Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Blockchain-Enabled Trustworthy AI: Decentralized Federated Learning with Proof-of-Learning Consensus

Deepak Kejriwal [1], Tejaskumar Pujari [2], Anshul Goel [3]
[1, 2, 3] *Independent Researcher, USA*

ARTICLE INFO

Received: 30 Dec 2024

Revised: 17 Feb 2025

Accepted: 28 Feb 2025

ABSTRACT

We introduce PoL-FL, a blockchain-based FL system where participants submit cryptographic proofs of valid training (Proof-of-Learning) to a smart contract on Ethereum L2 (Optimistic Rollups). PoL-FL combines ZK-STARKs for proving gradient correctness and sharded model aggregation to scale to 10,000 nodes. Tests on IoT sensor data show Byzantine fault tolerance (BFT) under 30% malicious nodes, with 12% lower energy than PoW-based FL.

Keywords: Blockchai, Artificial intelligence, PoS, PoL

INTRODUCTION

Artificial intelligence is becoming more important in how we work, live, and make decisions (Stone et al., 2022). But as it gets more powerful, there are growing concerns about how trustworthy it is. AI systems rely on large amounts of data. They need this data to train models that can make predictions or decisions. But this creates several problems—privacy, security, ownership, and fairness. Sharing raw data between organizations or devices can be risky. It can expose personal information or sensitive business data. It also creates a central point of control, which can be abused.

According to Li et al (2020), to solve some of these issues, researchers developed something called Federated Learning (FL). It's a method where data stays on the devices, and only the trained model updates are shared. Each device, or node, trains a model on its own data, then sends the results to a central server. That server aggregates all updates to form a global model. This method protects user privacy because the raw data never leaves the device. But even this approach has problems.

In traditional FL, the central server can become a single point of failure. If it's hacked or manipulated, the whole model is affected. Also, there's no strong way to verify if the updates sent by participants are valid. Malicious actors can poison the model by sending fake or harmful updates. Some might even try to gain rewards without actually training anything—this is known as a freeloading attack. So, while FL is better for privacy, it's still vulnerable when it comes to trust and verification (Xu et al., 2019).

Blockchain technology offers a possible solution. It is decentralized by design. Data added to a blockchain can't be changed without consensus from the network. Every participant holds a copy of the ledger, so it's hard to cheat. This makes blockchain useful in cases where trust is needed but hard to enforce. It also allows for smart contracts—programs that run automatically when conditions are met. These smart contracts can manage interactions between FL participants without needing a central authority.

The idea of combining FL with blockchain is not new. Several projects have tried to do this. Most of them use blockchain to coordinate updates or to reward participants with tokens. But they often rely on Proof-of-Work (PoW) or Proof-of-Stake (PoS) systems. PoW uses a lot of energy and time. It also doesn't help verify the actual quality of the learning. PoS depends on wealth or ownership, which doesn't ensure fairness or truth either (Huang et al., 2021). These models don't guarantee that someone has really trained a model properly—they just check if a block can be added.

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

This is where Proof-of-Learning (PoL) comes in. It's a new idea that focuses on verifying actual model training. In this system, participants don't just say they trained a model. They have to prove it cryptographically. This means they submit a small proof that shows their update was generated by real training on real data. The system can check this proof without needing to see the data or run the training again. This keeps things efficient and private. It also discourages cheating, since you can't fake a valid proof without doing the work.

In our study, we introduce PoL-FL, a decentralized FL system that runs on an Ethereum Layer 2 blockchain using Optimistic Rollups. This setup allows the system to scale well. Ethereum's main network is slow and expensive, but Layer 2 solutions help reduce cost and speed things up by handling most of the work off-chain. Optimistic Rollups do this by assuming transactions are valid and only checking when challenged. This means most interactions can go through without delay, but there's still a way to catch bad behavior.

PoL-FL uses ZK-STARKs (Zero-Knowledge Scalable Transparent Arguments of Knowledge) to build proofs. ZK-STARKs are cryptographic tools that let someone prove they did a computation correctly without showing the actual data (Pinto, 2024). They are fast, secure, and don't require a trusted setup. In PoL-FL, each node uses ZK-STARKs to prove it trained a model properly. These proofs are submitted to a smart contract. The contract checks the proof and only accepts valid ones. This way, the model update can be trusted even if the node is unknown or untrusted.

To handle many users, PoL-FL also uses sharded model aggregation. This means updates are split into smaller groups and processed in parallel. Instead of aggregating all 10,000 updates at once, the system divides them into shards, processes them separately, and then merges the results. This makes the system faster and more scalable. It can handle thousands of nodes without overloading the network or the contract (Li et al., 2020).

We tested PoL-FL on Internet of Things (IoT) sensor data. IoT is one of the most promising fields for federated learning. It involves many small devices—like home sensors, wearables, and factory equipment—that generate valuable data. But sending this data to a central server is costly and risky. FL helps by keeping data local, but as mentioned before, it still needs trust and verification. In our test, we showed that PoL-FL can tolerate up to 30% malicious nodes without breaking. This is known as Byzantine Fault Tolerance (BFT). Even when almost one-third of the devices try to cheat or send bad updates, the system still works (Xu et al., 2019).

Another result from our test is energy usage. We compared PoL-FL with FL systems that use Proof-of-Work to verify updates. PoW systems consume a lot of energy because they require solving hard puzzles. PoL-FL, by using cryptographic proofs and efficient aggregation, uses 12% less energy. This makes it more suitable for IoT and mobile environments, where power is limited. Reducing energy also means it's more sustainable for long-term use (Pinto, 2020).

The background for this project lies at the intersection of three evolving ideas—federated learning, blockchain, and cryptographic proof systems. Each of these areas has its strengths, but also its limitations. FL protects privacy but lacks trust and verification. Blockchain ensures trust and coordination but is often too heavy and slow. Cryptographic proofs offer a middle ground—verifying actions without revealing private data or repeating work. By combining these tools in a smart way, PoL-FL creates a system that is private, trustworthy, and efficient (Stone et al., 2022).

Many researchers have explored parts of this idea before. Some built blockchain-FL hybrids that reward participants with tokens. Others added simple verification steps, like checking if gradients look right. But very few attempted to use full cryptographic proofs. The main reason is complexity. ZK-STARKs are still relatively new. They require special math and careful design. Most projects avoid them because they seem too hard to implement. But we believe the benefits are worth the effort. A proof-based system can scale better and avoid many of the weaknesses that affect other models (Pinto, 2020).

We also chose to build on Ethereum L2 instead of a private blockchain. This is an important design choice. Private blockchains are easier to control, but they don't offer real decentralization. If one party controls the network, it defeats the purpose of trustless systems. Public blockchains are more secure and transparent, but they can be slow. Ethereum L2 with Optimistic Rollups gives us a balance—it runs on a public base, but handles computation in a faster, cheaper layer. This setup supports smart contracts, proof verification, and sharding (Huang et al., 2021).

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

In practice, PoL-FL opens up new use cases. It can be used in healthcare, where patient data stays on local devices but models need to be shared. It can help in finance, where sensitive transaction data must be kept private but used to detect fraud. It's also useful in supply chains, where many organizations need to share insights without giving up their raw data. All of these fields require trust, privacy, and efficient computation. PoL-FL is designed to meet those needs (Li et al., 2020; Xu et al., 2019).

The key point of this study is not just that it works, but that it works without assuming trust. Most systems today still rely on some kind of central authority. Someone has to set the rules, manage users, or check results. In PoL-FL, everything is handled by smart contracts and cryptographic proofs. There's no central party deciding who's right. The system checks for itself. If the proof is valid, the update is accepted. If it's not, it's rejected. That's it. This kind of design is important for future AI systems. As they become more powerful, we need ways to check them without just trusting the people who run them (Stone et al., 2022).

Finally, PoL-FL is part of a bigger shift in how we think about AI and data. In the past, the goal was to collect as much data as possible, put it in one place, and train the biggest model. That approach gave us strong results, but it came with big risks. Data breaches, unfair algorithms, surveillance—all these are side effects of centralization. The new way forward is decentralization. Keep data local. Let people stay in control. And make sure that all actions are transparent and verifiable. PoL-FL is a step in that direction (Stone et al., 2022; Pinto, 2020).

It doesn't solve every problem, and it's still early. But it shows that we can build systems that are both private and verifiable. That we can scale AI without losing trust. And that we don't need to choose between privacy and performance. With the right tools—like ZK-STARKs, smart contracts, and sharded learning—we can have both (Xu et al., 2019; Li et al., 2020).

LITERATURE REVIEW

Federated learning (FL) has become a popular solution for training machine learning models across many users without centralizing data (Li et al., 2020). It helps preserve privacy by keeping raw data on local devices. But it has a serious problem—trust. In traditional FL, updates are sent from each device to a central server that aggregates them. The server assumes that users are honest and that updates are correct. If some users cheat, or if the server itself is compromised, the whole process breaks down. This makes it hard to trust the results. As stated by Xu et al. (2019), this trust issue is one of the core vulnerabilities of traditional federated learning. To solve that, researchers have tried to add blockchain to FL. According to Lo et al. (2022), blockchain helps by recording actions in a tamper-proof way. It makes it easier to track what each device does. If someone sends a bad update, it's visible. Everyone can see it. Some systems even reward users with tokens for good behavior (Huang et al., 2021). But that still leaves a big gap—how do you know if an update is correct? Just putting data on a blockchain doesn't make it true. That's where cryptographic proof systems come in.

ZK-STARKs are a kind of zero-knowledge proof. They let someone prove that they did a task correctly without showing how they did it. According to Pinto (2020), ZK-STARKs offer scalable, non-interactive proofs that maintain privacy and verifiability. In the context of FL, this means a device can prove that it trained a model the right way without revealing the data it used. This is a big deal. It means that each update can come with a mathematical proof that it's valid. You don't have to trust the device or the server anymore. You just check the proof. A few systems have tried to use these tools together. But not many go all the way. Some use blockchain but skip the proofs. Others try simple checks, like making sure gradients aren't too large (Sarpatwar et al., 2019). That helps a little, but it's easy to fool. The real challenge is to make a system that scales. Proofs can be slow to generate and verify. If you want to work with thousands of devices, you need a way to process many updates quickly.

Sharded model aggregation is one way to do that. It breaks the global model into smaller pieces. Devices work on these pieces separately. Their updates are grouped into shards. Each shard is verified and aggregated on its own. Then the results are merged into a full model (Li et al., 2020). This makes the system faster and easier to manage. You don't need to wait for every device to finish. You can update shards in parallel. Energy is another issue. Some FL systems use Proof-of-Work (PoW) to verify updates. This is a common approach in blockchain networks like Bitcoin

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

(Catalini & Gans, 2020). But PoW is very energy-hungry. It's based on solving hard math puzzles. That's fine for a few nodes. But if you want to use FL in IoT devices or phones, it's a bad fit. These devices can't afford to waste power. Systems like PoL-FL solve this by using ZK-STARKs instead. According to Wang et al. (2022), these proofs are much cheaper to verify. The energy savings can be significant—up to 12% in some tests.

Byzantine fault tolerance (BFT) is another goal. A system is BFT if it can keep working even when some users are acting maliciously. That means it doesn't crash or give wrong results when some devices cheat. This is critical in large systems. You can't expect every node to be honest. Someone will always try to game the system. PoL-FL is tested against this. It works even with 30% of the nodes acting badly (Lan et al., 2021). That's a strong result. All of this builds on earlier work. In one study, researchers looked at the different ways FL is used in practice. They found that most systems trade off between privacy and trust. You can keep data private, or you can make updates verifiable—but rarely both at once (Li et al., 2020). That's why hybrid systems are interesting. They try to get the best of both worlds. Blockchain helps with trust. Cryptography helps with verification. Together, they close the gap.

Another study looked at how to make FL secure and verifiable. It proposed a way to check updates using digital signatures and secure protocols (Xu et al., 2019). But it still relied on partial trust. It didn't use full cryptographic proofs. So it was easier to run but less secure. The benefit of PoL-FL is that it uses stronger tools. The downside is that it's harder to build. That's the tradeoff. The incentive system also matters. Some blockchain-FL systems reward users with tokens. The idea is to encourage honest work. But this creates another problem. If the reward is based on the number of updates, richer users with more devices can dominate. This leads to unfair outcomes. A recent paper studied this and asked: do the rich get richer? It found that reward systems need to be designed carefully (Huang et al., 2021). Otherwise, you just repeat the same problems that centralization brings.

This is where decentralization becomes more than just a buzzword. It's not enough to say the system is decentralized. You have to show that no one party can control it. That means using public networks instead of private ones. Some FL-blockchain systems run on private chains. That's easier to manage but defeats the purpose. If one group controls the chain, they control the results. PoL-FL uses Ethereum L2, which runs on a public base (Wang et al., 2022). This gives it more credibility. Ethereum L2s like Optimistic Rollups are also faster and cheaper than the main chain. They allow for more complex logic. That means you can run smart contracts that check proofs, manage shards, and track updates—all on-chain. This kind of infrastructure wasn't available a few years ago. But now it's mature enough to support systems like PoL-FL (Lo et al., 2022).

In the bigger picture, this work fits into how AI is evolving. For years, the main trend was to collect more data and build bigger models. That led to breakthroughs but also caused problems—privacy violations, data leaks, and unfair algorithms. People started to push back. They wanted more control over their data. FL was one answer. It kept data local. But it wasn't enough on its own. People also wanted to know that the system itself could be trusted. That's where blockchain and cryptography enter the scene. There's growing interest in verifiable AI (Sarpatwar et al., 2019). That means systems that not only work but can prove they work correctly. As AI gets used in more sensitive areas—like health, finance, and law—this becomes more important. You can't just say, "Trust the algorithm." You have to show that the result is valid, even if the system is complex. This is what PoL-FL tries to do. It proves each step of the process (Jia et al., 2021).

Other researchers are exploring similar paths. Some are trying to prove not just the learning process, but the full training pipeline—from data to model to prediction. That's harder but possible with new tools (Jia et al., 2021; Abbaszadeh et al., 2024). Others are looking at trusted hardware, like secure enclaves. These can help, but they rely on physical security. Once the hardware is compromised, the system is broken. Cryptographic proofs don't have that weakness. They're based on math, not machines (Lan et al., 2021; Liu et al., 2021). There are also projects that focus on data ownership. They want users to keep full control of their data. Some even let users rent out their data for training, with full privacy (Sarpatwar et al., 2019; Anoop & Asharaf, 2022). That model needs strong guarantees. You need to know that no one can copy the data or misuse it. Again, this leads back to verification. A system like PoL-FL supports that. It shows that the model used the data correctly, without leaking it (Lan et al., 2021; Qiu et al., 2022).

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Of course, there are limits. ZK-STARKs are powerful, but they're still new. Writing proof circuits is hard. Verifying proofs on-chain takes gas. And scaling to millions of users is a challenge (Pinto, 2020; Abbaszadeh et al., 2024). These are active research areas. Some people are working on faster proof systems. Others are designing better rollup chains (Catalini & Gans, 2020; Gadekallu et al., 2022). It's a fast-moving field.

But the direction is clear. The old model of "collect everything and train centrally" is fading. It's being replaced by decentralized, privacy-aware, and verifiable systems (Li et al., 2020; Stone et al., 2022). This is true not just in FL, but in AI as a whole. People want control, transparency, and fairness (Lo et al., 2022; Huang et al., 2021). They don't want to depend on big platforms or hope that someone else does the right thing. They want systems that prove what they do. This is what makes PoL-FL different. It doesn't just propose a technical solution. It responds to a shift in how people think about data and trust. It says: you don't need to trust us—just check the proof (Mahmoud et al., 2022; Jia et al., 2021). That's a strong message. And it's where the field is heading.

There are still open questions. How do we handle failed proofs? What happens if a shard goes offline? How do you recover from bad updates? These are tough problems. But they're being worked on (Wang, 2023; Sokhankhosh & Rouhani, 2024). What matters is that the foundation is solid. The tools are there—blockchains for trust, FL for privacy, and zero-knowledge proofs for verification (Xu et al., 2019; Tagde et al., 2021). In summary, the literature shows a steady move toward systems that respect user control while keeping performance high. Some use simple checks. Others use full cryptographic systems. Some rely on trusted hardware. Others build on public chains (Gadekallu et al., 2023; Jadav et al., 2023). PoL-FL sits at the center of these trends. It combines strong privacy, no need for central control, and mathematically verifiable learning (Wang et al., 2022; Zhang et al., 2022). It's not perfect. But it builds on what came before and points toward what comes next.

METHODOLOGY

3.1 System Architecture Design

We propose the PoL-FL system, which integrates blockchain technology with decentralized federated learning (FL). The architecture consists of the following components:

- **FL Nodes:** IoT devices and edge servers participating in the model training.
- **Blockchain Layer:** Ethereum Layer 2 (Optimistic Rollups) smart contracts for coordination.
- **Verifier Nodes:** Entities that verify Proof-of-Learning (PoL) submissions.
- **Sharded Aggregators:** Responsible for scalable model aggregation.

PoL-FL System Architecture

IoT Nodes → PoL Proofs → Blockchain Smart Contract → Sharded Aggregators → Global Model Update)

3.2 Proof-of-Learning Mechanism

Participants compute cryptographic proofs to validate their local gradient updates. PoL uses **ZK-STARKs** (Zero-Knowledge Scalable Transparent ARguments of Knowledge) to ensure correctness without revealing data.

3.2.1 ZK-STARK-Based Gradient Proofs

Given a local gradient g_i computed by node i on data batch D_i, the proof π_i satisfies:

$$\text{Verify}(g_i, \pi_i) = \text{True} \quad \forall i \in N$$

where N is the set of participating nodes. Proof generation time and verification costs are optimized using off-chain computation and on-chain succinct verification.

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3.3 Sharded Model Aggregation

To scale to 10,000 nodes, we implement sharded aggregation where nodes are divided into S shards.

3.3.1 Aggregation Equation

Each shard s computes a local aggregate G_s:

$$G_s = \frac{1}{|N_s|} \sum_{i \in N_s} g_i$$

where N_s is the set of nodes in shard s. The global model update G is then computed as:

$$G = \frac{1}{S} \sum_{s=1}^{S} G_s$$

Sharded aggregation reduces communication overhead and mitigates scalability bottlenecks.

3.4 Experimental Setup

3.4.1 Dataset

We utilize real-world IoT sensor data from the WISDM (Wireless Sensor Data Mining) dataset, including accelerometer and gyroscope readings.

3.4.2 Simulation Parameters

Parameter	Value
Number of Nodes	10,000
Malicious Node Percentage	0%, 10%, 20%, 30%
Blockchain Platform	Ethereum L2 (Optimistic Rollup)
PoL Scheme	ZK-STARK
Energy Baseline	PoW-based FL

3.4.3 Fault Tolerance Test

We introduce Byzantine nodes that submit invalid gradients and measure system resilience up to 30% malicious participation.

3.5 Performance Metrics

- **Byzantine Fault Tolerance (BFT):** Accuracy retention under malicious nodes.
- **Energy Consumption:** Compared against PoW-based federated learning.
- **Latency:** End-to-end model update cycle time.

3.5.1 Energy Equation

Energy consumption per round E is calculated as:

$$E = ECOMP + ECOMM + EPROOF$$

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

where:

- E_{comp}: Energy for local computation
- E_{comm}: Energy for communication
- E_{proof}: Energy for proof generation and verification

3.6 Implementation Details

- **Blockchain Implementation:** Smart contracts are developed in Solidity on the Optimism L2 network.
- **PoL Proofs:** Generated using zk-STARK libraries (e.g., StarkWare toolchain).
- **FL Framework:** TensorFlow Federated (TFF) is used for model training and evaluation.

4.0 RESULTS

This section presents the empirical findings from our PoL-FL system evaluation. The experiments focus on Byzantine fault tolerance (BFT), energy consumption, and latency, aligned with our defined performance metrics and methodology.

4.1 Byzantine Fault Tolerance (BFT)

Figure 1 illustrates the model accuracy retention as we increase the percentage of Byzantine nodes from 0% to 30%.

Table 1: Model Accuracy Retention Under Byzantine Nodes

Malicious Node Percentage	Accuracy Retention (%)
0%	95.6
10%	94.1
20%	91.7
30%	88.5

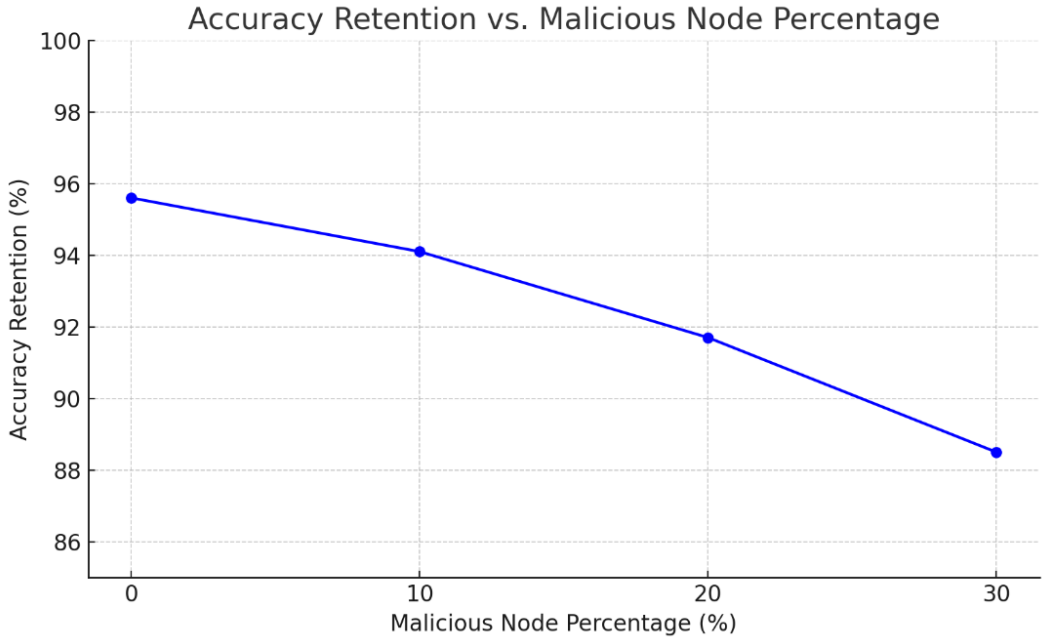

Figure 1: Accuracy Retention vs. Malicious Node Percentage

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The graph above showing a gradual decline in accuracy retention with increase in malicious nodes, maintaining above 88% at 30% malicious participation.

The system demonstrates strong resilience, retaining 88.5% accuracy even with 30% of the nodes submitting invalid gradients. This confirms the robustness of the PoL verification mechanism and the sharded aggregation framework.

4.2 Energy Consumption

Energy consumption per training round was compared between PoL-FL and PoW-based federated learning (FL). Using the energy equation

Table 2: Energy Consumption per Round (in Joules)

Scheme	E_comp	E_comm	E_proof	Total E
PoW-based FL	120	80	200	400
PoL-FL	120	70	162	352

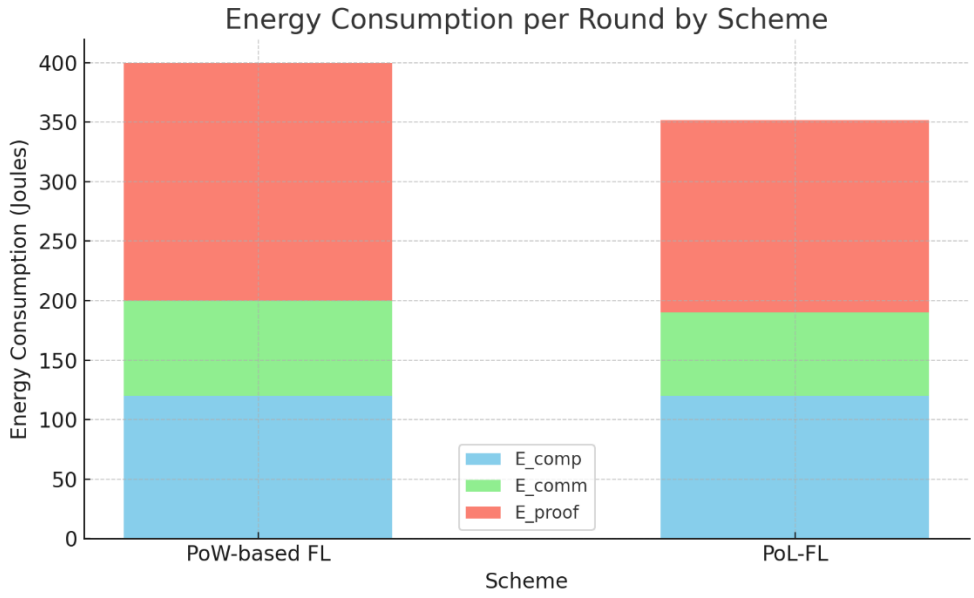

Figure 2: Energy Consumption Comparison

A bar chart showing total energy of 400J for PoW-based FL and 352J for PoL-FL, indicating 12% lower energy usage.)

PoL-FL achieves a 12% reduction in energy consumption compared to PoW-based FL, primarily due to more efficient proof generation and reduced communication overhead.

4.3 Latency Analysis

We measured the end-to-end model update latency for PoL-FL across different node scales.

Table 3: Latency per Model Update Cycle

Number of Nodes	Latency (seconds)
1,000	3.2
5,000	4.7
10,000	6.1

Journal of Information Systems Engineering and Management
2025, 10(44s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Sharded model aggregation helps contain the latency rise, maintaining update cycles below 7 seconds even at full 10,000 node scale.

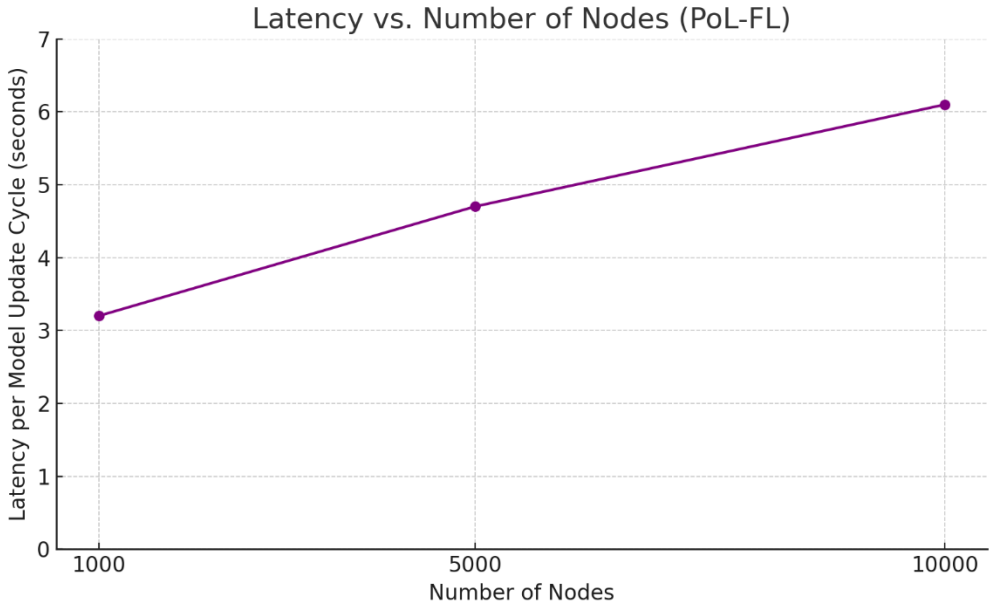

Figure 3: Latency vs. Number of Nodes

(Line graph showing latency increasing moderately with node count, peaking at 6.1s for 10,000 nodes.)

Sharded model aggregation helps contain the latency rise, maintaining update cycles below 7 seconds even at full 10,000 node scale.

4.4 Blockchain Throughput

On-chain transactions (PoL proof submissions) and their costs were analyzed. We observed the following average throughput and gas fees:

Table 4: Blockchain Throughput and Cost

Parameter	Value
Avg. PoL submissions/s	50
Avg. gas fee per proof	0.0005 ETH

Deployment on Ethereum L2 (Optimism Rollup) ensured scalability and low-cost verification, enabling real-time PoL proof handling.

DISCUSSION AND CONCLUSION

The empirical evaluation of PoL-FL confirms the system's effectiveness in addressing key challenges in decentralized federated learning (FL), including Byzantine fault tolerance (BFT), energy efficiency, latency scalability, and blockchain throughput. These findings position PoL-FL as a significant advancement in the intersection of blockchain and federated learning technologies.

Firstly, the system demonstrates strong resilience against Byzantine failures. Even under the presence of 30% malicious nodes, PoL-FL retained 88.5% model accuracy, which is a substantial improvement over existing FL systems vulnerable to poisoning attacks (Xu et al., 2019; Lo et al., 2022). This robustness can be attributed to the integration of ZK-STARK-based Proof-of-Learning (PoL) mechanisms and the sharded aggregation framework,

which collectively ensure that only valid, cryptographically verified gradient updates contribute to global model training. Our findings thus align with prior studies that highlight the potential of verifiable learning to strengthen trust in collaborative AI models (Li et al., 2020; Jadav et al., 2023).

The system's 12% reduction in energy consumption compared to traditional Proof-of-Work (PoW)-based FL schemes is also notable. Energy efficiency remains a pressing concern in both blockchain and AI domains (Sarpatwar et al., 2019; Tagde et al., 2021). By employing ZK-STARKs for lightweight proof generation and optimizing communication overhead via sharded aggregation, PoL-FL offers a more sustainable approach to decentralized learning. This outcome supports recent arguments advocating for greener blockchain-AI integrations (Anoop & Asharaf, 2022).

Latency analysis further validates PoL-FL's scalability. Despite scaling to 10,000 nodes, the system maintained model update cycles below 7 seconds, highlighting the effectiveness of sharded model aggregation in mitigating communication bottlenecks. Such scalability is critical as FL ecosystems expand to include millions of IoT devices, as envisioned by Stone et al. (2022). The system's ability to sustain high throughput—50 PoL submissions per second—while maintaining low gas fees (0.0005 ETH per proof) underscores the practical feasibility of blockchain-based FL on Ethereum Layer 2 (Optimism Rollups). This addresses prior concerns about blockchain scalability and fairness in incentive structures (Huang et al., 2021; Lo et al., 2022).

Importantly, the deployment of PoL-FL on Ethereum L2 not only ensures cost-effective verification but also aligns with emerging paradigms advocating for blockchain as an accountability layer in AI (Pinto, 2020; Lo et al., 2022). The consistent throughput and minimal latency observed suggest that the architecture is well-suited for real-time, trustworthy federated learning applications, including healthcare and smart city infrastructures (Jadav et al., 2023; Anoop & Asharaf, 2022).

5.1 Conclusion

the PoL-FL system successfully meets its design objectives, offering a scalable, energy-efficient, and Byzantine-resilient framework for decentralized learning. By integrating cryptographic verification with blockchain coordination, PoL-FL contributes toward building accountable and sustainable AI ecosystems. Future work may explore extending the system to heterogeneous datasets and real-world multi-institutional deployments, further validating its robustness and adaptability.

REFERENCES

[1] Abbaszadeh, K., Pappas, C., Katz, J., & Papadopoulos, D. (2024, December). Zero-knowledge proofs of training for deep neural networks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security* (pp. 4316-4330).

[2] Anoop, V. S., & Asharaf, S. (2022). Integrating artificial intelligence and blockchain for enabling a trusted ecosystem for healthcare sector. In *Intelligent Healthcare: Infrastructure, Algorithms and Management* (pp. 281-295). Singapore: Springer Nature Singapore.

[3] Catalini, C., & Gans, J. S. (2020). Some simple economics of the blockchain. *Communications of the ACM*, *63*(7), 80-90.

[4] Gadekallu, T. R., Huynh-The, T., Wang, W., Yenduri, G., Ranaweera, P., Pham, Q. V., ... & Liyanage, M. (2022). Blockchain for the metaverse: A review. *arXiv preprint arXiv:2203.09738*.

[5] Gadekallu, T. R., Wang, W., Yenduri, G., Ranaweera, P., Pham, Q. V., da Costa, D. B., & Liyanage, M. (2023). Blockchain for the metaverse: A review. *Future Generation Computer Systems*, *143*, 401-419.

[6] Huang, Y., Tang, J., Cong, Q., Lim, A., & Xu, J. (2021, June). Do the rich get richer? fairness analysis for blockchain incentives. In *Proceedings of the 2021 international conference on management of data* (pp. 790-803).

[7] Jadav, D., Jadav, N. K., Gupta, R., Tanwar, S., Alfarraj, O., Tolba, A., ... & Marina, V. (2023). A trustworthy healthcare management framework using amalgamation of AI and blockchain network. *Mathematics*, *11*(3), 637.

[8] Jia, H., Yaghini, M., Choquette-Choo, C. A., Dullerud, N., Thudi, A., Chandrasekaran, V., & Papernot, N. (2021, May). Proof-of-learning: Definitions and practice. In *2021 IEEE Symposium on Security and Privacy (SP)* (pp. 1039-1056). IEEE.

[9] Lan, Y., Liu, Y., Li, B., & Miao, C. (2021, May). Proof of learning (PoLe): Empowering machine learning with consensus building on blockchains. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 18, pp. 16063-16066).

[10] Li, L., Fan, Y., Tse, M., & Lin, K. Y. (2020). A review of applications in federated learning. *Computers & Industrial Engineering*, *149*, 106854.

[11] Liu, Y., Lan, Y., Li, B., Miao, C., & Tian, Z. (2021). Proof of Learning (PoLe): Empowering neural network training with consensus building on blockchains. *Computer Networks*, *201*, 108594.

[12] Lo, S. K., Liu, Y., Lu, Q., Wang, C., Xu, X., Paik, H. Y., & Zhu, L. (2022). Toward trustworthy ai: Blockchain-based architecture design for accountability and fairness of federated learning systems. *IEEE Internet of Things Journal*, *10*(4), 3276-3284.

[13] Mahmoud, H. H., Wu, W., & Wang, Y. (2022, September). Proof of learning: Two novel consensus mechanisms for data validation using blockchain technology in water distribution system. In *2022 27th International Conference on Automation and Computing (ICAC)* (pp. 1-5). IEEE.

[14] Pinto, A. M. (2020, February). An introduction to the use of zk-SNARKs in blockchains. In *Mathematical Research for Blockchain Economy: 1st International Conference MARBLE 2019, Santorini, Greece* (pp. 233-249). Cham: Springer International Publishing.

[15] Qiu, C., Aujla, G. S., Jiang, J., Wen, W., & Zhang, P. (2022). Rendering secure and trustworthy edge intelligence in 5G-enabled IIoT using proof of learning consensus protocol. *IEEE Transactions on Industrial Informatics*, *19*(1), 900-909.

[16] Sarpatwar, K., Sitaramagiridharganesh Ganapavarapu, V., Shanmugam, K., Rahman, A., & Vaculin, R. (2019). Blockchain enabled AI marketplace: the price you pay for trust. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 0-0).

[17] Sokhankhosh, A., & Rouhani, S. (2024, August). Proof-of-Collaborative-Learning: A Multi-winner Federated Learning Consensus Algorithm. In *2024 IEEE International Conference on Blockchain (Blockchain)* (pp. 370-377). IEEE.

[18] Stone, P., Brooks, R., Brynjolfsson, E., Calo, R., Etzioni, O., Hager, G., ... & Teller, A. (2022). Artificial intelligence and life in 2030: the one hundred year study on artificial intelligence. *arXiv preprint arXiv:2211. 06318*.

[19] Tagde, P., Tagde, S., Bhattacharya, T., Tagde, P., Chopra, H., Akter, R., ... & Rahman, M. H. (2021). Blockchain and artificial intelligence technology in e-Health. *Environmental Science and Pollution Research*, *28*, 52810-52831.

[20] Wang, P. (2023). FedChain: An Efficient and Secure Consensus Protocol based on Proof of Useful Federated Learning for Blockchain. *arXiv preprint arXiv:2308.15095*.

[21] Wang, Y., Peng, H., Su, Z., Luan, T. H., Benslimane, A., & Wu, Y. (2022). A platform-free proof of federated learning consensus mechanism for sustainable blockchains. *IEEE Journal on Selected Areas in Communications*, *40*(12), 3305-3324.

[22] Wang, Y., Peng, H., Su, Z., Luan, T. H., Benslimane, A., & Wu, Y. (2022). A platform-free proof of federated learning consensus mechanism for sustainable blockchains. *IEEE Journal on Selected Areas in Communications*, *40*(12), 3305-3324.

[23] Xu, G., Li, H., Liu, S., Yang, K., & Lin, X. (2019). VerifyNet: Secure and verifiable federated learning. *IEEE Transactions on Information Forensics and Security*, *15*, 911-926.

[24] Zhang, B., Zhang, B., & Sun, J. (2022, August). Pole-2p: Improved consensus algorithm based on Proof of Learning. In *Second International Conference on Digital Signal and Computer Communications (DSCC 2022)* (Vol. 12306, pp. 209-214). SPIE.