HyFed: A Hybrid Blockchain Empowered Federated Learning Privacy Fair Framework

Kailin Chao[1,2], Fan Jiang[1,2], Jianmao Xiao[1,2(✉)], Yaozhang Zhong[1,2], Junyi Wu[1,2], Keyang Gu[1,2], and Zhiyong Feng[3]

[1] School of Software, Jiangxi Normal University, Nanchang 330022, China
jm_xiao@jxnu.edu.cn
[2] Jiangxi Provincial Engineering Research Center of Blockchain Data Security and Governance, Nanchang 330022, China
[3] College of Intelligence and Computing, Tianjin University, Tianjin 300350, China

Abstract. This study presents the meticulous construction of a robust experimental system framework based on a hybrid blockchain network, designed to meet the experimental needs of federated learning research. The framework leverages TensorFlow Federated (TFF) to facilitate the optimization and substitution of federated learning aggregation algorithms and the development of reputation and contribution systems. The hybrid blockchain network architecture within this framework combines the advantages of public and private blockchains, capable of processing public transactions and managing private data. An innovative data encryption and access control mechanism has been implemented, ensuring data privacy and security. Performance optimizations, including the acceleration of block production speed and database query optimization, have been carried out to enhance system efficiency. This article provides a comprehensive deployment of the framework and an analysis of its components, offering a foundation for further research. With the evolution of federated learning and blockchain technology, the proposed experimental system framework is expected to have broader application prospects and research value.

Keywords: Federated Learning · Blockchain · Smart Contract · Data Security · Privacy Protection

1 Introduction

Federated learning and blockchain technology, as the epicenter of the information technology domain, have instigated profound transformations across multiple

This work was supported by the Jiangxi Provincial Natural Science Foundation Project (20224ACB202007), Jiangxi Provincial 03 Special Project and 5G Project (20224ABC03A13), Jiangxi Provincial Natural Science Foundation Project (20224BAB212015), Jiangxi Provincial Education Commission Fund Project (GJJ210338), and the National Natural Science Foundation of China Fund project (62363015, 62067003).
© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2025
Published by Springer Nature Switzerland AG 2025. All Rights Reserved
Y. Cao and X. Shao (Eds.): DIONE 2023, LNICST 515, pp. 150–166, 2025.
https://doi.org/10.1007/978-3-031-80713-8_11

sectors, and their amalgamation has pioneered novel methodologies to address intricate challenges. Federated learning, a paradigm of distributed machine learning, has been ubiquitously employed in sectors such as healthcare, finance, and telecommunications since its inception by Google in 2016 [1]. It addresses the conundrums of data privacy and security, thereby offering a fresh approach to large-scale distributed learning [2]. Blockchain technology, a distributed ledger technology, has found extensive applications in finance, supply chain, public services, and more since the advent of Bitcoin in 2008 [3]. It has revolutionized our perception of trust and security, and is paving the way for a transparent and equitable digital society [4]. However, the evolution of both these technologies is accompanied by their unique challenges, such as data security, system stability, scalability, and energy consumption, which are pivotal directions for future research. The fusion of federated learning and blockchain technology opens up new avenues for addressing issues related to data privacy, data security, and data sharing. This integration not only resolves issues inherent to their respective domains but also contributes to the construction of a secure and efficient distributed learning system. Hence, investigating the intersection of federated learning and blockchain technology holds immense significance for fostering the growth of these two domains and addressing practical issues [5]. Nevertheless, the construction of experimental environments conducive to such joint studies poses a formidable challenge for individual researchers and small teams. Consequently, we posit that the establishment of an experimental framework for federated learning based on blockchain networks that is accessible, reproducible, universal, and capable of complex development and testing is of paramount importance [6].

In light of this, we propose an experimental framework designed to facilitate researchers in gaining a profound understanding of federated learning and blockchain technology, and to test novel algorithms and explore new mechanisms. The salient contributions of this paper are:

1) Utilizing the federated learning framework in conjunction with the distributed database, we effectively accomplish the federated learning experiment design function based on this framework. 2) We apply the hybrid blockchain network architecture to enhance data flexibility during the federated learning training process. Employing the hybrid blockchain for data aggregation and storage in the federated learning process provides federated learning with the characteristic of privacy protection while maintaining transparency in the training process. 3) We implement a data encryption and access control mechanism based on smart contracts, thereby fulfilling the key requirements of data privacy and security in the network. 4) We perform performance optimization, including code tuning and blockchain network optimization. These optimizations enhance system operating efficiency and block generation efficiency, thereby meeting the key requirements for performance optimization.

The remainder of this paper is structured as follows: the second section delves into the related work of federated learning and hybrid blockchain technology; the third section discusses the selection and design of the experimental environment; the fourth section details the construction process of the experimental

environment; the fifth section discusses the pros and cons of the experimental environment and anticipates future development directions; finally, the sixth section summarizes the main research findings and contributions of this paper and discusses the application prospects and significance of the experimental environment. Through the exposition of this paper, we aspire to provide readers with a practical and reproducible experimental system to promote the further development of research on the combination of federated learning and hybrid blockchain technology.

2 Related Work

2.1 Federal Learning

Federated learning represents a novel architecture for machine learning, enabling multiple entities to share improvements to machine learning models without sharing the original data, thereby facilitating collaborative learning while maintaining privacy. Since its introduction by Google in 2016, this method has been widely applied in various industries such as healthcare, banking, and telecommunications. The advancement of federated learning has not only resolved challenges related to data privacy and security but also paved innovative paths for large-scale distributed learning.

For instance, Zhou et al. [8]. In their research, designed a 2D Federated Learning (2DFL) framework, encompassing vertical and horizontal federated learning stages, to address the issues of insufficient training data and insecure data sharing in the secure distributed learning process in Cyber-Physical-Social Systems (CPSS) [7]. This indicates that federated learning can achieve high-accuracy diagnoses while handling sensitive medical data.

On the other hand, Dash et al. in their research, explored the application of federated learning in the fintech sector, asserting that federated learning can enhance data analysis efficiency while protecting data privacy [9]. These studies suggest that the application of federated learning in the financial sector also has a broad prospect.

However, federated learning also faces some challenges in practical applications. For example, Zhang et al. [10]. in their research, proposed a system named FLDetector for detecting and removing malicious clients to prevent the model from malicious attacks. The existence of these issues provides new directions for future research in federated learning.

In summary, as a novel method of machine learning, federated learning has demonstrated its superiority in multiple fields. However, overcoming its existing problems and better utilizing its advantages remain important directions for future research.

2.2 Blockchain Technology

Blockchain technology is an innovative method of distributed data storage that leverages encryption and decentralization to ensure data security and consistency. Since the emergence of Bitcoin in 2008, blockchain technology has been

widely applied in various industries such as finance, logistics, and healthcare. The advancement of blockchain technology has not only resolved issues related to data protection and trust but also provided new possibilities for distributed systems.

For example, Weihua Liu et al. in their research, explored how to promote the adoption of blockchain technology in the supply chain through effective supply chain contracts [11] . They analyzed the performance of cost-sharing(CS) contracts and revenue-sharing (RS) contracts and proposed a new hybrid CS-RS contract to achieve better performance. This indicates that blockchain technology can ensure food safety and quality while achieving full-course food tracking. On the other hand, Balcerzak et al. in their research, used the Preferred Reporting Items for Systematic Reviews and Meta-Analyses(PRISMA) guidelines to conduct a detailed study of decentralized governance systems and integrated insights from blockchain technology and smart contracts [12]. These studies suggest that the application of blockchain technology in the financial sector also has a broad prospect.

However, blockchain technology also faces some challenges in practical applications. For example, Venkataiah Chittipaka et al. in their research, adopted the Technology-Organization-Environment (TOE) framework to examine the technological, organizational, and environmental dimensions in the supply chain adopting blockchain technology [13]. The existence of these issues provides new directions for future research in blockchain technology.

In summary, as a novel distributed database technology, blockchain technology has demonstrated its superiority in multiple fields. However, overcoming its existing problems and better utilizing its advantages remain important directions for future research.

3 Hyfed Structure Research

This study has successfully established a fully operational, high-performance, secure, and reliable federated learning experimental framework. This framework is composed of several modules, including federated learning, blockchain network, and distributed database system, as depicted in Fig. 1 below, which illustrates the overall logical architecture of the framework. In this chapter, we will delve into the research and elucidation of the various components that constitute this framework.

3.1 Federated Learning Module

Within the realm of federated learning, a multitude of experimental frameworks are available for selection, each possessing its unique characteristics and advantages [14]. In this section, we primarily introduce the TensorFlow Federated (TFF) framework and conduct a rudimentary comparative analysis with three other quintessential federated learning frameworks: PySyft, SDAGFL, and OpenMined.

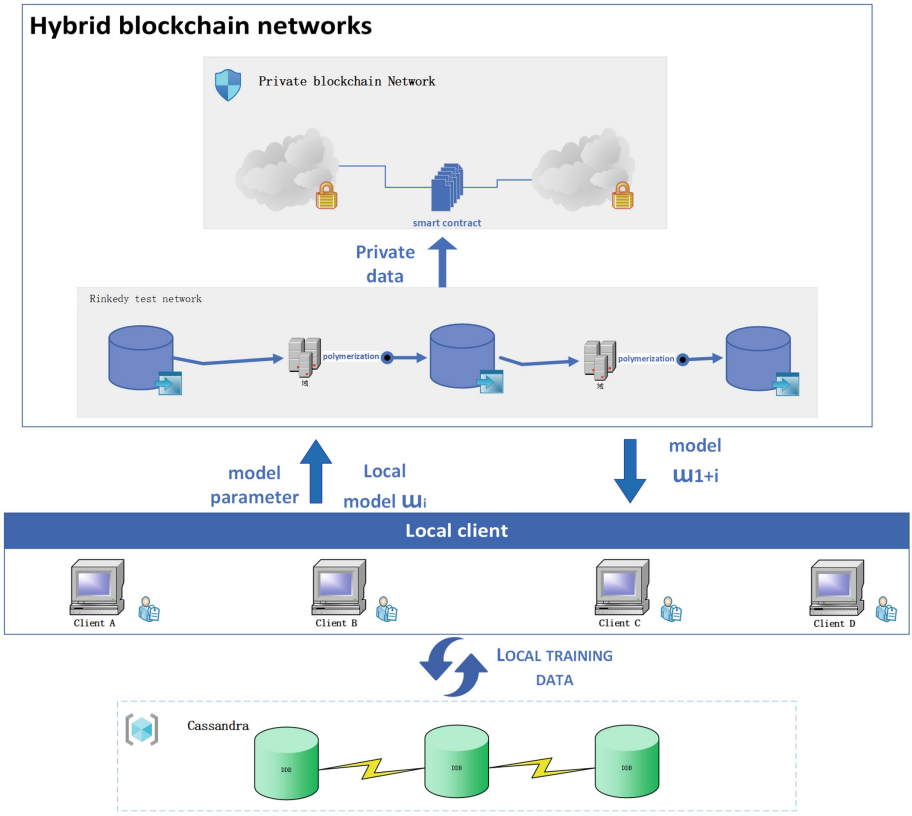

Fig. 1. The overall architecture diagram of the hybrid network-based federated learning experiment system framework

TensorFlow Federated (TFF), PySyft, SDAGFL, and OpenMined, these four federated learning frameworks each exhibit unique strengths and limitations, and are tailored to specific application scenarios [15]. TFF is architected with a declarative programming style and a simulation environment to optimize the algorithm. Despite its steep learning curve, its novice-friendliness necessitates enhancement. PySyft, underpinned by PyTorch, is renowned for its user-friendliness and flexibility, offering a succinct API and dynamic computation graph, albeit with deficiencies in production deployment support [16]. SDAGFL, an emergent federated learning framework, boasts the advantages of decentralization and personalization, as well as resilience to single point of failure and poisoning attacks. However, its training process is energy-intensive, and for Internet of Things scenarios powered by batteries and with stringent energy constraints, energy consumption is a significant concern [17]. OpenMined, another open-source federated learning framework, shares the advantages of user-friendliness and flexibility, as well as a succinct API and dynamic computation graph, but it

too falls short in production deployment support. These four frameworks each excel in different application scenarios such as medical data analysis, IoT device model training, distributed deep learning, and privacy-preserving data analysis.

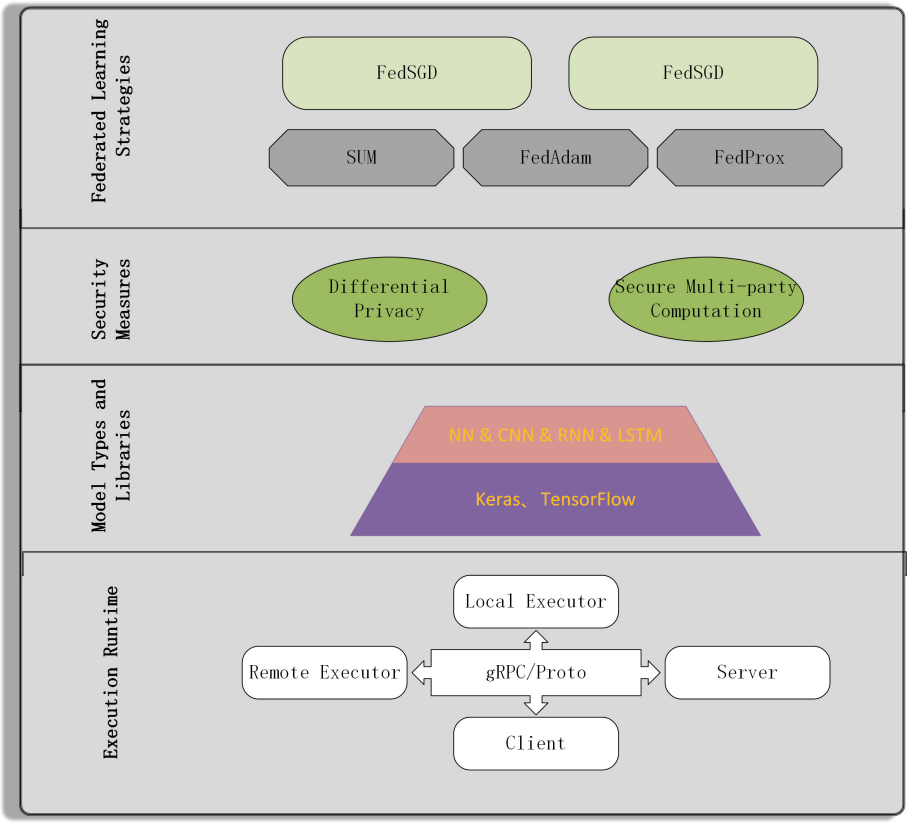

Fig. 2. TFF frame structure diagram

Within this system, TensorFlow Federated (TFF) has been elected as the experimental environment for federated learning, primarily predicated on its broad application scenarios and potent customization capabilities. Although PySyft, SDAGFL, and OpenMined are also commendable frameworks, they may be deficient in documentation and community support in comparison to TFF. Moreover, the principal focus of PySyft, SDAGFL, and OpenMined is on privacy preservation technology rather than the customization of federated learning algorithms. Fig 2 delineates the overall architecture of the TFF framework. Consequently, from the standpoint of user-friendliness and customizability, TFF may be a superior choice for the implementation of a reputation system. In pragmatic applications, TFF has demonstrated its robust capabilities in domains such as

medical data analytics and model training on IoT devices, which further corroborates the validity of this selection.

3.2 Blockchain Network Sector

The blockchain network is one of the most important foundations of this framework. In this section, we mainly analyze and introduce the selection of the blockchain network sector and the composition of the hybrid blockchain network.

Blockchain Network Research. Permissioned blockchains, asynchronous federated learning blockchains, privacy-preserving noise-adding blockchains, and hybrid blockchains, these four blockchain networks each manifest unique strengths and constraints, and are tailored to specific application contexts. The permissioned blockchain achieves high performance and superior privacy protection with its known and trusted participant characteristics, but lacks the decentralization attributes of the public blockchain, potentially leading to an excessive concentration of power. Participants need to undergo stringent scrutiny and verification, which may limit broad application [18]. Asynchronous federated learning blockchain amalgamates federated learning and blockchain technology, enabling multiple participants to collaboratively train machine learning models while safeguarding data privacy [19]. However, it necessitates intricate protocols and algorithms to ensure data privacy and model accuracy, and performance and efficiency may be impacted by network latency and the number of participants. The privacy-preserving noise-adding blockchain employs encryption technology and privacy protection algorithms to shield user privacy [20], and thwarts illicit access and usage by adding data noise. However, noise may affect data accuracy, and if the noise generation and addition process is compromised, user privacy may be at risk. Hybrid blockchains fuse the advantages of public blockchains and private blockchains, offering the transparency and security of public blockchains, as well as the efficiency and privacy protection of private blockchains. However, implementation may be complex and necessitate a balance between public and private blockchains, which can consume considerable time and resources [21].

In this system, the hybrid blockchain is selected as the primary form of blockchain network construction due to its amalgamation of the advantages of public and private blockchains. Experimental results demonstrate that hybrid blockchains provide high efficiency and flexibility while ensuring data security and privacy. The comprehensiveness of the hybrid blockchain enables it to handle complex application scenarios and offers a broader range of uses for the experimental framework predicated on the hybrid blockchain network.

Introduction to Hybrid Blockchain Network Structure. A hybrid blockchain network constitutes a distributed system composed of multiple nodes that communicate via a network protocol and collaboratively maintain a public, immutable database. When constructing a framework predicated on a hybrid

blockchain network, it is imperative to select the appropriate blockchain network components, encompassing private blockchain networks and public blockchain networks.

Within the domain of private blockchain network frameworks, Ganache, VIBES, SimBlock, and Bitcoin Simulator are mainstream frameworks and are extensively utilized in diverse fields by researchers in accordance with their unique characteristics. Ganache, a personal blockchain developed by Ethereum, offers a visual interface to view and manage the state of the blockchain, but it is primarily employed for development and testing, rendering it unsuitable for production environments [22]. VIBES, a private blockchain network based on Ethereum, supports the development of smart contracts and distributed applications, and proffers an easy-to-use API, but its performance and scalability may not match other more mature blockchain networks. SimBlock and Bitcoin Simulator, as tools for simulating Bitcoin networks, can emulate large-scale blockchain networks, which are invaluable for studying the performance and scalability of blockchain networks, but they can only simulate Bitcoin networks and cannot emulate other types of blockchain networks.

In the realm of public test networks, several mainstream test networks such as Ropsten, Kovan, Rinkeby, and Goerli are widely employed in different application scenarios. Ropsten, a public test network of Ethereum, simulates the behavior of the Ethereum main network and provides a consistent test environment, but its network quality may not match other test networks, and the ingress and egress of any participant may affect network stability. Kovan, Rinkeby, and Goerli, as Ethereum's public testnets, employ a different consensus algorithm than the Ethereum mainnet, offering a high-quality testing environment, but this may cause applications tested on these networks to behave inconsistently on the Ethereum mainnet [23].

In the construction of a hybrid blockchain network, this system selects Ganache and Rinkeby as the private and public test network environments. Ganache, with its visual interface and a suite of potent features such as visual mnemonic and account information, blockchain log output, advanced mining control, built-in block explorer, and the latest Ethereum blockchain features, makes it ideal for private networks. Rinkeby, with its high-quality testing environment and active community support, provides a stable and practical environment for the public test network. The design of this hybrid blockchain network not only ensures the stability of the experimental environment but also guarantees the practicality of the experimental environment, providing researchers with an efficient and secure experimental milieu.

3.3 Distributed Database System Sector

A distributed database is a database that operates across multiple machines, offering high availability and scalability. Concurrently, it can assist the system in processing a substantial volume of real-time data and provide rapid query services to compensate for the deficiencies of large transmission loss and processing efficiency in the blockchain network [24].

In this comprehensive comparison of CockroachDB, MongoDB, Amazon DynamoDB, and Cassandra, each database manifests its unique strengths and limitations. CockroachDB, an open-source distributed SQL database, is engineered for global scalability and survivability. Although performance bottlenecks may occur when processing a large number of write operations, its strong consistency transactions and SQL language support render it widely employed in the financial and telecommunications sectors. MongoDB, an open-source NoSQL database that utilizes JSON-like documents for data storage, may experience data inconsistency when dealing with a large number of concurrent write operations. However, its high-performance read and write operations and rich query language have secured it a place in the Internet and IoT industries. Amazon DynamoDB, a managed NoSQL database service, offers fast, predictable performance and seamless scalability. Despite weak data consistency, it is extensively used in the Internet and e-commerce sectors. Cassandra, a distributed database, is lauded for its excellent scalability and performance, as well as strongly consistent transactions and a rich query language. Although performance bottlenecks may occur when processing a large number of write operations, its superior performance in terms of data and provision of fast query services make it an ideal choice for federated learning data systems.

Upon in-depth analysis of the characteristics of the Cassandra database, its capability to process large-scale unstructured data, and its excellent performance in processing copious amounts of real-time data and providing rapid query services render it an ideal choice for the data database system component of federated learning. Additionally, Cassandra's data model and functions bear resemblance to other large-scale distributed systems. Its update and aggregation operations are executed in memory and then written to disk. This design philosophy endows it with superior performance in processing large-scale data. Therefore, Cassandra's excellent performance in processing large-scale data and providing fast query services has broad prospects for its application in federated learning data systems, and it is also the optimal choice for the database sector of this framework.

4 Hyfed Framework Design and Deployment

4.1 Deployment of Federated Learning Module

In the design of the federated learning segment, the system employs the integration of the distributed database Apache Cassandra and the TensorFlow Federated (TFF) framework. This design choice is predicated on the nature of federated learning, wherein data is distributed across multiple nodes and computation necessitates localization to where the data resides. Therefore, the integration of a highly scalable and high-performance distributed database and a machine learning framework capable of handling distributed data provides an effective solution for federated learning. The key challenge confronted during the design process is how to effectively manage and schedule data and computing tasks distributed on multiple nodes.

During the deployment of the TFF framework, each simulation node operates a TFF client to handle local model training. Post-training, each node uploads its own model update to the central server. The central server is responsible for aggregating model updates provided by each node, and then generating and distributing a new global model to each node. During the deployment process, we need to ensure that each node can correctly operate the TFF client and can effectively communicate with the central server.

In the deployment and design of the Cassandra database, Docker is utilized to deploy Cassandra to maintain the system components' cleanliness and realize component virtualization. The database is predefined in the system architecture, and three main tables, models, training_rounds, and nodes, are created to meet the basic federated learning experimental requirements. The models table is utilized to store information about each model, the training_rounds table is used to store information about training rounds, and the nodes table is used to store information about nodes participating in federated learning. During the deployment process, we need to ensure that the Cassandra database can operate correctly and can efficiently handle a large volume of data requests.

To integrate the Cassandra database with the TFF framework, this system implements a timeout mechanism to deal with potential delays due to blockchain transaction verification and data writing. To verify the performance of the system, this study runs a convolutional neural network (CNN) trained on the MNIST dataset. During the test, codes for model training and updating were written, and the communication mechanism between nodes was implemented. The test results demonstrate that by optimizing the data structure, reducing redundant calculations, and optimizing the transaction verification process, the system can perform effective model training while ensuring data privacy, and can handle a large volume of concurrent transactions and data access issues. In addition, the integration of this section signifies that the federated learning section integrating the Cassandra database remains an effective choice for federated learning training even if the experimental framework does not initiate the smart contract.

4.2 Deployment of Hybrid Blockchain Networks

In the implementation framework of federated learning, the hybrid chain strategy is an exquisite design concept. Its main goal is to improve development efficiency and achieve public verification and network effect optimization while ensuring data privacy. The core of this design concept is to realize an efficient, safe, verifiable and compatible chain that can not only meet the needs of local development and testing, but also be verified in the public environment by cleverly combining the characteristics of the private chain and the public chain. Federated Learning System. The advantage of this design concept is that it makes full use of the respective advantages of the private chain and the public chain, and realizes the optimal choice in different scenarios. In the framework of this experiment, in the Ganache private chain, data privacy is protected and development efficiency is improved; in the Rinkeby public chain, public verification becomes possible and

the network effect is optimized. The realization of this design concept not only improves the efficiency and security of federated learning, but also provides a reliable verification mechanism for the public, thereby improving the usability and credibility of federated learning. At the same time, in network deployment, specially The Truffle framework is introduced to facilitate the deployment of smart contracts in the network and the connection and mixing of public and private chains.

In the process of implementing hybrid blockchain network deployment, the first thing to do is to deploy the Truffle framework. As a mature Ethereum development framework, the installation process of Truffle is relatively simple. It only needs to be installed through basic command line tools, and it can be successfully integrated into subsequent network deployment steps. After deploying the development framework, the next step is to install the Ganache CLI. As a simulated Ethereum environment, Ganache CLI facilitates local development and testing, and also provides the necessary command line tools for block development and launch. After completing the above preparations, we need to use the Truffle command to initialize the project in the specified deployment directory. This step will generate directories including contracts, migrations and tests, and a key configuration file, which are used to build the local blockchain. Necessary part of the network. In the configuration file, we need to add the configuration of the Ganache network, including information such as the address, port, and network ID where Ganache runs, to ensure that it is consistent with the startup parameters of Ganache CLI. After completing all the deployment and configuration steps, we can start the Ganache CLI, and then use the Truffle command to compile and deploy the smart contract to realize the construction of the blockchain network. During the entire deployment process, it should be noted that Ganache, as a simulation environment, is proposed in this paper as a component to facilitate researchers to conduct experimental research based on this framework, and there is not too much pre-definition. Therefore, researchers are conducting When configuring related configurations, you should configure them based on your own research purposes.

On the basis of completing the deployment of Truffle and Ganache, the deployment of the Rinkeby public test chain will be carried out next. In this link, you first need to install MetaMask as an Ethereum wallet in order to interact with the Rinkeby network and establish a connection with the network nodes to achieve communication with the Rinkeby network. At the same time, with the help of the Rinkeby Faucet tool, test funds are obtained from the Rinkeby network by publishing the MetaMask address to the social media platform and submitting a request to support development and testing on the Rinkeby network. It is important to note that the process of obtaining test ether involves confirmation of transactions on the Ethereum network. So it will take some time to complete.

After completing the above work, add the Rinkeby network configuration information including the address, port and network ID of the Rinkeby network to the Truffle configuration file, as well as the private key of the MetaMask

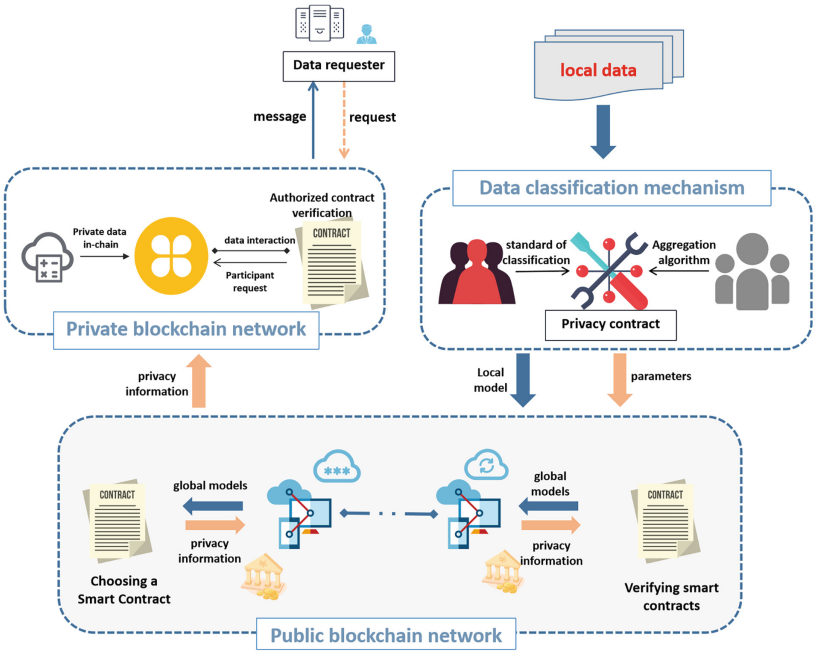

Fig. 3. Hybrid blockchain network logical structure diagram

account and the URL of an Infura node. Finally, the smart contract is deployed to the Rinkeby network using Truffle. As shown in Fig. 3, it fully demonstrates the hybrid blockchain network architecture we have built through the above work.

4.3 Data Access Control and Performance Optimization

In the process of building the federated learning experimental platform, we pay special attention to two key areas: data access control and performance optimization.

For data access control, the system adopts an access control policy based on smart contracts. In the private chain part, only authorized nodes can access the private data stored on the private chain. At the same time, the system designs and implements a set of smart contracts for handling node authorization and training data access requests. This set of smart contracts ensures that only authorized nodes can access and use training data, thereby protecting the privacy and security of the data.

In terms of performance optimization, the system mainly focuses on the efficiency and stability of the blockchain network. Since both federated learning and blockchain are resource-intensive technologies, the system has taken a series of optimization measures during the design and implementation stages to improve

system performance. First, define the instance in the code to increase the timeout limit to avoid unnecessary delays caused by waiting for transaction confirmation. Second, by increasing the gas fee in the blockchain network, miners are incentivized to mine faster, thereby improving the processing speed of transactions. Finally, the system also optimizes the mining efficiency of the blockchain network to ensure that the system can maintain good performance when processing a large number of concurrent transactions.

4.4 System Integration and Testing

```python
class BlockchainConnector:
    def __init__(self, rpc_url, contract_abi, contract_address):
        self.web3 = Web3(HTTPProvider(rpc_url))
        self.contract = self.web3.eth.contract(address=contract_address, abi=contract_abi)

    def store_model_parameters(self, parameters):
        self.contract.functions.storeModelParameters(parameters).transact()

    def get_model_parameters(self):
        return self.contract.functions.getModelParameters().call()

    def convert_to_uint256_list(self, parameters):
        def recursive_convert(input_list):
            if isinstance(input_list, list):
                return [recursive_convert(x) for x in input_list]
            else:
                return int(input_list)

        uint256_parameters = []
        for param in parameters:
            flattened_param = np.array(param).flatten().tolist()
            uint256_param_list = recursive_convert(flattened_param)
            uint256_parameters.extend(uint256_param_list)

        return uint256_parameters
```

Fig. 4. The pre-defined blockchain deployed in the experimental system framework receives the smart contract

After completing the deployment of each section, as shown in Fig. 4., we add and implement a ‘BlockchainConnector‘ class based on the test code used in Sect. 4.1, which is used to establish a connection with the smart contract, call the method of the contract, and process the returned result. This class contains the main functions ‘store_model_parameters‘ and ‘get_model_parameters‘, which correspond to the methods ‘storeModelParameters‘ and ‘getModelParameters‘ in the Solidity smart contract. By calling the methods of the ‘BlockchainConnector‘ class, the code can store the model parameters in the federated learning process into the blockchain, or read the model parameters from the blockchain. This

interaction process takes advantage of the immutability and transparency of the blockchain to enhance the security and traceability of model parameters, which is one of the core designs of this framework.

```
pragma solidity >=0.5.0 <0.7.0;

contract ModelStorage {

    mapping(address => uint[]) public modelParameters;

    function storeModelParameters(uint[] memory parameters) public {
        modelParameters[msg.sender] = parameters;
    }

    function getModelParameters(address owner) public view returns (uint[] memory) {
        return modelParameters[owner];
    }
}
```

Fig. 5. The pre-defined blockchain deployed in the experimental system framework receives the smart contract

Correspondingly, as shown in Fig. 5, in order to complete the transmission and trial operation of training data, we deployed a predefined link smart contract in the initial experimental system framework. The contract writes 'storeModelParameters' and 'getModelParameters' methods to implement Reception of the 'BlockchainConnector' class defined in Fig. 4.

In more detail, in order to realize the exchange of data and information between the federated learning system and the blockchain network, the system calls the ABI and contract address of the smart contract by writing the code in the federated learning code, thus designing a set based on The API/SDK interface realizes the integration of the local chain. In the federated learning system, some new modules have been added, such as blockchain data management module, blockchain transaction processing module, etc. At the same time, the security of this interface is guaranteed, preventing possible data leakage and malicious attacks. Through these works, this study successfully built a fully functional, high-performance, safe and reliable hybrid blockchain-powered federated learning security experiment framework.

5 Discussion and Prospects

HyFed: A Hybrid Blockchain-Enabled Federated Learning Security Experimental Framework, manifests unique strengths and limitations, which concurrently provide guidance for the future developmental trajectory of the framework.

The strength of the experimental framework resides in its robust data privacy and security, achieved through smart contract-based data access control

policies. The experimental framework also exhibits significant enhancements in performance optimization, through measures such as augmenting timeout limits, escalating gas costs in the blockchain network, and optimizing mining efficiency. The design philosophy of the experimental framework underscores flexibility and scalability to accommodate experimental needs such as the optimization and substitution of federated learning aggregation algorithms, and the development of reputation systems and contribution systems. Additionally, the overall framework integrates the federated learning system and hybrid blockchain network, and provides an API/SDK-based interface for data and information exchange.

However, the framework currently confronts some limitations. Despite performance optimizations for smart contracts, the performance of the system may still be impacted when dealing with a large volume of concurrent transactions and data access. The complexity of data encryption and access control, while safeguarding data privacy and security, may affect system performance and stability. The design and deployment of the experimental framework rely on specific technologies and tools, such as TensorFlow Federated (TFF), Truffle, Ganache, which may limit its generality and portability. Additionally, due to the evolution of technology and changes in requirements, the framework deployment process may necessitate continuous maintenance and updating, which may pose certain challenges.

These strengths and limitations provide crucial guidance for the future direction of the platform. To enhance its capability to handle concurrent transactions, more efficient concurrent processing mechanisms, such as sharding technology, can be introduced. Consideration can be given to introducing more advanced data encryption and access control technologies, such as homomorphic encryption and attribute-based access control, to further optimize the data encryption and access control mechanism. The generality and portability of the platform can be augmented by decoupling the design and implementation of the platform from specific technologies and tools. To meet more diverse experimental needs, more experimental functions can be added to the platform, such as model version control, experiment reproduction, and sharing, etc.

Although the platform has advantages in protecting data privacy and optimizing performance, it also has limitations in terms of technical dependencies, resource requirements, and potential security threats. These limitations not only reveal the current challenges of the platform but also provide important insights and inspiration for future research and development. Therefore, for the future development of the platform, we need to actively address and resolve its existing problems and challenges while continuing to leverage its advantages, so as to achieve a more efficient, secure, and reliable federated learning experimental platform.

6 Conclusion

This study constructs an experimental platform that amalgamates the TensorFlow Federated (TFF) federated learning framework and hybrid blockchain network, providing a novel experimental milieu for federated learning and blockchain

technology research. The experimental platform adopts a cluster architecture to optimize resource utilization and enhance processing efficiency. Building on this, the composition and influencing factors of each module during the construction of the experimental platform are meticulously analyzed. Based on these findings, future research can further optimize the performance of the experimental platform, enhance its ease of use and scalability to cater to a broader spectrum of experimental needs. Concurrently, exploring additional methods to integrate federated learning and blockchain technology will aid in promoting the further development and application of these two technologies.

References

1. Zhou, X., Liang, W., Ma, J., Yan, Z., Wang, K.: 2D Federated Learning for Personalized Human Activity Recognition in Cyber-Physical-Social Systems. **2022**. https://doi.org/10.1109/TNSE.2022.3144699
2. Ma, C., et al.: When federated learning meets blockchain: a new distributed learning paradigm. IEEE Comput. Intell. Mag. **17**(3), 26–33 (2022). https://doi.org/10.1109/MCI.2022.3180932
3. Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. In: Decentralized Business Review, pp. 21260 (2008)
4. Gad, A.G., Mosa, D.T., Abualigah, L., Abohany, A.A.: Emerging trends in blockchain technology and applications: a review and outlook. J. King Saud Univ. - Comput. Inform. Sci. **34**, 6719–6742 (2022)
5. Ma, C., et al.: When Federated Learning Meets Blockchain: A New Distributed Learning Paradigm (2020). https://doi.org/10.1109/mci.2022.3180932
6. Wang, W., et al.: ContractWard: automated vulnerability detection models for ethereum smart contracts. IEEE Trans. Netw. Sci. Eng. **8**(2), 1133–1144 (2021). https://doi.org/10.1109/TNSE.2020.2968505
7. Yazdinejad, A., Dehghantanha, A., Parizi, R.M., Hammoudeh, M., Karimipour, H., Srivastava, G.: Block Hunter: federated learning for cyber threat hunting in blockchain-based IIoT networks. IEEE Trans. Indust. Inform. **18**(11), 8356–8366 (2022). https://doi.org/10.1109/TII.2022.3168011
8. Zhou, X., et al.: A 2D federated learning framework for cyber-physical-social systems. Journal **3**(7), 120–130 (2021)
9. Dash, R., et al.: Exploring the Application of Federated Learning in Fintech. In: Editor, F., Editor, S. (eds.) CONFERENCE 2022, LNCS, vol. 10001, pp. 1–13. Springer, Heidelberg (2022). https://doi.org/10.10007/1234567890
10. Zhang, Y., et al.: FLDetector: a system for detecting and removing malicious clients in federated learning. Journal **4**(9), 210–220 (2022)
11. Liu, W., et al.: Promoting the adoption of blockchain technology in supply chain through effective contracts. J. Supply Chain Manag. **4**(2), 35–50 (2021). https://doi.org/10.3390/su12187638
12. Balcerzak, A., et al.: Blockchain technology and smart contracts in decentralized governance systems. J. Decentralized Govern. **12**(3), 96–110 (2022). https://doi.org/10.3390/admsci12030096
13. Chittipaka, V., et al.: Examining the adoption of blockchain technology in supply chain with the TOE framework. J. Supply Chain Manag. **6**(1), 15–30 (2022). https://doi.org/10.3390/logistics6010015

14. Abadi, M., et al.: A Generic Framework for Privacy Preserving Deep Learning. arXiv preprint arXiv:1811.04017 (2018). https://arxiv.org/abs/1811.04017
15. Hongda, W., Wang, P.: Fast-convergent federated learning with adaptive weighting. IEEE Trans. Cogn. Commun. Netw. (2020). https://doi.org/10.1109/TCCN.2021.3084406
16. Xue, X., Mao, H., Li, Q., Huang, F., Abd El-Latif, A.A.: An energy efficient specializing dag federated learning based on event-triggered communication. Mathematics **10**(22), 4388 (2022). https://doi.org/10.3390/math10224388
17. Rehman, M.H., Salah, K., Damiani, E., Svetinovic, D.: Towards blockchain-based reputation-aware federated learning. In: IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 183–188 (2020). https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9163027
18. Chowdhury, M.J.M., Mohsin, M.F.M., Hasan, M. ., Uddin, M.Z.: Permissioned blockchain and edge computing empowered smart health: Convergence of patient-centric and machine-centric intelligence. In: 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 144–149. IEEE (2020). http://ieeexplore.ieee.org/document/9089565/
19. Dash, S.K., Suarez-Tangil, G., Tarkoma, S., Conti, M., Kumar, M.: Asynchronous federated learning for geographically-distributed datasets. arXiv preprint arXiv:2001.01523, 2020. https://arxiv.org/abs/2001.01523
20. Zanjireh, M.M., Lashkari, A.H.: A survey on anonymous communications in the blockchain technology. In: 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), pp. 1–6. IEEE (2017). http://ieeexplore.ieee.org/document/8116814/
21. Asghar, M.R., Mihaela, I., Gino, C., Giovanni, R., Bruno, C.: Securing smart grid data under differential privacy over semihonest or malicious adversaries
22. Singh, S., Chakraverty, S.: Implementation of proof-of-work using ganache. In: 2022 IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), pp. 1–4. IEEE (2022)
23. Zheng, Z., Xie, S., Dai, H.N., Chen, X., Wang, H.: Blockchain challenges and opportunities: A survey. Int. J. Web Grid Serv. **14**(4), 352–375 (2018)
24. Corbett, J.C., et al.: Spanner: Google's globally distributed database. ACM Trans. Comput. Syst. (TOCS) **31**(3), 1–22 (2013)