mathematics

MDPI

Review

Decentralized Federated Learning for Private Smart Healthcare: A Survey †

Haibo Cheng [1,2,3], Youyang Qu [2,3,*], Wenjian Liu [1,*], Longxiang Gao [2,3] and Tianqing Zhu [1]

[1] Faculty of Data Science, City University of Macau, Macau, China; d23092110113@cityu.edu.mo (H.C.); tqzhu@cityu.edu.mo (T.Z.)
[2] Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China; gaolx@sdas.org
[3] Shandong Provincial Key Laboratory of Computing Power Internet and Service Computing, Shandong Fundamental Research Center for Computer Science, Jinan 250014, China
* Correspondence: quyy@sdas.org (Y.Q.); andylau@cityu.edu.mo (W.L.)
† This paper is an extension of a conference paper. Unleashing the Potential of Decentralized Federated Learning in Healthcare: A Comprehensive Survey on Insights and Challenges. In Proceedings of the 2024 IEEE 9th International Conference on Data Science in Cyberspace (DSC), Jinan, China, 23 August 2024.

Abstract: This research explores the use of decentralized federated learning (DFL) in healthcare, focusing on overcoming the shortcomings of traditional centralized FL systems. DFL is proposed as a solution to enhance data privacy and improve system reliability by reducing dependence on central servers and increasing local data control. The research adopts a systematic literature review, following PRISMA guidelines, to provide a comprehensive understanding of DFL's current applications and challenges within healthcare. The review synthesizes findings from various sources to identify the benefits and gaps in existing research, proposing research questions to further investigate the feasibility and optimization of DFL in medical environments. The study identifies four key challenges for DFL: security and privacy, communication efficiency, data and model heterogeneity, and incentive mechanisms. It discusses potential solutions, such as advanced cryptographic methods, optimized communication strategies, adaptive learning models, and robust incentive frameworks, to address these challenges. Furthermore, the research highlights the potential of DFL in enabling personalized healthcare through large, distributed data sets across multiple medical institutions. This study fills a critical gap in the literature by systematically reviewing DFL technologies in healthcare, offering valuable insights into applications, challenges, and future research directions that could improve the security, efficiency, and equity of healthcare data management.

Keywords: decentralized federated learning (DFL); medical data privacy; data security; communication efficiency; heterogeneity; incentive mechanisms; application scenarios

MSC: 68T01; 68P27

Academic Editor: Daniel-Ioan Curiac

Received: 7 March 2025
Revised: 30 March 2025
Accepted: 31 March 2025
Published: 15 April 2025

Citation: Cheng, H.; Qu, Y.; Liu, W.; Gao, L.; Zhu, T. Decentralized Federated Learning for Private Smart Healthcare: A Survey. *Mathematics* **2025**, *13*, 1296. https://doi.org/10.3390/math13081296

1. Introduction

As we stepped into the Healthcare 4.0 era [1], the emergence of the Medical Internet of Things (MIoT) and Internet of Healthcare Things (IoHT) devices has become the focus of attention in the health field [2]. The medical data can be obtained via lightweight physical medical [3] devices, such as wearable sensors distributed at the network's edge to collect patient data. They empower traditional fields such as healthcare, medical care,

and public health. Moreover, the medical data capture systems would generate massive electronic health records (EHRs), and EHRs contain a large amount of high-quality, valuable data for medical institutions. However, EHR management is inherently decentralized, as stakeholders are distributed between patients, medical institutions, and government institutions in some cases [4]. Therefore, there is a problem of data silos among medical institutions, and the privacy of medical data makes institutions reluctant to share their data directly. Additionally, gathering clinical data including symptoms of diseases and medical records from individual medical facilities presents a significant challenge [5,6].

The concept of federated learning (FL) was first proposed by Google in 2016 [7]. It was originally intended to solve the problem of collecting data on multiple Android devices to train a model while avoiding infringement of user privacy. Based on this, the paper "FL of Deep Networks using Model Averaging" was also published. FL uses distributed machine learning and deep learning technology to keep all parties' data locally during the entire training and interaction process. No swapping and merging of original data, only gradient updates [8]. This means that one of the most apparent advantages of FL is its ability to build joint models without sharing raw data. Medical institutions do not need to share data in medical scenarios but can make more accurate predictions or classifications only by providing model parameters. It has the potential to address these challenges by allowing multiple entities to collaboratively train a shared predictive model while preserving the privacy of their respective training data [5,9].

Although FL plays a vital role in the medical field, FL in the traditional scheme with a central server still faces challenges. Transferring raw patient data to a central server may violate privacy laws such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) [10,11]. At the same time, customers may not want to rely on a third party to manage the shared model. Because server-side computations often lack transparency, it is difficult for users to verify the computations performed. Specifically, cloud computing has been argued to decrease users' sovereignty over their data and models [12]. On the other hand, FL needs a trustworthy central server to aggregate the model parameters uploaded via the device and distribute the global model to all devices [13]. During training, updated information about the model is continuously reported to a third-party or central server. The third party can continuously collect data from all participants in different rounds [14]. However, it does not rule out that a malicious central server has the opportunity to analyze and derive the original information and attempts to cause gradient leakage by inferring gradients or destroy the model to conduct white-box attacks, such as member inference attacks [15]; once the central server is crashed by attackers, the FL training stops. The simple FL framework is mainly based on a centralized architecture, which relies heavily on a single central server and is more prone to single points of failure. Another risk centralization brings is server failure or delay due to server overload or an increased number of devices. Highly encrypted parameters are shared to a central server for training, making the FL framework's central node face communication pressure and bandwidth bottlenecks [16–18]. It may also lead to high computing costs, thus limiting the practicality and scalability of this technology [19]. Given the sensitivity of medical data, each hospital wants to pursue an autonomous model for its own regulatory compliance and customize it to its own data, so multi-institutional collaboration in centralized FL faces difficulties in creating a central model [20–22].

To overcome these challenges, the DFL framework proposed as an extension of FL aims to reduce reliance on a central server. DFL provides a novel approach to training global models by facilitating direct client interaction and eliminating the need for a central server [23]. DFL framework enhances privacy protection by minimizing the exposure of data and model information to a central entity. And improved capabilities against single

points of failure [10,23]. By allowing clients to interact directly, DFL reduces network overhead and makes the network more resilient to failures—enhancing efficiency and robustness and reducing overall network communication and computing costs. Additionally, the decentralized FL framework removes significant limitations in regulating FL by allowing model heterogeneity; each participant can have a private model of any architecture [24]. The approach works well with heterogeneous data sources and preserves the model autonomy of each actor [20,25]. Despite these advancements, DFL still faces several challenges (Figure 1). The specific contents are described in detail in Section 4 below.

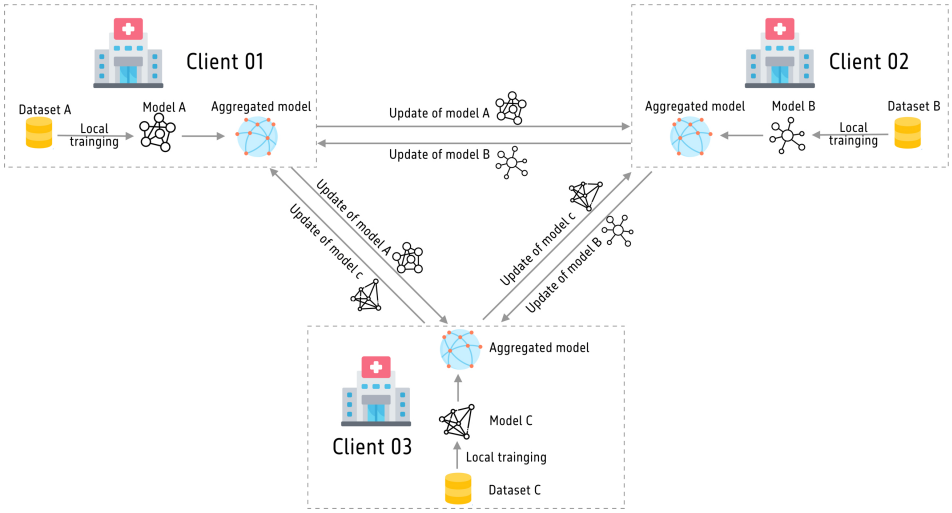

Figure 1. Overall training process for DFL.

This research describes the unique advantages and detailed classification of decentralized architecture in federated learning, comprehensively reviews its applications and challenges in the medical field, and systematically proposes strategies and solutions to different challenges. For example, it combines a decentralized federated learning architecture with new technologies such as differential privacy, blockchain, communication optimization algorithms, and reward mechanisms, providing new ideas and methods for data privacy protection and efficient collaboration in smart healthcare. This research is organized as follows. Section 2 introduces the research methods and PRISMA guidelines. Section 3 proposes a classification framework for decentralized federated learning. Section 4 focuses on analyzing the key challenges and coping strategies. Next, Section 5 presents applications and case studies in smart healthcare. Section 6 discusses future research directions and finally summarizes the contributions and innovations of the paper.

2. Research Method

A thorough examination of DFL in the medical sector within the existing literature needs to be conducted, revealing a research gap in elucidating the foundational tenets of this nascent methodology. Moreover, prior studies have yet to scrutinize DFL from challenging and potential resolution perspectives, neglecting to conduct comparative analyses across diverse medical contexts. Given these insights, this survey formulated a set of research questions (RQs) to explore the existing literature thoroughly, which aims to provide an extensive and structured overview of all papers relevant to DFL in the medical field.

2.1. Research Questions

RQ1: What are the current methods to achieve decentralization in FL in the medical field?

RQ2: What are the main obstacles and challenges faced by DFL when applied in healthcare settings? How is it solved?

RQ3: What are the specific case studies of DFL in the healthcare sector?

2.2. Search Process

We chose to follow the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines to conduct our review. We utilized several common databases, including the IEEE Xplore, Scopus, and Web Of Science libraries, employing advanced search options and Boolean expressions (and, or). The literature search phrases were applied to titles in each database. Specifically, we used the following search queries to retrieve relevant:

"DFL" or "Blockchain-based Learning" or "P2P FL" or "Peer-to-Peer FL".
And "Health" or "Healthcare" or "Medical" or "Medicine" or "Clinical"

2.3. Inclusion and Exclusion Criteria

However, the surveyed query terms return many irrelevant works to this review, which lie outside the present scope or cover completely unrelated topics. Thus, we excluded papers as follows.

- The term "decentralized" mentioned in some articles mainly refers to the decentralized approach to data processing and storage, rather than the architecture without a central server. In such articles, it is mainly emphasized that each participating node (hospital) has storage control of its own data and local models, and data processing is completed locally. The central server only aggregates and trains global models. Although the design of the entire system conforms to the principle of decentralization, it is beyond the scope of this article.
- Describe the centralized implementation of FL and cover a topic other than DFL.
- The discussion solely revolves around the blockchain without integration with FL.
- Discussing DFL in non-medical domains (such as industry, communication, etc.).
- Unrelated papers mistakenly returned via the query (Figure 2).

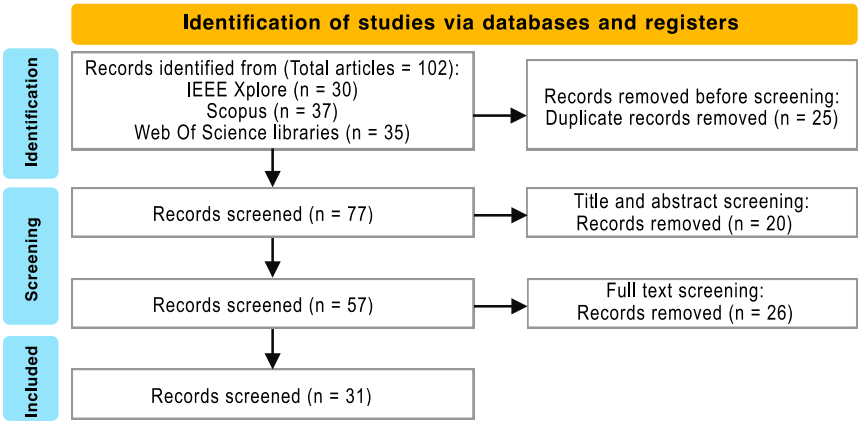

Figure 2. Overall search process for DFL.

3. Fundamentals and Taxonomy

This section outlines a comprehensive taxonomy for DFL based on different approaches, broadly categorized into two classes: traditional distributed computing methods, such as peer-to-peer (P2P) methods, and methods utilizing blockchain technology [26].

3.1. Traditional Distributed Methods (TD-FL)

One solution to cancel centralized federated learning is to adopt traditional distributed computing technologies, such as the decentralized architecture of peer-to-peer (P2P) networks [27], so that participating nodes, such as medical devices, can directly collaborate

to complete model training. There is no clear distinction between client and server roles in a decentralized architecture. Each client has the potential to act as a server, taking turns in each round to aggregate the current global model and transmit it to other clients in a random manner. Specifically depending on the network's configuration and protocols [27,28]. This approach solves the single point of failure and communication bottleneck problems associated with traditional FL star architecture by designing serverless architectures with different network structures. In DFL, the network topology describes the organization of participants and determines their communications. Three network topologies are used in DFL: fully connected networks, partially connected, and node clustering [29].

Fully connected architecture builds a decentralized FL framework characterized by flexibility because every node can communicate with each other directly [30,31]. However, a link must be established with every existing node upon adding a new node. Consequently, the communication overhead escalates with the network's node count. Despite the elevated communication cost, this topology boasts high reliability and robustness. Even in the event of node or link failures, the network can continue to operate effectively [32].

In partially connected architecture, nodes communicate with their two immediate neighbors in the ring topology [33], it can more easily detect inconsistencies or aberrations in the data it receives by comparing incoming data from both directions, which helps isolate malicious node updates. The dynamic structure is another partially connected network in which the interaction and data sharing between nodes is based on the current network conditions, and task requirements change dynamically. This network topology's characteristics are suitable for distributed learning environments that require a high degree of adaptability and flexibility.

3.2. Blockchain-Based Methods (BC-FL)

Blockchain is a peer-to-peer distributed ledger technology and architecture platform that has now developed into a basic technology for various decentralized applications [34]. It consists of multiple blocks linked using cryptographic hash functions. Each contains a set of transactions or records and is cryptographically linked to the previous block. This structure ensures the immutability and integrity of the data because modifying the contents of a block requires modifying all subsequent blocks, which is computationally infeasible [35]. Due to the inherent characteristics of blockchain technology, such as traceability, security, and decentralization [36], it is particularly suitable for the privacy protection of medical data [37]. Another solution to eliminating centralized federated learning is blockchain-based federated learning. By decentralized storage and management of model parameters through blockchain technology, each participant can lead the aggregation process in a specific learning round. This decentralized structure eliminates the need for a central server. Participants are called miners, and they submit locally trained model updates to the blockchain network and verify legitimacy through consensus algorithms such as PoS. The verified parameters are aggregated and calculated according to the model aggregation rules defined by the smart contract to generate a new global model [38]. Blockchain is built through a decentralized system of peer-to-peer trust [39,40]. As a ledger with tamper-proof properties, collective maintenance, and traceability, blockchain can replace the central server to decentralize the coordination process in FL, thus resisting single points of failure and illegal tampering attacks [3,38]. Therefore, blockchain has attracted widespread attention in the medical field, mainly for storing electronic medical records (EMRs) and protecting patient privacy [39,41]. The consensus mechanism of the blockchain network (such as PoW and PoS) selects one or more nodes as aggregation nodes, allowing each client to communicate directly with other clients and no longer rely on a central server for coordination communication and data synchronization between various clients. For

instance, Zhou et al. [42] introduced the combination of FL and blockchain to achieve decentralized operation. They improved the consensus mechanism to filter participating nodes and eliminate those with poor performance to ensure the overall performance of the training model. Smart contracts run on blockchain platforms serve as a centralized mechanism that, instead of a central server, receives model parameters from the distributed ledger, aggregates them, and sends them back to the client's associated ledger, ensuring the reliability and transparency of business execution [43,44].

In the field of blockchain systems, various consensus algorithms have been developed to enhance security and address Byzantine behavior. The Algorand [45] algorithm combines proof of stake (PoS) with Byzantine fault tolerance (BFT) and utilizes a verifiable random function (VRF) to select a set of nodes for the committee. Additionally, DeepChain [46] introduces a collaborative training method with an incentive mechanism that prioritizes the privacy of local gradients. It promotes fairness in collaborative training by incentivizing participants to act honestly in gradient collection and parameter updates, and it demonstrates that participants are incentivized to behave correctly with high probability. However, consensus mechanisms such as proof of work (PoW) and proof of stake (PoS) may introduce significant computational overhead and affect system performance while ensuring security. BlockFL [47] introduces an end-to-end delay model that calculates PoW delay and minimizes the delay by adjusting PoW difficulty or block generation rates. It facilitates the exchange and verification of the local learning model updates and solves the additional delay caused by the blockchain network. In addition, Feng et al. [48] proposed blockchain-assisted FL (BAFL), which controls the block generation rate to reduce delays and dynamically adjusts training times to prevent transaction overloads. It enables devices to upload their local models during global aggregation for fast convergence.

Many examples prove that combining blockchain technology and DFL can achieve transparency and Tamper resistance in data operations and model updates. For instance, Jatain et al. [49] developed a framework for securely aggregating private healthcare data, offering an effective way to train machine learning models. Kim et al. [47] introduced a blockchain-based FL (FL) architecture wherein participating clients store their local model updates on blocks. All client nodes function as miners, enabling them to access and aggregate updates via smart contracts and eliminating the necessity for a centralized server. Peng et al. introduced VFChain, a verifiable and auditable FL (FL) framework tailored to blockchain-based systems. Verifiability is ensured through the blockchain's selection of a committee tasked with collectively aggregating models and recording verifiable evidence within the blockchain. Blockchain technology can also be combined with a decentralized topology. For example, Wang et al. [50] mentioned that the consistent hashing algorithm builds a ring topology between nodes participating in the FL system. Each node in the network is assigned a hash value that determines its position on the virtual ring. A consistent hashing algorithm helps distribute trusted nodes evenly across the ring to ensure that each node handles approximately the same amount of load. Feng et al. [51] proposed a blockchain-based, privacy-preserving FL (FL) framework that utilizes smart contracts, rather than a central server to manage global model updates. Similarly, Kim et al. [47] developed BlockFL, a framework for FL that supports decentralized global model aggregation to enhance privacy.

3.3. Comparison and Summary of TD-FL and BC-FL

Traditional decentralized serverless federated learning (TD-FL) and blockchain-based federated learning (BC-FL) are two typical distributed machine learning paradigms, showing significant differences in architecture design, performance, and security. TD-FL adopts a peer-to-peer (P2P) structure to reduce communication overhead and computing costs [52].

However, due to the lack of a global trust mechanism, DFL is vulnerable to model poisoning, data tampering, and collusion attacks [53]. DFL mainly relies on local privacy protection technologies such as differential privacy and secure multi-party computing to reduce the risk of data leakage, but it is still difficult to completely resist the interference of malicious nodes. For example, malicious gradient injection can cause model convergence deviation or privacy leakage, and it is susceptible to Sybil attacks or malicious node interference. In comparison, BC-FL records model updates in an immutable distributed ledger through a blockchain network, and 51% of the nodes must be compromised before data can be tampered with, which significantly improves the system's anti-attack capability [54]. However, this enhanced security comes at the expense of efficiency. BCDFL requires transaction verification and consensus calculations between all participants, resulting in high communication and computing overheads, which limits the scalability of the system. Experimental studies have shown that in an environment with 100 clients, 10 rounds of training per round, and each model size of 10 MB, the communication overhead of DFL is about 1000 MB, the single round delay is 2 s, and the model converges within 50 rounds; meanwhile, the communication overhead of BCDFL is as high as 5000 MB, the single round delay is 10 s, and the model converges in 80 rounds [54]. Therefore, DFL is suitable for IoT and edge computing scenarios with high computing-efficiency requirements, while BCDFL is suitable for applications with high security requirements, such as medical care. Given the advantages and disadvantages of both, the hybrid solution is one of the optimization directions. For example, a lightweight blockchain architecture can be used to store only key model updates to reduce communication and storage overhead [55], or verifiable computation technology can be combined to improve training efficiency while ensuring security (Figure 3).

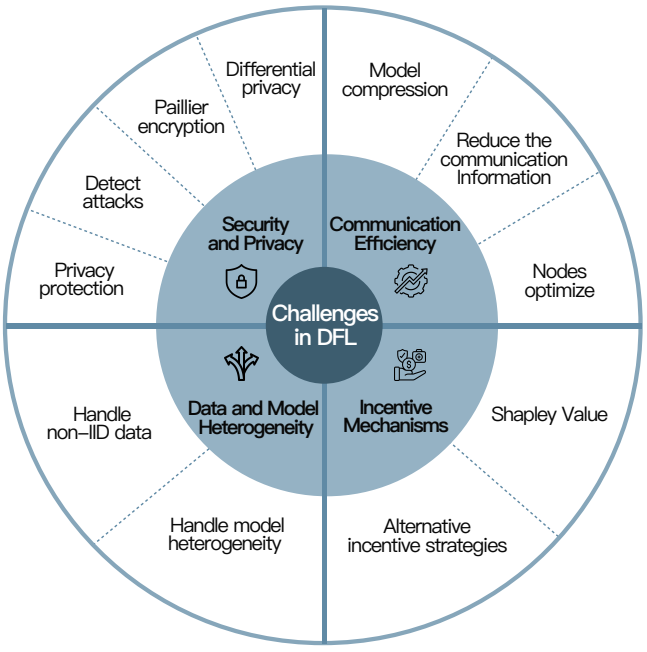

Figure 3. Key challenges for DFL.

4. Challenges

Although DFL offers advantages over standard FL in several aspects, it still faces critical challenges in terms of security and privacy, communication efficiency and cost, data and model heterogeneity, and incentive mechanisms. This section outlines the critical challenges in DFL to analyze the relevant methods and mathematical formulas in the existing literature. A summary of the relevant symbols used in this article is provided in Table 1.

Table 1. Frequent notations and descriptions.

Notations	Descriptions
Pr	The probability distributions of the query result
S	The subset of possible outputs for the query result
ϵ	The privacy budget
Δf	The sensitivity of the query function $f(x)$
η	The noise value
p	The probability density
$\mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right)$	Sampling from a Laplace distribution with a mean of 0 and a scale parameter of $\frac{\Delta f}{\epsilon}$
δ	The tolerance parameter that controls the probability of privacy failure
σ	The standard deviation
Δq	The sensitivity of the scoring function, indicating the impact of a single-data-point', s, change in the data set on the scoring function.
x and x'	The adjacent data sets
O	The set of candidate outputs, which includes all possible outputs
o	The particular output selected from the candidate output set O
o'	Another candidate output from the set, O, used for normalization
λ	The private key
lcm	Least common multiple
$\mathbb{Z}_{n^2}^*$	The set of all integers that are relatively prime to n^2
m	Plaintext message
c	Ciphertext
ϕ_k	The Shapley value of node k
$S \subseteq N \backslash \{k\}$	The subset S of all nodes, excluding node k
$\lvert S \rvert !$	The number of permutations of the nodes in coalition S
$\lvert N \rvert !$	The total number of permutations of all nodes
N	The number of total clients
$v(S)$	The utility function of coalition S, which represents the total value obtained via the nodes in coalition S when cooperating

4.1. Security and Privacy

Decentralized approaches typically operate under the assumption of semi-honesty regarding adversaries. However, this assumption might prove overly optimistic when applied to real-world scenarios [56]. During the information exchange process between various participants, local data may still be leaked during model updates and aggregation. Since there is no central node to manage or verify updates from different participating nodes, this increases the risk of Byzantine attacks on each participating node [57], making it difficult to ensure robustness in a decentralized FL environment. Additionally, malicious actors intentionally upload counterfeit models and pose significant threats. Similar to FL, in DFL, differential privacy and homomorphic encryption remain key privacy protection techniques. Traditional FL faces a range of adversarial attacks, where malicious nodes flood other nodes with random or redundant data. Decentralized FL approaches also face similar threats. Data poisoning [58] is designed to harm the accuracy of the global model. In addition, in DFL, each participant has access to these parameters. It can be exploited by malicious actors to perform model inversion attacks, where private training sets can be inferred from model parameters. A trusted DFL security measure has been proposed to protect against adversarial attacks [59]. This method resists model poisoning attacks by randomly sampling weights within a given range for transmission. The proposed algorithm shows better performance in the presence of attacked agents than the system without the introduction of a trust mechanism. In addition, existing BC-FL frameworks use consensus and trust mechanisms to strengthen the system's defense against potential attacks [60]. For example, Shayan et al. [54] used Multi-Krum's label-flipping attack, which effectively

rejects model updates that deviate significantly from the majority of update directions and it can handle malicious updates from up to 30% of malicious clients. BAFLtackles Gaussian reverse attacks by ensuring security through the Blockchain [48]. Additionally, BytoChain demonstrates its security by resisting random poisoning and reverse poisoning attacks. It maintains accuracy even when 40% of the data holders are attackers [61]. These methods demonstrate their effectiveness against adversarial attacks in BC-FL systems.

Privacy protection technologies are mainly divided into anonymization-based technologies and cryptographic methods. Anonymization-based technologies reduce the identifiability of individual data by perturbing or aggregating data or model updates. Among them, differential privacy (DP) is a standard method that protects privacy by adding noise to gradients or model parameters to limit the impact of attackers speculating on individual samples. Cryptographic methods rely on encryption technology to ensure that data remains confidential during transmission and calculation. Homomorphic encryption (HE) allows calculations to be performed directly on encrypted data, ensuring that the server cannot access plaintext data.

4.1.1. Differential Privacy

Differential privacy (DP) is a mathematical framework for ensuring data privacy protection that is used to avoid leaking an individual's private information during data publishing or use [62]. The core idea of differential privacy is that the result of any query will not be significantly affected by a single data point; that is, an attacker cannot infer whether a specific individual exists in the database based solely on the query result.

The probability distributions of the query result satisfy the following condition. x and x' are adjacent data sets, meaning the difference between these two data sets is only one element. $f(x)$ is the result of applying the query function to the data set, such as a statistic or a computed model. S is any subset of possible outputs for the query result. ϵ is the privacy budget. e^{ϵ} is an amplification factor that describes the privacy protection strength of the query result. It controls the strength of privacy protection: a smaller ϵ provides stronger privacy protection but also means that more noise will be added to the query results.

$$\Pr[f(x) \in S] \leq e^{\epsilon} \Pr[f(x') \in S] \tag{1}$$

In order to achieve differential privacy, specific information of the data set is usually hidden by adding noise to the query results. The three common methods for achieving differential privacy are the Laplace mechanism, the Gaussian mechanism, and the exponential mechanism [63].

Laplace Mechanism

Suppose we have a query function, $f(x)$, that acts on a data set, x, and generates an output. To ensure differential privacy, we need to add noise to the query result. The noise comes from the Laplace distribution. First, define the sensitivity of the query function, which represents the maximum impact of a change in a single data point in the database on the query result. Among them, x and x' are adjacent data sets that differ by only one data point. The sensitivity, Δf, of the query function, $f(x)$, is defined as follows:

$$\Delta f = \max_{x,x'} \|f(x) - f(x')\| \tag{2}$$

ϵ is the privacy budget. A smaller ϵ will increase the noise, thereby providing stronger privacy protection. $b = \frac{\Delta f}{\epsilon}$ is the scale of the noise, which determines the size of the noise. The probability density function of the Laplace distribution is as follows:

$$p(\eta) = \frac{1}{2b} \exp\left(-\frac{|\eta|}{b}\right),\ b = \frac{\Delta f}{\epsilon} \tag{3}$$

Generate a noise value, η, that conforms to the Laplace distribution. First, a random number, $U \in [-0.5, 0.5]$, is generated using uniform distribution. Then, the uniform distribution is converted into Laplace distribution using logarithmic transformation. $\text{sign}(U)$ is the sign function of U, indicating the positive or negative value of the noise.

$$\eta = -b \cdot \text{sign}(U) \cdot \ln(1 - 2|U|) \tag{4}$$

Add noise to the query result $f(x)$ through the Laplace mechanism:

$$f(x) + \eta, \eta \sim \text{Lap}\left(\frac{\Delta f}{\epsilon}\right) \tag{5}$$

Gaussian Mechanism

The Gaussian mechanism is another commonly used method for achieving differential privacy [64]. It protects data privacy by adding noise drawn from a Gaussian (normal) distribution to the query results [65]. The mechanism controls privacy protection by adjusting the standard deviation of the noise.

x and x' are adjacent data sets, differing by only a single data point. Δf is the sensitivity of the query function, measuring the maximum impact of a change in a single data point on the query result. The sensitivity, Δf, of the query function, $f(x)$, is defined as follows:

$$\Delta f = \max_{x,x'} \|f(x) - f(x')\|_2 \tag{6}$$

Δf is the sensitivity of the query function. ϵ is the privacy budget, which controls the strength of privacy protection. A smaller ϵ results in larger noise, providing stronger privacy protection. δ is the tolerance parameter that controls the probability of privacy failure. It is typically set to a very small value, such as 10^{-6}. The formula for calculating the standard deviation σ is as follows:

$$\sigma = \frac{\Delta f \cdot \sqrt{2\ln(1.25/\delta)}}{\epsilon} \tag{7}$$

Define the probability distribution of noise. The noise η is sampled from a Gaussian distribution, which means the noise is sampled from $\mathcal{N}(0, \sigma^2)$; its probability density function is as follows:

$$p(\eta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\eta^2}{2\sigma^2}\right) \tag{8}$$

The noise η is sampled from the probability distribution using a standard random number generation method. A random number, $Z \sim \mathcal{N}(0, 1)$, is generated from the standard normal distribution (with mean 0 and standard deviation 1). The noise η is then obtained by multiplying this random number by σ. The generated noise η follows a Gaussian distribution $\mathcal{N}(0, \sigma^2)$.

$$\begin{aligned} \eta &= Z \cdot \sigma \\ \eta &\sim \mathcal{N}(0, \sigma^2) \end{aligned} \tag{9}$$

Add noise η from a Gaussian distribution with standard deviation σ to the query result $f(x)$ in order to achieve differential privacy.

$$f(x) + \eta, \eta \sim \mathcal{N}(0, \sigma^2) \tag{10}$$

Exponential Mechanism

Unlike the Laplace and Gaussian mechanisms, which protect privacy by adding noise, the exponential mechanism selects outputs based on a probability distribution [65]. This distribution ensures that the output with the highest score has the greatest probability of being chosen while still allowing for a non-zero probability of selecting other outputs. This probabilistic selection process guarantees that, even if an attacker knows the scoring function, they cannot determine the exact result of the query.

Δq represents the sensitivity of the scoring function, indicating the impact of a single data point's change in the data set on the scoring function. x and x' are adjacent data sets, and o and o' are candidate outputs. $q(x, o)$ is the scoring function that evaluates the "quality" of the data set x and the candidate output o. The sensitivity is calculated as follows:

$$\Delta q = \max_{x,x'} \max_{o,o'} |q(x, o) - q(x', o')| \tag{11}$$

In the exponential mechanism, the probability of selecting an output is governed by an exponential function. Outputs with higher scores result in larger values for the exponential function, thereby increasing their probability of being selected. Conversely, outputs with lower scores yield smaller values for the exponential function, but these values remain positive, meaning that such outputs still have a non-zero probability of being selected. ϵ represents the privacy budget, which controls the strength of privacy protection; the smaller the value, the stronger the privacy protection.

$$e^{\frac{\epsilon q(x,o)}{2\Delta q}} \tag{12}$$

O denotes the set of candidate outputs, which includes all possible outputs. o represents a particular output selected from the candidate output set O. o' is another candidate output from the set O used for normalization (to ensure that the total probability of selecting all outputs sums to 1). The final selection probability $\Pr[o]$ is as follows:

$$\Pr[o] = \frac{e^{\frac{\epsilon q(x,o)}{2\Delta q}}}{\sum_{o' \in O} e^{\frac{\epsilon q(x,o')}{2\Delta q}}} \tag{13}$$

4.1.2. Paillier Encryption Algorithm

The Paillier homomorphic encryption algorithm is an asymmetric encryption algorithm proposed by Pascal Paillier in 1999 [66]. Its uniqueness lies in its homomorphic property; that is, it can directly operate on encrypted data without decryption. This makes it widely used in fields such as data privacy protection and secure multi-party computing. Data are processed in an encrypted state, which facilitates the aggregation of encrypted local model weights, thereby generating global model weights without the need for decryption. It is only decrypted when required, thus protecting data privacy.

Key generation.

Select two large prime numbers, p and q. Compute $n = p \times q$. Compute λ, where λ is the output of Carmichael's totient function, defined as $\lambda = \text{lcm}(p - 1, q - 1)$. Select an integer, g, such that $g \in \mathbb{Z}^*_{n^2}$.

The public key is (n, g), and the private key is λ.

Encryption of a Plaintext Message *m*

Select a random integer, r, satisfying $r \in \mathbb{Z}_n^*$. Using the public key (n, g), compute the ciphertext c as follows:

$$c = g^m \times r^n \bmod n^2 \tag{14}$$

Encryption algorithm correctness verification.

Define u as follows:

$$u = c^\lambda \bmod n^2 \tag{15}$$

$$u = (g^m \cdot r^n)^\lambda \bmod n^2 \tag{16}$$

Given that $\lambda = \text{lcm}(p-1, q-1)$, according to Carmichael's function property, for any $a \in \mathbb{Z}_n^*$

$$a^\lambda \equiv 1 \bmod n \tag{17}$$

Substituting " $r^\lambda = kn + 1$ into the expansion

$$(r^n)^\lambda \equiv 1 \bmod n^2 \tag{18}$$

Rewriting u by separating g^m and r^n,

$$u = (g^m)^\lambda \cdot (r^n)^\lambda \bmod n^2 \tag{19}$$

$$u = (g^m)^\lambda \cdot 1 \bmod n^2 \tag{20}$$

$$u = (kn+1)^m \bmod n^2 \tag{21}$$

Define the function $L(x) = \frac{x-1}{n}$, which is used to transform encrypted data.

Applying the function $L(x)$ to the encrypted value u:

$$L(u) = L((kn+1)^m \bmod n^2) = \frac{[(kn+1)^m \bmod n^2] - 1}{n} = \frac{(1 + m \cdot kn \bmod n^2) - 1}{n} = km \tag{22}$$

The modular inverse of $L(g^\lambda \bmod n^2)$. Firstly, compute $L(g^\lambda \bmod n^2)$. Calculate g^λ under modulo n^2, and apply the L function. Using the definition of $L(x)$, where $L(x) = \frac{x-1}{n}$,

$$L(g^\lambda \bmod n^2) = \frac{(g^\lambda \bmod n^2) - 1}{n} = \frac{(1 + kn \bmod n^2) - 1}{n} \tag{23}$$

Compute the modular inverse of $L(g^\lambda \bmod n^2)$:

$$\mu = \text{modinv}(L(g^\lambda \bmod n^2), n) \tag{24}$$

$$\mu = \left(\frac{(1 + kn \bmod n^2) - 1}{n}\right)^{-1} = (k \bmod n^2)^{-1} \bmod n \tag{25}$$

Recovering the Original Message, m. The plaintext message m is retrieved from the ciphertext, c using the following:

$$m = L(u) \times \mu \bmod n \tag{26}$$

4.1.3. Strategies for Detecting Attacks

Hacking can diminish institutions' motivation to engage in federated modeling collaborations. This underscores the necessity for a preemptive stance on safeguarding data

privacy. It highlights the imperative for machine learning researchers to proactively identify and address vulnerabilities to prevent the potential exploitation of such weaknesses (Table 2).

Table 2. A summary of security and privacy in DFL.

Ref.	Challenges	Solutions	ML Methods	Metrics	Compared to the Baseline Algorithms	Framework
[3]	Model parameters are stolen by attackers to infer the original private clinical data	(1) Adaptive differential privacy algorithm (2) A consensus protocol based on the gradient validation	CNN	Accuracy	BlockFL Original FL	Blockchain-based FL method
[20]	Privacy disclosure	Differential privacy stochastic gradient descent (DP-SGD)	LeNet5 MLP CNN	Accuracy A macro-average accuracy	FedAvg FML-proxy	Proxy-based FL (ProxyFL)
[23]	Privacy disclosure	(1) Blockchain provides a tamper-proof recording and verification mechanism for data operations (2) The ciphertext-policy attribute-based encryption (CP-ABE) encryption method that allows cryptors to define who can decrypt the ciphertext ensures that only authorized users have access to sensitive data	NA	NA	NA	A novel model of individual-initiated auditable access control (IIAAC)
[42]	Privacy disclosure	Federated distillation	NN	Loss Accuracy	FedAtt DS2PM FedAvg	Federated distillation and blockchain-empowered secure knowledge sharing (FDBC-SKS)
[56]	Model misconduct, such as submitting incorrect or tampered models	A framework with three components, auditing, coefficient, and performance detectors, that detects model misconduct by comparing models with historical data	Logistic regression	Precision Recall F1-score Execution time	NA	The model misconduct detection framework
[67]	Byzantine attack	Dual-way updating mechanism to isolate malicious nodes	NN	Accuracy	Cyclic institutional incremental learning (CIIL)	Robust DFL (RDFL)
[68]	Poisoning attack	Using secure multi-party computation (SMPC) encryption inference to exclude malicious models before model aggregation	ResNet18	Accuracy Time cost	NA	Secure multi-party computation (SMPC)
[69]	Data tampering	A practical Byzantine fault-tolerant (PBFT) consensus algorithm is used in a hierarchical cross-chain architecture	NA	Accuracy	NA	Cross-chain-empowered FL framework
[70]	Privacy disclosure	The generative adversarial network (GAN) is used to generate synthetic data to simulate the statistical properties of the real data without exposing the raw data	GAN	Accuracy	FedAvg FedProx FedBN	A DFL strategy using experience replay and GANs

Gouissem et al. [67] developed bidirectionally updated collaborative decision-making. A bidirectional update mechanism allows each node to receive updates from its neighbors and use these updates to update its model in two directions—clockwise (CW) and counterclockwise (ACW). This dual-path update allows each node to compare two potential updates to its current model, with each node evaluating which neighbor update is more beneficial and selecting the update that best improves model performance. This approach reduces the risk of a single malicious node adversely affecting the entire network. It can effectively isolate and ignore updates from Byzantine nodes after multiple iterations. Kuo et al. [56] developed a detection framework with three components: auditing, coefficient, and performance detectors. Among them, the audit detector compares the submitted model with historical models to check for duplication or unusual absence of data (e.g., empty gradient vectors). Coefficient detectors analyze changes in model parameters

during iterations to detect unexpected changes that may indicate data tampering. The performance detector evaluates a model's performance using metrics such as the area under the receiver operating characteristic curve (AUC) to identify significant deviations from expected performance levels. The system checks local models submitted from any site in a specific iteration. If any of the three detectors (audit, coefficients, or performance) detects an anomaly or violation, the model is marked as "misbehaving".

Several strategies have been proposed for detecting malicious gradients or models in blockchain environments. Chang et al. [3] designed a consensus protocol based on gradient verification. This protocol establishes a verification committee to check for abnormalities in the gradient data uploaded via each edge device. It identifies and removes malicious gradients uploaded by their associated MIoT devices through a consensus process among edge nodes, filtering out tampered-with or intentionally misleading gradient data to aggregate only qualified gradients in order to generate a reliable global model. Kalapaaking et al. [68] used the SMPC protocol, a secure multi-party computation protocol on the blockchain node. It ensures that poisoned local models are identified before aggregating and that only models that pass verification are aggregated to generate the final global model. After that, the blockchain network uses a consensus mechanism to validate the aggregated global model. This ensures that all nodes agree on the validity and integrity of the global model. Once validated, the global model is distributed back to the clients for the next round of training. Chen et al. [71] introduced a model aggregation method that considers the contribution weights of participants In DFL (contribution-weighted aggregation, FL-CWA); the method also mentioned that, after the local training is completed, a verifier is selected to verify the local model sent via each medical institution to the blockchain network, and the contribution value of each model is calculated. If the model's accuracy does not reach the dynamic accuracy threshold, the model will be excluded from participating in subsequent aggregations. The model that passes the verification is based on the contribution value to the global model as a weight, and the global model is obtained by weighting the average of all the local models that have passed the verification. Roy et al. [72] proposed that, before the global model is transmitted to each node as well, the blockchain verification manager (BVM) will verify and confirm it. This process ensures the integrity and credibility of the model, thereby providing confidence in the prediction results.

4.1.4. Strategies for Privacy Protection

While detecting and preventing malicious attacks are important measures to protect system security, inevitably some attacks may not be fully detected or prevented. At this time, it is particularly important to strengthen privacy protection measures as a supplementary means to reduce the possible damage to privacy after a successful attack.

In Section 3, blockchain as a distributed ledger system, was discussed for its role in mitigating single-point-of-failure issues in server-based architectures. Its immutability ensures data integrity while tracking previous records facilitates transparent and traceable FL. Smart contracts enforce security verifications, preventing malicious behavior or erroneous updates to the global model [73]. El Rifai et al. [74] integrated the FL and blockchain for the first time in a medical setting and proposed a smart contract to realize transparency and immutability while sharing knowledge. The powerful capabilities make it an attractive option for protecting sensitive information in DFL.

Technologies such as encryption algorithms in cryptography protect data privacy while allowing trusted third-party agents to perform data operations, establishing a secure interaction between data owners and recipients. Proxy re-encryption is a special encryption method that allows data to be converted from one encrypted form to another without decryption [72]. This means data can be securely re-encrypted and passed on to another

user without exposing the original data content. Wang et al. [75] combined AES symmetric encryption and public key encryption technology. The data provider uses the Advanced Encryption Standard (AES) symmetric encryption algorithm for data encryption, and the encrypted data is uploaded to the InterPlanetary File System (IPFS) to obtain the hash. The key and hash are encrypted using the recipient's public key and sent to the recipient. After the receiver decrypts them, the data are downloaded through the IPFS hash. This process ensures that data are encrypted during transmission and storage, and only the recipient can decrypt and access the data, ensuring the security of data sharing. Wang et al. [76] employed the Paillier homomorphic encryption algorithm to encrypt the local model, but it will still be decrypted by other clients, causing privacy leak issues. In this regard, they proposed using a public–private key pair (PPK) server to encrypt the homomorphically encrypted private key (SKhomo) with each client's public key and upload it to the blockchain. By centrally managing private key distribution, it is ensured that only authorized clients use their own private keys to decrypt these encrypted SKhomo and obtain the keys used for decryption.

Kalra et al. [20] used differential privacy (DP) technology when training the proxy model to ensure that, even if the proxy model is made public or shared, the information of any single data point cannot be accurately inferred. However, the differential privacy methods employed can reduce data utility, thereby diminishing the effectiveness of smart healthcare [77]. To address this issue, Chang et al. [3] proposed an adaptive differential privacy algorithm that optimizes the balance between data privacy and utility. This algorithm not only protects data privacy but also maintains model accuracy as much as possible by dynamically adjusting the amount of noise added to the data gradient during the FL process. In this way, even if an attacker can access the model parameters stored on the blockchain, it will be difficult to accurately infer the original clinical data.

Unlike differential privacy, which requires a trade-off between privacy protection strength and data accuracy, masking technology, as another privacy protection method, may not have much impact on the usability of data. Wang et al. [76] proposed a method of applying a mask on local model parameters. Even if other clients decrypt the model, they will get masked parameters, and unauthorized users will not be able to obtain accurate model parameters. The original data are protected.

Malicious attackers may associate specific data with specific identities and conduct targeted attacks through known client identities, such as model poisoning or inference attacks, leading to data privacy leaks. Therefore, it is critical to ensure that each client's identity remains anonymous or safe when submitting model updates. Roy et al. [72] applied digital certificates as electronic certificates to confirm ownership of a public key. Each certificate contains the public key, as well as the signature of the certificate authority (CA), proving that the holder is trusted. Digital certificates are used to verify identities in data communications and ensure that both parties to the communication are legitimate. Additionally, the Elliptic Curve Digital Signature Algorithm (ECDSA) can also be used as a digital signature technology [72]. Participants use ECDSA to sign data, proving the source and integrity of the data and thereby increasing their credibility. The Hyperledger framework is an open-source platform for building blockchain applications. Integrating ECDSA into Hyperledger can enhance the authentication process to ensure that participants' identities are trustworthy. Kang et al. [69] introduced the assurance that data authentication and unforgeability are ensured by utilizing the practical Byzantine fault tolerance (PBFT) consensus algorithm within the hierarchical cross-chain architecture for lightweight consensus [78]. PBFT's role involves rigorous auditing and authentication of all data by delegates (i.e., miners). Additionally, the decentralized nature of consortium blockchains, paired with digitally signed transactions, prevents attackers from impersonating users or

compromising the system. This combination guarantees the integrity and unforgeability of the data.

Kalra et al. [20] put forward the proxy model method. Specifically, each participant maintains two models, a private model and a public proxy model. The private model handles sensitive data directly, while the proxy model propagates and exchanges information in the network. The proxy model and the private model are jointly trained through the deep mutual learning (DML) method. The two models are jointly optimized by minimizing the cross-entropy loss and KL divergence loss so that the proxy model can learn the characteristics of the private model and, in turn, provide auxiliary information to enhance the performance of private models. This structural design helps isolate sensitive data and models involved in network communication, reducing the risk of directly leaking sensitive information.

A learning strategy was introduced by Pennisi et al. [70]. Each node initially trains a privacy-preserving GAN consisting of a generator and a discriminator. The generator's task is to create synthetic data with similar statistical properties to real data, while the discriminator attempts to differentiate between synthetic and real data. Through the adversarial training of the generator and the discriminator, the generator learns the distribution of real data and can generate synthetic data that maintains statistical properties, but it does not expose the specific content of the original data (Table 3).

Table 3. A summary of communication efficiency in DFL.

Ref.	Challenges	Solutions	ML Methods	Metrics	Compared to the Baseline Algorithms	Framework
[10]	(1) A large number of increased clients leads to higher communication costs (2) Resource limitations for the client side (wearables)	Knowledge distillation reduces the complex teacher model to a student model to run on wearable devices	DNN	Sensitivity (Sen) Specificity (Spe) Geometric mean (Gmean)	FedAvg	Real-time decentralized FL framework
[14]	(1) The growing size of the blockchain affects its scalability (2) Blockchain storage performance is limited by a single node	Upload the trained local model to the InterPlanetary File System (IPFS)	CNN	Accuracy	Training with individual nodes	Blockchain-Based Privacy-Preserving Medical Data Sharing Scheme (MPBC)
[42]	(1) Node load (2) Consensus process lacks efficiency	Selecting the master node based on node load, filtering nodes capable of participating in consensus in blockchain through reinforcement learning	NN	Loss Accuracy	FedAtt DS2PM FedAvg	Federated distillation and blockchain-empowered secure knowledge sharing (FDBC-SKS)
[75]	There is a redundancy when the data transfer between the nodes	(1) The large-scale data processing tasks are decomposed into small parts and processed in parallel on multiple processing nodes in the map-reduce framework (2) The Ring-All Reduce algorithm in parallel computing optimizes communication	GAN	Inception score (IS) Earth mover's distance (EMD)	Traditional centralized FL methods Decentralized FL frameworks that utilize gossip algorithms	Ring-topology DFL (RDFL)
[70]	Concept drift	Experience replay method	GAN	Accuracy	FedAvg FedProx FedBN	A DFL strategy using experience replay and GANs
[79]	The parameter exchange between nodes causes a communication burden	Decentralized stochastic gradient tracking (DSGT) through the gradient-tracking mechanism maintains the global gradient tracking at each node, reducing the overall communication frequency	NN	Optimality gap Accuracy Communication rounds	Traditional FL (FL) FL with DSGD FL with DSGT	NA

Table 3. *Cont.*

Ref.	Challenges	Solutions	ML Methods	Metrics	Compared to the Baseline Algorithms	Framework
[80]	The device in a wireless body area network (WBAN) consumes energy while transmitting	The Stable WBAN-Miner Association (WMA) heuristic algorithm maximizes the utility function of the entire system	QNN	Test loss Energy consumption	BlockFL HFEL BFL schemes	Efficient and privacy-preserving blockchain-based FL framework
[81]	Network communication burden	Knowledge distillation	ResNet18 ResNet50 DenseNet121	Balanced accuracy Class-specific accuracy Log loss	NA	Decentralized AI training algorithm (DAITA)
[82]	Communication and computational latency between the distributed nodes	The asynchronous consensus mechanism is integrated to handle changes in communication and computational delays between distributed nodes.	U-Net model	Dice similarity coefficient (DSC) Training time	Traditional centralized learning Federated averaging Consensus-driven fully DFL	NA
[83]	Miner disconnections during consensus execution in blockchain	A robust consensus based on PoS improvement that a robust proof of stake (RPoS) allows nodes participating in the consensus to go offline or enter during the execution of the consensus	CNN	Accuracy Convergence speed	Traditional FL without blockchain integration FL with PoW consensus mechanisms	Blockchain-Enabled Secure FL Architecture
[84]	(1) Differences in communication capacity between different nodes (2) Network heterogeneity leads to communication difficulties	(1) Edge servers with greater computing power are the nodes instead of mobile devices (2) Lin–Kernighan–Helsgaun (LKH) algorithm-based bottleneck traveling salesman problem (BTSP) solver	CNN	Accuracy Training time	Traditional decentralized ring-based FL as DFL	Decentralized, efficient, and privacy-enhanced federated edge learning (DEEP-FEL)
[85]	The heterogeneity of the data leads to inefficient model training	A dynamic clustering mechanism that groups clients based on data similarity	ClusterGAN CNN LSTM	Accuracy Training runtime	Dynamic clustering algorithm Dynamic-Fusion FL algorithm	A "Low-Overhead Clustered FL" approach
[86]	Differences in communication capacity between different nodes	Data-sharing scheme based on Ring-Allreduce	NA	Accuracy	FedAvg Gossip learning	Robust and privacy-preserving decentralized deep FL (RPDFL)

4.2. Communication Efficiency and Cost

In DFL, participants can connect to different nodes instead of simply connecting to a central server. The network topology of the parties may be more complex, which may result in more data needing to be transferred between the parties. It can cause a significant burden in scenarios with limited network bandwidth or high communication costs, such as hospital networks across geographical locations [79]. The massive increase in the number of clients and network topology adjustments increases the complexity and overhead of communication [10]. Therefore, transferring model parameters among data nodes consumes significant communication overhead [79]. Additionally, client resource limitations are another factor affecting communication efficiency. Wearable devices in a wireless body area network (WBAN) have limited computing resources [80]. The signals of mobile devices are uneven, which has become a limiting factor for model training [87].

When blockchain is used as decentralized storage and replaces the central server, it can also result in high communication costs. For example, the blockchain Ethereum requires a gas fee, which can be significantly high for large models [75]. Because nodes frequently communicate to exchange model parameter updates and verification data, the communication volume of some nodes could be too large. This may cause network congestion or increase latency, affecting the overall consensus efficiency. Additionally, certain FL schemes based on blockchain fail to account for participant dynamics, such as node dropouts and new joinings [88,89]. The consensus mechanism and immutability of blockchain may limit the flexibility and scalability of FL, thereby affecting the efficiency and real-time performance of FL.

4.2.1. Strategies for Model Compression

To overcome the communication problem of the decentralized FL framework, one of the current research focuses is on researching novel communication compression or model compression techniques to reduce communication pressure. For instance, Hu et al. made use of the gossip algorithm to enhance bandwidth utilization and model segmentation to alleviate communication pressure [90]. Singh et al. [80] suggested using quantized neural networks (QNNs) to reduce computational and storage requirements by reducing the computational precision of model parameters from standard 32-bit floating point precision to lower precision, thereby reducing energy consumption.

Additionally, knowledge distillation is employed in DFL to improve the performance on non-IID data sets. Knowledge distillation was first proposed by Hinton et al. [91]. In the context of FL, knowledge distillation methods are widely utilized to facilitate collaboration among clients while safeguarding privacy. Knowledge distillation only needs to transmit the output of the teacher model instead of the entire data set or model parameters, greatly reducing the network communication burden. This improves the overall training efficiency and performance of the model [81]. Numerous research works on knowledge distillation in DFL have been conducted. Among these techniques, Zhou et al. [42] defined a novel framework called federated distillation and blockchain-empowered secure knowledge sharing (FDBC-SKS) that reduces the size of model parameters that must be communicated over the network, facilitating efficiency improvements. Wang et al. [75] utilized Kullback–Leibler (KL) divergence function to compare the prediction results (soft labels) of the teacher model with the prediction results of the student model trained on the local data set. The relative "distance" between the two model output distributions is considered a gap in the student model's loss function, making the student model's predictions closer to those of the teacher model. In this way, the student model can learn helpful information from the teacher model without directly accessing other node data. Baghersalimi et al. [10] employed knowledge distillation for model compression to transfer the complex knowledge and information in the teacher DNN to the simplified student DNN, thereby reducing the model size and making it compatible with devices with limited memory and power significantly reducing communication costs.

4.2.2. Strategies for Reducing the Communication Information in Nodes

On the blockchain, each transaction produces input and output states. Wang et al. [76] introduced the cut-through method, which makes transaction records more concise and reduces the storage overhead of the blockchain by deleting unnecessary intermediate states and redundant data in these transactions. It helps improve the efficiency of the blockchain.

The communication scheme that Kalra et al. [20] presented handles data transmission and aggregation. First, nodes split their data into two parts: one part is sent to randomly selected nodes, and the other part is retained. This data parallel processing improves the speed and efficiency of data aggregation. Then, the information flow is controlled through the weighted adjacency matrix in the network. The PushSum algorithm adjusts the data flow according to the network structure and the nodes' importance. By precisely controlling the amount of data each node sends to other nodes and only sending part of the data to other nodes instead of the complete data, the communication burden on the network is significantly reduced.

Using the InterPlanetary File System (IPFS) to limit data transmission between nodes to encrypted keys and hash values is another way to reduce communication costs. Wang et al. [75] used an IPFS-based data-sharing scheme that significantly reduces the amount of data transferred directly between data nodes because the data transfer is limited to encrypted keys and hashes, rather than complete data files. Zhang et al. [14] pointed

out that IPFS technology uses hash encryption to generate immutable permanent IPFS addresses for massive data as well. This is equivalent to reducing raw data for IPFS address uploads. The actual data are stored locally by each data owner. This method relieves the storage pressure between nodes or the blockchain and reduces communication costs.

4.2.3. Strategies for Nodes Optimize

Wang et al. [75] combined the Ring-AllReduce algorithm and the consistent hashing algorithm to optimize communication and load balancing jointly. The Ring-AllReduce algorithm is a communication optimization algorithm used in parallel computing in distributed systems. The core idea of the algorithm is to organize all nodes into a logical ring structure. In the ring topology, each node moves along the ring. Data are passed along the path and only communicates directly with its immediate neighbors to implement data reduction operations. This method limits the scope of data transmission, thereby improving the efficiency of parallel computing. A consistent hashing algorithm distributes trusted nodes evenly across the ring, and each node in the network is assigned a hash value that determines its position on the virtual ring to ensure that each node handles approximately the same load quantity.

However, when mobile devices are utilized as nodes in a ring network topology, they often rely on battery power and must engage in frequent communication within the end-to-end structure. This can pose challenges for mobile devices to sustain stable and uninterrupted participation in training tasks. Furthermore, individual nodes within the ring topology possess varying communication capabilities. A large number of mobile devices or excessive message transmissions to a single user may exacerbate communication bottlenecks. Meanwhile, Lian et al. [87] addressed the ring construction problem using an LKH algorithm-based solver to minimize communication overhead and improve overall efficiency through a layered architecture. They regarded the ring construction problem as an abstraction of the asymmetric bottleneck traveling salesman problem (BTSP) and used a Lin–Kernighan–Helsgaun (LKH) algorithm to optimize the ring topology. They proposed BLKH, an LKH algorithm-based BTSP solver. Specifically, the weight of the longest edge in the path should be reduced so that the weight of the edge with the largest communication overhead in the ring is minimized. The maximum communication time between two nodes is made as short as possible, thereby finding the most effective data transmission path. In addition, Lian et al. [87] utilized a layered architecture, which is edge servers with muscular signal strength replacing mobile devices. Firstly, the mobile devices of each medical institution are trained locally based on the patient data received. After local training, the parameters are transmitted to the edge server for preliminary aggregation, and the corresponding edge model is generated. Multiple edge models go through a global aggregation process to generate a new global model and send it to the mobile device to update its local model. The layered architecture reduces dependence on the signal strength and stability of mobile devices through the intermediary role of edge servers. The edge server layer performs preliminary aggregation to reduce the burden of direct communication and alleviate communication bottlenecks.

Tedeschini et al. [82] used an asynchronous consensus mechanism to handle changes in communication and computing delays among distributed nodes. In asynchronous consensus, nodes can perform operations after receiving confirmations from a few other nodes without waiting for confirmations from slower nodes. This flexibility enables the system to reach consensus without requiring all nodes to synchronize simultaneously and ensures that a consistent state is eventually reached, thereby improving the robustness and efficiency of the training process.

Lu et al. [79] applied a method called decentralized stochastic gradient tracking (DSGT). It effectively tracks the global gradient at each node through the gradient tracking mechanism, which can reduce the overall communication frequency. The algorithm performs local updates for several iterations and then enables node communications. Each node can rely on local accumulated gradients for more accurate model updates rather than frequently relying on external data. Accordingly, the communication rounds of exchanging the common interest of parameters can be saved significantly.

Kalra et al. [20] applied the Exponential Communication Protocol of Assran, in which each client only communicates with other clients whose distance increases exponentially. By reducing the number of nodes that need to communicate in each round, the high communication overhead caused by frequent data exchange is effectively reduced.

Tedeschini et al. [82] applied the Message Queuing Telemetry Transport(MQTT) lightweight protocol, nodes will only receive data updates for the specific topics they are interested in and not for other topics. This subscription mechanism effectively improves bandwidth utilization efficiency. As a lightweight protocol with low bandwidth and an unstable network design, MQTT has low protocol overhead and is more suitable for real-time data transmission.

In blockchain-based approaches, miners can aggregate parameters in distributed manners, while the systems guarantee the aggregated data's accuracy and the miners' integrity via a consensus protocol. Nonetheless, most current methodologies rely on the proof of work (PoW) consensus [14], which entails performing redundant hash computations and consuming significant energy, leading to subpar performance [83]. The proof of stake (PoS) consensus has been proposed to solve the high energy consumption and low throughput issues associated with PoW. However, a drawback of PoS is its inability to handle offline nodes during the negotiation of random seeds by all participants, which can disrupt the consensus process and slow down the rate of federal learning convergence in blockchain systems. Zhang et al. [83] had developed a robust proof of stake (RPoS) to address this. This enhanced consensus mechanism allows nodes to join or leave the consensus execution without causing system halts or significant slowdowns. RPoS leverages publicly verifiable secret sharing (PVSS), through which encrypted shares of the secret can be independently stored and retrieved. If nodes go offline, the available shares can still reconstruct the random seed, ensuring uninterrupted consensus execution even when specific nodes are inactive and thereby preventing system crashes or interruptions. In contrast to the resource-intensive nature of the proof-of-work (PoW) protocol, Chang et al. [3] introduced a protocol built upon the advancements of Algorand. By employing a Byzantine agreement protocol through which only select miners validate new blocks in each training round, we effectively minimize the communication overhead among miners. This approach significantly enhances consensus efficiency.

Compared with the traditional proof of work (PoW) mechanism, in which participants must engage in resource-intensive competition to determine the block proposer, the consensus mechanism enhanced via reinforcement learning algorithms such as the Deep Q-Network can save many computing resources and achieve efficient computing load distribution. Zhou et al. [42] modeled optimal consensus node selection and load distribution in blockchain network problems as a Markov decision process (MDP), a mathematical framework for describing decision problems. In this framework, each consensus node can be viewed as an agent that needs to make decisions based on the state of the environment to maximize long-term accumulated rewards. The DQN algorithm can train these agents (i.e., consensus nodes) and find the optimal node selection strategy to achieve fair participation and load distribution in the consensus process (Table 4).

Table 4. A summary of heterogeneity and incentive mechanisms in DFL.

Ref.	Challenges	Solutions	ML Methods	Metrics	Compared to the Baseline Algorithms	Framework
[10]	Data heterogeneity	Adaptive ensemble learning	DNN	Sensitivity (Sen) Specificity (Spe) Geometric mean (Gmean)	FedAvg	Real-time decentralized FL framework
[20]	Model heterogeneity	Each participant uses a private model trained independently in a proxy model; a publicly available proxy model acts as a medium for information exchange	LeNet5MLPCNN	Accuracy, macro-average accuracy	FedAvg FML-proxy	Proxy-based FL (ProxyFL)
[23]	Individual users are reluctant to share data	Provide personalized feedback to the data owners who share the data; personalized feedback includes the results of comparative analysis with peer users or a group of users	NA	NA	NA	A novel model of individual-initiated auditable access control (IIAAC)
[71]	Lack of incentive	Staged reward mechanism	NN	Accuracy Loss	Federated average algorithm (FL-AVG)	FL with contribution-weighted aggregation (FL-CWA)
[69]	In the case of client information asymmetry, no consideration is given to motivate users to contribute fresh sensing data	An incentive mechanism based on contract theory with data freshness	NA	Accuracy	NA	Cross-chain-empowered FL framework
[70]	Data heterogeneity	The thetic data generated via GAN enhance data diversity and allow the model to learn from diverse data	GAN	Accuracy	FedAvg FedProx FedBN	A DFL strategy using experience replay and GANs
[79]	Data heterogeneity	Decentralized stochastic gradient tracking (DSGT) algorithm	NN	Optimality gap Accuracy Communication rounds	Traditional FL (FL) FL with DSGD FL with DSGT	NA
[92]	Assessing the quality of client features to determine which features should contribute more during the distillation process	Cyclic model transfer and feature attention-based multi-teacher knowledge distillation	AlexNetCNNLeNet5	Accuracy	FedAvg FedProx FedBN FedAP	FedTAM

4.3. Data and Model Heterogeneity

Single-source data sets do not represent real-world clinical or patient data diversity, and training with single-source data sets limits model scalability and introduces bias. This bias may lead to erroneous judgments or predictions in actual clinical applications [93]. In contrast, using diverse data from multiple sources to train models can provide a more comprehensive understanding of patient characteristics and diseases by covering diverse geographical regions, ethnicities, age groups, and disease conditions. Therefore, the diversity of model training data is crucial to improving models' accuracy, generalizability, especially in medical fields that are sensitive to diversity and individual differences. In addition, the differences in storage, computing, and communication capabilities of different devices in a distributed environment result in data characteristics and distribution that may be very different [94]. Even for the same disease, different patients have different physiological characteristics, and local data will show non-independent and identical distributions [84]. However, since MIoT devices are widely distributed on the open network edge, the clinical data they collect may be diverse. Then, the local gradient trained on this kind of data will deviate from the global convergence trend [3]. The problem of non-independent and identically distributed data causes the features learned during the model training process to differ, thus increasing model heterogeneity. This heterogeneity increases the complexity of integrating different models. When each node uses models of different architectures, even the same features may be learned at different layers or locations in the model [70]. In addition, the lack of a central organization to coordinate and standardize the model's training process in DFL may further exacerbate the feature misalignment problem. With FedAvg as the most common model, feature misalignments

may cancel each other out or distort each other during the merging process, resulting in poor learning effects [95].

In order to handle non-IID data, Lian et al. [84] put forward a model-layered training method: the model is divided into base layers and personalization layers. Each device uses the patient's individual characteristics for local training, updates the parameters of the personalization layer, and implements the training of the personalized model. The base layer is responsible for global updates and uploads the base layer parameters to the blockchain for global aggregation. In this way, the typical data characteristics can be captured through the global update of the base layer, and the data characteristics of each device can be captured through the personalization layer.

Similar to model hierarchical training, Kalra et al. [20] implemented a method to combine private models with proxy models. Private models are trained directly on local data. They can be optimized according to each node's specific needs and data characteristics without being restricted to using the same model structure as other nodes. The proxy model acts as a unified interface to exchange information between nodes. Through the method of deep mutual learning, private models and proxy models can learn from each other.

The distribution of non-independent and identically distributed data on different nodes may differ significantly, leading to model-forgetting problems [96]. Pennisi et al. [70] presented the experience replay mechanism, which uses synthetic data as a playback buffer to simulate the data scenario of previous nodes. This means the model can learn from the experience collected from each device, rather than just relying on the data of the current device or user. It can help the model better adapt to data heterogeneity.

Baghersalimi et al. [10] propounded an approach of adaptive ensembling to handle the heterogeneity of models between different hospitals. Specifically, each hospital trained a local model on its data set and exchanged local model weights with other participating hospitals. The model ensemble learning strategy compares disease detection accuracy between local weights and a blend of global and local weights on validation data. The weights that yielded higher detection accuracy were accepted for further use. The weight of each model in the final ensemble was dynamically adjusted according to the model performance. Models with better performance have a more significant influence when weighted together into a global model. Adaptive ensemble learning allows each hospital to adjust the ensemble model according to its specific data characteristics. The exchange and dynamic adjustment of model weights enable adaptive ensemble learning to effectively handle the models' heterogeneity between different hospitals.

Since data distribution between clients changes, each client needs to selectively aggregate features learned from other clients. The key challenge is to accurately evaluate the quality of these features to determine which features should contribute more to the ensemble process. A novel FL approach incorporates a feature attention mechanism [92]. Clients can calculate the similarity between the features of their models and the features of other client models to determine which features are most useful for their own models and selectively obtain the most relevant knowledge from other clients.

Another way of processing data is to cluster them according to the data type. Baghersalimi et al. [10] integrated a clustering algorithm based on disease (epilepsy) types to form hospital clusters of the same type of disease (epilepsy). This helps the system reduce unnecessary model transfer costs by limiting model transfer to only hospitals within the same cluster (hospitals that handle similar types of epilepsy).

4.4. Incentive Mechanisms

In a DFL framework, quantifying each client's contribution to model training helps to allocate limited data storage and computing power resources reasonably. Clients who

contribute more will be allocated more resources and given greater weight or more frequent update opportunities in their model training to obtain higher priority. However, due to concerns about data privacy and security and the fact that model training and parameter updates in FL involve a large amount of computing resources, such as data transmission and the maintenance of computing resources, the reduction in client participation enthusiasm affects the overall performance and model quality of the FL system. Shapley values, a concept from cooperative game theory, are widely used to measure the marginal contribution of each participant (node or player) in a cooperative setting [97]. In DFL, where multiple nodes compete for computational resources, Shapley values offer a fair mechanism for evaluating each node's contribution. By fairly distributing rewards and assigning Shapley values, it becomes possible to identify the nodes that contribute most significantly to the global model. This enables prioritizing high-contributing nodes for further training or giving them higher weight during model aggregation. Consequently, this approach enhances training efficiency and improves the overall quality of the global model.

ϕ_k represents the Shapley value of node k, indicating its contribution to the entire cooperation. $S \subseteq N \backslash \{k\}$ denotes a subset, S, of all nodes, excluding node k i.e., the set of all coalitions without node k. The Shapley value of node k is computed by summing over all such subsets, S. $|S|!$ represents the number of permutations of the nodes in coalition S. It accounts for the influence of the order within the subset. $(|N| - |S| - 1)!$ represents the number of permutations of the remaining nodes after node k joins the coalition. $|N|!$ represents the total number of permutations of all nodes, i.e., the factorial of the total number of nodes. $v(S)$ denotes the utility function of coalition S, which represents the total utility or value obtained via the nodes in coalition S when cooperating. $v(S \cup \{k\}) - v(S)$ represents the marginal contribution of node k when joining coalition S. Finally, by considering the permutations of each possible subset and the order in which nodes join, the contribution of node k to the global utility is as follows:

$$\phi_k = \sum_{S \subseteq N \backslash \{k\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [v(S \cup \{k\}) - v(S)] \tag{27}$$

In addition to the Shapley value, which ensures fair distribution and motivates all parties to participate, other complementary incentives also play a crucial role in encouraging full participation in FL tasks. These incentives help align the interests of all stakeholders, promoting collaboration and driving the success of the FL framework. For example, Cong et al. [23] provided personalized feedback on health status as a reward for participants who provide personal wearable device data. In addition, setting up a reasonable reward mechanism is also a standard method. For instance, methods used to incentivize participation in FL encompass reputation mechanisms [97] and optimal pricing strategies [98]. However, if the incentive mechanism is not designed properly, it may result in participants gaining benefits while facing greater risks, which may undermine the fairness and impartiality of the entire FL system, further eroding the enthusiasm of participants.

To ensure the fairness and rationality of reward distribution. Chen et al. [71] proposed a staged reward mechanism to ensure the fairness and rationality of reward distribution. This mechanism divides rewards into multiple stages instead of distributing them simultaneously. Distributing rewards in multiple stages ensures that participants can get fair returns at different stages, which helps maintain their long-term interest and participation.

In situations marked by unequal access to information, most studies do not explore how to motivate users to contribute up-to-date sensing data for healthcare services. Incentive mechanisms based on contract theory can address information asymmetry by

designing contracts to ensure all parties benefit from transactions. However, there is still the problem of ignoring data freshness. The age of information (AoI) has been widely used as an effective metric to quantify the freshness of destination data [99]. Kang et al. [69] employed the age of information (AoI) as a robust measure of data freshness. They introduce an AoI-based contract theory model utilizing prospect theory (PT) to incentivize the user-centric sharing of sensing data. This model leverages PT to assess the subjective value of the service provider more accurately.

The reward mechanism needs to be traced back to a specific client before rewards can be sent to that client. However, in DFL, the client's anonymity makes it impossible for the reward mechanism to track the client who uploaded the local model. Thus, the reward mechanism cannot ensure that the client's contribution is accurately identified and rewarded. This will significantly reduce the reward mechanism's fairness and incentive effect. Wang et al. [76] designed a reward mechanism based on ring signature (RMBRS) in an anonymous environment and verified the rationality of the reward mechanism through game theory methods. Specifically, the client selects the number of public keys for the ring signature and generates and uploads the ring signature. The reward agency verifies the ring signature and distributes the reward according to the weight so that the degree of reward is inversely proportional to the degree of anonymity. Therefore, the client can trade between the reward amount and anonymity according to its own needs.

Additionally, a reward mechanism in the blockchain encourages users to provide more high-quality data, which is conducive to the entire learning process and the establishment of a federated ecosystem. As another standard incentive method, in a blockchain-based system, Zhang et al. [83] utilized the participate incentive mechanism (PIM) and aggregation incentives mechanism (AIM). PIM provides token incentives and continuous payment incentives for medical institutions that join a training epoch. AIM provides payment incentives for well-behaved miners. Miners can point out parameters that tend to be malicious, and they will be rewarded with tokens after multi-party verification. Furthermore, in order to reward, the punishment mechanism will greatly reduce the possibility of malicious miners participating in a randomized proof of stake (RPoS) and gradually losing their own tokens as malicious behavior accumulates (Table 5).

Table 5. A summary of case studies in DFL.

Category	Ref.	Data Set	Performance	Summary
Wearable Device Data	[23]	Health-related data from wearable devices(private)	NA	The health data-sharing system supported via DFL provides personal feedback to users through data analysis, and shared models
	[92]	Physical activity monitoring (PAMAP2)	The average accuracy over multiple standard data sets is better than traditional FL methods	FedTAM combined recurrent model transfer and feature attention mechanism to customize personalized models for customers in environments with non-iid personal health monitoring data distribution
	[85]	Wearable stress and affect detection (WESAD)	Higher accuracy, reduced training runtime compared to baseline methods	A personalized, low-overhead clustered FL algorithm for stress level identification that improves accuracy and reduces training time
	[80]	MIT-BIH arrhythmia	The proposed scheme outperforms BlockFL, HFEL, and BFL, achieving 15.1%, 9.03%, and 15.35% more utility on average; consumes 12.87%, 7.6%, and 13.18% less energy on average than that of BlockFL, HFEL, and BFL, respectively	The algorithm based on the blockchain-supported FL model uses the wireless body area network (WBAN) to use the physiological sensor data collected via local devices to monitor the real-time health indicators of patients
Cardiovascular Diseases	[67]	Physikalisch-Technische Bundesanstalt Database electrocardiogram (PTBDB ECG)	Detection accuracy: above 95%	A server-less FL training mechanism verified the effectiveness in detecting irregular heart rhythms; dual-way update mechanism resisted Byzantine attacks
	[3]	Pima Indians Diabetes	With the same privacy budget, the accuracy of this method is slightly lower than Original FL, but with enhanced privacy protection	Blockchain-based FL for smart healthcare achieves high model accuracy in acceptable running time in diabetes monitoring while also showing good performance in reducing the privacy budget consumption and resisting poisoning attacks
	[100]	UCI Heart Disease Dataset	The DeFedHDP model achieved an accuracy of 90% in heart disease prediction, converging faster than the FedAVG (centralized FL) method to this accuracy	As a fully DFL method, DeFedHDP solves the privacy protection problem in heart disease prediction by introducing a differential privacy mechanism, an online aggregation strategy, and a single-point slot machine feedback strategy

Table 5. *Cont.*

Category	Ref.	Data Set	Performance	Summary
	[56]	Edinburgh Myocardial Infarction Dataset (EMIDEC)	DFL framework that generates different types of model misbehaviors through simulators and develops audit, coefficient, and performance detectors to efficiently identify misbehaviors in FL improves the reliability of healthcare modeling	DFL allowed participants to optimize the heart disease risk prediction model through local training, leading to higher accuracy early
	[101]	Four distinct T1D data sets—OhioT1DM, ABC4D, CTR3, and REPLACE-BG (all containing CGM records)	The asynchronous architecture with GluADFL addresses data heterogeneity while maintaining prediction accuracy, achieving mean absolute errors 15–20% lower than those of conventional centralized approaches	GluADFL as an asynchronous DFL method for blood glucose prediction, solves the "cold start" problem of patients with type 1 diabetes under privacy protection and demonstrates excellent prediction accuracy on multiple data sets
Neurological Disorders	[10]	(1) EPILEPSIAE (2) TUHEEGSeizureCorpus (TUSZ)	(1) Gmean: 88.72%, higher than FedAvg (2) Gmean: 85.83%, higher than FedAvg	A decentralized FL framework using adaptive ensemble learning and knowledge distillation addresses the non-IID challenge of hospital data and meets the resource constraints of wearable systems
	[79]	Proprietary data set consisting of patients diagnosed with Alzheimer's disease (AD) and mild cognitive impairment (MCI)	(1) Decentralized stochastic gradient descent (DSGD) reduces the computational burden compared to centralized GD/SGD processing. (2) Decentralized stochastic gradient tracking (DSGT) offers the advantage of dealing with non-identical data sets compared with (DSGD)	Decentralized non-convex optimization for FL to extract patients' features from hospital data sets; data privacy could be preserved better than the centralized case
Respiratory diseases	[70]	Tuberculosis classification contains two distinct sources (1) Montgomery County X-ray set (2) the Shenzhen Hospital X-ray set	Classification accuracy: 83.41%, better than FedAvg, FedProx, and FedBN	DFL inspired by experience replay and generative adversarial concepts achieves comparable performance to non-FL in the non-IID medical data scenario
	[102]	(1) Society for Imaging Informatics in Medicine COVID-19(SIIM-COVID-19) (2) Valencian Region Medical Image Bank COVID-19 (BIMCV COVID-19)	A detection model using vision transformers achieved high AUC scores of 0.92 and 0.99, respectively	A point-to-point FL (P2PFL) framework based on the Vision Transformer (ViT) model to address the classification of COVID-19 versus normal cases in chest X-ray (CXR) images
Cancer	[20]	(1) Kvasir (2) Camelyon-17	(1) Average accuracy: approximately 83.4% (2) Average accuracy: 81.1%; higher than FedAvg, the FML-proxy	A scheme for DFL called ProxyFL outperforms existing alternatives with much less communication overhead and stronger privacy on cancer diagnostic problems using gigapixel whole-slide histology images
	[69]	Breast cancer Wisconsin (BCW)	After 25 iterations, the prediction accuracy reaches 93.71%	A privacy-preserving framework based on DFL (FL) enhances privacy protection in the healthcare metaverse
	[71]	Breast image data set	FL-CWA achieved slightly higher training accuracy at each learning rate relative to FL-AVG; when the number of attackers increased, FL-CWA could still maintain high training accuracy and achieve low losses, while FL-AVG training accuracy was significant, and the loss gradually increased	Blockchain-based contribution-weighted aggregation FL outperforms centralized learning methods and FL average aggregation in terms of breast image classification model accuracy and system security
	[82]	Brain Tumor Segmentation (BraTS) for brain tumor segmentation	Saving 20% of training time compared to synchronous methods	Federated and decentralized learning tools with MQTT protocol show the reliability in brain tumor segmentation and support smart medical diagnosis
	[103]	Collect colon, head and neck, liver, and ovarian data from (1) The Cancer Genome Atlas (TCGA) (2) Gene Expression Omnibus (GEO)	AdFed outperforms other FL-based methods, achieving a better performance in cancer survival prediction (AUC = 0.605) compared to the average AUC of 0.554	AdFed, an integrated framework based on DFL, performs better than traditional FL in evaluating and predicting the survival of cancer patients

5. Case Studies

DFL has shown significant promise in healthcare applications, particularly in wearable devices, cardiovascular disease prediction, and cancer detection. In these contexts, DFL allows multiple healthcare institutions or wearable devices to train machine learning models collaboratively without sharing sensitive patient data. This ensures privacy and data security while enabling the development of robust models for early diagnosis and personalized treatment plans. DFL also enhances model generalization by leveraging diverse data sets across different demographics and medical conditions.

5.1. Wearable Device Data

Data collected via wearable devices, such as smart bands, are distributed across individual devices. Through DFL, the data collected via these wearable devices remains on the local devices, addressing data ownership issues. Additionally, multiple devices can locally train personalized predictive models, enhancing the model's ability to provide individualized predictions. Jiang et al. [85] utilized the WESAD (Wearable Stress and Affect Detection) data set, which includes data samples of physiological signals, such as heart rate (ECG) and respiration rate (RESP), collected from participants under emotional stimuli (e.g., stress,

pleasure, and anger). DFL allows participants to generate personalized models based on the detection data, providing a more accurate representation of individual stress patterns. Mou et al. [92] employed the Public Human Activity Recognition Dataset (PAMAP2), in which wearable devices recorded activity data across 22 different real-world scenarios (e.g., walking, running, and cycling), increasing the model's applicability in real-world settings. Furthermore, Cong et al. [23] used wearable devices to monitor and record participants' resting heart rate, total sleep duration, and activity-related energy expenditure to assess users' health conditions. In this context, the DFL framework offers privacy protection, allowing users to view their health indicators alongside comparative data from their records or peers while also receiving personalized feedback, all while ensuring the privacy of their data.

In the early diagnosis of complex diseases such as cardiovascular diseases, mental disorders, and cancer, DFL can leverage multi-party data for collaborative training, effectively improving prediction accuracy. Data from various medical institutions can contribute different perspectives, making the global model more accurate in early disease prediction [104].

5.2. Cardiovascular Diseases

Cardiovascular diseases are the leading cause of death globally. Due to patient privacy concerns and the centralized nature of current FL methods, collaborative research on heart disease prediction (HDP) faces significant obstacles in computational health systems [100]. The Physikalisch Technische Bundesanstalt (PTB) Diagnostic ECG Database contains electrocardiogram (ECG) signals from multiple patients, annotated with various types of heart diseases and arrhythmia. Gouissem et al. [67] proposed the use of Robust DFL with Collaborative Decisions under the PTB dataset to control error propagation. The results showed that the model for users converged, achieving an accuracy of over 95% in heart abnormality detection. Additionally, Kuo et al. [56] used the Edinburgh Myocardial Infarction Dataset (EMIDEC), which includes clinical data such as blood pressure, cholesterol levels, and ECG from patients with heart disease. The experimental results demonstrated that, compared to traditional centralized learning methods, DFL allowed participants to optimize the heart disease risk prediction model through local training, leading to higher accuracy in early myocardial infarction diagnosis. Singh et al. [80] employed the MIT-BIH Arrhythmia Dataset, which contains ECG data from multiple patients and is annotated with different types of arrhythmia. The study showed that the DFL model improved utility by 15.1%, 9.03%, and 15.35% over the Block FL (BlockFL), Hierarchical FL (HFEL), and Basic FL (BFL) schemes, respectively, highlighting its superior performance in arrhythmia prediction. Wei et al. [100] conducted experiments using the UCI Heart Disease Dataset for heart disease prediction. The DeFedHDP model mentioned in the research achieved an accuracy of 90% in heart disease prediction, converging faster than the FedAVG (centralized FL) method to this accuracy. Furthermore, the decentralized approach achieved performance similar to that of centralized methods, with better privacy protection.

DFL also plays a significant role in the early warning and prediction of diabetes. Type 1 Diabetes (T1D) patients often struggle to obtain effective Blood Glucose (BG) prediction models due to the lack of sufficient BG data from Continuous Glucose Monitoring (CGM), presenting a significant "cold start" problem in patient care. Piao et al. [101] pioneering work introduced GluADFL, an innovative blood glucose prediction framework employing Asynchronous DFL. Their comprehensive evaluation across four distinct T1D data sets—OhioT1DM, ABC4D, CTR3, and REPLACE-BG (all containing CGM records)—demonstrated the model's superior performance in cross-patient BG level prediction. The asynchronous architecture effectively addresses data heterogeneity while maintaining

prediction accuracy, achieving mean absolute errors 15–20% lower than conventional centralized approaches. Chang et al. [3] developed an adaptive differential privacy mechanism using the Pima Indians Diabetes Database. Their novel algorithm dynamically adjusts gradient noise levels through real-time sensitivity analysis, achieving an optimal balance between privacy preservation and model utility. The experimental results showed 82.7% classification accuracy with a privacy budget $\varepsilon = 3$, representing only a 1.8% accuracy degradation compared to non-private FL (84.5%). This strategic trade-off enables robust privacy protection without substantially compromising diagnostic effectiveness.

5.3. Neurological Disorders

DFL has become crucial for multimodal neurological analysis, addressing key clinical challenges such as epileptic seizure prediction using privacy-preserving EEG data and early Alzheimer's disease screening through collaborative neuroimaging analysis. The EPITEPSIAE and TUH EEG Seizure Corpus (TUSZ) data sets are extensively used in automated seizure detection research. Baghersalimi et al. [10] demonstrated that the DFL (DFL) method, using an ensemble strategy, improved the Gmean metric from 77.58% to 85.83% on the TUSZ data set, and from 80.77% to 88.72% on the EPITEPSIAE data set, outperforming traditional FedAvg. This shows that DFL better handles non-IID data, improving detection accuracy. Gmean (geometric mean) is a key metric for binary classification, particularly in imbalanced data sets, balancing precision and recall to assess model performance. Lu et al. [79] conducted experiments with a clinical data set containing EHR data from Alzheimer's and mild cognitive impairment (MCI) patients from 20 hospitals. Their results showed that using the DFL framework with decentralized stochastic gradient tracking (DSGT) successfully addressed non-convex optimization issues. DSGT, which introduces auxiliary variables to improve algorithm performance, demonstrated faster convergence and reduced errors from data distribution differences compared to traditional decentralized stochastic gradient descent (DSGD).

5.4. Respiratory Diseases

Pennisi et al. [70] used the Montgomery County X-ray set and the Shenzhen Hospital X-ray set, combining experience replay with privacy-preserving Generative Adversarial Networks (GANs) in a DFL approach. Their method achieved an accuracy of 83.41% in tuberculosis classification, which is close to the accuracy of centralized training, and outperformed traditional FL methods such as FedAvg and FedProx. Chetoui et al. [102] proposed a point-to-point FL (P2PFL) framework based on the Vision Transformer (ViT) model to address the classification of COVID-19 versus normal cases in chest X-ray (CXR) images. Specifically, a detection model using vision transformers was applied to two data sets, SIIM-COVID-19 and BIMCV COVID-19, which include chest X-rays and CT scan images. The model achieved high AUC scores of 0.92 and 0.99, respectively.

5.5. Cancer

DFL is widely used for early cancer screening and personalized treatment recommendations. By analyzing patient data and treatment history, it can perform prognostic assessments for various scenarios, including breast cancer, lung cancer, melanoma detection, and brain tumor segmentation.

Tissue section images are essential imaging tools in pathology. The TissueMNIST data set contains images of tumor and normal tissue sections, which are utilized for cancer diagnosis. Research shows that the Blockchain-Based FL global model can improve overall model performance by up to 10% compared to a single machine learning model in TissueMNIST [68].

Accurately assessing patient survival is one of the key objectives in precision oncology [105]. Predictive models can assist physicians in selecting the best subsequent interventions based on individual patient characteristics. Chai et al. [103] collected cancer data from four types of cancers—colon, head and neck, liver, and ovarian—using The Cancer Genome Atlas (TCGA) and Gene Expression Omnibus (GEO) data bases. They employed a DFL framework, AdFed, to assess patient survival. The experimental results show that AdFed, utilizing distributed data, outperforms other FL-based methods, achieving a better performance in cancer survival prediction (AUC = 0.605) compared to the average AUC of 0.554 for the compared methods. Additionally, regularization methods were employed to help researchers strike a balance between data accessibility and patient privacy protection.

Pennisi et al. [70] utilized the ISIC 2019 Challenge data set, which includes skin lesion images for the automated detection and classification of skin lesions and promoting the early diagnosis and screening of skin cancer, particularly melanoma. The study demonstrated that DFL, combining experience replay and privacy-preserving generative adversarial networks (GANs), outperforms centralized FL methods such as FedAvg and FedProx in melanoma classification tasks, achieving higher accuracy. This highlights the promising potential of DFL in medical image analysis.

Kang et al. [69] applied DFL to the Wisconsin Diagnostic Breast Cancer (WDBC) data set for the benign-malignant classification of breast cancer, supporting the automation of cancer diagnosis methods. In the breast cancer prediction task, the model's accuracy improved from 87% using traditional FL methods (FedAvg) to 92.3%. By employing strategies such as experience replay, the study effectively mitigated the non-independent and identically distributed (non-i.i.d.) problem present in the WDBC data set, ensuring good generalization across multiple client nodes.

The Brain Tumor Segmentation (BraTS) data set is widely used in medical image analysis. Tedeschini et al. [82] showed that, compared to centralized learning (CL) methods, DFL improved automated brain tumor detection and segmentation using MRI images. The Dice Similarity Coefficient (DSC) for tumor region detection increased by 1.6%, indicating better segmentation accuracy. The experiments demonstrated that decentralized learning can effectively leverage diverse data sources, enhancing the model's generalization and segmentation performance.

6. Open Issues and Research Directions

DFL has broad prospects in the medical field. It can improve privacy protection, promote multi-center data collaboration, and promote personalized medicine. However, its widespread application still faces compatibility issues caused by system heterogeneity, data heterogeneity affecting model generalization ability, communication efficiency limiting system performance, and lack of effective incentive mechanism reducing participation enthusiasm. Future development directions mainly focus on optimizing data security strategies, enhancing cross-institutional collaboration capabilities, improving communication protocols, and establishing reasonable incentive mechanisms to promote the in-depth application of FL in the medical field.

6.1. Security and Privacy

Regarding data sharing and compliance across medical institutions, the blockchain-based data sharing framework combines multi-party secure computing and a trusted execution environment, which can ensure that data cannot be tampered with. Still, the storage and computing costs are high. How to balance security and computing costs is a direction worth exploring. In addition, although differential privacy and homomorphic

encryption can enhance privacy protection, they will reduce model accuracy and training efficiency. The privacy–accuracy balance can be optimized by dynamically combining adaptive differential privacy adjusting noise [106]. The proxy model can isolate sensitive data but may still leak feature distribution. It can be combined with federated distillation and feature perturbation to enhance security [107]. In addition, to prevent malicious nodes from inferring private information through model training, adversarial sample generation and decentralized anomaly detection mechanisms are combined to improve security. Identity management conflicts with anonymity when nodes join or leave. Complete anonymity is prone to malicious use, while traditional identity authentication faces the risk of centralization. Future research directions can combine zero-knowledge proof and group signature to achieve a balance [108].

6.2. Communication Efficiency and Cost

Due to the particularity of the medical environment, the traditional decentralized learning framework is susceptible to delays and bandwidth fluctuations, resulting in a decrease in the model convergence speed. Knowledge transfer that relies on knowledge distillation lacks robustness in scenarios with dynamic network topologies or frequent node joining and exiting, resulting in training interruptions or reduced efficiency. To this end, a dynamic scheduling mechanism based on reinforcement learning can be combined to adaptively adjust the communication frequency and parameter synchronization strategy according to the network status. At the same time, the dynamic topological features of the graph neural network (GNN) modeling are combined to enhance the adaptability of decentralized knowledge distillation in complex networks. In addition, the cut-through method simplifies transaction records; it does not effectively address the impact of dynamic node changes on consensus efficiency. The PushSum algorithm relies on a fixed weighted adjacency matrix and is difficult to adapt to changing network structures in real time. Future work may focus more on balancing efficiency and performance in a dynamic node environment, reducing communication costs while avoiding precision losses. Through dynamic network adaptability optimization such as the dynamic hierarchical quantization [109] and adaptive knowledge distillation (AKD) methods [110], model performance and communication efficiency are weighed to obtain scalability support in dynamic network structures.

6.3. Data and Model Heterogeneity

The current method has alleviated the heterogeneity problem to a certain extent, but it still faces the challenges of computational complexity and data authenticity. Future improvements may include the introduction of feature attention mechanisms to help clients screen the most useful features and effectively aggregate information from other nodes to improve the adaptability of the global model. In addition, dynamic regularization terms can be used to constrain the update direction of local models [111], thereby alleviating the impact of data heterogeneity on model training. At the same time, the introduction of contrastive loss can enhance the robustness of the model to non-IID data and improve generalization capabilities [112,113]. At the same time, to address the fusion problem of heterogeneous models, lightweight algorithms can be designed to achieve parameter aggregation, and cross-model knowledge transfer can be achieved through unsupervised representation learning such as FedProc [114] or meta-learning such as FedMeta [115], thereby improving the applicability of FL in heterogeneous environments.

6.4. Incentive Mechanisms

In the medical field, the balance between data privacy and incentive mechanisms is particularly critical. Although medical data sharing helps promote medical research,

individuals may worry about their data being abused or leaked due to the high sensitivity of medical data. Therefore, traditional incentive mechanisms, such as encouraging data sharing through monetary rewards or resource allocation, may struggle to overcome people's concerns about privacy leaks. DFL based on blockchain reward mechanisms usually relies on fixed reward blockchain points, without considering dynamic contribution evaluation, such as malicious nodes pretending to have high contributions to defraud rewards. It is a direction worth paying attention to in order to balance incentives and security while incentivizing high-contribution nodes and defending against Sybil attacks. In addition, traditional incentive mechanisms, such as contribution evaluation methods based on Shapley values, have high computational complexity and are difficult to adjust strategies in real time. It is a direction worth paying attention to in order to establish a master–slave game model between task publishing nodes and participating nodes and dynamically adjust payment budgets and contributions. In addition, existing methods such as POS consensus focus on single-round reward distribution and lack long-term incentives for node reputation and historical behavior. A hybrid incentive strategy that combines economic rewards such as tokens and non-economic incentives such as reputation levels, and a phased reward mechanism, can help encourage high-quality nodes to participate in the long term.

7. Conclusions

FL encounters various challenges, including central orchestration vulnerability in the classical client–server architecture, leading to single-point-of-failure risks. To tackle these issues, decentralized FL solutions (DFLs) have surfaced, enabling FL clients to collaborate and communicate without relying on a central server. We have aimed to offer a comprehensive, well-defined, and systematic perspective that organizes and synthesizes the existing literature and definitions, facilitating a comprehensive introduction to DFL for new researchers. This paper defines and discusses two different methods for building DFL: traditional distributed (TD-FL) methods and blockchain-based (BC-FL) methods. We extensively explored the DFL challenges covering security and privacy, communication efficiency, data and model heterogeneity, incentive mechanisms, and potential solutions. We have discussed various solutions that can mitigate these issues, such as advanced cryptographic techniques, blockchain integration, and adaptive learning models. It is important to note that our research uncovers a considerable number of diverse usage scenarios. DFL's ability to support personalized healthcare treatments through large, distributed data sets from multiple medical institutions demonstrates its potential for improving healthcare delivery. The systematic literature review fills an important gap in understanding the current landscape of DFL in healthcare, offering valuable insights for future research and implementation efforts in securing and optimizing FL systems within the medical sector.

Funding: This research is supported by the National Key R&D Program of China Grant No. 2022ZD0116800, Shandong Provincial Natural Science Foundation No. ZR202211150015, Taishan Scholars Program No. TSQNZ20230621 and TSQN202211214, Shandong Excellent Young Scientists Fund Program (Overseas) No. 2023HWYQ-113.

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Aceto, G.; Persico, V.; Pescapé, A. Industry 4.0 and health: Internet of things, big data, and cloud computing for healthcare 4.0. *J. Ind. Inf. Integr.* **2020**, *18*, 100129. [CrossRef]

2. Guo, H.; Li, W.; Meese, C.; Nejad, M. Decentralized Electronic Health Records Management via Redactable Blockchain and Revocable IPFS. In Proceedings of the 2024 IEEE/ACM Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), Wilmington, DE, USA, 19–21 June 2024; pp. 167–171.
3. Chang, Y.; Fang, C.; Sun, W. A Blockchain-Based Federated Learning Method for Smart Healthcare. *Comput. Intell. Neurosci.* **2021**, *2021*, 4376418. [CrossRef] [PubMed]
4. Carlos Ferreira, J.; Elvas, L.B.; Correia, R.; Mascarenhas, M. Enhancing EHR Interoperability and Security through Distributed Ledger Technology: A Review. *Healthcare* **2024**, *12*, 1967. [CrossRef] [PubMed]
5. Yang, Q.; Liu, Y.; Chen, T.; Tong, Y. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol. (TIST)* **2019**, *10*, 1–19. [CrossRef]
6. de Gomez, M.R.C. A Comprehensive Introduction to Healthcare Data Analytics. *J. Biomed. Sustain. Healthc. Appl.* **2024**, 44–53.
7. Konečnỳ, J.; McMahan, H.B.; Yu, F.X.; Richtárik, P.; Suresh, A.T.; Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv* **2016**, arXiv:1610.05492.
8. Zhang, Y.; Gai, K.; Qiu, M.; Ding, K. Understanding privacy-preserving techniques in digital cryptocurrencies. In Proceedings of the Algorithms and Architectures for Parallel Processing: 20th International Conference, ICA3PP 2020, New York, NY, USA, 2–4 October 2020; Proceedings, Part III 20; pp. 3–18.
9. Ran, A.R.; Wang, X.; Chan, P.P.; Wong, M.O.; Yuen, H.; Lam, N.M.; Chan, N.C.; Yip, W.W.; Young, A.L.; Yung, H.W.; et al. Developing a privacy-preserving deep learning model for glaucoma detection: A multicentre study with federated learning. *Br. J. Ophthalmol.* **2024**, *108*, 1114–1123. [CrossRef]
10. Baghersalimi, S.; Teijeiro, T.; Aminifar, A.; Atienza, D. Decentralized federated learning for epileptic seizures detection in low-power wearable systems. *IEEE Trans. Mob. Comput.* **2023**, *23*, 6392–6407. [CrossRef]
11. Heiyanthuduwage, S.R.; Altas, I.; Bewong, M.; Islam, M.Z.; Deho, O.B. Decision trees in federated learning: Current state and future opportunities. *IEEE Access* **2024**, *12*, 127943–127965. [CrossRef]
12. Santhosh, G.; De Vita, F.; Bruneo, D.; Longo, F.; Puliafito, A. Towards trustless prediction-as-a-service. In Proceedings of the 2019 IEEE International Conference on Smart Computing (SMARTCOMP), Washington, DC, USA, 12–15 June 2019; pp. 317–322.
13. Ma, C.; Li, J.; Ding, M.; Yang, H.H.; Shu, F.; Quek, T.Q.; Poor, H.V. On safeguarding privacy and security in the framework of federated learning. *IEEE Netw.* **2020**, *34*, 242–248. [CrossRef]
14. Zhang, H.; Li, G.; Zhang, Y.; Gai, K.; Qiu, M. Blockchain-based privacy-preserving medical data sharing scheme using federated learning. In Proceedings of the Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, 14–16 August 2021; Proceedings, Part III 14; pp. 634–646.
15. Bai, L.; Hu, H.; Ye, Q.; Li, H.; Wang, L.; Xu, J. Membership Inference Attacks and Defenses in Federated Learning: A Survey. *ACM Comput. Surv.* **2024**, *57*, 1–35. [CrossRef]
16. Yang, L.; Lu, Y.; Cao, J.; Huang, J.; Zhang, M. E-tree learning: A novel decentralized model learning framework for edge ai. *IEEE Internet Things J.* **2021**, *8*, 11290–11304. [CrossRef]
17. Kang, H.S.; Chai, Z.Y.; Li, Y.L.; Huang, H.; Zhao, Y.J. Edge computing in Internet of Vehicles: A federated learning method based on Stackelberg dynamic game. *Inf. Sci.* **2025**, *689*, 121452. [CrossRef]
18. Li, H.; Ge, L.; Tian, L. Survey: Federated learning data security and privacy-preserving in edge-Internet of Things. *Artif. Intell. Rev.* **2024**, *57*, 130. [CrossRef]
19. Stripelis, D.; Saleem, H.; Ghai, T.; Dhinagar, N.; Gupta, U.; Anastasiou, C.; Ver Steeg, G.; Ravi, S.; Naveed, M.; Thompson, P.M.; et al. Secure neuroimaging analysis using federated learning with homomorphic encryption. In Proceedings of the 17th International Symposium on Medical Information Processing and Analysis, Campinas, Brazil, 17–19 November 2021; Volume 12088, pp. 351–359.
20. Kalra, S.; Wen, J.; Cresswell, J.C.; Volkovs, M.; Tizhoosh, H.R. Decentralized federated learning through proxy model sharing. *Nat. Commun.* **2023**, *14*, 2899. [CrossRef]
21. Ben Shoham, O.; Rappoport, N. Federated learning of medical concepts embedding using behrt. *JAMIA Open* **2024**, *7*, ooae110. [CrossRef]
22. Babar, M.; Qureshi, B.; Koubaa, A. Review on Federated Learning for digital transformation in healthcare through big data analytics. In *Future Generation Computer Systems*; Elsevier: Amsterdam, The Netherlands, 2024.
23. Cong, R.; Ye, Y.; Wu, J.; Li, Y.; Chen, Y.; Bian, Y.; Tago, K.; Nishimura, S.; Ogihara, A.; Jin, Q. A Trustworthy Decentralized System for Health Data Integration and Sharing: Design and Experimental Validation. In Proceedings of the International Conference on Human-Computer Interaction, Copenhagen, Denmark, 23–28 July 2023; pp. 125–134.
24. Pavlova, E.; Melnikov, G.; Yanovich, Y.; Frolov, A. Unlocking potential of open source model training in decentralized federated learning environment. In *Blockchain: Research and Applications*; Elsevier: Amsterdam, The Netherlands, 2025; p. 100264.
25. Samikwa, E.; Di Maio, A.; Braun, T. DFL: Dynamic federated split learning in heterogeneous IoT. *IEEE Trans. Mach. Learn. Commun. Netw.* **2024**, *2*, 733–752. [CrossRef]
26. Gabrielli, E.; Pica, G.; Tolomei, G. A survey on decentralized federated learning. *arXiv* **2023**, arXiv:2308.04604.

27. Arthi, N.T.; Mubin, K.E.; Rahman, J.; Rafi, G.; Sheja, T.T.; Reza, M.T.; Alam, M.A. Decentralized federated learning and deep learning leveraging xai-based approach to classify colorectal cancer. In Proceedings of the 2022 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 18–20 December 2022; pp. 1–6.
28. Yuan, L.; Wang, Z.; Sun, L.; Philip, S.Y.; Brinton, C.G. Decentralized federated learning: A survey and perspective. *IEEE Internet Things J.* **2024**, *11*, 34617–34638. [CrossRef]
29. Beltrán, E.T.M.; Pérez, M.Q.; Sánchez, P.M.S.; Bernal, S.L.; Bovet, G.; Pérez, M.G.; Pérez, G.M.; Celdrán, A.H. Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges. *IEEE Commun. Surv. Tutorials* **2023**, *25*, 2983–3013. [CrossRef]
30. Liu, Y.; Ai, Z.; Sun, S.; Zhang, S.; Liu, Z.; Yu, H. Fedcoin: A peer-to-peer payment system for federated learning. In *Federated Learning: Privacy and Incentive*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 125–138.
31. Van Truong, V.; Quan, P.K.; Park, D.H.; Kim, T. Performance Evaluation of Decentralized Federated Learning: Impact of Fully and K-Connected Topologies, Heterogeneous Computing Resources, and Communication Bandwidth. *IEEE Access* **2025**, *13*, 32741–32755. [CrossRef]
32. Xiao, Y.; Ye, Y.; Huang, S.; Hao, L.; Ma, Z.; Xiao, M.; Mumtaz, S.; Dobre, O.A. Fully decentralized federated learning-based on-board mission for UAV swarm system. *IEEE Commun. Lett.* **2021**, *25*, 3296–3300. [CrossRef]
33. Chung, W.C.; Lin, Y.H.; Luo, J.A. Ring-Based Decentralized Federated Learning with Cosine Similarity Grouping. In Proceedings of the 2024 International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), Taichung, Taiwan, 9–11 July 2024; pp. 113–114.
34. Iansiti, M.; Lakhani, K.R. The truth about blockchain. *Harv. Bus. Rev.* **2017**, *95*, 118–127.
35. Wang, M.; Zhu, T.; Zuo, X.; Ye, D.; Yu, S.; Zhou, W. Blockchain-based gradient inversion and poisoning defense for federated learning. *IEEE Internet Things J.* **2023**, *11*, 15667–15681. [CrossRef]
36. Qu, Y.; Uddin, M.P.; Gan, C.; Xiang, Y.; Gao, L.; Yearwood, J. Blockchain-enabled federated learning: A survey. *ACM Comput. Surv.* **2022**, *55*, 1–35. [CrossRef]
37. Andrew, J.; Isravel, D.P.; Sagayam, K.M.; Bhushan, B.; Sei, Y.; Eunice, J. Blockchain for healthcare systems: Architecture, security challenges, trends and future directions. *J. Netw. Comput. Appl.* **2023**, *215*, 103633.
38. Qu, Y.; Pokhrel, S.R.; Garg, S.; Gao, L.; Xiang, Y. A blockchained federated learning framework for cognitive computing in industry 4.0 networks. *IEEE Trans. Ind. Inform.* **2020**, *17*, 2964–2973. [CrossRef]
39. Lin, F.; Xia, S.; Qi, J.; Tang, C.; Zheng, Z.; Yu, X. A parking sharing network over blockchain with proof-of-planned-behavior consensus protocol. *IEEE Trans. Veh. Technol.* **2022**, *71*, 8124–8136. [CrossRef]
40. Qin, Z.; Yan, X.; Zhou, M.; Deng, S. BlockDFL: A blockchain-based fully decentralized peer-to-peer federated learning framework. In Proceedings of the ACM Web Conference 2024, Virtual, 13–17 May 2024; pp. 2914–2925.
41. Jin, H.; Dai, X.; Xiao, J.; Li, B.; Li, H.; Zhang, Y. Cross-cluster federated learning and blockchain for internet of medical things. *IEEE Internet Things J.* **2021**, *8*, 15776–15784. [CrossRef]
42. Zhou, X.; Huang, W.; Liang, W.; Yan, Z.; Ma, J.; Pan, Y.; Kevin, I.; Wang, K. Federated distillation and blockchain empowered secure knowledge sharing for internet of medical things. *Inf. Sci.* **2024**, *662*, 120217. [CrossRef]
43. Zhaofeng, M.; Xiaochang, W.; Jain, D.K.; Khan, H.; Hongmin, G.; Zhen, W. A blockchain-based trusted data management scheme in edge computing. *IEEE Trans. Ind. Inform.* **2019**, *16*, 2013–2021. [CrossRef]
44. Ray, P.P.; Chowhan, B.; Kumar, N.; Almogren, A. BIoTHR: Electronic health record servicing scheme in IoT-blockchain ecosystem. *IEEE Internet Things J.* **2021**, *8*, 10857–10872. [CrossRef]
45. Gilad, Y.; Hemo, R.; Micali, S.; Vlachos, G.; Zeldovich, N. Algorand: Scaling byzantine agreements for cryptocurrencies. In Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai, China, 28–31 October 2017; pp. 51–68.
46. Weng, J.; Weng, J.; Zhang, J.; Li, M.; Zhang, Y.; Luo, W. Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive. *IEEE Trans. Dependable Secur. Comput.* **2019**, *18*, 2438–2455. [CrossRef]
47. Kim, H.; Park, J.; Bennis, M.; Kim, S.L. Blockchained on-device federated learning. *IEEE Commun. Lett.* **2019**, *24*, 1279–1283. [CrossRef]
48. Feng, L.; Zhao, Y.; Guo, S.; Qiu, X.; Li, W.; Yu, P. BAFL: A blockchain-based asynchronous federated learning framework. *IEEE Trans. Comput.* **2021**, *71*, 1092–1103. [CrossRef]
49. Jatain, D.; Singh, V.; Dahiya, N. Blockchain Base Community Cluster-Federated Learning for Secure Aggregation of Healthcare Data. *Procedia Comput. Sci.* **2022**, *215*, 752–762. [CrossRef]
50. Wang, Z.; Hu, Y.; Xiao, J.; Wu, C. Efficient ring-topology decentralized federated learning with deep generative models for industrial artificial intelligent. *arXiv* **2021**, arXiv:2104.08100.
51. Feng, C.; Liu, B.; Yu, K.; Goudos, S.K.; Wan, S. Blockchain-empowered decentralized horizontal federated learning for 5G-enabled UAVs. *IEEE Trans. Ind. Inform.* **2021**, *18*, 3582–3592. [CrossRef]
52. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and open problems in federated learning. *Found. Trends® Mach. Learn.* **2021**, *14*, 1–210. [CrossRef]

53. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [CrossRef]
54. Shayan, M.; Fung, C.; Yoon, C.J.; Beschastnikh, I. Biscotti: A blockchain system for private and secure federated learning. *IEEE Trans. Parallel Distrib. Syst.* **2020**, *32*, 1513–1525. [CrossRef]
55. Lu, Y.; Huang, X.; Dai, Y.; Maharjan, S.; Zhang, Y. Blockchain and federated learning for privacy-preserved data sharing in industrial IoT. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4177–4186. [CrossRef]
56. Kuo, T.T.; Pham, A. Detecting model misconducts in decentralized healthcare federated learning. *Int. J. Med. Inform.* **2022**, *158*, 104658. [CrossRef] [PubMed]
57. Cui, L.; Qu, Y.; Xie, G.; Zeng, D.; Li, R.; Shen, S.; Yu, S. Security and privacy-enhanced federated learning for anomaly detection in IoT infrastructures. *IEEE Trans. Ind. Inform.* **2021**, *18*, 3492–3500. [CrossRef]
58. Wan, Y.; Qu, Y.; Ni, W.; Xiang, Y.; Gao, L.; Hossain, E. Data and model poisoning backdoor attacks on wireless federated learning, and the defense mechanisms: A comprehensive survey. *IEEE Commun. Surv. Tutorials* **2024**, *26*, 1861–1897. [CrossRef]
59. Gholami, A.; Torkzaban, N.; Baras, J.S. Trusted decentralized federated learning. In Proceedings of the 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 8–11 January 2022; pp. 1–6.
60. Yang, Z.; Shi, Y.; Zhou, Y.; Wang, Z.; Yang, K. Trustworthy federated learning via blockchain. *IEEE Internet Things J.* **2022**, *10*, 92–109. [CrossRef]
61. Li, Z.; Yu, H.; Zhou, T.; Luo, L.; Fan, M.; Xu, Z.; Sun, G. Byzantine resistant secure blockchained federated learning at the edge. *IEEE Netw.* **2021**, *35*, 295–301. [CrossRef]
62. Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating noise to sensitivity in private data analysis. In Proceedings of the Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, 4–7 March 2006; Proceedings 3; pp. 265–284.
63. Yin, C.; Xi, J.; Sun, R.; Wang, J. Location privacy protection based on differential privacy strategy for big data in industrial internet of things. *IEEE Trans. Ind. Inform.* **2017**, *14*, 3628–3636. [CrossRef]
64. Hallaji, E.; Razavi-Far, R.; Saif, M.; Wang, B.; Yang, Q. Decentralized federated learning: A survey on security and privacy. *IEEE Trans. Big Data* **2024**, *10*, 194–213. [CrossRef]
65. McSherry, F.; Talwar, K. Mechanism design via differential privacy. In Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), Providence, RI, USA, 21–23 October 2007; pp. 94–103.
66. Paillier, P. Public-key cryptosystems based on composite degree residuosity classes. In Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques, Prague, Czech Republic, 2–6 May 1999; pp. 223–238.
67. Gouissem, A.; Abualsaud, K.; Yaacoub, E.; Khattab, T.; Guizani, M. Robust decentralized federated learning using collaborative decisions. In Proceedings of the 2022 International Wireless Communications and Mobile Computing (IWCMC), Dubrovnik, Croatia, 30 May–3 June 2022; pp. 254–258.
68. Kalapaaking, A.P.; Khalil, I.; Yi, X. Blockchain-based federated learning with SMPC model verification against poisoning attack for healthcare systems. *IEEE Trans. Emerg. Top. Comput.* **2023**, *12*, 269–280. [CrossRef]
69. Kang, J.; Wen, J.; Ye, D.; Lai, B.; Wu, T.; Xiong, Z.; Nie, J.; Niyato, D.; Zhang, Y.; Xie, S. Blockchain-empowered federated learning for healthcare metaverses: User-centric incentive mechanism with optimal data freshness. *IEEE Trans. Cogn. Commun. Netw.* **2023**, *10*, 348–362. [CrossRef]
70. Pennisi, M.; Salanitri, F.P.; Bellitto, G.; Spampinato, C.; Palazzo, S.; Casella, B.; Aldinucci, M. Experience Replay as an Effective Strategy for Optimizing Decentralized Federated Learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 3376–3383.
71. Chen, Y.; Lin, F.; Chen, Z.; Tang, C.; Jia, R.; Li, M. Blockchain-based Federated Learning with Contribution-Weighted Aggregation for Medical Data Modeling. In Proceedings of the 2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS), Denver, CO, USA, 20–22 October 2022; pp. 606–612.
72. Roy, S.; Bera, D. A blockchain-based Verifiable Aggregation for Federated Learning and Secure Sharing in Healthcare. In Proceedings of the 2023 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Jaipur, India, 17–20 December 2023; pp. 165–170.
73. Moulahi, W.; Jdey, I.; Moulahi, T.; Alawida, M.; Alabdulatif, A. A blockchain-based federated learning mechanism for privacy preservation of healthcare IoT data. *Comput. Biol. Med.* **2023**, *167*, 107630. [CrossRef] [PubMed]
74. El Rifai, O.; Biotteau, M.; de Boissezon, X.; Megdiche, I.; Ravat, F.; Teste, O. Blockchain-based federated learning in medicine. In Proceedings of the Artificial Intelligence in Medicine: 18th International Conference on Artificial Intelligence in Medicine, AIME 2020, Minneapolis, MN, USA, 25–28 August 2020; Proceedings 18; pp. 214–224.
75. Wang, Z.; Hu, Y.; Yan, S.; Wang, Z.; Hou, R.; Wu, C. Efficient ring-topology decentralized federated learning with deep generative models for medical data in ehealthcare systems. *Electronics* **2022**, *11*, 1548. [CrossRef]

76. Wang, C.; Wang, S.; Zhao, C.; Wang, W.; Hu, B.; Wang, Y.; Wang, L.; Chen, Z. Decentralized Reinforced Anonymous FLchain: a Secure Federated Learning Architecture for the Medical Industry. In Proceedings of the 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), Torino, Italy, 26–30 June 2023; pp. 396–405.
77. Rahman, M.A.; Hossain, M.S.; Islam, M.S.; Alrajeh, N.A.; Muhammad, G. Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach. *IEEE Access* **2020**, *8*, 205071–205087. [CrossRef]
78. Li, W.; Feng, C.; Zhang, L.; Xu, H.; Cao, B.; Imran, M.A. A scalable multi-layer PBFT consensus for blockchain. *IEEE Trans. Parallel Distrib. Syst.* **2020**, *32*, 1146–1160. [CrossRef]
79. Lu, S.; Zhang, Y.; Wang, Y. Decentralized federated learning for electronic health records. In Proceedings of the 2020 54th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 18–20 March 2020; pp. 1–5.
80. Singh, M.B.; Singh, H.; Pratap, A. Energy-Efficient and Privacy-Preserving Blockchain Based Federated Learning for Smart Healthcare System. *IEEE Trans. Serv. Comput.* **2023**, *17*, 2392–2403. [CrossRef]
81. Nguyen, T.; Dakka, M.; Diakiw, S.; VerMilyea, M.; Perugini, M.; Hall, J.; Perugini, D. A novel decentralized federated learning approach to train on globally distributed, poor quality, and protected private medical data. *Sci. Rep.* **2022**, *12*, 8888. [CrossRef]
82. Tedeschini, B.C.; Savazzi, S.; Stoklasa, R.; Barbieri, L.; Stathopoulos, I.; Nicoli, M.; Serio, L. Decentralized federated learning for healthcare networks: A case study on tumor segmentation. *IEEE Access* **2022**, *10*, 8693–8708. [CrossRef]
83. Zhang, W.; Li, P.; Wu, G.; Li, J. Privacy-Preserving Deep Learning in Internet of Healthcare Things with Blockchain-Based Incentive. In Proceedings of the International Conference on Knowledge Science, Engineering and Management, Singapore, 6–8 August 2022; pp. 302–315.
84. Lian, Z.; Wang, W.; Han, Z.; Su, C. Blockchain-based personalized federated learning for internet of medical things. *IEEE Trans. Sustain. Comput.* **2023**, *8*, 694–702. [CrossRef]
85. Jiang, S.; Firouzi, F.; Chakrabarty, K. Low-overhead clustered federated learning for personalized stress monitoring. *IEEE Internet Things J.* **2023**, *11*, 4335–4347. [CrossRef]
86. Tian, Y.; Wang, S.; Xiong, J.; Bi, R.; Zhou, Z.; Bhuiyan, M.Z.A. Robust and privacy-preserving decentralized deep federated learning training: Focusing on digital healthcare applications. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2023**, *21*, 890–901. [CrossRef]
87. Lian, Z.; Yang, Q.; Wang, W.; Zeng, Q.; Alazab, M.; Zhao, H.; Su, C. DEEP-FEL: Decentralized, efficient and privacy-enhanced federated edge learning for healthcare cyber physical systems. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3558–3569. [CrossRef]
88. Zhang, W.; Lu, Q.; Yu, Q.; Li, Z.; Liu, Y.; Lo, S.K.; Chen, S.; Xu, X.; Zhu, L. Blockchain-based federated learning for device failure detection in industrial IoT. *IEEE Internet Things J.* **2020**, *8*, 5926–5937. [CrossRef]
89. Peng, Z.; Xu, J.; Chu, X.; Gao, S.; Yao, Y.; Gu, R.; Tang, Y. Vfchain: Enabling verifiable and auditable federated learning via blockchain systems. *IEEE Trans. Netw. Sci. Eng.* **2021**, *9*, 173–186. [CrossRef]
90. Hu, C.; Jiang, J.; Wang, Z. Decentralized federated learning: A segmented gossip approach. *arXiv* **2019**, arXiv:1908.07782.
91. Hinton, G. Distilling the Knowledge in a Neural Network. *arXiv* **2015**, arXiv:1503.02531.
92. Mou, T.; Jiang, X.; Li, J.; Yan, B.; Chen, Q.; Zhang, T.; Huang, W.; Gao, C.; Chen, Y. FedTAM: Decentralized Federated Learning with a Feature Attention Based Multi-teacher Knowledge Distillation for Healthcare. In Proceedings of the 2023 IEEE 29th International Conference on Parallel and Distributed Systems (ICPADS), Danzhou City, 17–21 December 2023; pp. 1246–1253.
93. Ratwani, R.M.; Sutton, K.; Galarraga, J.E. Addressing AI algorithmic bias in health care. *JAMA* **2024**, *332*, 1051–1052. [CrossRef]
94. Veeramachaneni, V. Edge Computing: Architecture, Applications, and Future Challenges in a Decentralized Era. *Recent Trends Comput. Graph. Multimed. Technol.* **2025**, *7*, 8–23.
95. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
96. Ng, D.; Lan, X.; Yao, M.M.S.; Chan, W.P.; Feng, M. Federated learning: A collaborative effort to achieve better medical imaging models for individual sites that have small labelled datasets. *Quant. Imaging Med. Surg.* **2021**, *11*, 852. [CrossRef]
97. Shapley, L.S. A value for n-person games. *Contrib. Theory Games* **1953**, *2*, 307–317.
98. Zhan, Y.; Li, P.; Qu, Z.; Zeng, D.; Guo, S. A learning-based incentive mechanism for federated learning. *IEEE Internet Things J.* **2020**, *7*, 6360–6368. [CrossRef]
99. Kosta, A.; Pappas, N.; Angelakis, V. Age of information: A new concept, metric, and tool. *Found. Trends® Netw.* **2017**, *12*, 162–259. [CrossRef]
100. Wei, M.; Yang, J.; Zhao, Z.; Zhang, X.; Li, J.; Deng, Z. Defedhdp: Fully decentralized online federated learning for heart disease prediction in computational health systems. *IEEE Trans. Comput. Soc. Syst.* **2024**, *11*, 6854–6867. [CrossRef]
101. Piao, C.; Zhu, T.; Wang, Y.; Baldeweg, S.E.; Taylor, P.; Georgiou, P.; Sun, J.; Wang, J.; Li, K. Privacy Preserved Blood Glucose Level Cross-Prediction: An Asynchronous Decentralized Federated Learning Approach. *arXiv* **2024**, arXiv:2406.15346.

102. Chetoui, M.; Akhloufi, M.A. Peer-to-peer federated learning for COVID-19 detection using transformers. *Computers* **2023**, *12*, 106. [CrossRef]
103. Chai, H.; Huang, Y.; Xu, L.; Song, X.; He, M.; Wang, Q. A decentralized federated learning-based cancer survival prediction method with privacy protection. *Heliyon* **2024**, *10*, e31873. [CrossRef]
104. Adeniran, I.A.; Efunniyi, C.P.; Osundare, O.S.; Abhulimen, A.O. Data-driven decision-making in healthcare: Improving patient outcomes through predictive modeling. *Eng. Sci. Technol. J.* **2024**, *5*, 59–67.
105. Wong, E.S.; Choy, R.W.; Zhang, Y.; Chu, W.K.; Chen, L.J.; Pang, C.P.; Yam, J.C. Global retinoblastoma survival and globe preservation: A systematic review and meta-analysis of associations with socioeconomic and health-care factors. *Lancet Glob. Health* **2022**, *10*, e380–e389. [CrossRef]
106. Xie, H.; Zhang, Y.; Zhongwen, Z.; Zhou, H. Privacy-Preserving Medical Data Collaborative Modeling: A Differential Privacy Enhanced Federated Learning Framework. *J. Knowl. Learn. Sci. Technol.* **2024**, *3*, 340–350.
107. Chen, R.; Xia, H.; Wang, K.; Xu, S.; Zhang, R. KDRSFL: A knowledge distillation resistance transfer framework for defending model inversion attacks in split federated learning. *Future Gener. Comput. Syst.* **2025**, *166*, 107637. [CrossRef]
108. Liu, H.; Wei, J.; Xu, Z.; Zhao, Z. Authentication and Traceability for Federated Learning Models via Group Signatures. 2024. Available online: https://www.researchsquare.com/article/rs-4867383/v1 (accessed on 30 March 2025).
109. Azimi-Abarghouyi, S.M.; Fodor, V. Quantized hierarchical federated learning: A robust approach to statistical heterogeneity. *arXiv* **2024**, arXiv:2403.01540.
110. Zheng, S.; Hu, J.; Min, G.; Li, K. Mutual Knowledge Distillation based Personalized Federated Learning for Smart Edge Computing. *IEEE Trans. Consum. Electron.* **2024**, *1*. [CrossRef]
111. Qu, Y.; Ding, M.; Sun, N.; Thilakarathna, K.; Zhu, T.; Niyato, D. The Frontier of Data Erasure: A Survey on Machine Unlearning for Large Language Models. *Computer* **2025**, *58*, 45–57. [CrossRef]
112. Ye, M.; Fang, X.; Du, B.; Yuen, P.C.; Tao, D. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Comput. Surv.* **2023**, *56*, 1–44. [CrossRef]
113. Shi, Y.; Zhang, Y.; Xiao, Y.; Niu, L. Optimization strategies for client drift in federated learning: A review. *Procedia Comput. Sci.* **2022**, *214*, 1168–1173. [CrossRef]
114. Mu, X.; Shen, Y.; Cheng, K.; Geng, X.; Fu, J.; Zhang, T.; Zhang, Z. FedProc: Prototypical contrastive federated learning on non-IID data. *Future Gener. Comput. Syst.* **2023**, *143*, 93–104. [CrossRef]
115. Liu, X.; Deng, Y.; Nallanathan, A.; Bennis, M. Federated learning and meta learning: Approaches, applications, and directions. *IEEE Commun. Surv. Tutorials* **2023**, *26*, 571–618. [CrossRef]