2024 International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan) | 979-8-3503-8684-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCE-Taiwan62264.2024.10674219

An Implementation of Blockchain-based Federated Learning Framework

Po-Hsuan Hung*, Bing-Siang Wang*, Yu-Wei Chang*, Ted T. Kuo†, Li-Jen Wang†, and Chia-Yu Lin*
* *Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan*
† *College of Artificial Intelligence, National Yang Ming Chiao Tung University, Tainan, Taiwan*
Corresponding Author: Chia-Yu Lin (sallylin0121@ncu.edu.tw)

***Abstract*—While deep learning technologies are rising, information security is becoming increasingly important in the modern era. Federated Learning (FL) is a method that allows different clients to collaborate on training deep learning models while ensuring data privacy. However, traditional FL faces the challenge of a single point of failure with the aggregation server if the centralized server fails. Additionally, FL often encounters malicious model attacks, leading to a reduction in global model accuracy. To address these issues, we propose Scale BlockChain Federated Learning (ScaleBCFL) using Hyperledger Fabric (HLF). By leveraging blockchain technology, we decentralize the central server in traditional FL and create multiple shards to reduce communication costs significantly. Furthermore, using a blockchain handler obviates the necessity for users to invest time in learning how to write smart contracts. Instead, users can focus solely on customizing programs related to training, aggregation, and validating models.**

***Index Terms*—Blockchain, Federated Learning, Hyperledger Fabric**

I. Introduction

Clients' models are sent to a centralized server for aggregation in traditional federated learning. This causes a single point of failure, where the failure of the central server disrupts the entire FL process. Besides, validation of models using a verification dataset by a centralized server may raise privacy concerns. The verification dataset could contain sensitive information, limiting its use to publicly available datasets. Finding ways to eliminate the single point of failure and enable decentralized validation processes without compromising privacy is essential for advancing FL's effectiveness and security. The decentralization of the central server in FL has become a critical research topic for addressing these challenges.

There are many related studies on using blockchain to decentralize FL. For example, BLADE-FL by Ma, Chuan et al. [1], FL-Block by Qu, Youyang et al. and BFLC by Li, Yuzheng et al. [2] explored decentralizing public chains. There were also studies using Hyperledger Fabric (HLF) as their framework, such as ScaleSFL by Madill er al. [3], ChainsFL by Yuan, Shuo et al. [4] and FabricFL by Mothukuri et al. [5]. However, ChainsFL and FabricFL are aggregated on the client side, which makes synchronization between clients difficult. Although our research is based on ScaleSFL, we decentralize all operations from publishing tasks to training by HLF.

We propose Scale BlockChain Federated Learning (ScaleBCFL). This framework decentralizes FL with blockchain, which is implemented with Hyperledger Fabric (HLF), using channels of the HLF framework as both task publishing platform and aggregation model consensus platform between organizations, replacing traditional centralized servers. ScaleBCFL has the following functions.

(i) ScaleBCFL supports hierarchical FL [6].
(ii) FL task release platform and training-aggregation process are decentralized.
(iii) Allowing users to focus only on training, aggregation, and verification without caring about smart contracts.

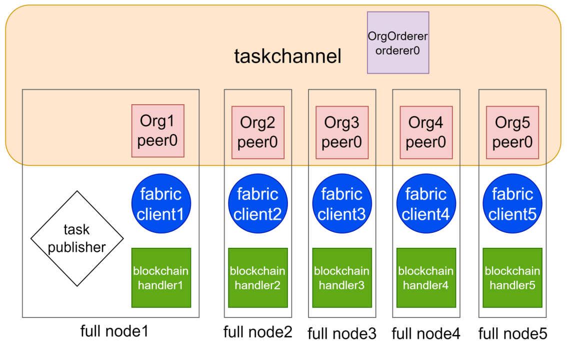

Fig. 1. System architecture.

II. System Architecture

The system architecture is shown in Fig. 1. In our framework, each organization maintains a node, specifically a full node. This node comprises several components: an HLF peer responsible for participating in HLF channels, a blockchain handler primarily tasks with federated learning model training and endorsing transaction requests sent from peers, a fabric client serving as the communication bridge between the blockchain handler and peers, and optionally, a task publisher to join different HLF channels and store the channel's account simultaneously. Inter-organization communication occurs via HLF channels, ensuring that any ledger modification on the channel requires approval from other organizations, thereby decentralizing our framework.

III. Methods

As shown in Fig. 2, the blockchain handler controls this system's operation. It writes the model address, hash, or start training message into the ledger through the fabric client and

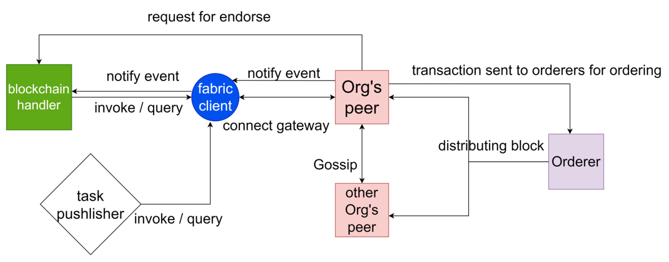

Fig. 2. Interaction between components

HLF peer. For other organizations' ledger writes, the endorsement plugin in the HLF peer sends endorsement requests to the blockchain handler for checking to identify malicious models or ledger writes that damage the system. The event generated after successful ledger writing is returned to the blockchain handler through the HLF peer and fabric client so that the blockchain handler can perform corresponding operations based on the written content, such as starting aggregation or a new training round.

The operation process of ScaleBCFL can be roughly divided into two major steps. "Task Publishing" and "Training-Aggregating". The first is Task Publishing. The organization's task publisher sends the FL task information to other organizations. Organizations respond to the task publisher on whether to participate. After confirming the participants, the task publisher determines the hierarchical structure according to the participants, establishes HLF channels based on the structure, and writes the structure into the ledger. Finally, each organization joins the corresponding channel according to the hierarchical structure in the ledger, installs the chaincode, and starts the fabric client. This completes the "Task Publishing" step. Next is "Training-Aggregating". The task publisher sends the training message to each organization through the channel and writes the initial model into the ledger. After the organization receives the training message, it sends it to the lower channel or uses the initial model in the ledger for local training. When the local training is complete, the address of the local model will be written into the ledger, and the peers of each organization will send the ledger writes to the blockchain handler to identify malicious models by measuring the L2 distance between the global model and the model being written. If the number of uploaded models is enough, each organization will aggregate these models individually to obtain candidate aggregation models. Upload the candidate aggregation model address and its hash to the ledger, and then use chaincode to set the model selected by most organizations as the new global model, thus completing a round of "Training-Aggregating".

IV. Discussion

A. Hierarchical Sharding

Hierarchical sharding allows for the independent operation of each shard, bringing several enhancements to traditional FL workflows. During device selection, smaller populations in each shard enhance the efficiency of updating and maintaining active devices. A more extensive sampling of devices from the worldwide population becomes feasible during aggregation [3].

B. Rewards Distribution

Although we have implemented ScaleBCFL using HLF, alternative approaches might explore platforms like Ethereum for implementing reward mechanisms. As training models in a permissionless manner could lead to reduced contributions, given the accessibility of models via a model hub without the need for significant computing resources, incorporating rewards becomes a crucial aspect of the workflow. While maintaining free access to models, it is essential to recognize and incentivize contributing computing resources by distributing rewards to the participating clients. Task contributors or interested clients may introduce additional incentives to enhance community involvement and stimulate contributions [3].

V. Conclusion

This paper proposes the ScaleBCFL framework to decentralize FL to solve the single point of failure problem. This framework uses Hyperledger Fabric to build a blockchain, allowing organizations to communicate with each other through channels, and uses the fabric client and endorse plugin to enable the blockchain handler to call different functions based on events or ledger writes in the channel. This framework not only supports hierarchical FL but also allows the endorse step of HLF to be checked by the blockchain handler, allowing users to easily customize training, aggregation, and verification programs without spending time and cost learning how to write smart contracts.

Acknowledgements

This work is sponsored by the National Science and Technology Council (NSTC) under the projects NSTC 110-2222-E-008-008-MY3 and NSTC 112-2622-8-A49-021.

References

[1] Chuan Ma, Jun Li, Long Shi, Ming Ding, Taotao Wang, Zhu Han, and H Vincent Poor, "When federated learning meets blockchain: A new distributed learning paradigm," *IEEE Computational Intelligence Magazine*, vol. 17, no. 3, pp. 26–33, 2022.

[2] Yuzheng Li, Chuan Chen, Nan Liu, Huawei Huang, Zibin Zheng, and Qiang Yan, "A blockchain-based decentralized federated learning framework with committee consensus," *IEEE Network*, vol. 35, no. 1, pp. 234–241, 2020.

[3] Evan Madill, Ben Nguyen, Carson K Leung, and Sara Rouhani, "Scalesfl: a sharding solution for blockchain-based federated learning," in *ACM International Symposium on Blockchain and Secure Critical Infrastructure*, 2022.

[4] Shuo Yuan, Bin Cao, Mugen Peng, and Yaohua Sun, "Chainsfl: Blockchain-driven federated learning from design to realization," in *IEEE Wireless Communications and Networking Conference*, 2021.

[5] Viraaji Mothukuri, Reza M Parizi, Seyedamin Pouriyeh, Ali Dehghantanha, and Kim-Kwang Raymond Choo, "Fabricfl: Blockchain-in-the-loop federated learning for trusted decentralized systems," *IEEE Systems Journal*, vol. 16, no. 3, pp. 3711–3722, 2021.

[6] Lumin Liu, Jun Zhang, SH Song, and Khaled B Letaief, "Client-edge-cloud hierarchical federated learning," in *ICC 2020-2020 IEEE international conference on communications (ICC)*. IEEE, 2020, pp. 1–6.