Blockchain-empowered Federated Learning: Challenges, Solutions, and Future Directions

JUNCEN ZHU, JIANNONG CAO, DIVYA SAXENA, SHAN JIANG, and HOUDA FERRADI, Department of Computing, The Hong Kong Polytechnic University, China

Federated learning is a privacy-preserving machine learning technique that trains models across multiple devices holding local data samples without exchanging them. There are many challenging issues in federated learning, such as coordinating participants' activities, arbitrating their benefits, and aggregating models. Most existing solutions employ a centralized approach, in which a trustworthy central authority is needed for coordination. Such an approach incurs many disadvantages, including vulnerability to attacks, lack of credibility, and difficulty in calculating rewards. Recently, blockchain was identified as a potential solution for addressing the abovementioned issues. Extensive research has been conducted, and many approaches, methods, and techniques have been proposed. There is a need for a systematic survey to examine how blockchain can empower federated learning. Although there are many surveys on federated learning, few of them cover blockchain as an enabling technology. This work comprehensively surveys challenges, solutions, and future directions for blockchain-empowered federated learning (BlockFed). First, we identify the critical issues in federated learning and explain why blockchain provides a potential approach to addressing these issues. Second, we categorize existing system models into three classes: decoupled, coupled, and overlapped, according to how the federated learning and blockchain functions are integrated. Then we compare the advantages and disadvantages of these three system models, regard the disadvantages as challenging issues in BlockFed, and investigate corresponding solutions. Finally, we identify and discuss the future directions, including open problems in BlockFed.

CCS Concepts: • **General and reference** → **Surveys and overviews;** • **Computing methodologies** → **Machine learning;** • **Computer systems organization** → *Distributed architectures*;

Additional Key Words and Phrases: Blockchain, federated learning, blockchain-based federated learning, client selection, incentive mechanisms

ACM Reference format:
Juncen Zhu, Jiannong Cao, Divya Saxena, Shan Jiang, and Houda Ferradi. 2023. Blockchain-empowered Federated Learning: Challenges, Solutions, and Future Directions. *ACM Comput. Surv.* 55, 11, Article 240 (February 2023), 31 pages.
https://doi.org/10.1145/3570953

240

This work was supported by the Research Institute for Artificial Intelligence of Things, The Hong Kong Polytechnic University, HK RGC TRS No. T41-603/20-R, HK RGC CRF No. C2004-21GF, HK RGC RIF No. R5060-19 and R5034-18, HK RGC GRF No. 15204921 and 15220020, The Hong Kong Jockey Club Charities Trust No. 2021-0369, and Germany/HK JRS No. G-PolyU506/21.

Authors' address: J. Zhu, J. Cao, D. Saxena, S. Jiang, and H. Ferradi, Department of Computing, The Hong Kong Polytechnic University, 11 Yuk Choi Road, Hung Hom, Hong Kong, China; emails: juncen.zhu@connect.polyu.hk, {csjcao, divsaxen}@comp.polyu.edu.hk, cs-shan.jiang@polyu.edu.hk, ferradih@gmail.com.

0360-0300/2023/02-ART240 $15.00
https://doi.org/10.1145/3570953

1 INTRODUCTION

Recent advances in computational power have accelerated the adoption of machine learning and artificial intelligence in various application areas, including computer vision [108], natural language processing [73], autonomous driving [25], and recommender systems [86]. Additionally, researchers have been working on advanced machine learning algorithms such as deep learning [46] and reinforcement learning [93]. The performance of these machine learning algorithms is highly dependent on the availability of large volumes of high-quality data for training high-accuracy models [103]. For instance, Facebook's target detection system uses up to 350 million images. Rich data is often vulnerable to privacy, high in volume, or both. Because of the privacy-sensitive information in data, the data owners are unwilling to share it, making it difficult to obtain a large amount of data. The data owners form islands isolated and disconnected from each other [33]. The data island problem severely hinders the advancement of machine learning.

To address the data island problem, McMahan et al. [67] proposed federated learning, which uses locally computed model updates to train a shared global model. Federated learning allows model training without data exchange among users, which significantly protects data privacy. In addition, federated learning connects the data islands and builds a healthy and sustainable data ecosystem among stakeholders with conflicts of interest. Federated learning is used in a wide range of applications, especially those involving confidential data or with stringent requirements on data privacy. The first and most impactful federated learning system is the machine learning model for word prediction (also known as Google's Gboard) proposed by Bonawitz et al. [9]. The model employs a federated averaging algorithm to continuously refine the model using without knowing the user's mobile phone data.

Federated learning is a trendy technology that connects the data islands to form a data ecosystem, fully discovering and dramatically amplifying big data's value. However, a series of challenging issues still need to be addressed in federated learning, such as lack of incentive mechanism [128], model security [114], and system heterogeneity [71]. While the research community has been developing advanced solutions to the challenges above, most existing ones consider a centralized approach in which a trustworthy central authority monitors model training. Such a centralized approach imposes serious disadvantages, including a single point of failure, vulnerability to attacks, lack of credibility, and difficulty in calculating rewards. For example, the contributions of the central server and other clients are difficult to estimate during the learning process, leading to challenges in reward distribution.

Blockchain technology [34], derived from the decentralized cryptocurrency system, has shown a remarkable impact on industry and academia. Blockchain also has the potential to address the issues caused by centralization in federated learning. By combining blockchain technology with smart contracts [94], users can perform authentic and traceable transactions without a central third party. As a result, we can build a decentralized and stable platform based on blockchain to empower federated learning systems. Blockchain-empowered federated learning (BlockFed) ensures data privacy, model security, computation auditability, etc.

BlockFed is broadly used in diverse fields, including industrial internet, intelligent transportation, smart healthcare, and wireless network infrastructure. BlockFed has shown tremendous success, and several impactful solutions have been proposed, such as BlockFL [43] and Deepchain [110]. In the literature, there are many surveys about blockchain [22, 51, 52, 132] and federated learning [7, 54, 102, 116, 123], respectively. However, these surveys mainly focus on the practical applications, challenging issues, and technical solutions of blockchain or federated learning. Despite their comprehensiveness in either area, they rarely explain the potential of using blockchain for federated learning. In other words, none of them systematically studies BlockFed providing an overview of the current research trends and future directions.

Table 1. Comprehensiveness Comparison of Surveys of Blockchain-empowered Federated Learning

Surveys	Comprehensive in topic coverage?					
	Applications	System models	Incentive mechanisms	Client selection	Consensus	Security & privacy
[70]	✓	✗	✓	✗	✗	✓
[2]	✗	✗	✗	✗	✗	✓
[47]	✓	✗	✓	✗	✗	✓
[83]	✗	✗	✓	✗	✓	✓
This work	✓	✓	✓	✓	✓	✓

More recently, several surveys on BlockFed are surging out [2, 47, 70, 83]. Table 1 shows the comparison of this work and the existing related surveys in terms of comprehensiveness. In terms of the applications of BlockFed, some surveys only consider a particular area, e.g., mobile edge computing [70], internet of things [2], and distributed machine learning [47], and fail to consider the common challenges among different applications. Moreover, some surveys fail to summarize the existing application areas [2, 83]. Furthermore, none of the existing surveys considers the different system models in various BlockFed studies. In addition, they fail to consider at least one of the significant challenges together with the potential solutions, e.g., incentive mechanism, client selection, consensus, and security and privacy.

In this work, we provide a comprehensive survey on the challenges, solutions, and future directions of BlockFed. In particular, we summarize the existing BlockFed applications and classify their employed models into three categories based on the client devices' involvement in blockchain and federated learning functions. We have found a large number of BlockFed-related studies, which reflects the importance of this survey. Based on the analysis of the system models, we identify the challenges and potential solutions of BlockFed, including incentive mechanism, client selection, consensus, and security and privacy. Finally, we discuss the research directions and open problems to motivate future research work.

The main contributions of this work are as follows:

- We demonstrate the system architectures and identify the challenges of federated learning. Then, we explain in detail why emerging blockchain technology is a highly potential approach to tackling the challenges.
- We categorize the BlockFed system models into three classes: decoupled, coupled, and overlapped, according to how the federated learning and blockchain functions are fulfilled in individual nodes. The categorization is based on a systematic literature review of the research work and applications of BlockFed.
- We compare the advantages and disadvantages of the three system models of BlockFed. We consider the disadvantages of the system models as the challenging issues of BlockFed and thoroughly investigate the state-of-the-art solutions.
- We identify and discuss the promising future directions and open problems of BlockFed.

Figure 1 depicts the organization of this survey. Section 2 succinctly introduces the concepts of federated learning and blockchain and points out the motivations of BlockFed. Section 3 summarizes the advanced and emerging applications of BlockFed. Section 4 categorizes three general system models for BlockFed based on the analysis of the BlockFed research and applications. Section 5 identifies the challenging issues of BlockFed and the potential solutions in the literature. Section 6 discusses the future research directions of BlockFed. Finally, Section 7 concludes this survey.

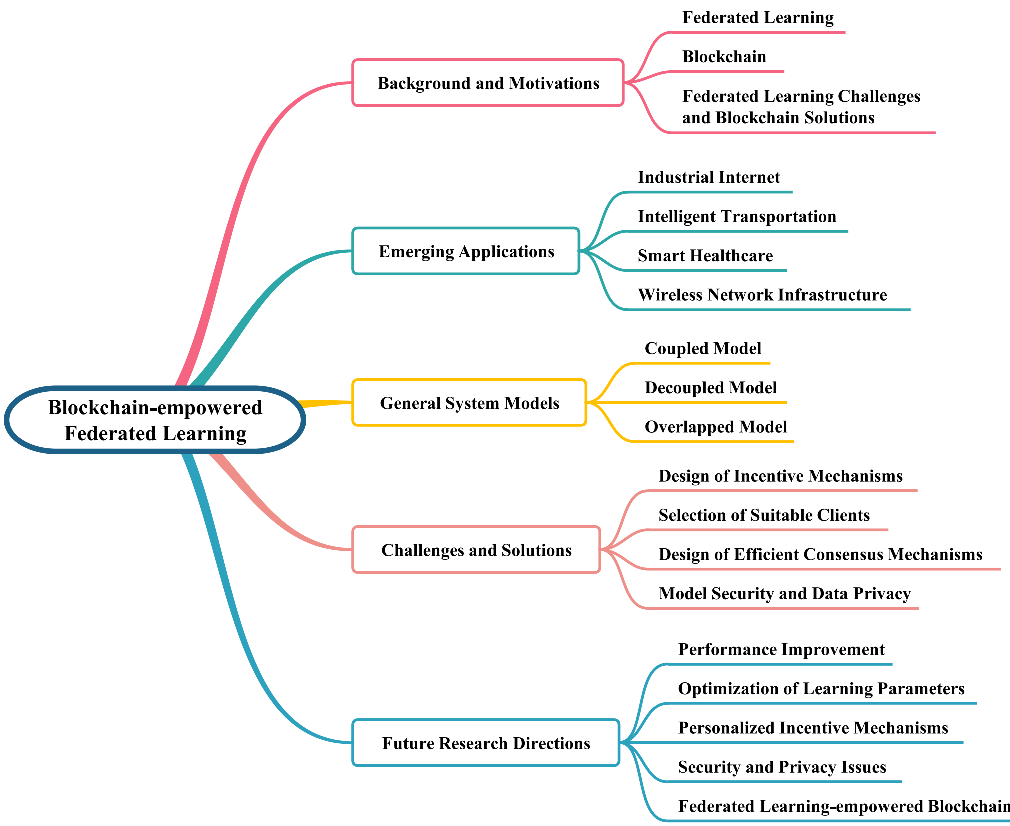

Fig. 1. Structure of this survey.

2 BACKGROUND AND MOTIVATIONS

In this section, we first introduce the definition of federated learning and blockchain. Then we summarize six challenging issues of federated learning and the inadequacies of the existing solutions. Finally, we discuss how blockchain can address each of the challenges.

2.1 Federated Learning

Federated learning was first introduced by McMahan et al. [67] in 2017: *"we call the method federated learning because the learning task is performed loosely by the participating devices (we call them the clients) under the coordination of the central server."* Instead of training machine learning models on a centralized server, federated learning makes the client devices train models locally. The critical innovation of federated learning is gathering trained local models rather than the original data. To this end, federated learning well respects data privacy. Generally speaking, federated learning runs round by roundas follows:

- *Client selection.* A new round starts. The coordinator selects some clients from sample devices based on certain criteria, e.g., historical activities, model quality, network bandwidth, and computation capabilities. The criteria are designed in advance to get the federated learning system rid of malicious client devices.
- *Client calculation.* Each selected client receives the shared global machine-learning model-from the coordinator and executes a model training program to update the local machine-learning model taking local data as input.

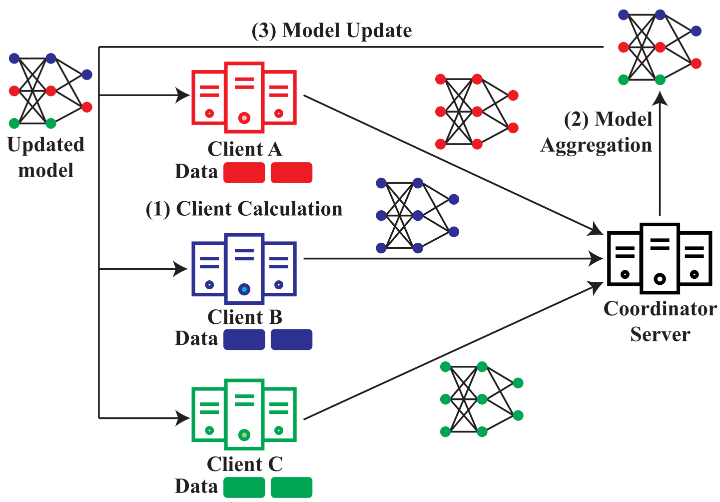

Fig. 2. Architecture of client-server federated learning. A centralized coordinator server aggregates the trained models from clients and computes the global model update.

- *Model aggregation.* The coordinator aggregates the models or updates from the client devices. To improve efficiency, the coordinator may stop the aggregation once a sufficient number of client devices have submitted the models or updates. From the client's perspective, they may slightly adjust the models before sending them to the coordinator for data privacy concerns.
- *Model update.* The coordinator updates the shared global model based on the models or updates from the client devices participating in the current round. The model update algorithm can be a simple average of the received models or updates. The design of the model update algorithm is critical for the convergence speed and accuracy of the final output model.
- *Convergence checking.* The coordinator calculates the model difference between two consecutive rounds. If the difference is smaller than a predefined threshold, the procedure ends. Otherwise, it goes back to the step of client selection.

Figure 2 illustrates the system architecture of client-server federated learning. During the procedures, a central server coordinates the model training process. First, each client trains a local machine-learning model using local data and sends it to the central server. Then, the central server aggregates the received local models into an updated global model and sends it back to the clients. In client-server federated learning, a reliable and robust central server is needed; however, it is not always available [101]. To this end, the federated learning system may not include a central coordinator but is designed as a peer-to-peer (P2P) network.

Figure 3 elaborates on the system architecture of P2P federated learning. This model further improves data security because the clients communicate directly with each other without assistance from a third authority. Model aggregation and update are performed via direct communication among client devices. However, P2P federated learning is not widely adopted in real-world applications because of the stringent requirements of intensive computation and networking resources for message encryption, decryption, and transmission. Moreover, P2P federated learning almost shares the same challenging issues with client-server federated learning, which will be discussed in the later sections. As a result, this work only considers client-server federated learning.

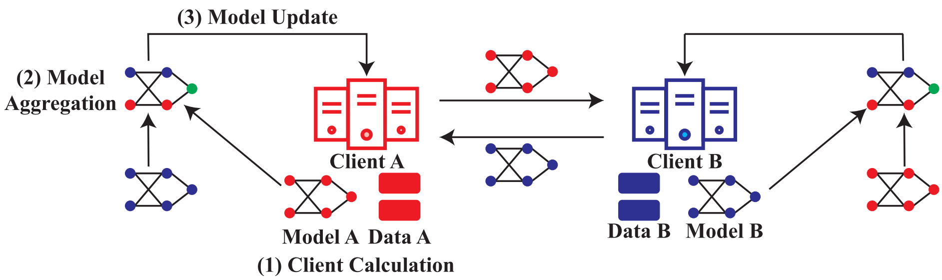

Fig. 3. Architecture of peer-to-peer federated learning. All the nodes are identical. The global model update is computed in a decentralized manner.

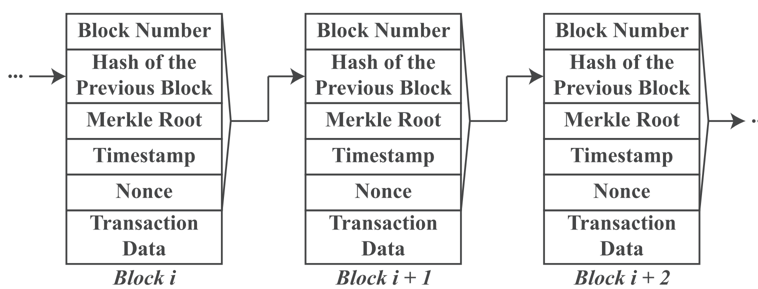

Fig. 4. Typical structure of a blockchain.

2.2 Blockchain

A blockchain is a distributed ledger consisting of a sequence of data blocks in which each data block contains a set of verifiable transactions [35]. Figure 4 depicts the blockchain structure. Each data block in a blockchain except the first one contains the hash value of the previous block's header, making the blocks linked one by one. Blockchains employ unique consensus mechanisms, e.g., proof of work (PoW) and proof of stake (PoS), run by a P2P network of nodes to make it difficult to generate but easy to verify each data block. For example, in PoW, the blockchain nodes compete to find a *nonce* whose hash value is smaller than a target *difficulty*. The nonce-generation process is complex due to the preimage resistance property of hash functions, while the nonce-verification is easy because of the polynomial computation complexity. The data in a blockchain is nearly impossible to be modified because of the chain structure, difficulty in block generation, and decentralized consensus in P2P networks. Generally, a blockchain can be regarded as a technical solution for maintaining a reliable database in a decentralized and trustless manner. Specifically, blockchain has the following four characteristics:

- *Decentralization.* A blockchain does not require a central authority to manage keys, confirm transactions, generate blocks, etc.
- *Immutability.* No user can independently decide to modify the transactions or blocks because of the cryptographic basis of blockchains. Anyone who wants to change the information in the blockchain must attack 51% of the nodes in the network.
- *Traceability.* Blockchain records the input and output of each transaction so that the data changes can be easily tracked.
- *Openness.* Everyone can query the data in blockchains through open interfaces.

Due to the distinctive characteristics above, blockchain has been widely used in finance [21], supply chain [111, 112], etc., to establish trust among stakeholders with different interests.

Table 2. Challenging Issues of Federated Learning and Solutions with and without Blockchain

Federated learning challenges	Techniques	Solution without blockchain	Solution with blockchain
Lack of incentives	Game theory	[48, 105, 115, 119, 121]	[62, 87, 95, 110, 125, 130]
	Auction theory	[20, 56, 91, 122]	
Statistical heterogeneity	Meta-learning	[41, 113]	[40, 90, 127]
	Multi-task learning	[16]	
	Proximal term	[50]	
System heterogeneity	Model-based federated learning	[67]	[99, 127]
	Optimization	[97]	
	Data-driven resource allocation	[120]	
Model security	Differential privacy	[1]	[43, 65, 69, 78, 110, 133]
	Homomorphic encryption	[1, 10, 124]	
	Secure multi-party computation	[10, 124]	
Data privacy	Differential privacy	[68]	[6, 38, 80]
	Secure multi-party computation	[10]	
High communication overhead	Parameter optimization	[9]	[60, 66]

Similarly, blockchain is expected to benefit federated learning by running the coordinator server using blockchain technology [83]. In this way, the blockchain-based federated learning system can escape from the single point of failure and provides better data privacy and model security. Furthermore, the native tokens of blockchain can incentivize the devices to join the federated learning process.

2.3 Federated Learning Challenges and Blockchain Solutions

In this subsection, we identify the challenges of federated learning, existing solutions and inadequacies, and how to overcome the limitations using blockchain technology. In particular, we categorize federated learning's challenging issues into six classes and summarize the approaches to addressing them, especially the solutions that use and do not use blockchain. The taxonomy is shown in Table 2. Note that extra challenges are brought despite the benefits of blockchain, e.g., the performance issue and complexity in design. The challenges and solutions are discussed in Section 5 instead of this section to organize this paper better. In the following, we explain the challenging issues of federated learning one by one.

2.3.1 Lack of Incentive Mechanisms. When a client device participates in a federated learning task, it will consume a certain amount of resources, including computation, network bandwidth, and battery power. In addition, federated learning frameworks also face various security risks. More specifically, curious parameter servers can learn the private information of clients' training data through generative adversarial networks. Furthermore, the trained global models can be leaked to other devices that never join the training process. Because of these security concerns, clients are more reluctant to be engaged in federated learning tasks unless they can get a sufficient return. Therefore, appropriate incentives must be developed to motivate the clients to contribute data and join the federated learning process.

The inventive mechanism is expected to reward the participants if they contribute more using their local data. Therefore, it is essential to evaluate the contributions of different data providers to calculate an appropriate and reasonable distribution of profits. Extensive literature focuses on designing incentives for federated learning based on client contributions, which can be summarized into data quality and quantity.

Designing incentives has been a hot research topic in crowdsourcing systems. Xu et al. [115] proposed the fundamental incentive mechanism of time-related mobile crowd perception task scenarios. A universal system model of mobile crowdsensing was designed based on the reverse auction framework, which derives the incentives mechanisms for the cases of single and multiple time windows. Li et al. [48] outlined the communication market's incentive mechanism between open and sealed markets of devices. Stackelberg game-based and auction-based incentive mechanisms were designed for open and sealed markets, respectively, considering fairness and trustworthiness. Wang et al. [105] studied the incentive mechanism of temperature setting in the shared space of smart buildings. The authors proposed collecting the occupants' preferences for keeping, increasing, and decreasing the room temperature and deciding the temperature by maximizing the overall thermal comfort of all occupants. Zhan et al. [122] proposed a large-scale incentive mechanism based on bargaining methods. An economic model was designed to reflect the interaction among the crowdsensing task initiator and smart devices. Then, a dual decomposition method was proposed to reduce the computation workload while respecting the smart devices' privacy.

More recently, game and auction theories have been widely adopted to design incentive mechanisms. For instance, Liu et al. [56] proposed a novel auction-based incentive mechanism to encourage light vehicle nodes to participate in data caching in the vehicular network. Song et al. [91] proposed an efficient and effective metric based on Shapley value, called contribution index, to evaluate the contributions of different clients to the federated learning process. Gradient-based methods were designed to avoid extra model training and reduce the time to calculate the contribution indexes. Zeng et al. [119] consider multi-dimensional and dynamic edge resources in federated learning and propose a novel multi-dimensional incentive framework for federated learning. They use game theory to derive an optimal policy for each client and use the expected utility to guide the parameter server to select the optimal client to train a machine-learning model. Ding et al. [20] proposed a multi-dimensional contract-theoretic approach to design the optimal incentive mechanism for parameter servers in the presence of clients' multi-dimensional private information, including training costs and communication delays. Zhan et al. [121] proposed a game-based incentive mechanism for federated learning platforms for big data analysis on mobile clients. The platform first publishes a task and issues corresponding rewards. In order to maximize its utility, each mobile client decides its level of participation, i.e., the amount of training data, by considering the rewards and energy costs it gets.

However, the approaches above are still inadequate from the following perspectives. First, nearly all the current solutions necessitate a trustworthy and central authority to monitor client behaviors and arbitrate their rewards. However, a central authority is vulnerable to attacks and suffers from a single point of failure. Second, the central authority is only sometimes credible in independent public auditing and decision-making. Finally, the role of the central authority differs from that of the clients, making it difficult to estimate the rewards compared with the clients.

Blockchain solutions. The fundamental reason for the issues above is the centralization of the federated learning system, which can be well addressed with the help of blockchain. The new problem is that high-performance devices cannot be fully motivated to participate in the federated learning system in production and real-world applications because they cannot get substantial rewards. This situation can be significantly solved through the incentive mechanism provided by the blockchain system. As an infrastructure, the blockchain provides rewards to users. The high availability of blockchain can also strengthen the confidence of the devices to participate in the training process. Since forcing terminal devices to go online on time is unnecessary, blockchain provides sufficient flexibility for terminal devices so that high-performance devices can still preferentially select after returning from other tasks. Blockchain can also implement a reputation-driven incentive mechanism. Reputation is an important indicator of customer selection during federated learning. Clients with higher reputations are more likely to bring high-quality and reliable training to federated learning tasks. For example, Zhao et al. [130] proposed a blockchain-based reputation system for federated learning of home appliance manufacturers to train machine learning models based on customer data. An incentive mechanism was designed to reward participation in the federated learning process and punish possible poisoning attacks. Zhang et al. [125] proposed a horizontal federated learning incentive mechanism called RRAFL, based on reputation and reverse auction theory, to incentivize parties to actively participate and allow the requester to choose reliable, high-quality data participants.

2.3.2 Statistical Heterogeneity. Statistical heterogeneity refers to the variant distributions of the data from client devices. In Google's GBoard, the clients use different languages that contribute data with highly deviated distributions in the word prediction task [9]. Statistical heterogeneity dramatically increases the complexity of problem modeling, theoretical analysis, and empirical evaluation of solutions [15]. The challenges of statistical heterogeneity are twofold as follows. On the one hand, it is challenging to model the heterogeneous data when training federated models from unevenly distributed data among clients [5]. On the other hand, it is not easy to analyze the convergence of related training processes [49]. In the following, we introduce the recent work addressing the two challenges.

Machine learning, especially meta-learning and multi-task learning, has shown great success in modeling heterogeneous data; such ideas have recently been extended to federated learning successfully [16, 41, 113]. For example, MOCHA [90] is an optimized framework specially designed for federated learning. It can be personalized by learning an independent but related model for each device while taking advantage of shared representations.

Statistical heterogeneity also presents new challenges when analyzing convergence behavior in a federated environment. For instance, when the data between the devices in the network is erratic, methods, such as federated averaging and federated stochastic gradient descent, can hardly converge [67]. To solve this problem, Li et al. [50] proposed the FedProx algorithm based on the interaction between system heterogeneity and statistical heterogeneity and used a different metric to capture the statistical heterogeneity in the network to provide convergence guarantees for convex and non-convex functions. Some heuristic methods also solve statistical heterogeneity by sharing local device data or some server-side proxy data [30, 31].

Despite these recent advances, there are still significant challenges in developing robust, scalable, and automated data heterogeneity modeling methods in federated learning. For example, when modeling the data distribution, it is also essential to consider the performance issues. In particular, the naive solution usingthe total loss function may implicitly benefit from specific equipment or data unfavorable to certain equipment because the learned model canbe biased toward equipment with numerous data. In addition to the issue of fairness, we note that the accountability and interpretability aspects of federated learning are also worth exploring. Some solutions require sending local data to the server, which violates the key privacy assumptions of federated learning. It is also possible to send shared proxy data to all devices; however, such a process requires remarkable communication overhead [57, 75, 128].

Blockchain solutions. The use of distributed optimization algorithms solves the limitations of existing solutions. For example, Smith et al. [90] modified the distributed dual coordinate ascent framework with high communication efficiency and proposed a multi-task federated learning program. Moreover, the blockchain can be deployed with various distributed optimization algorithms.

For instance, the blockchain-based distance-weighted joint average algorithm proposed by Zhang et al. [127] has been verified to have good feasibility and accuracy in addressing the data heterogeneity issue of federated learning.

2.3.3 System Heterogeneity. System heterogeneity refers to the substantial differences in computation speed, networking bandwidth, power supply, storage capacity, etc., of the devices in the federated learning network. Also, the network scale of each device and system-related constraints usually support only a small percentage of devices being immediately active. Due to connectivity and energy limitations, it is common for active devices to drop the training task during a given iteration. These system-level functions greatly exacerbate challenges, such as reducing customer churn and fault tolerance. Therefore, a high-performance federated learning method must tolerate heterogeneous hardware and be robust enough to discard devices in the communication network.

In response to the issues above, Mcmahan et al. [67] proposed a federated learning practice model based on model averaging and conducted extensive experience evaluation considering five different model architectures and four datasets. Tran et al. [97] formulated an optimization problem that captures the trade-off between communication and computational cost. The authors exploited the problem structure and transformed the formulated non-convex problem into three convex subproblems that can be easily solved. Zhan et al. [120] proposed an experience-driven computing resource allocation scheme to improve federated learning's energy efficiency by reducing the CPU cycle frequency of faster mobile devices. However, these solutions need to transmit data to the central server, which may cause the central server to be overloaded. Moreover, the security and privacy of the central server have no guarantee.

Blockchain solutions. Enabling blockchain also solves the problem of system heterogeneity. For example, Zhang et al. [127] proposed a blockchain-based federated learning system for fault detection in the industrial internet of things (IIoT), which can achieve client data's verifiable integrity. Moreover, the blockchain's consensus mechanism ensures the training data's correctness.

2.3.4 Model Security. Although clients will provide their private data for training models, we cannot guarantee that the data is accurate and useful. Malicious clients may deliberately provide incorrect data to destroy the final training models. We call such malicious clients active attackers. Also, the model may be leaked by untrustworthy servers [98]. Servers deployed by service providers are considered passive attackers. The attacker's purpose is to destroy the security requirements of the learning model, i.e., confidentiality, integrity, and availability [26, 114].

First, the attacker can steal sensitive information in the training data and break confidentiality by disclosing the model information and its prediction results. Second, the threats to integrity and availability mainly concentrate on the federated learning model's output, which will seriously affect the model's regular use. Integrity threat means the attacker induces the model's behavior to output the specified classification label during the prediction process. Third, availability threats are mainly used to prevent users from obtaining the correct output of the model or interfere with users' access to certain functions.

The existing solutions to addressing the model security issues usually employ differential privacy (DP), homomorphic encryption (HE), and secure multi-party computation (SMC). The basic idea of DP is to add noise to sensitive personal information to protect data [23]. In federated learning, to avoid information leakage through reversing data retrieval, DP is introduced to add noise to the parameters uploaded by clients [109]. HE can be used to encrypt the parameters of the local models. After the server receives the model, it uses the additive homomorphic attributes to calculate the sum of the statistical values, which realizes the model aggregation and update [124]. Then, the clients can use their private keys to decrypt the parameters in the updated global model. In terms of SMC, it is a technique for multiple parties to jointly compute a function over their inputs without knowing the inputs [11]. SMC is an important technique to preserve the data and even model privacy during model aggregation in federated learning [124].

Generally, model security can be guaranteed by integrating security methods or changing learning strategies. For example, combining the DP, HE, and SMC methods with federated learning ensures the security of the local and global models. However, the security mechanism still produces many adverse effects in federated learning, such as the cost of DP, the computational complexity of the encryption system, and the communication cost of multi-party aggregation.

Blockchain solutions. The current solutions for model security cannot eliminate the possibility of the central server stealing data or tampering to damage the model. The blockchain does not require a central server for model aggregation to eliminate security risks as a distributed system. Moreover, the identity verification, traceability, durability, anonymity, and high scalability of the blockchain also ensure the security of the model.

2.3.5 Data Privacy. Although federated learning allows sharing models instead of raw data, communicating model updates during training still displays sensitive information to a third party or central server. Moreover, malicious clients also can infer other sensitive information from shared parameters. Therefore, privacy is still a significant issue in federated learning. Clients' privacy is usually vulnerable to two types of attacks: model extraction and reverse model attacks. Through the model extraction attack, the attacker tries to steal the model's parameters and destroy the confidentiality of the model. For example, malicious clients perform predictive queries on the shared model and then extract the model. Florian et al. [96] attacked BigML and Amazon machine learning online services, extracted almost the same model, and proved that the same attack applies to multiple machine learning methods. Through the model reverse attack, the attacker tries to obtain the statistical information of the training data set from the model, thereby obtaining the user's private information. Although current methods aim to use SMC or DP to enhance the privacy of federated learning, these methods usually provide privacy at the cost of reduced model performance or system efficiency [10, 68]. Understanding and balancing these tradeoffs theoretically and empirically is a massive challenge for realizing a private federated learning system.

Blockchain solutions. To eliminate the possibility of the central server stealing user privacy and prevent any client from using the global model to reconstruct another client's private data, client-level differential privacy for federated learning [68] has been proposed. A single client's update on the aggregated global model can be hidden by adding random Gaussian noise. In the case of distributed federated learning, we also let each client add noise locally. In other words, each client adds a certain amount of Gaussian noise locally after the local gradient descent step and submits the model to the blockchain. The noise level is calculated locally so that the blockchain's aggregate noise can achieve client-level differential privacy. Finally, the global model summarized on the blockchain can be encrypted, and only the participating clients have the decryption key, thereby protecting the model from public attacks.

2.3.6 High Communication Overhead. Since training calculations are distributed among devices connected to the internet, expensive communications are the critical bottleneck in the federated network [9]. When combined with the privacy issues of sending raw data, raw data must be generated on each device. A federated network may contain many devices, such as millions of smartphones. Due to limited resources, such as bandwidth, energy, and power, communication in the network may be slower than local computing [100]. In order to adapt the model to the data generated by the devices in the federated network, it is essential to develop an effective communication method that iteratively sends small messages or model updates as part of the training process instead of sending the entire network data set. Moreover, we believe that communication efficiency can still be further improved by reducing the total number of communication rounds and reducing the size of the message transmitted in each round.

Blockchain solutions. The shortcomings of existing solutions can be optimized by choosing a suitable consensus mechanism and adjusting the blockchain's basic parameters. The BlockFL system proposed by Kim et al. optimizes the end-to-end model delay by adjusting the block generation rate of the blockchain [43]. Note that asynchronous operation and channel optimization increase the transmission rate and reduce the delay. Moreover, they also proposed a method of calculating delay by adjusting the difficulty of PoW. Lu et al. [60] developed a hybrid blockchain architecture consisting of a permissioned blockchain and a local directed acyclic graph (DAG). Moreover, they used deep reinforcement learning for node selection to propose an asynchronous federated learning scheme, which improves transmission efficiency significantly. Majeed et al. [66] employed permissioned blockchain and improved the transmission process at the channel level. More specifically, the concept of Hyperledger Fabric was employed, in which individual channels are responsible for training each global machine learning model.

3 EMERGING APPLICATIONS OF BLOCKFED

In this section, we investigate the emerging critical applications of BlockFed and categorize them into four domains, i.e., industrial internet, intelligent transportation, smart healthcare, and wireless network infrastructure. There are three common characteristics of the domains. First, the problems in the domains are complex to solve and even challenging to model, while machine learning or data-driven methods is applicable. Second, they are often involved with networked systems where nodes, e.g., internet of things (IoT) devices and vehicles, interact to achieve common goals. Finally, the concerned systems are privacy-sensitive and vulnerable to various attacks. In the following, application domains are introduced one by one with a summary and several representative works.

3.1 Industrial Internet

Industrial internet refers to the interconnected sensors, equipment, actuators, etc., that are intelligent for real-time status monitoring and self-adaptive decision-making [126]. The industrial internet is essential for the digitization and intelligentization of the modern manufacturing industry. Federated learning has been applied to the industrial internet to address many complex problems such as predictive maintenance [92] and smart metering [106]. However, the interconnected industrial devices demand a privacy-preserving platform to train federated learning models because of the sensitivity of industrial data [32]. For example, industrial data can be used to infer the cost and even productivity of particular factories. Meanwhile, there are many security concerns because of the severe vulnerabilities of low-capability devices. In recent years, the industrial sectors have proposed integrating blockchain technology with federated learning and came up BlockFed, a promising solution for the industrial internet to train machine learning models in a secure and privacy-preserving way.

Lu et al. [58] used BlockFed to share data on the industrial internet. They formulated the data-sharing problem into a machine-learning problem by incorporating privacy-preserved federated learning and incorporated federated learning in the consensus process of a permissioned blockchain. The computing work for consensus can also be used for federated training. Zhang et al. [127] studied the device failure detection problem in the IIoT. They proposed a system architecture of blockchain-based federated learning, enabling the verifiable integrity of client data. The data heterogeneity issue was addressed using a centroid distance-weighted federated averaging algorithm. Qu et al. [82] studied the poisoning attacks, poor performance, and data resources problems in IoT. They proposed a novel federated learning-based framework for big data-driven cognitive computing and designed a blockchain-based cognitive computing paradigm for Industry

4.0. Lu et al. [61] considered the unreliability of communication channels, limitation of computation resources, and difficulties in establishing trust among users in IIoT. They proposed a federated learning framework authorized by the blockchain for collaborative computing, significantly improving the system's reliability, security, and privacy.

3.2 Intelligent Transportation

Intelligent transportation refers to a safe and coordinated transportation network enabled by real-time monitoring of traffic conditions through advanced technologies, including wireless communications, integrated sensing, and machine learning [3]. In intelligent transportation systems, smart vehicles continuously exchange data with other vehicles, roadside units, and base stations. Federated learning is widely adopted in intelligent transportation, especially the internet of vehicles (IoVs). An example application is to predict the driving status of heavy vehicles and monitor the flight status of drones [55]. However, the existing intelligent transportation system incurs unique safety hazards due to the privacy concerns of vehicle information and high demands of communication efficiency [76]. The research community and industry found BlockFed to be an indispensable technology for intelligent transportation systems.

Hua et al. [28] optimized the intelligent control method in the heavy transportation track system. They proposed a BlockFed method to implement federated learning among distributed agents that own heterogeneous data. Pokhrel et al. [76] proposed an autonomous blockchain-based federated learning design for privacy-aware and efficient vehicle communication networks in which local vehicle machine learning models are exchanged and verified in a distributed manner. Lu et al. [60] studied the privacy and transmission workload issues among IoVs. To reduce transmission load and solve provider privacy issues, they proposed a new architecture based on federated learning and hybrid blockchain, composed of permissioned blockchain and local DAGs. Chai et al. [13] studied the data security and privacy issues in data sharing on IoVs. They proposed a hierarchical blockchain-based federated learning framework for data sharing to learn environmental data through machine learning methods and share the learning model. Wang et al. [107] solved the problem of unreliability of central decision-makers in the field of drones for mobile crowd perception by proposing a blockchain-based secure federated learning framework.

3.3 Smart Healthcare

Smart healthcare is an intelligent infrastructure that leverages sensors, real-time networking, and intelligent algorithms to perceive, transmit, and digest information [89]. With advanced technologies, such as loT, big data, cloud computing, and artificial intelligence, medical care becomes more efficient and convenient [77]. However, advanced digital technologies also raise serious data security and privacy concerns. More specifically, the hospitals and research community have been devoted to developing techniques to protect users' medical data from leakage and improper usage. BlockFed is one of the promising techniques that enable training machine learning models with high data security and privacy.

Kumar et al. [45] studied privacy protection in medical data sharing. They proposed a blockchain-based federated learning framework for collecting numerous data from various hospitals in a reliable way. The data normalization and capsule neural network techniques were proposed to address the data heterogeneity and model training issues. Zhao et al. [131] designed a blockchain and federated learning-based system which utilizes a reputation mechanism to help home appliance manufacturers train machine learning models based on customer data and improve the smart home system. Rahman et al. [84] studies data privacy issues and the lack of high-quality training data sets in the internet of health things. They proposed a lightweight hybrid federated learning framework and used blockchain and smart contracts to manage edge training.

3.4 Wireless Network Infrastructure

Wireless network infrastructure is the intelligent wireless network that contains a wireless router or access point and enables other computers to connect wirelessly. The employed technologies include but are not limited to IoT [72], edge computing [79], and 5G [4, 59]. High communication overhead and security concerns are the two main issues that hinder the development of this field [24], and BlockFed has an excellent performance in solving them. Furthermore, the research community and industry have considered integrating federated learning and blockchain as the intelligent and security infrastructure into future wireless networks.

Qu et al. [81] studied the inefficiency and poisoning attack in fog computing. They proposed a novel blockchain-enabled federated learning framework that allows a blockchain-based global learning model to update terminal device exchanges locally. Such a method alleviates communication problems and eliminates poisoning attacks. Shen et al. [88] proposed a novel attribute inference attack that utilized the unexpected attribute leakage in the blockchain-assisted federated learning for intelligent edge computing. A dynamic client selection strategy was proposed to accelerate the client selection process and improve the attack accuracy. Prkhrel et al. [76] proposed a new blockchain-based federated learning framework for the future sixth-generation network of drones using wireless mobile mining machines for disaster response systems. The future research directions were identified, including sophisticated mobility models, mobility-aware efficient verification, and analysis and privacy leakage risks. Cui et al. [17] proposed a system that combines IoT devices, edge nodes, remote cloud, and blockchain. They also designed a new algorithm in which the blockchain-assisted compression algorithm based on federated learning is applied to the content cache to predict the cache file.

To the best of our knowledge, industrial internet, intelligent transportation, smart healthcare, and wireless network infrastructure are the major application areas of BlockFed. They are all critical applications for a smart and connected society. In the future, more advanced applications will be developed based on BlockFed, especially intelligent applications with stringent data security and privacy requirements. Such applications include but are not limited to fatigue status detection for workers in smart construction and efficiency enhancement for supply chains in smart logistics.

4 GENERAL SYSTEM MODELS OF BLOCKFED

The broad applications in various areas articulate the great potential of BlockFed. However, the application essentially employs different BlockFed system models. We categorized the system models of application scenarios above into three: decoupled, coupled, and overlapped, as shown in Table 3. The categorization is based on the functions of the nodes in the system. We can observe that the decoupled model is the most popular system model in BlockFed applications. The reason is that the blockchain and federated learning subsystems can be considered separately, making the system implementation easier. However, the separation also increases the deployment cost and affects the system performance. To this end, some work employs coupled or overlapped models, balancing the implementation overhead, deployment cost, and system performance. The definitions of the three system models are as follows:

- *Decoupled model.* For each node, it works either in federated learning or blockchain. No nodes work in both systems. The system consists of a blockchain subsystem and a federated learning subsystem.
- *Coupled model.* All the nodes work in both federated learning and blockchain. The functions of federated learning and blockchain are mixed in each node.
- *Overlapped model.* A portion of nodes work in both federated learning and blockchain. The nodes' roles can adjust dynamically.

Table 3. Employed System Models of BlockFed Applications and Work

Application domain	Representative work	General model
Industrial Internet	[58]	Decoupled Model
	[127]	Decoupled Model
	[82]	Coupled Model
	[61]	Overlapped Model
Intelligent Transportation	[28]	Overlapped Model
	[76]	Decoupled Model
	[60]	Decoupled Model
	[13]	Overlapped Model
	[107]	Coupled Model
Smart Healthcare	[45]	Decoupled Model
	[131]	Decoupled Model
	[84]	Overlapped Model
Wireless Network Infrastructure	[81]	Coupled Model
	[88]	Decoupled Model
	[76]	Decoupled Model
	[17]	Decoupled Model

4.1 Decoupled Model

In the decoupled model, the system nodes are only responsible for model training or packaging transactions to deliver the blocks. For instance, Chai et al. [13] proposed a hierarchical system consisting of a blockchain system and a federated learning system for autonomous vehicles.

Figure 5 shows the BlockFed decoupled system model. We consider the first round of training. The decoupled models have two kinds of nodes: blockchain nodes and client devices. First, in training, client devices such as mobile phones, personal computers, laptops, etc., use their data to generate local models. Then the local model is encrypted and uploaded to the blockchain nodes. The blockchain nodes will verify each other. After the verification is successful, the local model is aggregated by the contract, and the aggregated model and training records are written into the blockchain. Client devices that provide local models will receive data rewards and aggregated global models. The terminal devices use their data to update the aggregate model generated in round $n-1$ and upload the updated model to the blockchain network if it is the n-th round of training. The node performs the same verification and aggregation process as in the first round again, and the clients can benefit. When the accuracy of the model reaches the set threshold, training stops. The general procedures of the BlockFed decoupled model are as follows:

- *Client calculation.* Each client device uses data to train the model locally.
- *Model upload.* The client devices encrypt the local models and upload them to the connected blockchain nodes.
- *Model verification.* The blockchain nodes verify the models from the client devices.
- *Model aggregation.* The smart contract aggregates the local models and writes the aggregated model in the blockchain.
- *Model update.* The smart contract sends the aggregated models to clients.
- *Reward distribution.* The smart contract sends data rewards to the client devices participating in the training process.

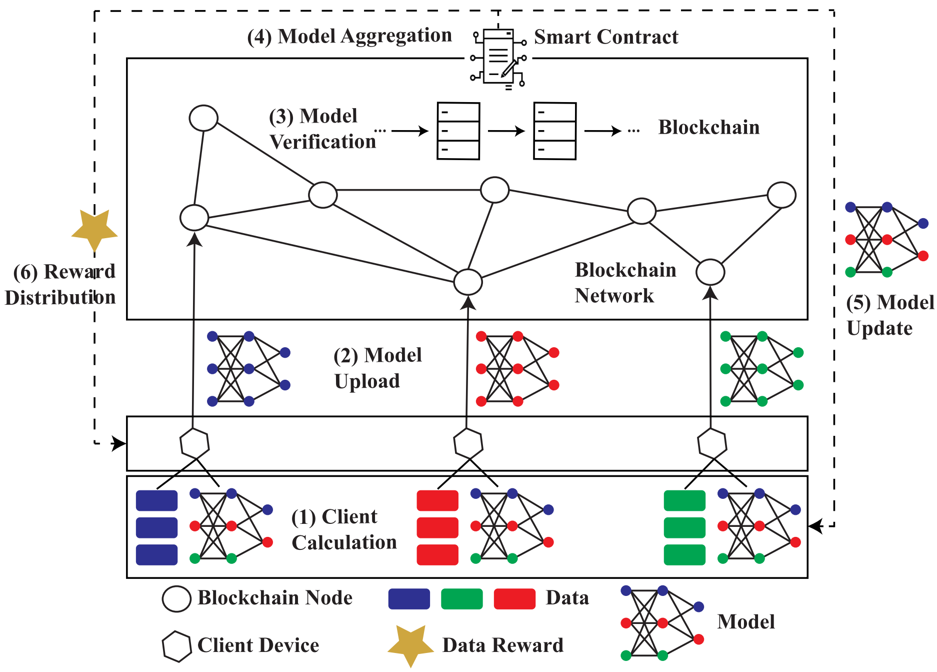

Fig. 5. The decoupled model of BlockFed. The whole system can be divided into a blockchain subsystem and a federated learning subsystem with interactions as follows: (1) the federated learning subsystem uploads the local models to the blockchain subsystem, and (2) the blockchain subsystem returns the updated global model and rewards to the federated learning subsystem.

4.2 Coupled Model

In the coupled model of BlockFed [107], all the nodes in the system are responsible for both blockchain and federated learning. On the one hand, all the nodes own the data and train the local machine-learning models using the data. The local models are broadcast to the whole network. On the other hand, the nodes maintain a blockchain and make the consensus to verify the local modelsand update the global model. Figure 6 depicts the coupled model of BlockFed. In this model, the nodes are called composite nodes and perform model training and aggregation, transaction generation and verification, and block generation and validation simultaneously. The nodes also regularly communicate to synchronize the machine-learning models, transactions, and blocks. The general procedures of the BlockFed coupled model are as follows:

- *Local calculation.* Each node uses its data to train the model locally. Different nodes can have variant data distributions and local models.
- *Model broadcast.* Each node encrypts the local model and broadcasts the encrypted model to the other nodes in the BlockFed network.
- *Model verification.* The nodes verify the models from each other. The verification includes abnormal model detection and signature validation.
- *Model aggregation.* The smart contract aggregates the local models and writes the aggregated model in the blockchain.
- *Model update.* The smart contract distributes the aggregated models to all the nodes, which will be used to initialize next-round *local calculation.*
- *Reward distribution.* The smart contract sends the data reward and consensus reward to the nodes participating in the training and consensus processes, respectively.

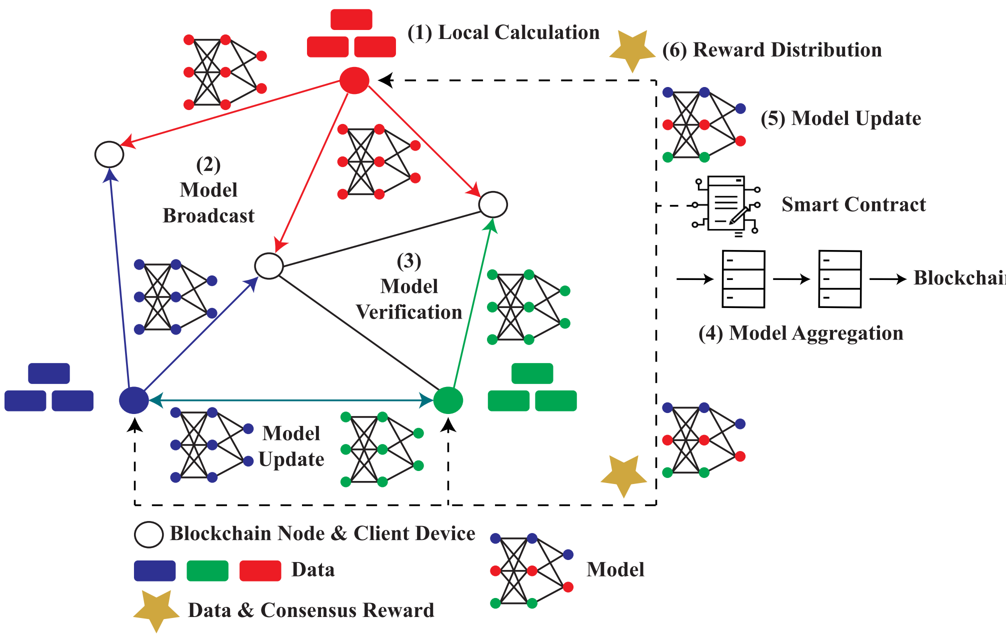

Fig. 6. The coupled model of BlockFed. The blockchain and federated learning subsystems are tightly coupled and seamlessly integrated. All the nodes in the system are identical and responsible for both blockchain and federated learning. They perform both model training and consensus tasks.

4.3 Overlapped Model

Figure 7 elaborates on the general procedure of the BlockFed overlapped model. This model has three types of nodes: blockchain node, client device, and composite node. The composite node acts as both a blockchain node and a client device. Besides the different functions, a node may change its type over time. Such an overlapped model balances the decoupled and coupled models and takes advantage of both. However, it is not easy to configure the network due to the high complexity. The general procedures of the overlapped model are as follows:

- *Local calculation.* The client devices and composite nodes use the data to train the machine-learning models locally.
- *Model exchange.* The client device uploads the encrypted local model to the blockchain node while the composite node broadcasts the local model to the connected nodes.
- *Model verification.* The blockchain and composite nodes verify the models from the client devices and composite nodes.
- *Model aggregation.* The smart contract aggregates the local models and writes the aggregated model in the blockchain.
- *Model update.* The smart contract sends the aggregated models to all the nodes.
- *Reward distribution.* There are data and consensus rewards. The smart contract sends the data reward to the client devices and composite nodes participating in the training process. Moreover, the consensus rewards will be distributed to the blockchain nodes and composite nodes participating in the consensus process.

The above three models have different advantages and disadvantages, making them suitable for applications in different scenarios with different requirements. Table 4 compares the three general system models in detail.

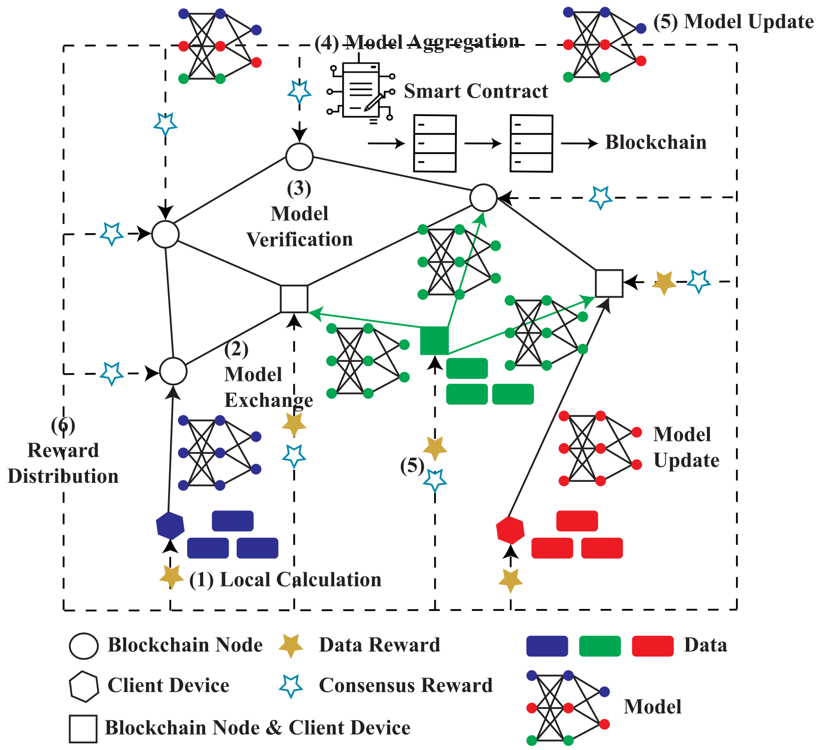

Fig. 7. The overlapped model of BlockFed. It makes the balance between the coupled and decoupled model. The responsibilities of different nodes are highly variant and dynamic.

Table 4. Advantages and Disadvantages of General System Models

Models	Advantages	Disadvantages
Decoupled model	Low resource demand for nodes Simplicity in design & configuration	High communication overhead Difficult to design incentive mechanism Difficult to select suitable clients
Coupled model	Low communication overhead Simplicity in design & configuration	High resource demands for nodes Difficult to design consensus mechanism Extra security and privacy concerns
Overlapped model	Optimization of role assignment Balance of coupled & decoupled models	Complexity in design and configuration Difficult to design incentive mechanism Difficult to design consensus mechanism Extra security and privacy concerns

Regarding the decoupled model, each node only participates in a single subsystem, federated learning, or blockchain. Compared to participation in both subsystems, the decoupled model does not demand rich resources from the nodes. Moreover, the design and configuration of individual subsystems are adequate for the entire BlockFed system, showing simplicity in design and configuration. In terms of the disadvantages, the interconnected large number of nodes results in high communication overhead. Second, it is challenging to design the incentive mechanisms because of the difficulties in evaluating the contributions of the blockchain nodes and federated learning nodes. Third, the federated learning subsystem has a loose connection with the blockchain system, making it hard to find trustworthy federated learning nodes.

The coupled model enjoys low communication overhead because the topology of the whole system is the same as the ones of individual subsystems. Because of the seamless integration of blockchain and federated learning, the coupled model is also simple in design and configuration. However, each node in the coupled model needs to perform the intensive tasks of model training and consensus, which demands rich computation, storage, and networking resources. In addition, the design consensus mechanism is challenging because it needs to consider seamless integration with the federated learning functions. Besides, each node in the coupled model takes more responsibilities than those in individual subsystems, raising extra concerns about model security and data privacy.

As for the overlapped model, it achieves the balance between coupled and decoupled models. The assignment of blockchain and federated learning functions to the nodes can be optimized based on their available resources, security level, etc. Despite the distinctive advantages, the overlapped model also suffers from similar inadequacies with the coupled and decoupled models. Designing the incentive and consensus mechanisms for the overlapped model is also challenging. The overlapped model intrinsically incurs high complexity in design and configuration because different nodes take highly variant roles in the whole system.

We can observe that all the general system models incur certain challenges, e.g., selecting suitable clients, designing consensus mechanisms, addressing extra security and privacy concerns, and designing incentive mechanisms. The research community has been developing innovative solutions to tackle the challenges. In the next section, we introduce the challenges and potential solutions of BlockFed in detail.

5 CHALLENGES AND SOLUTIONS IN BLOCKFED

Even if blockchain solves some challenges in federated learning, BlockFed still encounters four main challenges. In this section, we discuss the challenges and state-of-the-art solutions.

5.1 Design of Incentive Mechanisms

Nowadays, data has become invaluable assets of large companies, enterprises, and organizations. Intrinsically, they are not willing to share the data with others because of the great value. Moreover, the data may contain sensitive information, e.g., personal medical records and financial records. Therefore, a critical problem in federated learning is motivating different parties to participate in the learning process. In BlockFed, parties with abundant data and computation resources can get more benefits than those with limited resources because of the significant contributions to model training and consensus. As a result, BlockFed can motivate the large parties to actively share their local updates and the small ones to compete. However, most existing solutions are still designed based on token rewards, which can hardly reflect the values of data and computation resources. Furthermore, it is not easy to choose a particular type of token, e.g., bitcoins or ethers, because of the fluctuating prices and system reliability concerns. Therefore, Kim et al. [43] proposed using data rewards in BlockFed. Because data is an essential asset in federated learning, rewarding clients with data or model updates are feasible. Despite the inspiring idea, Kim et al. failed to present a detailed reward mechanism. Lu et al. [62] proposed a BlockFed method for crowdsourcing tasks, in which clients who behave correctly are rewarded while those malicious clients are punished. They proposed a promising scheme to prevent malicious clients from simply copying and reporting the performance of other clients. In addition, they have created an incentive strategy game for customers to ensure correct behaviors.

Another problem is that it requires specialized hardware, such as the Trusted Execution Environment of Intel SGX, and many computing resources for encryption to ensure that customers are correctly rewarded. Since federated learning is mainly performed on mobile devices, it is best to avoid computationally intensive cryptography or dedicated hardware. Toyoda et al. [95] proposed a system to update the competition model to ensure fair customer compliance and increase their income. Each customer selected in a particular round will pick the most popular model update from the previous round and merge it into their model. The customer's profit is determined in the next round of customer voting. A better model means they have a better chance of voting in the next round resulting in increasing their reward. The next round of model updates cannot be changed because customers have contested and voted on their models. This mechanism ensures that customers will consciously act honestly to ensure their interests without strict encryption and specialized hardware.

5.2 Selection of Suitable Clients

In BlockFed, the system needs to select participants after encouraging many participants to participate in training. The system's first consideration is participants with abundant resources, a stable communication network, and a high reputation. Nishio et al. [71] proposed a method of participant selection, which includes two steps. The first step is the resource check, immediately sending a resource query message to the screened client, asking about its local resources and the data size related to the training task. The second step is to let the coordinator use this information to estimate the time required for each client to calculate the local model update and the time required to upload the update. After that, the coordinator will decide which customer to choose based on these estimates. However, they only considered resource constraints and customer selection issues while ignoring the reliability of workers.

Fairness indicators are essential to assess the reliability of customers in order to achieve high-quality customer selection. Previous work used reputation to measure the responsibility or credibility of an entity in certain activities based on its past behavior. Inspired by this, Kang et al. [36] used reputation to measure customer reliability, thereby ensuring reliable customer choice. The reason is that high-reputation clients will bring high-quality data, i.e., high-precision and reliable data, for model training and generate reliable local model updates for federated learning tasks [19, 39]. Therefore, to better perform the federated learning task, each task publisher chooses a reputable client with high precision and reliable local data to reduce the influence of attackers [87]. Rehman et al. [99] proposed a fine-grained reputation system based on blockchain. Access to reputation information is provided to all clients of the FL system through a front-end that uses Ethereum's public blockchain and smart contract technology to calculate and determine trusted aggregations of reported reputation scores. Next, they report the hash of the reputation score to the on-chain smart contract. The smart contract then aggregates and calculates the reputation of each client and cloud server for client selection.

5.3 Design of Efficient Consensus Mechanisms

Federated learning transfers the problem of data sharing to model sharing, which brings many benefits to data sharing because only sharing the data model without sharing the original data helps protect the privacy of the data owner. Because federated learning allows sharing of models instead of data, using existing consensus for model sharing will bring high computation and communication costs and limited additional contributions to model sharing. Although PoW is still the mainstream method used in this field, it is impractical to deploy PoW directly in the BlockFed system because efficiency is one of the critical indicators of BlockFed [80]. Therefore, the design of the consensus algorithm is still a challenging issue in BlockFed.

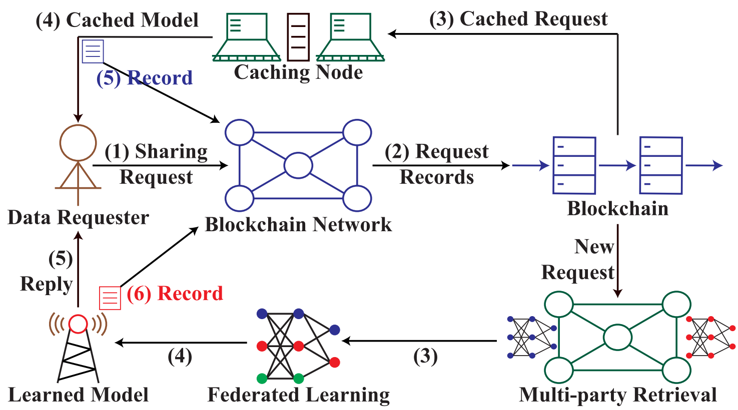

Fig. 8. System model proposed in [58].

To provide high efficiency for the BlockFed consensus mechanism, Lu et al. [58] proposed a proof of training quality (PoQ) consensus protocol for federated learning. As shown in Figure 8, PoQ is the first work that combines the data model training with the consensus process. It leverages the node's computation resources to guarantee the quality of model training. Qu et al. [80] proposed a unique consensus algorithm named proof of federated learning (PoFL). It uses learning tasks instead of hash puzzles to achieve consensus, making the consensus process energy-efficient. The key techniques of PoFL lie in the reverse game-based data trading mechanism and privacy-preserving model verification mechanism based on homomorphic encryption and secure two-party computation. In addition, some studies use PoS as the consensus mechanism [6, 78, 85] for BlockFed. Li et al. [53] proposed a committee consensus mechanism to select a small number of honest nodes to validate the model updates and run the consensus mechanism. The committee is small in scale, making the BlockFed system of high performance. Meanwhile, the committee nodes are incentivized to behave honestly because a lot of rewards will be given only in case of honest behavior. Furthermore, the whole network will eventually deny invalid blocks proposed by the committee. Cao et al. [12] proposed the first DAG-based blockchain consensus protocol for BlockFed. A three-layer structure was designed consisting of the layers of federated learning, directed acyclic graph, and application. Then, two model-controlling and updating algorithms were developed to proceed with the DAG consensus.

To summarize, there is a bunch of work studying high-efficiency blockchain mechanisms for BlockFed systems. However, most of the existing studies are still not dedicated to the federated learning scenarios, making the topic of BlockFed consensus understudied. It is highly demanded to develop high-efficiency consensus mechanisms for BlockFed.

5.4 Model Security and Data Privacy

Unlike traditional security analysis, we mainly analyze the model security and data privacy in the BlockFed systems. The reason is that federated learning aims to obtain a trained model, so the model's safety is paramount. However, since the model is open to clients participating in the training, the model's privacy does not need to be considered. Second, although clients share the trained model instead of the original data in federated learning, many attacks can still steal private data. Therefore, this section analyzes the model security and data privacy attacks in BlockFed and summarizes the existing solutions.

5.4.1 Model Security. Model security refers to the confidentiality, integrity, and availability of the trained global models. It is expected that the global models are always available only to those with access rights. Furthermore, the performance of the global model should not degrade in the presence of various attacks. In particular, two attacks, i.e., byzantine attacks and poisoning attacks, can undermine the model security. In the following, we introduce them separately.

Byzantine attack refers to arbitrary behaviors, e.g., intentional message delay and omission, taken by the participating nodes in BlockFed systems. The byzantine attack is the primary attack in distributed systems and can be regarded as the worst-case targetless attack on a given set of computing nodes. A malicious Byzantine client may exhibit completely arbitrary behavior and adjust its output to have a distribution similar to the correct model update, which makes detecting [8, 14, 15, 117] challenging. If most devices have similar calculation methods and communication resources in the BlockFed system, the system will specify a fixed learning time for each round, eliminating Byzantine attacks. However, the wide application of federated learning determines its more comprehensive application scope. Different situations have different devices with different calculation methods and communication resources, resulting in different devices having different processing times. The fixed learning time method is unsuitable, so a Byzantine flexible learning model is required. An effective way to resist Byzantine attacks is to limit the number of clients in training and enable the sharding blockchain protocol [18, 44, 64, 118].

Zhou et al. [133] are the first to raise the Byzantine attack of BlockFed. They developed a shard-based blockchain protocol to protect better the privacy of model updates and gradient aggregation to solve this problem. The overall learning [65] proposed by Majeed et al. limits the number of clients and therefore has a good performance in resisting Byzantine attacks because the number of clients is limited, and it is easy to manage and coordinate. Kim et al. [43] selected customers based on local learning accuracy and participation frequency. By selecting customers that meet the criteria, the model can reduce the harmful effects of Byzantine attacks. Preuveneers et al. [78] proposed a model that can detect abnormal behavior of devices. By removing malicious devices, the model can resist Byzantine attacks. However, in the research of Nagar et al., the blockchain of the alliance has multiple permissions and runs in a relatively private environment [69]. If the authorities do not cooperate in trust and cooperation, reaching an appropriate consensus will not be easy. Therefore their model cannot resist Byzantine attacks. Other research [6, 37, 43, 58, 66, 74, 82, 85, 129] can only resist attacks to a certain extent.

Poisoning attack refers to the behaviors that degrade the performance of trained global models. According to the attacker's target, poisoning attacks can be divided into random and targeted attacks [29]. Random attacks aim to reduce the accuracy of the federated learning model. In contrast, targeted attacks aim to promote the federated learning model to output the target label specified by the opponent. Generally, because the attacker has specific goals to achieve, targeted attacks are much more complicated than random attacks. Poisoning attacks in training may appear in data or models. The poison source can be the data during the local data collection period or the model during the local training period. Both poisoning attacks try to modify the target model's behavior in unpleasant ways at a high level. If the adversary compromises the federated learning server, they easily carry out a targeted and untargeted poisoning attack on the trained model. During the federated learning training phase, clients may have intentional or unintentional malicious behavior [38]. Deliberately malicious clients may submit incorrect model updates, causing federated learning model updates to fail. Unexpected malicious clients may upload model updates because their low-quality training data may negatively affect global model updates. When the central server aggregates these local model updates to update the global model, it leads to low accuracy or even useless global models. All these malicious actions, intentionally or unintentionally, may poison the federated learning model. In summary, the current federated learning model relies on a trust mechanism, which makes it vulnerable to poisoning attacks [110].

In the BlockFed system, all existing models can resist poisoning attacks. The address of the poisoning attack comes from the consensus mechanism of the blockchain. In the BlockFed system, all learning clients can be regarded as blockchain users or miners, and the rest are ordinary users. In each training round, one of the clients can be the winner selected through various consensus mechanisms [42]. Then, the winner collects the latest updated model and broadcasts it to the blockchain network for verification by other miners. After verification, the blockchain will save the verified model updates and use them for further processing while detecting and discarding fake data. Moreover, there is a trust management mechanism in some BlockFed systems to reward or punish trusted or malicious users. Since it can identify forged and maliciously tampered models, it can resist poisoning attacks.

5.4.2 Data Privacy. The primary type of attack that undermines data privacy is the inference attack. The inferred attacks can be divided into white-box attacks, which means full access to the federated learning model, and black-box attacks, which means only the federated learning model can be queried. They usually do not tamper with the target model but can cause it to produce the wrong output (target/non-target) or collect evidence about the model's characteristics. In federated learning, when the target model is deployed as a service, the server's model not only suffers from the same evasion attack as the conventional federated learning settings, but the model broadcast step in federated learning makes the model accessible by any malicious client. Therefore, federated learning needs extra effort to defend against white-box attacks.

In the BlockFed system, inference attacks mainly include tracking and reconstruction. Tracking refers to collecting model features or inferring user privacy data through the model, and reconstruction results from interfering with the target model to output wrong labels. Some models can completely resist inference attacks [6, 65, 80]. In these studies, the heterogeneity of learning models determines that they can resist inference attacks. The reason is that each device has a unique learning task, and no one knows which tasks are running on other devices. On the other hand, some models [38, 66, 78] cannot resist inference attacks. The model proposed by Preuveneers et al. [78] is the only one that can detect anomalies. However, they must collect sensitive data for analysis and learning to improve detection accuracy. The collected data reveals some sensitive information and even reveals more private information.

6 FUTURE RESEARCH DIRECTIONS

In this section, we introduce the unresolved issues in the BlockFed systems. Based on the investigation of state-of-the-art research, we observed that there are still performance defects in the BlockFed system. For instance, the learning parameters are difficult to choose, member selection and incentive mechanisms are not flexible enough, and security and privacy levels are insufficient. Furthermore, we indicate that federated learning-based blockchain is understudied in the literature. This section aims to clarify future research directions to inspire readers and researchers in this underdeveloped field.

6.1 Performance Improvement

Although numerous consensus protocols have been developed for federated learning in existing systems, their performance remains insufficient because they overlooked the significant impact of federated learning clients on the consensus protocols. Furthermore, there are many newly proposed federated learning algorithms, e.g., federated edge learning [54] and personalized federated learning [27], that demand new efficient blockchain consensus protocols. It is essential to adjust the traditional blockchain consensus algorithms to make them more energy-efficient, suitable, and specific for federated learning. One potential solution is substituting federated learning tasks for the random number search problem. The device can achieve PoW consensus by adjusting the learning accuracy threshold. To this end, all available computing resources can be allocated to training and learning activities, and devices that cannot compete can contribute to integration. Blockchain and federated learning are mutually beneficial when designed carefully. No additional computing power is consumed when the PoW program is implemented via the federated learning task. In this way, resources can be conserved while efficiency is increased.

Furthermore, although the federated system's primary focus should be on learning efficiency, other metrics such as privacy protection, energy consumption, and block generation rate should also be considered. In this case, the blockchain is used as the underlying architecture of federated learning. As a result, it increases computing costs and possible privacy and security risks. Another critical factor is striking the optimal balance between academic success and other metrics. In certain instances, genetic algorithms using non-dominated sorting will achieve equilibrium. By carefully designing the algorithm and applying it to the BlockFed scheme, we can change the predefined objective function parameters to generate a versatile and universal optimization model, balancing academic success and the intent of other indicators.

6.2 Optimization of Learning Parameters

In some learning algorithms, it is common to presume that the collaborator or server already has a model and then train using that model, but this is only sometimes possible. Even in BlockFed, determining who provides the initial model and its parameters takes much work. Additionally, users would have difficulty selecting machine learning model parameters and configuring the optimizer due to their inability to access or validate distributed training data.

A fundamental problem in federated learning is establishing the criteria for training termination. In current federated learning, the end condition for training may be a predefined threshold set by the coordinator. The training process is complete when the global model's parameters exceed the threshold or the threshold's error is within a defined range. This condition can be defined in advance by including a threshold in the smart contract. However, another end situation allows the coordinator to decide based on the qualified model's condition. In this situation, the training's end condition can change throughout the training. Unfortunately, no universal algorithm can accurately explain how conditions change at the end of training for various models. As a result, even if smart contracts are allowed, this issue will remain unsolved.

6.3 Personalized Incentive Mechanisms

While we examined some client selection approaches and motivation mechanisms in Section 5, these current studies are inflexible, lack versatility, and do not promote customized training based on training conditions. Therefore, we suggest implementing a personalized incentive mechanism to promote further participation in the learning process by all terminal devices.

The reward function in a conventional blockchain system is a coin, which typically represents a set number of awards for the winner or group of winners. However, in BlockFed, clients have varying computational capabilities, and their contribution is also contingent on the quality of the data. Unlike conventional blockchain systems, however, all clients have contributed to the integration process, and all contributors should be compensated in the future. As a result, we suggest using a customized reward structure in which all clients collaborate to reach a consensus, and everybody is a winning team member. In this case, using the token as an example, the token would be distributed proportionately to all. Additionally, the benefits can be dual. They will pay the winners in tokens to use the latest global model for devices not chosen as clients during the learning process. The incentive system can take on various types, including data rewards and even computational power rewards for particular scenarios.

6.4 Security and Privacy Issues

While BlockFed is resistant to most known attacks, some unique security and privacy concerns require immediate attention. On the one hand, due to the scheme's model update verification's limitations in the non-IID environment, a more accurate verification scheme should be developed for the non-IID data set to improve the poisoning attack's detection efficiency. Additionally, the credibility threshold can be dynamically configured using advanced machine-learning techniques to mitigate the harmful effects of malicious clients. On the other hand, federated learning has always been concerned about the privacy of model changes. While the use of blockchain technology will ensure the validity of data, it can introduce additional complications. Both clients can access and verify each other's model updates. If a malicious client performs a stealth attack in secret, the data that the public can access becomes a drawback for the device. As a result, an improved sharing model that protects privacy is critical.

6.5 Federated Learning-empowered Blockchain

This work discusses how blockchain can benefit and address the challenges in federated learning. That is, blockchain can significantly empower federated learning. However, blockchain and federated learning should be mutually beneficial, which means that federated learning can also empower blockchain systems. To the best of our knowledge, federated learning-empowered blockchain is understudied. From our perspectives, the potentials of federated learning in blockchain lie in two aspects.

On the one hand, federated learning can enrich blockchain functionalities. Since 2008, cryptocurrency has been regarded as the most and only representative application of blockchain. In recent years, machine learning is getting trendy, and we may raise the question of whether we can deploy machine learning models or outsource machine learning tasks on the blockchain to improve the security and privacy of machine learning. We believe federated learning is a machine learning paradigm suitable for blockchain systems. In the literature, machine learning services can be provided via blockchain-based federated learning systems reliably.

On the other hand, the trained models of federated learning can improve the performance of blockchain systems. In cryptocurrency public blockchain, the miners should decide their mining strategies. Wang et al. [104] proposed a reinforcement learning-based approach to learn the optimal mining strategy. In mobile blockchains, resource management is critical and Luong et al. [63] proposed a deep learning-based approach. These machine learning models can be extended to federated learning with enhanced privacy.

7 CONCLUSION

Blockchain-empowered federated learning (BlockFed) has recently attracted widespread attention. Despite a bunch of research on BlockFed, there is no comprehensive survey summarizing the state-of-the-art research and identifying the future directions. Therefore, we reviewed the challenges, solutions, and future directions for BlockFed. In this survey, we first categorized the fundamental challenges in federated learning and solutions based on blockchain. Afterward, we presented the emerging and important BlockFed applications and classified the system models based on the functionalities of the participating nodes. Then, we elaborate on the challenges and solutions of BlockFed. Finally, we discussed the open problems and potential research directions of BlockFed. This survey will help researchers continue optimizing the BlockFed system and accelerate the implementation of current research.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, Vienna, Austria, 308–318.

[2] Mansoor Ali, Hadis Karimipour, and Muhammad Tariq. 2021. Integration of blockchain and federated learning for internet of things: Recent advances and future challenges. *Computers & Security* 108 (2021), 102355.

[3] Moayad Aloqaily, Ismaeel Al Ridhawi, and Mohsen Guizani. 2022. Energy-aware blockchain and federated learning-supported vehicular networks. *IEEE Trans. Intell. Transp. Syst.* 23, 11 (2022), 22641–22652. DOI: https://doi.org/10.1109/TITS.2021.3103645

[4] Saeed Hamood Alsamhi, Faris A. Almalki, Fatemeh Afghah, Ammar Hawbani, Alexey V. Shvetsov, Brian Lee, and Houbing Song. 2021. Drones' edge intelligence over smart environments in B5G: Blockchain and federated learning synergy. *IEEE Transactions on Green Communications and Networking (TGCN)* 6, 1 (2021), 295–312.

[5] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. 2019. Federated learning with personalization layers. *CoRR* abs/1912.00818 (2019), 1–13. http://arxiv.org/abs/1912.00818.

[6] Sana Awan, Fengjun Li, Bo Luo, and Mei Liu. 2019. Poster: A reliable and accountable privacy-preserving federated learning framework using the blockchain. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, London, UK, 2561–2563.

[7] Paolo Bellavista, Luca Foschini, and Alessio Mora. 2021. Decentralised learning in federated deployment environments: A system-level survey. *ACM Computing Surveys (CSUR)* 54, 1 (2021), 1–38.

[8] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 119–129. https://proceedings.neurips.cc/paper/2017/hash/f4b9ec30ad9f68f89b29639786 cb62ef-Abstract.html.

[9] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, H. Brendan McMahan, et al. 2019. Towards federated learning at scale: System design. In *Second Conference on Machine Learning and Systems (MLSys)*. mlsys.org, Stanford, CA, 1–15.

[10] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, Dallas, TX, USA, 1175–1191.

[11] Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. 1996. Adaptively secure multi-party computation. In *Twenty-eighth Annual ACM Symposium on Theory of Computing (STOC)*. ACM, New York, USA, 639–648.

[12] Mingrui Cao, Long Zhang, and Bin Cao. 2021. Towards on-device federated learning: A direct acyclic graph-based blockchain approach. *CoRR* abs/2104.13092 (2021). arXiv:2104.13092 https://arxiv.org/abs/2104.13092.

[13] Haoye Chai, Supeng Leng, Yijin Chen, and Ke Zhang. 2020. A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems (T-ITS)* 22, 7 (2020), 3975–3986.

[14] Lingjiao Chen, Hongyi Wang, Zachary Charles, and Dimitris Papailiopoulos. 2018. DRACO: Byzantine-resilient distributed training via redundant gradients. In *International Conference on Machine Learning (ICML)*, Vol. 80. PMLR, 902–911.

[15] Yudong Chen, Lili Su, and Jiaming Xu. 2017. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems (POMACS)* 1, 2 (2017), 1–25.

[16] Luca Corinzia and Joachim M. Buhmann. 2019. Variational federated multi-task learning. *CoRR* abs/1906.06268 (2019), 1–12.

[17] Laizhong Cui, Xiaoxin Su, Zhongxing Ming, Ziteng Chen, Shu Yang, Yipeng Zhou, and Wei Xiao. 2022. CREAT: Blockchain-assisted compression algorithm of federated learning for content caching in edge computing. *IEEE Internet Things J.* 9, 16 (2022), 14151–14161. DOI: https://doi.org/10.1109/JIOT.2020.3014370

[18] Hung Dang, Tien Tuan Anh Dinh, Dumitrel Loghin, Ee-Chien Chang, Qian Lin, and Beng Chin Ooi. 2019. Towards scaling blockchain systems via sharding. In *International Conference on Management of Data (SIGMOD)*. ACM, New York, USA, 123–140.

[19] Sergi Delgado-Segura, Cristian Tanas, and Jordi Herrera-Joancomartí. 2016. Reputation and reward: Two sides of the same bitcoin. *Sensors* 16, 6 (2016), 776.

[20] Ningning Ding, Zhixuan Fang, and Jianwei Huang. 2020. Optimal contract design for efficient federated learning with multi-dimensional private information. *IEEE Journal on Selected Areas in Communications (JSAC)* 39, 1 (2020), 186–200.

[21] Saulo Dos Santos, Japjeet Singh, Ruppa K. Thulasiram, Shahin Kamali, Louis Sirico, and Lisa Loud. 2022. A new era of blockchain-powered decentralized finance (DeFi) - a review. In *IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, New York, USA, 1286–1292.

[22] Maya Dotan, Yvonne-Anne Pignolet, Stefan Schmid, Saar Tochner, and Aviv Zohar. 2021. Survey on blockchain networking: Context, state-of-the-art, challenges. *ACM Computing Surveys (CSUR)* 54, 5 (2021), 1–34.

[23] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation (TAMC)*. Springer, Berlin, Germany, 1–19.
[24] Chaosheng Feng, Bin Liu, Keping Yu, Sotirios K. Goudos, and Shaohua Wan. 2021. Blockchain-empowered decentralized horizontal federated learning for 5G-enabled UAVs. *IEEE Transactions on Industrial Informatics (TII)* 18, 5 (2021), 3582–3592.
[25] Andreas Geiger, Philip Lenz, and Raquel Urtasun. 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, New York, USA, 3354–3361.
[26] Robin C. Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *CoRR* abs/1712.07557 (2017), 1–7.
[27] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. 2020. Lower bounds and optimal algorithms for personalized federated learning. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., Long Beach, CA, 2304–2315.
[28] Gaofeng Hua, Li Zhu, Jinsong Wu, Chunzi Shen, Linyan Zhou, and Qingqing Lin. 2020. Blockchain-based federated learning for intelligent control in heavy haul railway. *IEEE Access* 8 (2020), 176830–176839.
[29] Ling Huang, Anthony D. Joseph, Blaine Nelson, Benjamin I. P. Rubinstein, and J. Doug Tygar. 2011. Adversarial machine learning. In *4th ACM Workshop on Security and Artificial Intelligence*. ACM, New York, USA, 43–58.
[30] Li Huang, Yifeng Yin, Zeng Fu, Shifa Zhang, Hao Deng, and Dianbo Liu. 2018. LoAdaBoost: Loss-based AdaBoost federated machine learning on medical data. *CoRR* abs/1811.12629 (2018), 1–16.
[31] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. 2018. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data. *CoRR* abs/1811.11479 (2018), 1–6.
[32] Bin Jia, Xiaosong Zhang, Jiewen Liu, Yang Zhang, Ke Huang, and Yongquan Liang. 2021. Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT. *IEEE Transactions on Industrial Informatics (TII)* 18, 6 (2021), 4049–4058.
[33] Shan Jiang, Jiannong Cao, Julie A. McCann, Yanni Yang, Yang Liu, Xiaoqing Wang, and Yuming Deng. 2019. Privacy-preserving and efficient multi-keyword search over encrypted data on blockchain. In *IEEE International Conference on Blockchain (Blockchain)*. IEEE, New York, USA, 405–410.
[34] Shan Jiang, Jiannong Cao, Hanqing Wu, and Yanni Yang. 2020. Fairness-based packing of industrial IoT data in permissioned blockchains. *IEEE Transactions on Industrial Informatics (TII)* 17, 11 (2020), 7639–7649.
[35] Shan Jiang, Jiannong Cao, Hanqing Wu, Yanni Yang, Mingyu Ma, and Jianfei He. 2018. BlocHIE: A blockchain-based platform for healthcare information exchange. In *IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, New York, USA, 49–56.
[36] Jiawen Kang, Zehui Xiong, Dusit Niyato, Shengli Xie, and Junshan Zhang. 2019. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal (IoT-J)* 6, 6 (2019), 10700–10714.
[37] Jiawen Kang, Zehui Xiong, Dusit Niyato, Dongdong Ye, Dong In Kim, and Jun Zhao. 2019. Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory. *IEEE Transactions on Vehicular Technology (TVT)* 68, 3 (2019), 2906–2920.
[38] Jiawen Kang, Zehui Xiong, Dusit Niyato, Yuze Zou, Yang Zhang, and Mohsen Guizani. 2020. Reliable federated learning for mobile networks. *IEEE Wireless Communications* 27, 2 (2020), 72–80.
[39] Jiawen Kang, Rong Yu, Xumin Huang, Maoqiang Wu, Sabita Maharjan, Shengli Xie, and Yan Zhang. 2018. Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet of Things Journal (IoT-J)* 6, 3 (2018), 4660–4670.
[40] Latif U. Khan, Shashi Raj Pandey, Nguyen H. Tran, Walid Saad, Zhu Han, Minh N. H. Nguyen, and Choong Seon Hong. 2020. Federated learning for edge networks: Resource optimization and incentive mechanism. *IEEE Communications Magazine* 58, 10 (2020), 88–93.
[41] Mikhail Khodak, Maria-Florina F. Balcan, and Ameet S. Talwalkar. 2019. Adaptive gradient-based meta-learning methods. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., New York, USA, 5917–5928.
[42] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. 2017. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual International Cryptology Conference (CRYPTO)*. Springer, Berlin, Germany, 357–388.
[43] Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. 2019. Blockchained on-device federated learning. *IEEE Communications Letters* 24, 6 (2019), 1279–1283.
[44] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ewa Syta, and Bryan Ford. 2018. OmniLedger: A secure, scale-out, decentralized ledger via sharding. In *IEEE Symposium on Security and Privacy (SP)*. IEEE, New York, USA, 583–598.

[45] Rajesh Kumar, Abdullah Aman Khan, Jay Kumar, Noorbakhsh Amiri Golilarz, Simin Zhang, Yang Ting, Chengyu Zheng, Wenyong Wang, et al. 2021. Blockchain-federated-learning and deep learning models for covid-19 detection using CT imaging. *IEEE Sensors Journal* 21, 14 (2021), 16301–16314.
[46] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.
[47] Dun Li, Dezhi Han, Tien-Hsiung Weng, Zibin Zheng, Hongzhi Li, Han Liu, Arcangelo Castiglione, and Kuan-Ching Li. 2022. Blockchain for federated learning toward secure distributed machine learning systems: A systemic survey. *Soft Computing* 26, 9 (2022), 4423–4440.
[48] Peng Li and Song Guo. 2015. Incentive mechanisms for device-to-device communications. *IEEE Network* 29, 4 (2015), 75–79.
[49] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. 2021. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning (ICML)*. PMLR, 6357–6368.
[50] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Third Conference on Machine Learning and Systems (MLSys)* 2 (2020), 429–450.
[51] Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo, and Qiaoyan Wen. 2020. A survey on the security of blockchain systems. *Future Generation Computer Systems (FGCS)* 107 (2020), 841–853.
[52] Xi Li, Zehua Wang, Victor C. M. Leung, Hong Ji, Yiming Liu, and Heli Zhang. 2021. Blockchain-empowered data-driven networks: A survey and outlook. *ACM Computing Surveys (CSUR)* 54, 3 (2021), 1–38.
[53] Yuzheng Li, Chuan Chen, Nan Liu, Huawei Huang, Zibin Zheng, and Qiang Yan. 2020. A blockchain-based decentralized federated learning framework with committee consensus. *IEEE Network* 35, 1 (2020), 234–241.
[54] Wei Yang Bryan Lim, Nguyen Cong Luong, Dinh Thai Hoang, Yutao Jiao, Ying-Chang Liang, Qiang Yang, Dusit Niyato, and Chunyan Miao. 2020. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials* 22, 3 (2020), 2031–2063.
[55] Hong Liu, Shuaipeng Zhang, Pengfei Zhang, Xinqiang Zhou, Xuebin Shao, Geguang Pu, and Yan Zhang. 2021. Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing. *IEEE Transactions on Vehicular Technology (TVT)* 70, 6 (2021), 6073–6084.
[56] Jiaqi Liu, Wei Wang, Deng Li, Shaohua Wan, and Hui Liu. 2019. Role of gifts in decision making: An endowment effect incentive mechanism for offloading in the IoV. *IEEE Internet of Things Journal (IoT-J)* 6, 4 (2019), 6933–6951.
[57] Sin Kit Lo, Yue Liu, Qinghua Lu, Chen Wang, Xiwei Xu, Hye-Young Paik, and Liming Zhu. 2022. Towards trustworthy AI: Blockchain-based architecture design for accountability and fairness of federated learning systems. *IEEE Internet of Things Journal (IoT-J)* 5 (2022), 1–8.
[58] Yunlong Lu, Xiaohong Huang, Yueyue Dai, Sabita Maharjan, and Yan Zhang. 2019. Blockchain and federated learning for privacy-preserved data sharing in industrial IoT. *IEEE Transactions on Industrial Informatics (TII)* 16, 6 (2019), 4177–4186.
[59] Yunlong Lu, Xiaohong Huang, Ke Zhang, Sabita Maharjan, and Yan Zhang. 2020. Blockchain and federated learning for 5G beyond. *IEEE Network* 35, 1 (2020), 219–225.
[60] Yunlong Lu, Xiaohong Huang, Ke Zhang, Sabita Maharjan, and Yan Zhang. 2020. Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles. *IEEE Transactions on Vehicular Technology (TVT)* 69, 4 (2020), 4298–4311.
[61] Yunlong Lu, Xiaohong Huang, Ke Zhang, Sabita Maharjan, and Yan Zhang. 2020. Low-latency federated learning and blockchain for edge association in digital twin empowered 6G networks. *IEEE Transactions on Industrial Informatics (TII)* 17, 7 (2020), 5098–5107.
[62] Yuan Lu, Qiang Tang, and Guiling Wang. 2018. On enabling machine learning tasks atop public blockchains: A crowdsourcing approach. In *IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, New York, USA, 81–88.
[63] Nguyen Cong Luong, Zehui Xiong, Ping Wang, and Dusit Niyato. 2018. Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach. In *IEEE International Conference on Communications (ICC)*. IEEE, New York, USA, 1–6.
[64] Loi Luu, Viswesh Narayanan, Chaodong Zheng, Kunal Baweja, Seth Gilbert, and Prateek Saxena. 2016. A secure sharding protocol for open blockchains. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, New York, USA, 17–30.
[65] Umer Majeed and Choong Seon Hong. 2019. EFLChain: Ensemble learning via federated learning over blockchain network: A framework. In *Proceedings of the KIISE Korea Computer Congress*. 845–847.
[66] Umer Majeed and Choong Seon Hong. 2019. FLchain: Federated learning via MEC-enabled blockchain network. In *20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, New York, USA, 1–4.
[67] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 1273–1282.

[68] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning differentially private recurrent language models. In *6th International Conference on Learning Representations (ICLR)*. OpenReview.net, 1–14.

[69] Anudit Nagar. 2019. Privacy-preserving blockchain based federated learning with differential data sharing. *CoRR* abs/1912.04859 (2019), 1–9.

[70] Dinh C. Nguyen, Ming Ding, Quoc-Viet Pham, Pubudu N. Pathirana, Long Bao Le, Aruna Seneviratne, Jun Li, Dusit Niyato, and H. Vincent Poor. 2021. Federated learning meets blockchain in edge computing: Opportunities and challenges. *IEEE Internet of Things Journal (IoT-J)* 8, 16 (2021), 12806–12825.

[71] Takayuki Nishio and Ryo Yonetani. 2019. Client selection for federated learning with heterogeneous resources in mobile edge. In *IEEE International Conference on Communications (ICC)*. IEEE, New York, USA, 1–7.

[72] Safa Otoum, Ismaeel Al Ridhawi, and Hussein Mouftah. 2021. Securing critical IoT infrastructures with blockchain-supported federated learning. *IEEE Internet of Things Journal (IoT-J)* 9, 4 (2021), 2592–2601.

[73] Daniel W. Otter, Julian R. Medina, and Jugal K. Kalita. 2020. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)* 32, 2 (2020), 604–624.

[74] Jonathan Passerat-Palmbach, Tyler Farnan, Robert Miller, Marielle S. Gross, Heather Leigh Flannery, and Bill Gleim. 2019. A blockchain-orchestrated federated learning architecture for healthcare consortia. *CoRR* abs/1910.12603 (2019), 1–6.

[75] Zhe Peng, Jianliang Xu, Xiaowen Chu, Shang Gao, Yuan Yao, Rong Gu, and Yuzhe Tang. 2021. Vfchain: Enabling verifiable and auditable federated learning via blockchain systems. *IEEE Transactions on Network Science and Engineering (TNSE)* 9, 1 (2021), 173–186.

[76] Shiva Raj Pokhrel and Jinho Choi. 2020. Federated learning with blockchain for autonomous vehicles: Analysis and design challenges. *IEEE Transactions on Communications (TCOM)* 68, 8 (2020), 4734–4746.

[77] Dawid Połap, Gautam Srivastava, and Keping Yu. 2021. Agent architecture of an intelligent medical system based on federated learning and blockchain technology. *Journal of Information Security and Applications (JISA)* 58 (2021), 102748.

[78] Davy Preuveneers, Vera Rimmer, Ilias Tsingenopoulos, Jan Spooren, Wouter Joosen, and Elisabeth Ilie-Zudor. 2018. Chained anomaly detection models for federated learning: An intrusion detection case study. *Applied Sciences* 8, 12 (2018), 2663.

[79] Guanjin Qu, Naichuan Cui, Huaming Wu, Ruidong Li, and Yuemin Ding. 2021. ChainFL: A simulation platform for joint federated learning and blockchain in edge/cloud computing environments. *IEEE Transactions on Industrial Informatics (TII)* 18, 5 (2021), 3572–3581.

[80] Xidi Qu, Shengling Wang, Qin Hu, and Xiuzhen Cheng. 2021. Proof of federated learning: A novel energy-recycling consensus algorithm. *IEEE Transactions on Parallel and Distributed Systems (TPDS)* 32, 8 (2021), 2074–2085.

[81] Youyang Qu, Longxiang Gao, Tom H. Luan, Yong Xiang, Shui Yu, Bai Li, and Gavin Zheng. 2020. Decentralized privacy using blockchain-enabled federated learning in fog computing. *IEEE Internet of Things Journal (IoT-J)* 7, 6 (2020), 5171–5183.

[82] Youyang Qu, Shiva Raj Pokhrel, Sahil Garg, Longxiang Gao, and Yong Xiang. 2020. A blockchained federated learning framework for cognitive computing in industry 4.0 networks. *IEEE Transactions on Industrial Informatics (TII)* 17, 4 (2020), 2964–2973.

[83] Youyang Qu, Md Palash Uddin, Chenquan Gan, Yong Xiang, Longxiang Gao, and John Yearwood. 2022. Blockchain-enabled federated learning: A survey. *ACM Computing Surveys (CSUR)* 55 (2022), 1–33. https://doi.org/10.1145/3524104

[84] Mohamed Abdur Rahman, M. Shamim Hossain, Mohammad Saiful Islam, Nabil A. Alrajeh, and Ghulam Muhammad. 2020. Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach. *IEEE Access* 8 (2020), 205071–205087.

[85] Paritosh Ramanan and Kiyoshi Nakayama. 2020. BAFFLE: Blockchain based aggregator free federated learning. In *IEEE International Conference on Blockchain (Blockchain)*. IEEE, New York, USA, 72–81.

[86] Paul Resnick and Hal R. Varian. 1997. Recommender systems. *Communications of the ACM (CACM)* 40, 3 (1997), 56–58.

[87] Muhammad Shayan, Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. 2020. Biscotti: A blockchain system for private and secure federated learning. *IEEE Transactions on Parallel and Distributed Systems (TPDS)* 32, 7 (2020), 1513–1525.

[88] Meng Shen, Huan Wang, Bin Zhang, Liehuang Zhu, Ke Xu, Qi Li, and Xiaojiang Du. 2020. Exploiting unintended property leakage in blockchain-assisted federated learning for intelligent edge computing. *IEEE Internet of Things Journal (IoT-J)* 8, 4 (2020), 2265–2275.

[89] Saurabh Singh, Shailendra Rathore, Osama Alfarraj, Amr Tolba, and Byungun Yoon. 2022. A framework for privacy-preservation of IoT healthcare data using federated learning and blockchain technology. *Future Generation Computer Systems (FGCS)* 129 (2022), 380–388.

[90] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S. Talwalkar. 2017. Federated multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., New York, USA, 4424–4434.
[91] Tianshu Song, Yongxin Tong, and Shuyue Wei. 2019. Profit allocation for federated learning. In *IEEE International Conference on Big Data (BigData)*. IEEE, New York, USA, 2577–2586.
[92] Gian Antonio Susto, Andrea Schirru, Simone Pampuri, Seán McLoone, and Alessandro Beghi. 2014. Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics (TII)* 11, 3 (2014), 812–820.
[93] Richard S. Sutton and Andrew G. Barto. 1998. Reinforcement learning: An introduction. *IEEE Transactions on Neural Networks (TNN)* 9, 5 (1998), 1054–1054.
[94] Palina Tolmach, Yi Li, Shang-Wei Lin, Yang Liu, and Zengxiang Li. 2021. A survey of smart contract formal specification and verification. *ACM Computing Surveys (CSUR)* 54, 7 (2021), 1–38.
[95] Kentaroh Toyoda and Allan N. Zhang. 2019. Mechanism design for an incentive-aware blockchain-enabled federated learning platform. In *IEEE International Conference on Big Data (Big Data)*. IEEE, New York, USA, 395–403.
[96] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction APIs. In *25th USENIX Security Symposium*. USENIX Association, California, USA, 601–618.
[97] Nguyen H. Tran, Wei Bao, Albert Zomaya, Minh N. H. Nguyen, and Choong Seon Hong. 2019. Federated learning over wireless networks: Optimization model design and analysis. In *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, New York, USA, 1387–1395.
[98] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, Rui Zhang, and Yi Zhou. 2019. A hybrid approach to privacy-preserving federated learning. In *12th ACM Workshop on Artificial Intelligence and Security*. ACM, New York, USA, 1–11.
[99] Muhammad Habib ur Rehman, Khaled Salah, Ernesto Damiani, and Davor Svetinovic. 2020. Towards blockchain-based reputation-aware federated learning. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, New York, USA, 183–188.
[100] C. H. Van Berkel. 2009. Multi-core for mobile phones. In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, New York, USA, 1260–1265.
[101] Paul Vanhaesebrouck, Aurélien Bellet, and Marc Tommasi. 2017. Decentralized collaborative learning of personalized models and collaboration graph. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 509–517.
[102] Omar Abdel Wahab, Azzam Mourad, Hadi Otrok, and Tarik Taleb. 2021. Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems. *IEEE Communications Surveys & Tutorials* 23, 2 (2021), 1342–1397.
[103] Senzhang Wang, Jiannong Cao, and Philip S. Yu. 2022. Deep learning for spatio-temporal data mining: A survey. *IEEE Trans. Knowl. Data Eng.* 34, 8 (2022), 3681–3700. DOI: https://doi.org/10.1109/TKDE.2020.3025580
[104] Taotao Wang, Soung Chang Liew, and Shengli Zhang. 2021. When blockchain meets AI: Optimal mining strategy achieved by machine learning. *International Journal of Intelligent Systems* 36, 5 (2021), 2183–2207.
[105] Tao Wang, Yunjian Xu, Chathura Withanage, Lan Lan, Selin Damla Ahipaşaoğlu, and Costas A. Courcoubetis. 2016. A fair and budget-balanced incentive mechanism for energy management in buildings. *IEEE Transactions on Smart Grid (TSG)* 9, 4 (2016), 3143–3153.
[106] Yi Wang, Qixin Chen, Dahua Gan, Jingwei Yang, Daniel S. Kirschen, and Chongqing Kang. 2018. Deep learning-based socio-demographic information identification from smart meter data. *IEEE Transactions on Smart Grid (TSG)* 10, 3 (2018), 2593–2602.
[107] Yuntao Wang, Zhou Su, Ning Zhang, and Abderrahim Benslimane. 2020. Learning in the air: Secure federated learning for UAV-assisted crowdsensing. *IEEE Transactions on Network Science and Engineering (TNSE)* 8, 2 (2020), 1055–1069.
[108] Zhengwei Wang, Qi She, and Tomas E. Ward. 2021. Generative adversarial networks in computer vision: A survey and taxonomy. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–38.
[109] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q. S. Quek, and H. Vincent Poor. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security (TIFS)* 15 (2020), 3454–3469.
[110] Jiasi Weng, Jian Weng, Jilian Zhang, Ming Li, Yue Zhang, and Weiqi Luo. 2019. DeepChain: Auditable and privacy-preserving deep learning with blockchain-based incentive. *IEEE Transactions on Dependable and Secure Computing (TDSC)* 18, 5 (2019), 2438–2455.
[111] Hanqing Wu, Jiannong Cao, Yanni Yang, Cheung Leong Tung, Shan Jiang, Bin Tang, Yang Liu, Xiaoqing Wang, and Yuming Deng. 2019. Data management in supply chain using blockchain: Challenges and a case study. In *28th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, New York, USA, 1–8.

[112] Hanqing Wu, Shan Jiang, and Jiannong Cao. 2022. High-efficiency blockchain-based supply chain traceability. *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*. Early Access (2022), 1–11.
[113] Peng Xiao, Samuel Cheng, Vladimir Stankovic, and Dejan Vukobratovic. 2020. Averaging is probably not the optimum way of aggregating parameters in federated learning. *Entropy* 22, 3 (2020), 314.
[114] Guowen Xu, Hongwei Li, Sen Liu, Kan Yang, and Xiaodong Lin. 2019. VerifyNet: Secure and verifiable federated learning. *IEEE Transactions on Information Forensics and Security (TIFS)* 15 (2019), 911–926.
[115] Jia Xu, Jinxin Xiang, and Dejun Yang. 2015. Incentive mechanisms for time window dependent tasks in mobile crowdsensing. *IEEE Transactions on Wireless Communications (TWC)* 14, 11 (2015), 6353–6364.
[116] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 2 (2019), 1–19.
[117] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning (ICML)*. PMLR, 5650–5659.
[118] Mahdi Zamani, Mahnush Movahedi, and Mariana Raykova. 2018. RapidChain: Scaling blockchain via full sharding. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, New York, USA, 931–948.
[119] Rongfei Zeng, Shixun Zhang, Jiaqi Wang, and Xiaowen Chu. 2020. FMore: An incentive scheme of multi-dimensional auction for federated learning in MEC. In *IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, New York, USA, 278–288.
[120] Yufeng Zhan, Peng Li, and Song Guo. 2020. Experience-driven computational resource allocation of federated learning by deep reinforcement learning. In *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, New York, USA, 234–243.
[121] Yufeng Zhan, Peng Li, Kun Wang, Song Guo, and Yuanqing Xia. 2020. Big data analytics by CrowdLearning: Architecture and mechanism design. *IEEE Network* 34, 3 (2020), 143–147.
[122] Yufeng Zhan, Yuanqing Xia, and Jinhui Zhang. 2018. Incentive mechanism in platform-centric mobile crowdsensing: A one-to-many bargaining approach. *Computer Networks* 132 (2018), 40–52.
[123] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. 2021. A survey on federated learning. *Knowledge-Based Systems (KBS)* 216 (2021), 106775.
[124] Jiale Zhang, Bing Chen, Shui Yu, and Hai Deng. 2019. PEFL: A privacy-enhanced federated learning scheme for big data analytics. In *IEEE Global Communications Conference (GLOBECOM)*. IEEE, New York, USA, 1–6.
[125] Jingwen Zhang, Yuezhou Wu, and Rong Pan. 2021. Incentive mechanism for horizontal federated learning based on reputation and reverse auction. In *The Web Conference (WWW)*. ACM, New York, USA, 947–956.
[126] Peiying Zhang, Yanrong Hong, Neeraj Kumar, Mamoun Alazab, Mohammad Dahman Alshehri, and Chunxiao Jiang. 2021. BC-EdgeFL: A defensive transmission model based on blockchain-assisted reinforced federated learning in IIoT environment. *IEEE Transactions on Industrial Informatics (TII)* 18, 5 (2021), 3551–3561.
[127] Weishan Zhang, Qinghua Lu, Qiuyu Yu, Zhaotong Li, Yue Liu, Sin Kit Lo, Shiping Chen, Xiwei Xu, and Liming Zhu. 2020. Blockchain-based federated learning for device failure detection in industrial IoT. *IEEE Internet of Things Journal (IoT-J)* 8, 7 (2020), 5926–5937.
[128] Zhebin Zhang, Dajie Dong, Yuhang Ma, Yilong Ying, Dawei Jiang, Ke Chen, Lidan Shou, and Gang Chen. 2021. Refiner: A reliable incentive-driven federated learning system powered by blockchain. *Proceedings of the VLDB Endowment (PVLDB)* 14, 12 (2021), 2659–2662.
[129] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2020. iDLG: Improved deep leakage from gradients. *CoRR* abs/2001.02610 (2020), 1–5.
[130] Yang Zhao, Jun Zhao, Linshan Jiang, Rui Tan, Dusit Niyato, Zengxiang Li, Lingjuan Lyu, and Yingbo Liu. 2020. Privacy-preserving blockchain-based federated learning for IoT devices. *IEEE Internet of Things Journal (IoT-J)* 8, 3 (2020), 1817–1829.
[131] Yang Zhao, Jun Zhao, Linshan Jiang, Rui Tan, Dusit Niyato, Zengxiang Li, Lingjuan Lyu, and Yingbo Liu. 2020. Privacy-preserving blockchain-based federated learning for IoT devices. *IEEE Internet of Things Journal (IoT-J)* 8, 3 (2020), 1817–1829.
[132] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen, and Huaimin Wang. 2018. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services (IJWGS)* 14, 4 (2018), 352–375.
[133] Sicong Zhou, Huawei Huang, Wuhui Chen, Pan Zhou, Zibin Zheng, and Song Guo. 2020. PiRATE: A blockchain-based secure framework of distributed machine learning in 5G networks. *IEEE Network* 34, 6 (2020), 84–91.

Received 1 July 2021; revised 6 July 2022; accepted 1 November 2022