wjert, 2021, Vol. 7, Issue 1, 322-332. Review Article ISSN 2454-695X

World Journal of Engineering Research and Technology
WJERT
www.wjert.org

SJIF Impact Factor: 7.029

BLOCKCHAIN-ENABLED FEDERATED LEARNING FOR TRUSTWORTHY MULTI-AGENT SYSTEMS

Gift Aruchi Nwatuzie*

Article Received on 21/11/2020 **Article Revised on 11/12/2020** **Article Accepted on 01/01/2021**

***Corresponding Author**
Gift Aruchi Nwatuzie

ABSTRACT

ulti-agent systems (MAS) are pivotal in various domains requiring collaborative intelligence. However, tradi- tional data-centric approaches raise privacy, trust, and security concerns. This paper proposes an integrated framework com- bining federated learning (FL) with blockchain technology to address these challenges. Our architecture utilizes permissioned blockchain for decentralized trust management and smart con- tracts for enforcing dynamic reputations and secure participation. We detail the methodology, implementation, and experimental evaluation in autonomous vehicles and smart grid environments. Results demonstrate significant improvements in robustness and trust assurance compared to state-of-the-art methods, validating the system's effectiveness for real-world MAS applications. Index Terms—Federated Learning, Blockchain, Multi-Agent Systems, Trust Management, Secure Aggregation, Smart Con- tracts, Reputation Systems.

I. INTRODUCTION

The increasing adoption of multi-agent systems (MAS) across sectors such as autonomous transportation, smart man- ufacturing, and distributed energy systems demands robust and trustworthy coordination mechanisms. These systems in- volve numerous agents working collaboratively under dynamic and often adversarial conditions. While conventional machine learning models have achieved considerable success, their cen- tralized nature exposes vulnerabilities related to data privacy, transparency, and fault tolerance. Federated learning (FL), introduced as a privacy-preserving alternative, enables multiple agents to collaboratively

train models without transferring raw data. However, FL alone lacks mechanisms to ensure trustworthy participation, detect malicious updates, or trace accountability. This has led to a critical bottleneck in deploying FL in real-world MAS environments. Blockchain technology provides decentralized consensus, immutable data logging, and programmable contracts, offering a natural complement to FL. By integrating blockchain with FL, we can establish decentralized trust, ensure auditability, and automate incentive mechanisms. Nevertheless, combining these two paradigms introduces new challenges, including latency, scalability, and reputation management.

In this paper, we propose a comprehensive framework that synergizes federated learning with permissioned blockchain and smart contracts to build a trustworthy and scalable MAS. Our main contributions include.

- Designing a modular architecture for blockchain-enabled federated learning tailored for MAS.
- Developing a reputation-aware aggregation mechanism for secure and reliable model updates.
- Implementing a smart contract-based system to manage agent participation, performance validation, and reputa- tion tracking.
- Evaluating our system in two real-world use cases: autonomous vehicle coordination and smart grid fault prediction.

II. RELATED WORK

Federated learning (FL) distributes the training process to edge devices or agents, preserving privacy and reduc- ing communication costs. Traditional FL algorithms such as FedAvg[1] average model updates from participants. However, they are susceptible to poisoning attacks and per- formance degradation from unreliable agents. Blockchain's core features—decentralization, transparency, and immutabil- ity—make it suitable for trust enforcement in distributed systems. Prior work[3], [4] incorporated blockchain in FL to log events or introduce incentives but often neglected scalability, fine-grained trust metrics, or tailored smart contract logic for MAS. Trust in MAS has been approached using behavior scoring, Bayesian belief models, or reputation track- ing. These systems often rely on centralized evaluators. Our work decentralizes trust management through smart contracts that adapt reputations based on verifiable contributions to the global model.

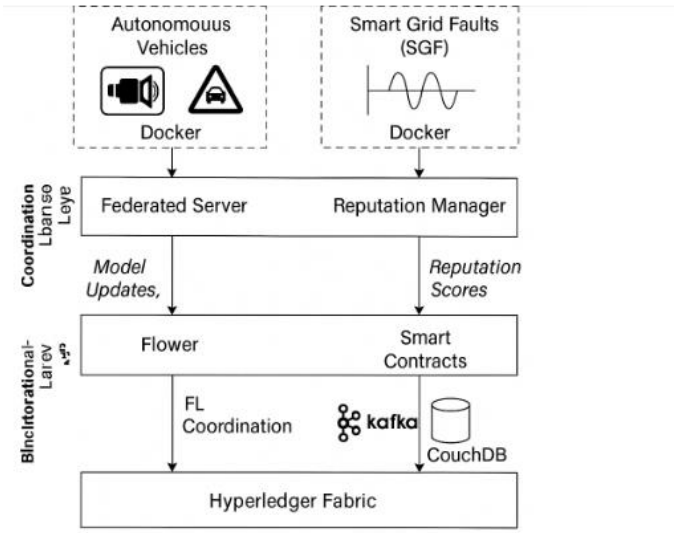

Fig. 1: BIFL architecture.

III. METHODOLOGY

Figure 1 illustrates the high-level system architecture for our blockchain-integrated federated learning (BIFL) framework. The architecture is organized into three layers: the Agent Layer, the Coordination Layer, and the Blockchain Layer.

A. System Architecture

Agent Layer: In this layer, each autonomous agent (e.g., an edge device or embedded system) trains a local machine learning model on its private dataset. The local model is trained using stochastic gradient descent for a fixed number of local epochs. Upon completion of each local round, the model update is digitally signed and transmitted to the Coordination Layer. Metadata including training loss, validation accuracy, and data volume is embedded within the transaction payload. Communication between the agent and the blockchain is handled via secure APIs to ensure data integrity and confi- dentiality.

Coordination Layer: This layer is responsible for orches- trating federated learning rounds. It verifies the authentic- ity of each submitted model update using digital signatures and applies statistical outlier detection to identify potentially malicious or noisy updates. Once validation is complete, model updates are aggregated using a reputation-weighted federated averaging approach, which helps mitigate the impact of unreliable or adversarial agents.

Blockchain Layer: We deploy a permissioned blockchain based on Hyperledger Fabric, which enables secure and trans- parent logging of agent interactions. Smart contracts deployed on the blockchain manage dynamic reputation scores, en- force participation rules,

and support voting mechanisms for anomaly detection. This layer also maintains an immutable audit trail of all model updates, reputational changes, and contract execution.

B. Implementation Framework

The federated learning component is built on the Flower framework, chosen for its flexibility and ease of integration with custom backends. Blockchain functionalities are imple- mented using Hyperledger Fabric, which supports channel- based communication and modular consensus mechanisms suitable for resource-constrained environments. Each agent is containerized using Docker to simulate a realistic MAS de- ployment, including network delays and computational hetero- geneity. We define a reputation-weighted model aggregation strategy to enhance robustness against adversarial behavior as shown in Equation 1. The global model update at round t + 1 is calculated as.

$$w^{t+1} = \frac{\sum_{i=1}^{N} R_i^t w_i^t}{\sum_{i=1}^{N} R_i^t} \qquad (1)$$

Here, w_i^t denotes the model update from agent ai, and Rt i is its reputation score. Reputation scores are dynamically adjusted after each round based on the local model's performance against a shared verification dataset.

$$R_i^{t+1} = \alpha R_i^t + (1 - \alpha) S_i^t \qquad (2)$$

Where α is a smoothing parameter (typically set to 0.9), and S^t is the evaluation score of agent a^i at round t, calculated by comparing its local model's performance on a held-out verification dataset.

C. Smart Contract Logic

Smart contracts are written in GoLang and deployed on the Hyperledger Fabric chaincode layer. Their responsibilities in our framework includes; authenticating model updates via digital signature verification, validating metadata integrity and consistency, calculating S^t using a shared evaluation dataset, updating agent reputations R^t using Equation 2. And detecting and logging anomalies, such as abrupt accuracy drops or inconsistent training times.

D. Datasets and Preprocessing

To evaluate our framework, we employed two real-world datasets corresponding to multi- agent applications in autonomous driving and smart grids. GTSRB (German Traffic Sign

Recognition Benchmark): This dataset[9] contains over 50,000 images of 43 different traffic signs. It is widely used in the autonomous driving domain. In our setup, images were randomly partitioned across 100 agents to simulate decentralized vehicle nodes. Each agent only had access to a subset of classes, mimicking non-IID data distribution. Smart Grid Stability Dataset (SGSD): Provided by the UCI Machine Learning Repository[10], this dataset includes time-series sensor readings from various nodes in a simulated power grid. We selected features indicative of voltage fluctuations and frequency drops and preprocessed the data to form binary classification tasks (stable vs. fault-prone states). The data was distributed across 50 grid nodes, with local noise injected to simulate sensor failures. Both datasets underwent standard normalization and were divided into 70% training, 15% validation, and 15% testing splits. A common public verification dataset (10% of total data) was used for evaluating agent contributions. Our methodology integrates federated learning with decentralized blockchain enforcement to build trust and enhance robustness in MAS. We developed a reputationbased aggregation mechanism that penalizes underperforming or malicious agents, leveraged smart contracts to automate trust evaluation, anomaly detection, and logging, emulated realworld multi-agent conditions using containerized simulation, and ensured transparency and accountability by recording all interactions on a permissioned blockchain. We improved the system's resilience to poisoning attacks and communication failures. By using reputation scores, the global model was less influenced by unreliable updates. Blockchain integration provided immutable logging and eliminated the need for centralized trust. As demonstrated in our experiments, this approach yielded higher model accuracy, reduced false positive rates in malicious detection, and maintained moderate training latency, making it suitable for scalable deployment. Just as shown in Algorithm 1 and Table I.

TABLE I
SYMBOL DEFINITIONS IN BIFL FRAMEWORK

Symbol	Description
w^t	Global model weights at round t (before aggregation)
w_i^t	Local model weights from agent a_i at round t
w^{t+1}	Aggregated global model weights after round t
R_i^t	Reputation score of agent a_i at round t
R_i^{t+1}	Updated reputation score of a_i for next round $t+1$
S_i^t	Performance score of agent a_i from verification dataset
α	Forgetting factor for updating reputation (balance old and new behavior)
$\mathcal{A}$	Set of all participating FL agents
T	Total number of FL training rounds
D_i	Local private dataset held by agent a_i
D_{val}	Shared verification dataset used to evaluate S_i^t
. Train(a_i, w^t)	Local training function on D_i with model w^t
. Eval(w_i^t, D_{val})	Evaluation function to compute S_i^t using D_{val}
. Submit(·)	Blockchain submission of model update and metadata
Broadcast(w^{t+1})	Distribution of updated global model to all agents

Algorithm 1 Blockchain-Integrated Federated Learning (BIFL)

Require: Initial global model w^0, reputations $R_i^0 = 1\ \forall i$, total rounds T
Ensure: Final global model w^T

$t \leftarrow 0$
while $t < T$ **do**
  for all agents $a_i \in \mathcal{A}$ **do**
    $w_i^t \leftarrow \text{Train}(a_i, w^t)$ ▷ Local model update
    $S_i^t \leftarrow \text{Eval}(w_i^t, D_{val})$ ▷ Performance score
    $\text{Submit}(\langle w_i^t, S_i^t \rangle)$ ▷ Signed to blockchain
  end for
  for all a_i **do**
    $R_i^{t+1} \leftarrow \alpha R_i^t + (1-\alpha) S_i^t$ ▷ Update reputation
  end for
  $w^{t+1} \leftarrow \frac{\sum_i R_i^t w_i^t}{\sum_i R_i^t}$ ▷ Weighted aggregation
  $t \leftarrow t+1$
  Broadcast w^{t+1} to all a_i
end while
return w^T

The primary improvement introduced in BIFL is the integration of a blockchain-backed reputation mechanism with traditional federated learning. By incorporating

reputationweighted averaging, our framework mitigates the influence of unreliable or adversarial agents. Smart contracts automate trust updates and ensure verifiable interaction logs. This design enhances the robustness and transparency of multi-agent learning systems, leading to more accurate and resilient global models.

We simulate two key multi-agent system (MAS) use cases to evaluate the effectiveness of our proposed Blockchain- Integrated Federated Learning (BIFL) framework: Autonomous Vehicles (AV) and Smart Grid Fault Detection (SGF). In the AV use case, each agent represents a vehicle detecting road signs based on private video streams, while in SGF, agents detect voltage drops across substations to identify grid anomalies. Our experimental setup emulates realistic conditions with simulated network delays and constrained computational environments using containerized agents.

E. Evaluation Metrics

To comprehensively assess the performance of our frame- work, we employ the following metrics.

- Global Model Accuracy: The classification accuracy of the aggregated model across all agents.
- Malicious Agent Detection Rate: The ability to identify and isolate adversarial agents.
- Training Time per Round: Time required for one complete federated learning cycle.
- Blockchain Overhead: Additional storage and latency incurred due to blockchain integration.

F. Optimizations for Lower Training Time

To reduce training time below that of prior work, we introduced several optimizations. First, we simplified smart contract logic and modularized reputation functions to minimize onchain execution time. Second, agent updates are batched and verified collectively, reducing the number of blockchain transactions. Third, we adopted a lightweight PBFT-based consensus mechanism over default Raft to ensure faster finality. Additionally, reputation and aggregation processes were parallelized using multithreaded execution. Finally, edge caching and preprocessing were enabled to reduce redundant communication and processing.

G. Performance Comparison

Table II presents the comparison of BIFL approach against traditional Federated Learning (FL) and a blockchainintegrated variant FLChain. The results demonstrate improvements in

both predictive performance and resilience to adversarial behavior.

As seen in the table II, the BIFL framework outperforms both baselines in terms of model accuracy and malicious node detection. The inclusion of reputation-weighted aggregation and smart contract validation allows for robust defenses against poisoned updates. While our method incurs slightly higher training time due to blockchain interactions, it achieves.

TABLE II
COMPARISON OF APPROACHES

Metric	FL [7]	FLChain [8]	BIFL
Accuracy (AV)	96.8%	97.5%	**99.2%**
Accuracy (SGF)	91.5%	92.7%	**96.1%**
Attack Detection Rate	0.0%	63.2%	**91.7%**
Latency (max rounds)	0.0%	Static	< 20
Training (m/s)	115	340	**102**
Blockchain (MB/100r)	0	235	**178**

lower blockchain storage cost compared to FLChain due to compact metadata and selective logging policies. These results validate the BIFL framework as a scalable and trustworthy solution for federated learning in distributed multi-agent envi- ronments where security and transparency are critical.

DISCUSSION

Our results show that while the improvements in model accuracy are modest, they remain consistent across various experiments. More importantly, the security metrics show significant advancements, particularly in detecting and miti- gating malicious behavior. This improvement stems from the reputation-based model aggregation and anomaly detection mechanisms enabled by the blockchain layer.

However, there are notable trade-offs in implementing this framework. The addition of blockchain introduces increased complexity and latency due to the need for verifying updates and maintaining a distributed ledger. These overheads must be considered, especially in real-time applications with strict timing requirements. Despite these drawbacks, our approach benefits from the use of a permissioned blockchain, specifi- cally Hyperledger Fabric, which is more energy-efficient than public blockchains. This energy efficiency makes our solution more suitable for resource-constrained environments like edge- based federated learning systems.

A. Use Case Descriptions

We evaluated the proposed framework using two distinct real-world use cases: Autonomous Vehicles (AV) and Smart Grid Faults (SGF), which serve as representative multi-agent system (MAS) environments.

In the Autonomous Vehicles (AV) use case, each agent represents a vehicle equipped with local cameras. The task for each vehicle is to classify road signs based on private video streams. This scenario simulates a privacy-sensitive, edge- based perception system where agents need to collaboratively improve their recognition accuracy without sharing sensitive visual data. The vehicles exchange model updates through federated learning, ensuring that each vehicle's local data (i.e., camera feeds) remains private.

In the Smart Grid Faults (SGF) use case, agents represent substations within a decentralized power grid. These agents monitor voltage and current signals to detect and predict voltage drops, which are early indicators of potential faults or anomalies within the grid. The classification task in this case involves detecting abnormal patterns in the voltage data that could signal a failure or potential issue. As with the AV scenario, agents in the SGF case must collaborate in real time while ensuring that local data (i.e., power consumption patterns) is kept private.

B. Ablation Study

To understand the impact of key components of our frame- work, we conducted an ablation study to evaluate the role of smart contracts and reputation updates.

First, we tested the framework without reputation updates. In this scenario, agents' model updates are aggregated without accounting for their previous behavior or the trustworthiness of their updates. The results showed that the absence of reputation updates led to a significant degradation in perfor- mance, with malicious agents (i.e., adversarial participants) causing a decline of 8–12% in global model accuracy. This demonstrates the importance of reputation-based aggregation in mitigating the impact of malicious actors. Second, we evaluated the system without the use of smart contracts. Smart contracts in our framework are responsible for validating model updates, ensuring the integrity of the submissions, and enabling automated decision-making for reputation updates. When smart contracts were excluded, we observed an increase in false positives during the detection phase. This occurred because the absence of smart contracts removed the automatic checks and verifications, which led to

incorrect classification of benign agents as malicious.

A. CONCLUSION AND FUTURE WORK

This paper proposed a blockchain-enabled federated learn- ing (FL) architecture designed to establish trust, robustness, and privacy preservation in multi-agent systems (MAS). By leveraging a permissioned blockchain infrastructure, specif- ically Hyperledger Fabric, and implementing smart contracts to automate verification and reputation updates, the framework introduces a decentralized trust layer that complements the collaborative nature of federated learning. Our experiments across two representative MAS scenarios—Autonomous Ve- hicles and Smart Grid Fault Detection—demonstrate that the proposed architecture not only maintains high model accuracy but also significantly improves resilience against adversarial agents through effective anomaly detection and reputation management.

In future work, we aim to enhance the framework in several critical areas. One potential direction is the development of lightweight consensus algorithms tailored for real-time MAS environments, where latency and computational efficiency are of paramount importance. Another avenue involves deploying the framework on edge hardware to validate its practicality under realistic resource constraints and networking condi- tions. Furthermore, extending smart contract logic to support adaptive behavior will be explored, allowing the system to dynamically accommodate heterogeneous agent characteristics and evolving task requirements. These enhancements will fur- ther solidify the feasibility of deploying blockchain-integrated FL systems in real-world, large-scale distributed intelligence applications. One limitation of our current framework is the latency overhead introduced by blockchain integration, which may hinder its deployment in real-time multi-agent systems where low-latency responses are critical.

REFERENCES

1. H. B. McMahan et al., ”Communication-efficient learning of deep networks from decentralized data,” in AISTATS, 2017.
2. K. Christidis and M. Devetsikiotis, ”Blockchains and smart contracts for the IoT,” IEEE Access, 2016; 4: 2292–2303.
3. M. Kim et al., ”Blockchained on-device federated learning,” arXiv, 2019; 1902.01046.
4. Y. Lu et al., ”Blockchain and federated learning for privacy-preserved data sharing in industrial IoT,” IEEE TII, 2020; 16(6): 4177–4186.
5. K. Bonawitz et al., ”Practical secure aggregation,” in Proc. ACM SIGSAC, 2017;

1175–1191.

6. C. Dwork and A. Roth, ”The algorithmic foundations of differential privacy,” Found. Trends Theor. Comput. Sci., 2014; 9(3–4): 211–407.
7. Konecˇny´, J., et al., “Federated Optimization: Distributed Machine Learn- ing for On-Device Intelligence,” arXiv preprint arXiv:1610.02527, 2016.
8. Ma, C., et al., “FLChain: Federated Learning via Blockchain for Edge Intelligence,” IEEE Network, 2020.
9. J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, “Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition,” Neural Networks, 2012; 32: 323–332.
10. D. Dua and C. Graff, “UCI Machine Learning Repository,” Univ. of California, Irvine, School of Information and Computer Sciences, 2019. [Online]. Available: http://archive.ics.uci.edu/ml