages and poisoning attacks, finding solutions to overcome communication failures in the network. The frequent exchange of model updates to the aggregator poses significant network traffic, which requires certain algorithms to address stragglers in the network such as model compression or sharing asynchronous updates. To address these challenges, researchers are developing novel algorithms for wise aggregation of model updates, client selection strategies to remove malicious nodes from the network and optimization techniques to optimize aggregation process and communication overheads.

6. Conclusions and Future directions

The concept of Resilient Federated Learning marks a milestone in the journey towards trustworthy, secure and collaborative AI systems. AI/ML models are employed in intelligent systems. This chapter highlighted how Distributed Ledger Technology assented to the position of such a system In the field of Federated Learning, Artificial Intelligence (AI) technologies like DLT, especially blockchain can be embedded into FL systems to solve the challenges posed by FL systems such as data poisoning, single point of failures etc. It has the benefit of helping In conjunction with the immutability, transparency and decentralized consensus of the distributed ledger technologies, DLT increases FL resilience. systems from adversarial threats, clients in the network with malicious intents, and server breakdowns. In the The integration of DLT and FL does not only make systems more robust but also trustworthy in the eyes of clients in cooperative system. Yet, implementation of DLT into FL indeed boosts the prospects of obtaining learning outcomes yet an analysis through a critical lens suggest that it might indeed be more problematic--depends on how you view the issue--more deeply divisive depending on the perspective you take. Apart from these, there are other challenges of scalability, energy consumption, and communication overhead which must be addressed ad carefully balanced. RFL and DLT propose these research avenues to be pursued by researchers in this field. It is an industry shift for example in the direction of optimization of consensus algorithms, which can be used to turn into a better decision-making system: faster and more scalable model updates. Yet another factor to consider is designing lightweight ledger architectures [35] for IoT devices with limited resources and the study of Hybrid models, which are built by merging different kinds of DLT like blockchain and DAGs with FL. There are several privacy- preserving techniques, homomorphic encryptions for example, can help in securing data while being processed and later sharing it with other parties. Different levels of security like secret sharing, homomorphic computation, and DP (Differential Privacy) can combine to further secure a system with respect to RFL. Sometimes human scribe will opt to go a step further and ensure resilience in RFL against advanced emerging threats like those in the quantum era. An important part of the puzzle and a key factor in creating trustworthy, scalable and future-ready systems is the use of cryptography based intelligent systems. Along these directions such innovations will be the starting points for scalable, trustworthy and really federated learning systems that are resilient and hence can be the driving force for next-generation health applications, finance and smart cities.

References

1. Peteiro-Barral, D., & Guijarro-Berdiñas, B. (2013). A survey of methods for distributed machine learning. *Progress in Artificial Intelligence*, *2*(1), 1–11. https://doi.org/10.1007/s13748-012-0035-5
2. Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., Rouayheb, S. E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., … Zhao, S. (2021). *Advances and Open Problems in Federated Learning* (No. arXiv:1912.04977). arXiv. https://doi.org/10.48550/arXiv.1912.04977
3. Soltani, R., Zaman, M., Joshi, R., & Sampalli, S. (2022). Distributed Ledger Technologies and Their Applications: A Review. *Applied Sciences*, *12*(15), 7898. https://doi.org/10.3390/app12157898
4. Li, K. (2025). A Blockchain-Integrated Federated Learning Approach for Secure Data Sharing and Privacy Protection in Multi-Device Communication. *Applied Artificial Intelligence*, *39*(1), 2442770. https://doi.org/10.1080/08839514.2024.2442770
5. Cachin, C., & Vukolić, M. (2017). *Blockchain Consensus Protocols in the Wild* (No. arXiv:1707.01873). arXiv. https://doi.org/10.48550/arXiv.1707.01873

6. Imteaj, A., Khan, I., Khazaei, J., & Amini, M. H. (2021). FedResilience: A Federated Learning Application to Improve Resilience of Resource-Constrained Critical Infrastructures. *Electronics*, *10*(16), 1917. https://doi.org/10.3390/electronics10161917
7. Liu, J., Huang, J., Zhou, Y., Li, X., Ji, S., Xiong, H., & Dou, D. (2022). From distributed machine learning to federated learning: A survey. *Knowledge and Information Systems*, *64*(4), 885–917. https://doi.org/10.1007/s10115-022-01664-x
8. Jovanovic, Z., Hou, Z., Biswas, K., & Muthukkumarasamy, V. (2024). Robust integration of blockchain and explainable federated learning for automated credit scoring. *Computer Networks*, *243*, 110303. https://doi.org/10.1016/j.comnet.2024.110303
9. Ferretti, S., Cassano, L., Cialone, G., D'Abramo, J., & Imboccioli, F. (2025). Decentralized coordination for resilient federated learning: A blockchain-based approach with smart contracts and decentralized storage. *Computer Communications*, *236*, 108112. https://doi.org/10.1016/j.comcom.2025.108112
10. Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). *Federated Optimization in Heterogeneous Networks* (No. arXiv:1812.06127). arXiv. https://doi.org/10.48550/arXiv.1812.06127
11. Cai, H.; Lam, N.S.; Qiang, Y.; Zou, L.; Correll, R.M.; Mihunov, V. A synthesis of disaster resilience measurementmethods andindices. Int. J. Disaster Risk Reduct. 2018, 31, 844–855.
12. Pursiainen, C. Critical infrastructure resilience: A Nordic model in the making? Int. J. Disaster Risk Reduct. 2018, 27, 632–641.
13. Alemzadeh, S.; Talebiyan, H.; Talebi, S.; Dueñas-Osorio, L.; Mesbahi, M. Resource Allocation for InfrastructureResilience usingArtificial Neural Networks. In Proceedings of the 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence(ICTAI), Baltimore, MD, USA, 9–11 November 2020; pp. 617–624.
14. Amini, M.H.; Nabi, B.; Haghifam, M.R. Load management using multi-agent systems in smart distribution network. InProceedings of the 2013 IEEE Power & Energy Society General Meeting, Vancouver, BC, Canada, 21–25 July 2013; pp. 1–5.
15. Chen, W.; Bhardwaj, K.; Marculescu, R. Fedmax: Mitigating activation divergence for accurate and communication-efficientfederated learning. arXiv 2020, arXiv:2004.03657.
16. C. Dwork. 'Differential Privacy: A Survey of Results'. In: Theory and Applicationsof Models of Computation. Ed. by M. Agrawal, D. Du, Z. Duan and A. Li. Berlin,Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–19. ISBN: 978-3-540-79228-4.
17. P. Paillier. 'Public-key cryptosystems based on composite degree residuosityclasses'. In: TAMC. Springer. 1999.
18. T. ElGamal. 'A public key cryptosystem and a signature scheme based ondiscrete logarithms'. In: IEEE Transactions on Information Theory 31.4 (1985),pp. 469–472.
19. C. Gentry and D. Boneh. A fully homomorphic encryption scheme. Vol. 20. 9.Stanford University Stanford, 2009.
20. I. Damgård, V. Pastro, N. Smart and S. Zakarias. 'Multiparty Computation fromSomewhat Homomorphic Encryption'. In: CRYPTO. 2012.
21. R. C. Geyer, T. Klein and M. Nabi. 'Differentially private federated learning: Aclient level perspective'. In: arXivpreprint arXiv:1712.07557 (2017).
22. K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quekand H. V. Poor. 'Federated learning with differential privacy: Algorithms andperformance analysis'. In: IEEE Transactions on Information Forensics andSecurity (2020).
23. L. T. Phong, Y. Aono, T. Hayashi, L. Wang and S. Moriai. 'Privacy-PreservingDeep Learning via Additively Homomorphic Encryption'. In: IEEE Transactionson Information Forensics and Security 13.5 (2018), pp. 1333–1345. DOI:10.1109/TIFS.2017.2787987.
24. R. Xu, N. Baracaldo, Y. Zhou, A. Anwar and H. Ludwig. 'HybridAlpha: AnEfficient Approach for Privacy-Preserving Federated Learning'. In: The 12thACM AISec. 2019.
25. C. Zhang, S. Li, J. Xia, W. Wang, F. Yan and Y. Liu. 'Batchcrypt: Efficienthomomorphic encryption for cross-silo federated learning'. In: {USENIX} ATC. 2020, pp. 493–506.
26. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). *Federated Learning with Non-IID Data*. https://doi.org/10.48550/arXiv.1806.00582
27. Brisimi, T. S., Chen, R., Mela, T., Olshevsky, A., Paschalidis, I. Ch., & Shi, W. (2018). Federated learning of predictive models from federated Electronic Health Records. *International Journal of Medical Informatics*, *112*, 59–67. https://doi.org/10.1016/j.ijmedinf.2018.01.007
28. Bhagoji, A. N., Chakraborty, S., Mittal, P., & Calo, S. (2019). *Analyzing Federated Learning through an Adversarial Lens* (No. arXiv:1811.12470). arXiv. https://doi.org/10.48550/arXiv.1811.12470

29. Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., & Suresh, A. T. (2021). *SCAFFOLD: Stochastic Controlled Averaging for Federated Learning* (No. arXiv:1910.06378). arXiv. https://doi.org/10.48550/arXiv.1910.06378
30. Minango, J., Carvajal Mora, H., Zambrano, M., Orozco Garzón, N., & Pérez, F. (2025). Distributed Ledger Technology in Healthcare: Enhancing Governance and Performance in a Decentralized Ecosystem. *Technologies*, *13*(2), 58. https://doi.org/10.3390/technologies13020058
31. Nezhadsistani, N., Moayedian, N. S., & Stiller, B. (2025). Blockchain-Enabled Federated Learning in Healthcare: Survey and State-of-the-Art. *IEEE Access*, *13*, 119922–119945. https://doi.org/10.1109/ACCESS.2025.3587345
32. Prathiba, S. B., Govindarajan, Y., Pranav Amirtha Ganesan, V., Ramachandran, A., Selvaraj, A. K., Kashif Bashir, A., & Reddy Gadekallu, T. (2024). Fortifying Federated Learning in IIoT: Leveraging Blockchain and Digital Twin Innovations for Enhanced Security and Resilience. *IEEE Access*, *12*, 68968–68980. https://doi.org/10.1109/ACCESS.2024.3401039
33. Javed, A. R., Hassan, M. A., Shahzad, F., Ahmed, W., Singh, S., Baker, T., & Gadekallu, T. R. (2022). Integration of Blockchain Technology and Federated Learning in Vehicular (IoT) Networks: A Comprehensive Survey. *Sensors*, *22*(12), 4394. https://doi.org/10.3390/s22124394
34. Ali, M., Karimipour, H., & Tariq, M. (2021). Integration of blockchain and federated learning for Internet of Things: Recent advances and future challenges. *Computers & Security*, *108*, 102355. https://doi.org/10.1016/j.cose.2021.102355
35. Fan, T., Chen, X., Dong, Y., Chen, X., Xuan, Y., & Jing, W. (2024). Lightweight Secure Aggregation for Personalized Federated Learning with Backdoor Resistance. *2024 Annual Computer Security Applications Conference (ACSAC)*, 810–825. https://doi.org/10.1109/ACSAC63791.2024.00071
36. Li, K., Zhang, Z., Pourkabirian, A., Ni, W., Dressler, F., & Akan, O. B. (2025). *Towards Resilient Federated Learning in CyberEdge Networks: Recent Advances and Future Trends* (No. arXiv:2504.01240). arXiv. https://doi.org/10.48550/arXiv.2504.01240
37. Lu, Y., Huang, X., Dai, Y., Maharjan, S., & Zhang, Y. (2020). Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT. *IEEE Transactions on Industrial Informatics*, *16*(6), 4177–4186. https://doi.org/10.1109/TII.2019.2942190
38. Chatterjee, P., Das, D., & Rawat, D. (2023). *Use of Federated Learning and Blockchain towards Securing Financial Services*. https://doi.org/10.36227/techrxiv.22155182.v1
39. Tao Sun, Dongsheng Li, and Bao Wang (2015), "Decentralized Federated Averaging", Journal of Latex Class Files, Vol 14, No 8, arXiv:2104.11375**.** https://doi.org/10.48550/arXiv.2104.11375
40. Huang, X., Ding, Y., Jiang, Z.L. *et al.* DP-FL: a novel differentially private federated learning framework for the unbalanced data. *World Wide Web* **23**, 2529–2545 (2020). https://doi.org/10.1007/s11280-020-00780-4
41. Cao, X., Fang, M., Liu, J., & Gong, N. Z. (2022). *FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping* (arXiv:2012.13995). https://doi.org/10.48550/arXiv.2012.13995
42. Nguyen, J., Malik, K., Zhan, H., Yousefpour, A., Rabbat, M., Malek, M., & Huba, D. (2022). *Federated Learning with Buffered Asynchronous Aggregation* (arXiv:2106.06639). arXiv. https://doi.org/10.48550/arXiv.2106.06639