UTS UNIVERSITY OF TECHNOLOGY SYDNEY

Utilizing Blockchain in Privacy Protection and Machine Unlearning

by Xuhan Zuo

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

under the supervision of Prof Tianqing Zhu and Prof Shui Yu

University of Technology Sydney
Faculty of Engineering and Information Technology

January 2025

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, Xuhan Zuo, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the *School of Computer Science*, *Faculty of Engineering and IT* at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

SIGNATURE: Production Note: Signature removed prior to publication.

[Xuhan Zuo]

DATE: 31st January, 2025

PLACE: Sydney, Australia

ABSTRACT

The rapid advancements in blockchain technology, machine learning, and the Internet of Things (IoT) have transformed various fields, including intelligent transportation, collaborative learning, and privacy protection. However, the increasing reliance on decentralized data-driven systems has also raised significant challenges related to privacy, security, and trust, particularly in environments where sensitive information must be shared across multiple parties. This thesis investigates the integration of blockchain technology with federated learning and machine unlearning to address these challenges, providing secure and privacy-preserving mechanisms for data collaboration and model management.

The thesis first examines the role of blockchain in intelligent transportation systems, proposing a dynamic location privacy-preserving scheme that leverages blockchain,Äôs immutability and transparency to protect vehicle location data while facilitating secure communication in transportation networks. This solution is analyzed from multiple perspectives, demonstrating its effectiveness in balancing location privacy with system efficiency.

Building on this foundation, the research extends to federated learning and machine unlearning, where blockchain is integrated to enhance security and trust in decentralized model training. A novel framework is introduced for blockchain-enhanced federated learning, enabling multiple organizations to collaborate securely while maintaining data privacy. This framework is further extended to incorporate machine unlearning, allowing for verifiable data removal without compromising model performance or violating privacy regulations.

A key focus of this thesis is on the application of these techniques in large language models (LLMs). A novel framework, Federated TrustChain, is proposed to facilitate secure and auditable LLM training and unlearning within a federated learning setting. By leveraging blockchain, this approach ensures an immutable and verifiable record of model updates and data removal requests, enhancing trust and accountability in distributed LLM training.

Further, this research addresses cross-organizational collaboration in federated learning, where institutions must train shared models without exposing proprietary data. A blockchain-based solution is designed to support secure and efficient data sharing, integrating smart contracts and differential privacy techniques to ensure compliance with data protection regulations while enabling trustworthy federated collaboration.

The proposed frameworks are extensively evaluated through real-world-inspired case

studies, including applications in healthcare and education, demonstrating their effectiveness, scalability, and practicality. The results highlight the improvements in privacy protection, security, and collaborative learning efficiency enabled by the integration of blockchain with federated learning and unlearning.

In conclusion, this thesis makes significant contributions to the field of privacy-preserving and secure collaborative learning, establishing a strong foundation for future research and real-world applications. By bridging blockchain, federated learning, and machine unlearning, this work paves the way for more secure, transparent, and privacy-aware intelligent systems, particularly in domains such as intelligent transportation, healthcare, and cross-organizational AI development.

Dedication

To our younger selves since 2013 . . .
To our future selves in 2024 . . .
To everyone dear to us, I want to say, Minghao, you are my sunshine. . . .

Acknowledgments

A journey of a thousand miles begins with a single step. This thesis began during the pandemic in 2020, amidst my illness. I extend my heartfelt thanks to all who supported me.

First and foremost, I would like to extend my heartfelt thanks to my supervisors, Prof. Tianqing Zhu, Prof. Shui Yu, and Prof. Wanlei Zhou, for their unwavering guidance and support throughout my research. I also express my sincere gratitude to A/Prof. Angela Huo, A/Prof. Bo Liu, and A/Prof. Dayong Ye for their encouragement, assistance, and guidance. Thank you, Prof. Zhu, for being the flame that brings light to my path. However, I must not forget the one who holds the lamp. My supervisors have stood steadfastly in the darkness, guiding me with unwavering support. I am deeply grateful for your mentorship and I am deeply grateful for their contribution to my academic and personal growth. My supervisors will always be my role models in the futurework.

I would like to express my gratitude to Dr. Wang Minghao, my academic endorser. "Whatever our souls are made of, his and mine are the same." We have supported each other for eleven years. During the darkest moments of my life, his support was like a beacon of light illuminating every corner of despair. We have crossed mountains and seas together and look forward to an even brighter future. You have been my light and warmth during those grey days. I am very fortunate to have met such an interesting and lovely soul. I look forward to the unknown scenery ahead, and I am eager to embark on this great adventure of life with you.

I want to thank my parents, my grandparents and my big family. Looking back on my twenty-one years of education, your spiritual support, and broad perspectives have encouraged me to explore the world with confidence. My parents have dedicated their hardworking lives to support my extraordinary youth. In this turbulent world, it is my family that provides me with endless comfort and silent inspiration. No matter where I am, my parents and hometown will always be my deepest concern on the journey ahead.

I want to thank Dr. Hsia, Jason, Jinze, Jingyuan, Kun C., MengQ., Meng R., Renee, Qian, Wanying H, Yue, Dr. Zed, and ZZZ for their help. I am grateful to Dr.CCZ, HX, Kun, Dr. Lefeng, Dr. Mond, and all the bro from the Lab for Cyber Privacy. Meeting you all has given me beautiful memories of eight years in Sydney. Unfortunately, good times are always short, and we are destined to part ways this Sydney winter. I sincerely hope you achieve more glory, and wish we would get sixpence with our moon when meet again.

"My dreams are worth fighting for by myself; my life is worth exploring by myself."

List of Publications

Related to the Thesis :

1. **X. Zuo**, M. Wang, D. Ye, S. Yu, "A Scheme of Dynamic Location Privacy-Preserving with Blockchain in Intelligent Transportation System", Submitted in The 24th International Conference on Algorithms and Architectures for Parallel Processing.

2. **X. Zuo**, M. Wang, T. Zhu, L. Zhang, S. Yu, W. Zhou, "Federated Learning with Blockchain-Enhanced Machine Unlearning: A Trustworthy Approach", Under review in IEEE Transaction on Services Computing.

3. **X. Zuo**, M. Wang, T. Zhu, L. Zhang, D. Ye, S. Yu, W. Zhou, "Federated TrustChain: Blockchain-Enhanced LLM Training and Unlearning", Under review in IEEE Transactions on Dependable and Secure Computing.

4. **X. Zuo**, M. Wang, T. Zhu, L. Zhang, S. Yu, W. Zhou, "Federated Learning with Blockchain for Cross-Organizational Collaboration", Under review in IEEE Internet of Things Journal.

Others :

4. M. Wang, T. Zhu, **X. Zuo**, D. Ye, S. Yu, W. Zhou, "Public and Private Blockchain Infusion: A Novel Approach to Federated Learning", IEEE Internet of Things Journal, 2024.

5. M. Wang, T. Zhu, **X. Zuo**, M. Yang, S. Yu, W. Zhou, "Differentially private crowdsourcing with the public and private blockchain", IEEE Internet of Things Journal, 2023.

6. M. Wang, T. Zhu, **X. Zuo**, D. Ye, S. Yu, W. Zhou, "Blockchain-based Gradient Inversion and Poisoning Defense for Federated Learning", IEEE Internet of Things Journal, 2023.

7. M. Wang, T. Zhu, **X. Zuo**, D. Ye, S. Yu, W. Zhou, "Advising Multi-agent System with Blockchain Network", IEEE Internet of Things Journal, 2023.
8. M. Wang, **X. Zuo**, T. Zhu, "Blockchain in 5G and 6G networks." (2020): 137-173.
9. S. Shen, T. Zhu, D. Ye, M. Wang, **X. Zuo**, A. Zhou, "A novel differentially private advising framework in cloud server environment", Concurrency and Computation: Practice and Experience, 2022, 34(7): e5932.

Table of Contents

List of Figures

Figure **Page**

List of Tables

Table **Page**

CHAPTER 1

INTRODUCTION

1.1 Background

The rapid advancement of technology has transformed various aspects of our lives, from communication and travel to learning and work. In this era of digital transformation, the convergence of cutting-edge technologies such as blockchain, machine learning [30], and the Internet of Things (IoT) has opened new possibilities for enhancing privacy, security, and trust across diverse domains. Integrating blockchain technology with federated learning, privacy protection, and machine unlearning offers novel solutions to the challenges of secure and trustworthy collaborative learning in an increasingly connected world [96].

Blockchain, a decentralized and immutable ledger technology, ensures secure and transparent data management [93]. By enabling tamper-proof record-keeping and secure data sharing among multiple parties, blockchain revolutionizes collaborative environments while addressing critical privacy and trust issues [49]. For example, in intelligent transportation systems, blockchain can protect sensitive information like vehicle location data while enabling efficient and secure communication within the network.

The rise of machine learning, particularly federated learning, has created new opportunities for collaborative model training while preserving data privacy [35]. Federated learning allows multiple parties to jointly develop machine learning models without sharing raw data, thereby mitigating privacy risks [45]. However, the growing concerns about data privacy and the right to be forgotten highlight increasing demand for effective

machine unlearning techniques [7]. The aim of machine unlearning is to remove or reduce the influence of specific data points of trained models, ensuring compliance with privacy regulations and maintaining user trust [52].

The integration of blockchain with machine learning has created a powerful framework for secure and trustworthy collaborative learning. Leveraging the immutability and transparency of blockchain allows for verifiable and auditable records of model updates and data removal processes. This combination of technologies addresses privacy protection and machine unlearning challenges in various domains, from intelligent transportation systems to cross-organizational collaboration.

In natural language processing, large language models (LLMs) have achieved remarkable breakthroughs, enabling advanced language understanding, generation, and translation [48]. However, training LLMs requires vast amounts of data, raising concerns about data privacy and the need for secure collaborative learning. Integrating blockchain with LLM training and unlearning in federated learning settings offers a promising approach to addressing these challenges, creating a secure and transparent framework for collaborative model development.

As the world becomes increasingly connected and data-driven, innovative solutions to ensure privacy, security, and trust are more critical than ever. The convergence of blockchain, machine learning, and privacy protection sets the stage for exploring cutting-edge approaches to address these challenges. This thesis aims to push the boundaries of secure and trustworthy collaborative learning by delving into these technologies and their applications, paving the way for a more privacy-preserving and trustworthy digital future.

1.2 Motivation

The increasing reliance on technology and data-driven approaches in various domains has brought forth significant challenges related to privacy, security, and trust. As the volume of sensitive data generated by IoT devices and shared across networks continues to grow, robust and innovative solutions to protect user privacy and ensure secure collaboration have become more critical than ever.

A primary motivation for this research is to address pressing concerns surrounding data privacy and the right to be forgotten. With stringent privacy regulations like the General Data Protection Regulation (GDPR) [59] and the California Consumer Privacy Act [8], organizations face increased pressure to develop effective mechanisms

for protecting user data and complying with data removal requests. Traditional privacy protection approaches often lack transparency and verifiability, making it difficult to establish trust among participants in collaborative environments.

The rise of federated learning as a privacy-preserving collaborative learning paradigm highlights the need for efficient and trustworthy machine unlearning techniques. Federated learning enables multiple parties to jointly train machine learning models without sharing raw data, mitigating privacy risks. However, the ability to remove the impact of particular data points of trained models is crucial for maintaining user privacy and complying with regulations. Existing machine unlearning techniques often fall short in terms of efficiency, scalability, and verifiability, hindering their practical application in real-world scenarios.

Another significant motivation is the potential of blockchain technology to enhance privacy, security, and trust in collaborative learning environments. The immutability, transparency, and decentralized nature of blockchain make it ideal for creating secure and auditable records of data sharing, model updates, and unlearning processes. Integrating blockchain with federated learning and machine unlearning can develop trustworthy and privacy-preserving collaborative learning frameworks capable of withstanding the increasing complexity and scale of modern systems.

The application of blockchain-based privacy protection and machine unlearning techniques in domains such as intelligent transportation systems and large language model training underscores the importance of this research. Intelligent transportation systems generate vast amounts of sensitive vehicle location data, necessitating robust privacy-preserving mechanisms to protect user privacy while enabling efficient and secure communication. Similarly, training large language models involves processing massive amounts of textual data, raising concerns about data privacy and the need for secure collaborative learning approaches.

Motivated by these challenges and the potential of blockchain technology to address them, this thesis explores innovative solutions at the intersection of blockchain, privacy protection, and machine unlearning. By developing secure, efficient, and trustworthy collaborative learning frameworks that leverage the strengths of blockchain and federated learning, this research seeks to advance privacy-preserving technologies and their practical application in real-world scenarios.

The proposed approaches in this thesis aim to provide tangible solutions to pressing issues of data privacy, trustworthy, and security in federated learning environments. By addressing these challenges head-on and demonstrating the effectiveness and scalability

of blockchain-based privacy protection and machine unlearning techniques, this research aspires to pave the way for a more secure and trustworthy future in collaborative learning and data sharing.

1.3 Research Objectives

This thesis aims to explore innovative solutions at the intersection of blockchain, privacy protection, and machine unlearning. The primary objectives of this research are as follows:

- Enhance privacy protection in intelligent transportation systems using blockchain technology: Develop a novel scheme for dynamic location privacy-preserving. This objective leverages the immutability and transparency of blockchain to create secure and verifiable privacy-preserving mechanisms. It aims to address the challenges of vehicle location privacy while enabling efficient and secure communication within the transportation network.

- Integrate blockchain with machine unlearning in federated learning environments: Develop a trustworthy approach for verifiable and efficient data removal. This objective seeks to enhance the efficiency, scalability, and verifiability of machine unlearning techniques by combining them with blockchain technology. It aims to create a secure and auditable record of data removal processes, ensuring privacy-preserving and trustworthy collaborative learning.

- Enhance large language model (LLM) training and unlearning in federated learning settings using blockchain: Develop a secure and auditable framework for collaborative model development. This objective addresses the challenges of data privacy and secure collaboration in LLM training by leveraging blockchain to create a transparent and verifiable framework for model updates and data removal. It aims to ensure efficient and verifiable unlearning mechanisms while maintaining data privacy.

- Enable secure and efficient cross-organizational collaboration in federated learning using blockchain: Develop a decentralized and trustworthy framework for collaborative learning. This objective focuses on ensuring data confidentiality, integrity, and privacy across multiple organizations. It aims to facilitate secure data sharing and model updates among participating organizations.

- Evaluate the proposed approaches through extensive experiments and case studies: Validate the practical applicability and performance of the proposed blockchain-based privacy protection and machine unlearning techniques. This objective involves demonstrating their effectiveness, scalability, and robustness in various domains and real-world scenarios. It aims to provide insights into the strengths and limitations of the proposed frameworks and techniques, guiding future research and development efforts.

By addressing these research objectives, this thesis aims to contribute significantly to the advancement of privacy-preserving technologies, machine unlearning, and secure collaborative learning frameworks. The proposed approaches and solutions are expected to have broad implications for various domains, including intelligent transportation systems, large language model training, and cross-organizational collaboration.

1.4 Major Contributions of This Thesis

This thesis makes several significant contributions to the fields of blockchain technology, privacy protection, and machine unlearning. The primary contributions of this research are as follows:

- A scheme of dynamic location privacy-preserving with blockchain in intelligent transportation system: This contribution addresses the challenges of vehicle location privacy by proposing a secure and verifiable privacy-preserving mechanism that leverages the immutability and transparency of blockchain. The proposed scheme integrates blockchain technology with privacy-preserving techniques, such as differential privacy, to ensure the protection of sensitive location data while enabling efficient and secure communication within the transportation network. This scheme provides a practical solution for maintaining user privacy in intelligent transportation systems, offering enhanced security and verifiability compared to traditional approaches.

- Federated Learning with Blockchain-Enhanced Machine Unlearning: A Trustworthy Approach: This contribution enhances the efficiency, scalability, and verifiability of machine unlearning techniques by combining them with blockchain technology. The proposed approach leverages the immutable and transparent nature of blockchain to create a secure and auditable record of data removal processes,

ensuring verifiable and efficient data removal in federated learning settings. By integrating blockchain with machine unlearning, this approach establishes a trustworthy framework for privacy-preserving and collaborative learning, enabling participants to maintain control over their data while benefiting from the collective knowledge of the federated learning system.

- Federated TrustChain: Blockchain-Enhanced LLM Training and Unlearning: This contribution addresses the challenges of data privacy and secure collaboration in LLM training by leveraging blockchain to create a transparent and verifiable framework for model updates and data removal. Federated TrustChain integrates blockchain technology with LLM training and unlearning processes, ensuring secure and auditable collaborative model development while maintaining data privacy. The framework enables efficient and verifiable unlearning mechanisms, allowing participants to remove specific data points or model contributions without compromising the overall performance of the LLM.

- Federated Learning with Blockchain for Cross-Organizational Collaboration: This contribution proposes a decentralized and trustworthy framework for collaborative learning that ensures data confidentiality, integrity, and privacy across multiple organizations. The proposed solution leverages the security and transparency features of blockchain to facilitate secure data sharing and model updates among participating organizations while maintaining the privacy of sensitive information. The blockchain-based framework enables efficient and verifiable cross-organizational collaboration, allowing organizations to jointly train machine learning models without compromising data privacy or trust.

By addressing these contributions, this thesis aims to push the boundaries of secure and trustworthy collaborative learning, paving the way for a more privacy-preserving and trustworthy digital future.

1.5 Thesis Organisation

The remaining of this thesis is organized as below:

- Chapter 2 provides an overview of the preliminary concepts and related work in the areas of blockchain, privacy protection, federated learning, and machine unlearning.

- Chapter 3 presents a novel scheme for dynamic location privacy-preserving using blockchain in intelligent transportation systems, addressing the challenges of vehicle location privacy.
- Chapter 4 introduces a trustworthy approach for integrating blockchain with machine unlearning in federated learning environments, ensuring verifiable and efficient data removal.
- Chapter 5 proposes Federated TrustChain, a blockchain-enhanced framework for LLM training and unlearning in federated learning, enabling secure and auditable collaborative model development.
- Chapter 6 presents a blockchain-based solution for enabling secure and efficient cross-organizational collaboration in federated learning, addressing the challenges of data privacy and trust.
- Chapter 7 concludes the thesis by summarizing the key findings, discussing the implications of the research, and outlining potential directions for future work.

CHAPTER 2

PRELIMINARY AND RELATED WORK

2.1 Preliminary

2.1.1 Blockchain

Blockchain technology is a decentralized ledger system that offers a secure and transparent method for recording transactions across multiple computers [73]. Its foundation relies on cryptography principles, ensuring that each entry in the ledger is immutable and verifiable [65]. While this technology is the backbone of cryptocurrencies like Bitcoin and Ethereum, it also extends its applications to secure transactional data in various sectors, including supply chain management, healthcare, and, as explored in this paper, federated learning environments.

A blockchain comprises a sequence of blocks, each containing a list of transactions. These blocks are interconnected through a cryptographic hash, linking each block to its predecessor and forming a chain. This structure is mathematically represented as $B_1 \rightarrow B_2 \rightarrow \ldots \rightarrow B_n$, where B_i symbolizes the i-th block in the chain, and n represents the total number of blocks. The integrity of the chain is preserved by consensus algorithms, which ensure that all instances of the distributed ledger are synchronized and in agreement on the transaction sequence.

2.1.2 Differential Privacy

Differential Privacy (DP) is a formal mathematical framework designed to protect individual data while enabling statistical analysis [14]. It ensures that the presence or absence of a single data point does not significantly affect the output of an algorithm, providing robust privacy guarantees.

2.1.2.1 Definition of Differential Privacy

A randomized algorithm $\mathscr{M}$ satisfies (ϵ, δ)-differential privacy if for any two adjacent datasets D and D', differing by at most one record, and for all subsets of outputs $S \subseteq \text{Range}(\mathscr{M})$, the following inequality holds:

$$\Pr[\mathscr{M}(D) \in S] \leq e^{\epsilon} \Pr[\mathscr{M}(D') \in S] + \delta. \tag{2.1}$$

Here, $\epsilon > 0$ is the privacy budget, controlling the degree of privacy, and $\delta \geq 0$ allows for a small probability of privacy leakage. When $\delta = 0$, the mechanism satisfies pure ϵ-differential privacy.

2.1.2.2 Sensitivity

To determine the noise required for DP, we define the sensitivity of a function, which measures the maximum impact that changing a single data point can have on the function's output.

Definition 1 (Global Sensitivity)**.** The global sensitivity of a function $f : \mathscr{D} \to \mathbb{R}^d$ is defined as:

$$\Delta f = \max_{D,D'} \|f(D) - f(D')\|, \tag{2.2}$$

where D and D' are adjacent datasets differing in at most one record.

2.1.2.3 Mechanisms for Achieving Differential Privacy

To satisfy differential privacy, noise is added to query results using the following mechanisms:

(1) Laplace Mechanism The Laplace mechanism ensures ϵ-differential privacy by adding noise sampled from a Laplace distribution:

$$\mathcal{M}(D) = f(D) + \text{Lap}(\lambda), \tag{2.3}$$

where $\lambda = \frac{\Delta f}{\epsilon}$, and $\text{Lap}(\lambda)$ is a random variable drawn from the Laplace distribution with mean zero and scale parameter λ.

Definition 2 (Laplace Distribution)**.** A random variable X follows a Laplace distribution, denoted as $X \sim \text{Lap}(\lambda)$, if its probability density function (PDF) is given by:

$$P(X = x) = \frac{1}{2\lambda} \exp\left(-\frac{|x|}{\lambda}\right). \tag{2.4}$$

(2) Gaussian Mechanism For (ϵ, δ)-differential privacy, the Gaussian mechanism is employed:

$$\mathcal{M}(D) = f(D) + \mathcal{N}(0, \sigma^2), \tag{2.5}$$

where $\mathcal{N}(0, \sigma^2)$ is a Gaussian distribution with mean zero and variance σ^2.

Definition 3 (Gaussian Distribution)**.** A random variable X follows a Gaussian distribution, denoted as $X \sim \mathcal{N}(0, \sigma^2)$, if its probability density function is:

$$P(X = x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{2.6}$$

The standard deviation σ is chosen based on the privacy parameters ϵ and δ, and the function sensitivity Δf.

(3) Exponential Mechanism For selecting outputs from a discrete range, the exponential mechanism is used. It ensures ϵ-differential privacy by selecting an output r according to the probability distribution:

$$\Pr[\mathcal{M}(D) = r] \propto \exp\left(\frac{\epsilon u(D, r)}{2\Delta u}\right), \tag{2.7}$$

where $u(D, r)$ is a utility function, and Δu is its sensitivity.

Definition 4 (Utility Function Sensitivity)**.** The sensitivity of a utility function $u(D,r)$ is:

$$\Delta u = \max_{D,D'} |u(D,r) - u(D',r)|. \tag{2.8}$$

(4) Renyi Differential Privacy Renyi Differential Privacy (RDP) is an alternative formulation based on Renyi divergence. It provides a more flexible privacy analysis, particularly under composition.

Definition 5 (Renyi Differential Privacy)**.** A mechanism $\mathscr{M}$ satisfies (α,ϵ)-Renyi differential privacy if for all adjacent datasets D and D':

$$D_\alpha(\mathscr{M}(D)\|\mathscr{M}(D')) \leq \epsilon, \tag{2.9}$$

where D_α is the Renyi divergence of order $\alpha > 1$.

(5) Local Differential Privacy (LDP) Local Differential Privacy (LDP) eliminates the need for a trusted central server by applying perturbation to each individual's data before sharing.

Definition 6 (Randomized Response)**.** A simple LDP mechanism for binary responses is given by:

$$P(Y = x) = \begin{cases} \frac{e^\epsilon}{1+e^\epsilon}, & \text{if } Y = X, \\ \frac{1}{1+e^\epsilon}, & \text{otherwise.} \end{cases} \tag{2.10}$$

where X is the original response, and Y is the reported (perturbed) response.

2.1.2.4 Composition of Differential Privacy

In practice, multiple queries are often applied to a dataset, leading to cumulative privacy loss. This is addressed by composition theorems:

Definition 7 (Sequential Composition)**.** If mechanisms $\mathscr{M}_1,\mathscr{M}_2,...,\mathscr{M}_k$ satisfy ϵ_i-DP individually, then their sequential composition satisfies:

$$\sum_{i=1}^{k} \epsilon_i\text{-DP}. \tag{2.11}$$

Definition 8 (Advanced Composition)**.** A stronger bound on cumulative privacy loss is given by:

$$\epsilon' = \sum_{i=1}^{k} \epsilon_i + 2\sqrt{2\log(1/\delta)} \sum_{i=1}^{k} \epsilon_i^2. \tag{2.12}$$

2.1.3 Dirichlet Mechanism

A Dirichlet mechanism, without any need for projection, maps:

$$\mathbb{P}[\mathscr{M}_D^{(k)}(p) = x] = \frac{1}{\mathrm{B}(kp)} \prod_{i=1}^{n-1} x_i^{kp_i - 1} \left(1 - \sum_{i=1}^{n-1} x_i\right)^{kp_n - 1} \tag{2.13}$$

where

$$\mathrm{B}(kp) = \frac{\prod_{i=1}^{n} \Gamma(kp_i)}{\Gamma\left(k \sum_{i=1}^{n} p_i\right)} \tag{2.14}$$

The multi-variable beta function serves as a Dirichlet mechanism, noteworthy for not requiring projection mappings. This mechanism, denoted by Dir and parameterized by k, balances between the precision of the Dirichlet mechanism and the degree of privacy it ensures. By methodically adjusting k, the privacy protection offered by the Dirichlet mechanism can be achieved.

Initially, the proposed approaches employ the Dirichlet mechanism to assess identity queries. A sensitive vector p becomes practically indistinguishable from its neighboring sensitive vectors due to the application of the Dirichlet process. Here, Δk signifies the sensitive data's storage region.

2.1.4 Federated Learning

Federated Learning (FL) is a machine learning paradigm where a model is trained across multiple decentralized devices (or clients) holding local data, without the need for exchanging the data itself. This approach is key for preserving data privacy and minimizing the necessity of central data storage. The concept can be formalized as follows:

Consider $\mathscr{D} = \{D_1, D_2, \ldots, D_n\}$ representing the distributed datasets across n clients, with D_i as the local dataset of the i^{th} client. The objective in FL is to train a global model $\mathscr{M}$ by aggregating locally computed updates, achieved by minimizing a loss function $\mathscr{L}$ over the model parameters θ:

$$\min_{\theta} \mathscr{L}(\theta) = \sum_{i=1}^{n} \frac{|D_i|}{|D|} \mathscr{L}_i(\theta) \tag{2.15}$$

where $|D_i|$ denotes the number of samples in D_i, $|D|$ the total number of samples across all clients, and $\mathscr{L}_i$ the loss function computed on the i^{th} client's dataset.

Federated Averaging (FedAvg) is a prominent algorithm in FL, where each client computes an update to the model based on its local data. These updates are averaged at a central server to update the global model. The update rule in FedAvg is given by:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \sum_{i=1}^{n} \frac{|D_i|}{|D|} \nabla \mathscr{L}_i(\theta^{(t)}) \tag{2.16}$$

Here, $\theta^{(t)}$ and $\theta^{(t+1)}$ represent the global model parameters before and after the update in round t, respectively, and η is the learning rate. This aggregation process ensures the global model learning based on the entire distributed dataset while keeping each client's data localized.

2.1.5 Machine Unlearning

Machine Unlearning is the process of removing or reducing the impact of specific data points from a trained model. In scenarios where data needs to be forgotten or corrected, machine unlearning ensures that the model no longer retains or reflects this information. A common approach to machine unlearning is retraining, which involves modifying the model to reflect the absence of the data to be unlearned.

The retraining approach typically requires identifying the specific data that needs to be unlearned and then retraining the model from scratch or from a certain point without including this data. Formally, if a model $\mathscr{M}$ is trained on a dataset $\mathscr{D}$, and a subset $\mathscr{D}_{\text{unlearn}} \subset \mathscr{D}$ needs to be unlearned, the model is retrained on the modified dataset $\mathscr{D} \setminus \mathscr{D}_{\text{unlearn}}$. This process could be described following:

$$\mathscr{M}_{\text{new}} = \text{Train}(\mathscr{M}, \mathscr{D} \setminus \mathscr{D}_{\text{unlearn}}) \tag{2.17}$$

Here, $\mathscr{M}_{\text{new}}$ represents the updated model after retraining. The retraining process ensures that the influence of $\mathscr{D}_{\text{unlearn}}$ is effectively removed from $\mathscr{M}_{\text{new}}$.

While retraining is conceptually straightforward, it can be computationally intensive, especially for large models and datasets. In the context of federated learning, retraining

for unlearning purposes must be coordinated across all participating clients, which adds to the complexity but is essential for maintaining the integrity and privacy compliance of the learning process.

2.1.6 Large Language Models (LLMs)

The fundamental architecture of LLMs is based on transformer models, characterized by a series of layers that systematically process the input text data [82]. At the key part of these models is the self-attention mechanism, a crucial feature that enables LLMs to assess the significance of each word in a sentence, thereby crafting responses that are contextually coherent [10]. The mathematical representation of an LLM's output can be succinctly expressed as:

$$O = F(I;\theta) \tag{2.18}$$

The equation represents the relationship where I is the input text, O the output generated by the model, F the function embodied by the LLM, and θ the set of parameters honed during training. The training process for LLMs involves fine-tuning these parameters (θ) to reduce the discrepancy between the model's output and the expected output, enhancing the model's precision in generating relevant responses. The sheer size of LLMs, with potentially billions of parameters, endows them with exceptional levels of language comprehension and production.

However, deploying LLMs is not without its hurdles. The need for extensive datasets for training and considerable computational power are significant barriers [25]. Furthermore, incorporating LLMs into federated learning environments brings extra challenges, including preserving model performance and privacy across decentralized data sources.

2.1.7 LoRA Fine-tuning

LoRA (Low-Rank Adaptation) [26] offers an innovative method for fine-tuning Large Language Models (LLMs) that balances efficiency with effectiveness, especially valuable in scenarios demanding model adaptability and computational thrift. Unlike traditional approaches that modify the original model parameters, LoRA adapts pre-trained LLMs to specific tasks through a low-rank decomposition technique. This method introduces additional trainable parameters, enabling the model to undergo task-specific adjustments without direct changes to its foundational parameters.

LoRA is particularly well-suited for federated learning environments, as it allows for efficient and targeted fine-tuning of LLMs across multiple participants without the need to share the entire model. Additionally, LoRA's low-rank decomposition approach makes it an ideal candidate for enabling effective unlearning mechanisms, as it allows for the selective modification of specific model components without affecting the overall model performance.

The core of LoRA's innovation lies in its approach to modifying the attention and feed-forward layers of transformer-based Large Language Models (LLMs) by integrating low-rank matrices. Specifically, for a weight matrix $W \in \mathbb{R}^{m \times n}$ within a transformer layer, LoRA introduces two smaller matrices, $A \in \mathbb{R}^{m \times k}$ and $B \in \mathbb{R}^{k \times n}$, where $k \ll \min(m, n)$. The adaptation of the original weight matrix W can be mathematically described as:

$$W' = W + AB \tag{2.19}$$

where W' denotes the adapted weight matrix, and AB is the low-rank update applied to W.

Such a strategy enables substantial customization of the model with only a modest increase in parameters, maintaining the extensive knowledge of the pre-trained LLM while introducing task-specific adjustments efficiently. During the fine-tuning phase, only the low-rank matrices A and B are updated, significantly lowering the computational demands typically seen with large-scale model training. Consequently, LoRA's approach to fine-tuning offers a scalable, resource-effective method for tailoring LLMs to diverse tasks and sectors. This is especially advantageous in federated learning settings, where computational efficiency and the flexibility to adapt to various tasks are paramount.

2.2 Related Work

2.2.1 Utilizing Blockchain in Location Privacy Protection

Current solutions for location privacy issues in the Internet of Vehicles (IoV) can be divided into two categories: level-based classification (event-level and user-level) and technique-based classification.

2.2.1.1 Level-based Classification

Location can indicate health, personal habits, and professional commitments by linking locations or paths visited. The Internet of Things (IoT) includes many technologies that

enable innovative services across applications. Vehicles have become active and efficient network nodes due to the car industry's extensive integration with IoT technologies and strong research base. Event-level scenarios require excellent time efficiency and reliability due to their complexity and variety. To simplify method analysis, prototypes always use event-level methods. The Internet of Vehicles event-level privacy protection using blockchain technology must not compromise system effectiveness. However, privacy protection can significantly impair vehicle operation management application operations. Data analysis quality, timeliness, depth, and trustworthiness are essential for traffic congestion forecast, decision-making, and worldwide perception positioning.

2.2.1.2 Technique-based Classification

In the Internet of Vehicles (IoV) domain, Crowd sensing LPPM is pivotal for user participation in mobile crowd-sensing, yet existing strategies often overlook the varied quality of task execution among participants [57, 95]. Legislation enforcement on data submission architecture is essential for data privacy, especially on edge servers [73], while k-anonymity and blockchain integration offer a solution for secure data contribution and reward receipt [53]. Singh and Kim's BCLPPM system, integrating a local dynamic blockchain with the main blockchain and employing the Intelligent Vehicle Trust Point (IVTP), ensures reliable vehicle-to-vehicle interactions [18, 90]. Additionally, the PBFT-based PoR (PPoR) consensus method addresses the challenges of V2V energy trading in vehicular networks [1, 2]. The integration of IoV with AI is revolutionizing intelligent transportation systems and enhancing VANET communication protocols [37, 64]. Blockchain architectures prove effective in detecting security attacks within IoV, offering enhanced privacy and efficiency [73].

2.2.2 Federated Learning and Machine Unlearning

2.2.2.1 Federated Learning with Machine Unlearning

The integration of federated learning and machine unlearning is a subject of increasing interest in current research. However, several studies do not adequately address the issue of trustworthiness. For instance, Liu et al.[41] delve into this area by proposing a rapid retraining method for efficient data removal in federated learning (FL), crucial for adhering to the 'right to be forgotten.' This approach highlights the challenges of implementing unlearning in decentralized systems like FL, where data remains local to participants. Our proposed method builds upon this by incorporating blockchain

technology into the FL framework for machine unlearning. This not only maintains FL efficiency but also adds a new level of transparency and accountability through the immutable ledger of blockchain, providing a verifiable and tamper-proof record of unlearning activities, crucial for IoT environments.

Wu et al.[78] address a definition of "Right to be Forgotten" in federated learning with a focus on federated unlearning. They outline efficient techniques for removing data influences from models, such as reverse SGA and EWC. While providing a solid unlearning framework, our method further enhances the process by integrating blockchain technology, ensuring verifiability and trustworthiness. This is particularly important for the sensitive nature of IoT applications, streamlining efficiency while fortifying the integrity of the unlearning process.

Wang et al.[68] focus on enforcing "the right to be forgotten" in federated learning systems, identifying the issue of information leakage due to model discrepancies before and after unlearning. Their work reviews existing studies proposes new taxonomies for unlearning algorithms, and highlights vulnerabilities to inference attacks. In contrast, our method integrates blockchain technology with federated unlearning, enhancing both the efficiency and security of the process. It ensures an immutable, transparent record of unlearning activities, thereby minimizing information leakage and unauthorized data access.

Wang et al.[69] introduce a method for selective unlearning in federated learning (FL) using channel pruning based on Term Frequency-Inverse Document Frequency (TF-IDF) for CNN classification models. This accelerates unlearning while maintaining model accuracy. Our approach, however, enriches the FL unlearning process by integrating blockchain technology, adding a layer of transparency and trust, essential for security and regulatory compliance in IoT applications.

Zhang et al.[91] present "FedRecovery," an efficient algorithm for machine unlearning in federated learning, overcoming the limitations of traditional retraining-based methods. It leverages differential privacy for indistinguishability in complex tasks like neural networks. Our method, by integrating blockchain technology, adds an essential layer of transparency and trustworthiness to the unlearning process, meeting the high security and compliance demands in IoT applications.

Liu et al.[39] propose "FedEraser," an approach in federated learning that efficiently removes specific training data from FL models, outperforming traditional methods in speed. It uses historical parameter updates for rapid model reconstruction. Our method enhances this by integrating blockchain technology into the federated unlearning process,

improving efficiency and significantly boosting transparency and trustworthiness, vital for handling sensitive data in IoT applications.

2.2.2.2 Blockchain-enhanced Federated Learning

The integration of blockchain in federated learning has been explored in various contexts, but the specific question of machine unlearning within this framework appears to be largely unaddressed. For instance:

Nguyen et al.[51] discuss FLchain, a novel integration of mobile-edge computing (MEC), federated learning (FL), and blockchain, which enhances privacy and security in decentralized AI training. This concept emphasizes collaborative training across multiple mobile devices without data exposure. In comparison, our research uniquely focuses on machine unlearning within this FLchain framework, a topic scarcely addressed in existing studies. Our approach uses blockchain not only for secure and private data management but also to enable trustworthy and verifiable unlearning processes in FL systems, advancing responsible AI practices in MEC environments.

Pokhrel et al.[54] introduce an autonomous blockchain-based federated learning system for vehicular networks, emphasizing decentralized, privacy-aware, and efficient communication. It combines on-vehicle machine learning and blockchain consensus mechanisms. However, our research augments this domain by incorporating machine unlearning into the blockchain-based federated learning (BFL) system, an aspect overlooked in their study. Our integration enhances both data processing efficiency and compliance with data regulations, making it highly relevant in dynamic environments like vehicular networks.

Li et al.[36] propose the Block-chain-based Federated Learning with Committee consensus (BFLC) framework, aiming to secure federated learning against malicious attacks. BFLC leverages blockchain for decentralized model storage and local model updates. Our research enriches this framework by integrating machine unlearning, a feature not included in BFLC. This addition not only addresses security concerns but also enables verifiable data removal, ensuring compliance in rapidly changing data environments.

Lu et al.[43] present a blockchain-based architecture for secure data sharing in the industrial Internet of Things, focusing on privacy in wireless networks. They integrate privacy-preserved federated learning into the blockchain consensus process. Our research extends this by embedding machine unlearning into the blockchain-enhanced federated

learning framework, addressing the need for dynamic data management and adherence to privacy regulations in the industrial IoT.

Shayan et al.[61] present Biscotti, a decentralized peer-to-peer approach for multi-party machine learning, using blockchain and cryptographic methods to enhance privacy and security, while addressing the limitations of federated learning. Our research further advances this concept by integrating machine unlearning capabilities into the decentralized learning framework, maintaining Biscotti's robust privacy and security features while adding the ability to manage and remove data responsibly, ensuring compliance with evolving privacy standards.

In summary, while various studies have explored blockchain's application in federated learning for enhancing security and privacy, our research takes a novel step by integrating machine unlearning into this paradigm. This integration addresses a crucial gap in the current literature and positions our approach as a pioneering solution in the realm of secure, dynamic, and responsible federated learning systems.

2.2.3 Large Language Models (LLMs) in Federated Learning and Machine Unlearning

2.2.3.1 Federated Learning with LLMs

In addressing the exhaustion of public data resources for LLM training, federated learning emerges as a potent solution. By enabling multiple participants to collaboratively train a model without sharing their raw data, federated LLM can access a wider array of diverse and representative datasets.

Chen et al. conclude the concept of federated Large-Scale Language Models (LLMs), which includes federated pre-training, fine-tuning, and prompt engineering, and explore the unique challenges and potential engineering strategies within this framework, highlighting its advantages over traditional LLM training approaches[13]. In Gupta et al. study[23], they introduce FILM, a novel attack methodology for federated learning of language models. They demonstrate for the first time the feasibility of recovering text data from large batch sizes and evaluating various defence strategies, thereby suggesting new directions for enhancing privacy in language model training. This paper [17] proposed an industrial federated learning framework, which is designed to facilitate the efficient training of large language models. This framework addressed the dual challenges of computational resources and data privacy. The LP-FL methodology prioritizes the reduction of model parameters within the federated learning framework [29], this

method employs Low-Rank Adaptation (LoRA) technology to construct compact learnable parameters, enabling effective local model fine-tuning and sustainable global model federation. FederatedScope-LLM (FS-LLM) provides a robust framework for optimizing LLM in a network. FS-LLM processes from data preparation to outcome assessment and facilitating diverse computational strategies [34]. Therefore, there are limitations among these frameworks due to the lack of transparency in the LLM training processes. [92] proposed an automated data quality control pipeline for federated fine-tuning of LLM, by utilizing data valuation algorithms, this pipeline assesses the quality of training samples across collaborative platforms, thereby enhancing model performance while preserving data privacy. There are also research concerns in the wireless field, [28] addresses significant challenges, including privacy concerns, inefficient data handling, and high communication costs, and demonstrates the effectiveness of these methods through simulations. In the Ro et al. research, they demonstrate that scale-invariant modifications to the Coupled Input Forget Gate (CIFG) and transformer models significantly enhance federated learning performance by improving convergence speeds and offering an improved privacy-utility trade-off [60].

2.2.3.2 Machine Unlearning with LLMs

The rapid advancements in large language models (LLMs) have led to remarkable breakthroughs in natural language processing and artificial intelligence. However, as these models are trained on vast amounts of data, they may inadvertently learn and perpetuate undesirable behaviors, biases, and harmful information. To address this issue, researchers have recently turned their attention to the concept of unlearning in LLMs.

In [85] paper, the authors explore the novel concept of unlearning in large language models (LLMs). They present a method that utilizes only negative examples to efficiently remove undesirable behaviors, demonstrating its effectiveness in alignment while significantly reducing computational resources compared to traditional reinforcement learning from human feedback (RLHF). While there is another paper introduces a data-driven unlearning approach for large language models (LLMs), utilizing a fine-tuning method informed by the importance of weights and relabeling during the pre-training phase of LLMs [89]. This method adjusts word embedding, involving identifying and neutralizing bias vectors within the embedding space to prevent biased associations. Wang et al.[70] proposed an unlearning framework called Knowledge Gap Alignment (KGA), emphasizing its capability to efficiently handle large-scale data removal requests with significant accuracy. However, the inability of KGA to guarantee the complete removal of

data influences also faces the challenge of maintaining extra data sets and models. Si et al.[62] explores the technical challenges of knowledge unlearning in large language models (LLMs), specifically introducing parameter optimization, parameter merging, and in-context learning as methods to efficiently remove harmful or biased data while maintaining the integrity of the models. This approach not only advances the field of responsible AI but also opens new avenues for enhancing data privacy and model impartiality. Huang et al. claim an innovation offset unlearning framework tailored for the black box LLM[27]. This framework effectively addresses the challenge of unlearning problematic training data in LLMs without requiring access to internal model weight, thus offering a versatile solution for adapting current unlearning algorithms.

2.2.3.3 Blockchain-enhanced LLM Training and Unlearning

Blockchain technology and artificial intelligence (AI) have emerged as two of the most transformative technologies of our time. The integration of these technologies has the potential to revolutionize various industries and address critical challenges faced by both domains. Recent research has explored the synergistic relationship between blockchain and AI, particularly focusing on the integration of blockchain with large language models (LLMs) and generative AI (GAI) techniques.

Luo et al.[44] introduce the concept of "Blockchain for LLM" (BC4LLM), which aims to empower LLMs with the superior security features of blockchain technology, enabling reliable learning corpora, secure training processes, and identifiable generated content. This paper presents emerging solutions that showcase the effectiveness of GAI in detecting unknown blockchain attacks and smart contract vulnerabilities, designing key secret sharing schemes, and enhancing privacy. Through a case study, they demonstrate that the generative diffusion model, a GAI approach, can significantly optimize blockchain network performance metrics, outperforming traditional AI approaches in terms of convergence speed, rewards, throughput, and latency [50]. Mboma et al. propose a novel approach to combat academic document fraud by integrating Large Language Models (LLMs), specifically the Bidirectional Encoder Representations from Transformers (BERT), with blockchain and Interplanetary File System (IPFS) technologies to pre-validate academic documents before certification [47]. LLMChain, a decentralized blockchain-based reputation system, assists users and entities in identifying the most trustworthy LLM for their specific needs while providing valuable information to LLM developers for model refinement [6]. This framework demonstrated through evaluation across two benchmark datasets, making it a significant contribution to the field of

trustworthy and transparent LLM assessment.

2.2.4 Comparative Analysis of Blockchain-Enhanced Federated Learning and Unlearning

Despite significant research in federated learning and machine unlearning, many existing studies lack a verifiable and immutable record of unlearning activities. Blockchain offers a promising solution by ensuring transparency, integrity, and decentralized trust in federated learning environments. However, integrating blockchain into FL and unlearning presents unique challenges, including scalability, consensus mechanisms, and data confidentiality.

Table 2.1 provides a comparative analysis of key studies on federated learning, unlearning, and blockchain integration. It highlights gaps in existing work and situates our research within the broader landscape.

Table 2.1: Comparison of Existing Work on Federated Learning, Machine Unlearning, and Blockchain Integration

Study	**FL/Unlearning Method**	**Blockchain Use**	**Transparency**	**Security Enhancement**
Liu et al. [41]	Rapid retraining for data removal	No	Limited	No
Wu et al. [78]	Reverse SGA and EWC for FL unlearning	No	No	No
Wang et al. [68]	Taxonomy of FL unlearning methods	No	No	No
Nguyen et al. [51]	FLchain for secure decentralized training	Yes	Partial	Medium
Pokhrel et al. [54]	Blockchain for vehicular FL	Yes	No	Medium
Li et al. [36]	BFLC for secure FL updates	Yes	No	Medium
Zhang et al. [91]	FedRecovery for unlearning with DP	No	No	No
Our Work	Blockchain-enhanced federated unlearning	Yes	Full	High

Blockchain as a Trust Layer for Federated Unlearning While traditional federated unlearning methods focus on selective forgetting, differential privacy, and retraining-based approaches, none incorporate an immutable and decentralized record of unlearning

activities. Our framework addresses this by leveraging blockchain's tamper-proof properties, ensuring that all unlearning operations are verifiable.

Challenges in Blockchain-Based Federated Learning Existing blockchain-integrated federated learning frameworks, such as FLchain [51] and BFLC [36], primarily focus on securing model updates rather than enabling verifiable unlearning. Key challenges include:

- Scalability: Large-scale federated learning requires efficient consensus mechanisms to prevent bottlenecks.
- Data Confidentiality: While blockchain offers transparency, model updates must remain privacy-preserving.
- Efficient Unlearning: Removing learned knowledge from a decentralized model is computationally complex.

Advancing Blockchain-Based Federated Learning and Unlearning Our research builds upon prior work by introducing:

- Fully Auditable Unlearning Records: Blockchain ensures that all data removal requests and unlearning operations are transparent, verifiable, and immutable.
- Hybrid Smart Contracts for Trust Management: Smart contracts automate federated model updates, unlearning verification, and participant rewards.
- Integration of Efficient Unlearning Mechanisms: Unlike existing methods, our approach employs gradient-based and modular forgetting techniques, combined with on-chain verification.

CHAPTER 3

A Scheme of Dynamic Location Privacy-Preserving with Blockchain in Intelligent Transportation System

3.1 Introduction

The Internet of Vehicles (IoV) is a prominent and integral part of the Internet of Things (IoT). IoV represents a network of automobiles equipped with sensors, software, and technologies that facilitate communication between them. The foundation of the IoV scenario is established by Vehicular Ad-hoc Networks (VANETs), which leverage a variety of protocols (e.g., IEEE 802.11P, IEEE 1609) and communication modalities, including Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I). IoV harnesses the capabilities of IoT to maximize utility, enabling innovative applications aimed at reducing accidents, alleviating traffic congestion, and providing other information services.

However, similar to other industries implementing big data, the IoV encounters significant challenges in data security and privacy during its advancement[94]. A paramount concern in IoV networks is the preservation of location privacy. The disclosure of a vehicle's precise position poses considerable risks to the traffic system and high-mobility vehicles on freeways. For instance, if an attacker compromises IoV systems in a high-speed vehicle, even minor disruptions can lead to significant traffic congestion. To mitigate these risks, Location Privacy Preserving Methods (LPPMs) have been developed to secure sensitive vehicle data, such as real-time location and trajectories. Current solutions for

location privacy in IoV can be categorized into several techniques: anonymity-based methods, Multi-agent Reinforcement Learning (MARL) methods[86], encryption-related methods, and Blockchain-based Location Privacy Preserving Methods (BCLPPMs).

Anonymity-based LPPMs in IoV aim to protect users' trajectory data over time. Event-level privacy, on the other hand, is concerned with safeguarding the user's position at each timestamp [46]. A limitation of these methods is their focus on mitigating data mining risks after user data has been compromised.

MARL solutions have addressed some of these issues in implementing location privacy-preserving approaches in IoV. Federated learning perspectives are also being explored on third-party platforms [87] [58]. Location privacy in trajectory data is crucial for intelligent transportation systems, reflecting users' movements, such as travel paths. However, these solutions often prove less feasible in actual industrial applications. Encryption-related solutions face challenges in demonstrating general effectiveness, especially when integrated into current business platforms. They struggle to maintain consensus among all platform users within a reasonable timeframe. Blockchain-based LPPMs, often combined with trust management systems [72], such as [38] and [63], typically focus on two main features: traceability and anonymity. However, these solutions frequently overlook the mobility aspect of vehicles in IoV, as highlighted in studies like [4]. The most effective method among the four categories mentioned earlier is the Blockchain-based Location Privacy Preserving Method (BCLPPM) [4].

Several platforms, such as those mentioned in [94], and [74], currently employ blockchain's distributed ledger and smart contract technology to ensure data integrity on the chain, while utilizing desensitization algorithms for privacy protection. However, there are three main limitations at the current stage:

- Centralized while they simulation: In BCLPPM, a decentralized system is employed for managing reputation, a feature recognized as advantageous in IoV for location-based crowdsourcing tasks. Roadside Units (RSUs) are configured to work in conjunction with vehicles within the IoV [40]. Each block within this system is capable of containing a set number of vehicle ratings. However, there is a concern that these ratings in the IoV might inadvertently disclose users' private information.

- Evaluation metrics is insufficient: From a performance perspective, Li et al. [36] consider the simulation results to demonstrate the feasibility and effectiveness of

Hyperledger, despite the lack of latency and throughput metrics in the proposed scheme.

- Experiments are not comprehensive: Further experiments are essential to identify the optimal settings for selecting and creating blocks, as well as to assess the scalability of the proposed plans. Once the feasibility and appropriate settings are established, it becomes crucial to ascertain whether they align with the demands of real-world Intelligent Transportation Systems (ITS).

To address the limitations previously discussed, we introduce a decentralized scheme for preserving the privacy of vehicle density location data. This work integrates the Dirichlet differential privacy mechanism with the capabilities of Vehicle to Infrastructure (V2I) communications. When vehicles register with Roadside Units (RSUs), the timestamp can signify a specific location at a given time, given that the coordinates of the RSUs are known to the intelligent transportation systems, as illustrated in Figure 3.1.

This scheme challenges conventional design principles by focusing not only on recognizing vehicle density but also on incorporating a flexible mechanism suitable for Blockchain-based Location Privacy Preserving Methods (BCLPPM) in practical Intelligent Transportation Systems (ITS). Moreover, the method allows for privacy sensitivity adjustments by a certified organization. A key challenge is to verify if the scheme meets the practical requirements for decentralized features. Specifically, the proposed privacy mechanism could be implemented within a smart contract. Once in place, assessing the performance of internal systems presents a significant challenge.

Based on the blockchain, this chapter has proposed and implemented a more effective and privacy focused scheme:

- This chapter addresses the location privacy problem in awareness of vehicle density via blockchain. The vehicle density definition comes from V2I communication in an intelligent transportation system, and the sensing network in IoV enable the V2I communication.

- This chapter raises various issues regarding the density of vehicles in intelligent transportation and the effect of these factors on the privacy of the vehicular location.

- This chapter analyzes the security properties and privacy. We evaluate the performance of the proposed scheme. The simulation results show that the vehicle

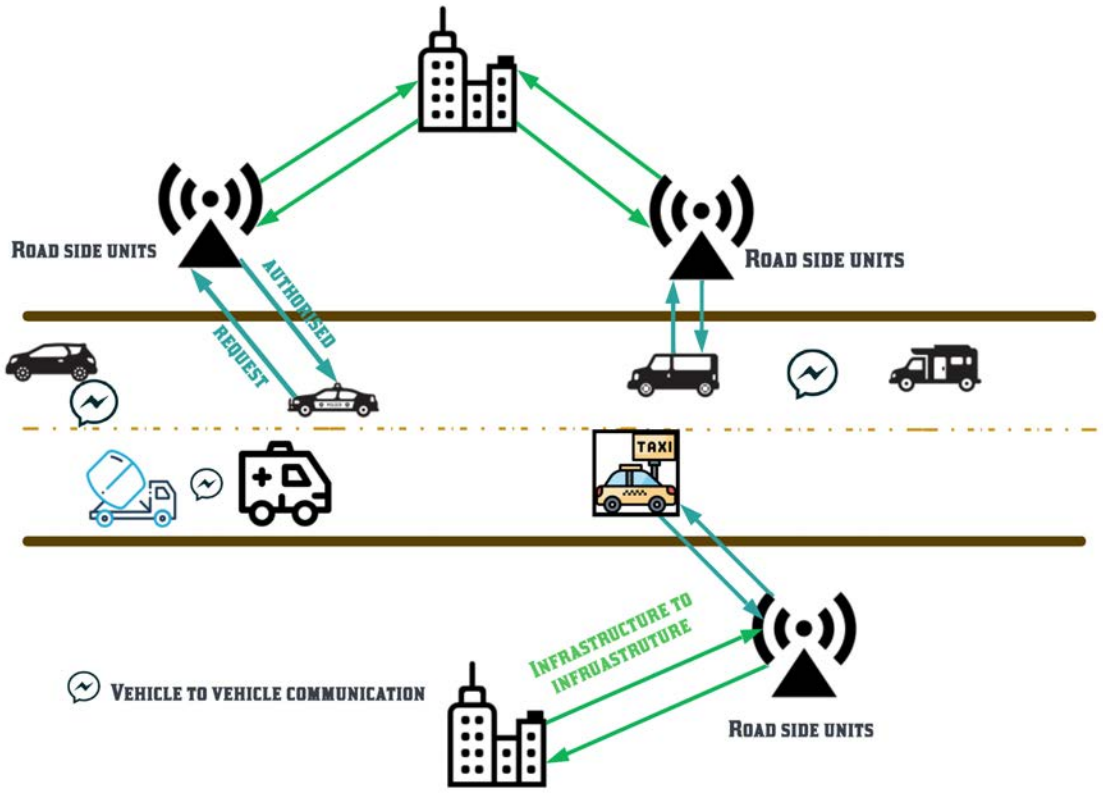

Figure 3.1: Overview scenario for Road Side Units Communication. The main components in this figure is road side units and vehicles, and how they communicate with each other.

density awareness method with multiple ratings proposed in this chapter can significantly validate the feasibility and efficiency.

3.2 Proposed approach

Differential privacy technologies have the potential for widespread application across large institutions through strategic integration. Within such institutions, which often consist of numerous departments, mutually trusted entities can expedite the process of certification exchange. This approach allows for efficient information sharing within a decentralized framework. Thus, while institutions may not share the most precise data on-chain, the utility of the data remains uncompromised.

The method we propose focuses on integrating the concepts of density and the standard deviation associated with Laplace noise. This has a direct relevance to urban

planning and is strongly correlated with the natural development of public transportation in urbanization. By offering viable solutions for the Internet of Vehicles (IoV), our method fosters an amalgamation of digitization, flexibility, and traffic management.

3.2.1 Consensus Mechanism

In our proposed system, the Road Side Units (RSUs) generate transaction records containing vehicle location updates and submit them to the blockchain network. To maintain data consistency and ensure transaction validity, Hyperledger Fabric's consensus mechanism is employed.

Hyperledger Fabric does not rely on traditional Proof-of-Work (PoW) or Proof-of-Stake (PoS) mechanisms. Instead, it utilizes a Practical Byzantine Fault Tolerance (PBFT)-like consensus approach in combination with an endorsement policy and an ordering service. In this setup, RSUs act as endorsing peers that validate transactions before they are ordered into blocks. Each transaction proposal must be endorsed by a predefined number of RSUs, as specified in the chaincode policy. Once a transaction collects enough endorsements, it is submitted to the ordering service.

The ordering service in Hyperledger Fabric is responsible for batching endorsed transactions into blocks and delivering them to all peers. This ensures a deterministic finality, meaning that once a block is committed to the ledger, it is immutable and does not require additional confirmations. The ordering service can be implemented using Raft, which provides a crash-fault tolerant consensus, or Kafka, which ensures message consistency.

By leveraging Hyperledger Fabric's pluggable consensus architecture, our system achieves a balance between security, scalability, and performance. The endorsement-based mechanism ensures that only authorized transactions are committed, while the ordering service guarantees an efficient and fault-tolerant transaction flow. This approach significantly reduces the computational overhead compared to traditional blockchain consensus models, making it well-suited for intelligent transportation systems where low latency and high throughput are required.

3.2.2 System modelling

To deploy hash-linked data storage, Hyperledger Fabric utilizes a specialized infrastructure type. Hyperledger Fabric is a permissioned blockchain platform that enables the development of distributed applications and networks. It follows a modular architecture,

which allows for customization and flexibility in designing blockchain solutions. The blocks are assembled such that the data is stored in a database-like structure, featuring multiple distinct hash instances, as illustrated in Figure 3.2. Hyperledger Fabric's architecture consists of several key components:

Peers: Peers are the fundamental elements of the network, responsible for maintaining the ledger and participating in the consensus process. They execute smart contracts (known as chaincode) and validate transactions.

Orderers: Orderers are responsible for establishing the order of transactions and creating blocks. They ensure the consistency and reliability of the transaction sequence across the network.

Chaincode: Chaincode is the smart contract in Hyperledger Fabric. It defines the business logic and rules for transaction processing and ledger updates. Chaincode is executed within a secure container environment on the peers.

Channels: Channels provide a mechanism for data isolation and confidentiality. They allow subsets of network participants to communicate and transact privately, without exposing their data to the entire network.

Hyperledger Fabric's modular design and permissioned nature enable fine-grained access control, privacy, and confidentiality. It supports the use of different consensus algorithms. By leveraging these architectural components and the hash-linked data storage mechanism, Hyperledger Fabric provides a robust and flexible platform for building blockchain-based applications in various domains. The overview of how proposed approach effects in blockchain is shown in Figure 3.2.

3.2.3 Security and Privacy Requirements

To ensure the integrity and confidentiality of the system, several security and privacy requirements must be met by the key participants in the process: the Certificate Authority (CA), vehicles, and roadside units (RSUs).

- **Certificate Authority:** The CA, whether a trustworthy organization or certified third-party corporation, plays a crucial role in maintaining the security of the system. It is responsible for implementing secure chain code, properly identifying and authenticating roadside units, and configuring the system with robust security measures. The CA must possess strong communication and computational capabilities to prevent unauthorized access and protect sensitive data.

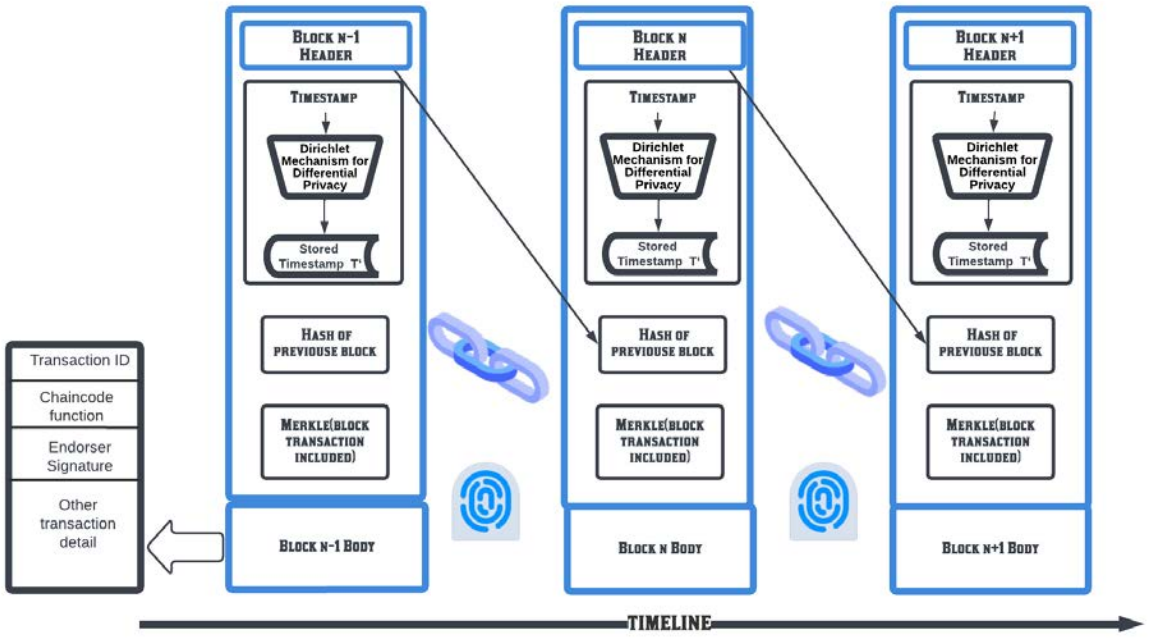

Figure 3.2: Overview proposed approach effects in blockchain

- **Vehicles:** Vehicles, as participants in the system, must ensure secure interactions with roadside units and maintain the privacy of their location data when cooperating with Location Based Services (LBS). The on-board components of the Internet of Vehicles (IoV) should provide secure Vehicle-to-Infrastructure (V2I) connectivity for transmitting traffic information updates. To protect user privacy, the vehicle density D used for traffic forecasting in the intelligent transportation system should be aggregated and anonymized, preventing the identification of individual vehicles or users. Secure communication protocols and encryption mechanisms should be employed to safeguard the transmitted data.

- **Roadside Units:** RSUs play a critical role in collecting real-time vehicle position data and must be properly certified and validated by authorized entities on the blockchain. The certification process ensures that only legitimate and trusted RSUs are part of the system, reducing the risk of rogue or compromised units. RSUs should securely store and transmit the collected vehicle position data, protecting it from unauthorized access or tampering. Secure communication channels and data encryption should be used when RSUs interact with vehicles and other system components. The strategic positioning of designated RSUs should consider both optimal data collection and the security of the infrastructure.

By addressing these security and privacy requirements for the CA, vehicles, and RSUs, the system can establish a robust and trustworthy framework for secure data collection, transmission, and storage. Implementing strong authentication mechanisms, secure communication protocols, and data encryption techniques will help protect sensitive information and maintain user privacy. Regular security audits and updates should be conducted to identify and address any potential vulnerabilities in the system. By prioritizing security and privacy, the intelligent transportation system can ensure the integrity of the collected data and maintain the trust of its users.

3.2.4 Initialization

According to Algorithm 1, the initial phase of our method is outlined. This approach's major objective is to guarantee that the registration procedure is successful. The initialization process comprises issuing IDs, which are designated as V_{id} and RSU_{id}, to unauthenticated RSUs and vehicles.The above step serves the improvement the Hyperledger authentication method that was initially developed. The V_{id} should be checked to see whether it is duplicated on the blockchain in the event that the registration process is unsuccessful. whether it is found to be duplicated, a new V_{id} should be provided. Afterwards, it has been established the V_{id}, the proposed mechanism will proceed to generate the token key_v for the vehicle V_i, as seen in lines 5 and 6. The process of verifying RSU_{id} is carried out in a manner that is comparable. During the registration process, the procedure checks that all V_{id} and RSU_{id} have been successfully merged into the pool. This prevents any ID clashes from occurring. The method is located on line 12. Furthermore, this phase acts as a pre-measure in Algorithm2, to avoid a possible excessive amount of SHA-256 duplication. RSU_{id} is the identifier that is used to record subsequent transactions on the blockchain.

3.2.5 Vehicle sign up with RSU

This procedure focuses on the vehicle registering its data with roadside units. Algorithm 2 aims to phrase the important message while the vehicle communicates with the roadside units. Due to the nature of the roadside device, its position and the vehicle's coordinates at the moment of registration are steady. Consequently the timestamp $Time$ and the RSU_{id} mean the position for the V_{id}. Furthermore, there is a function in blockchain to track the existing transactions, which is straightforward to collect the trajectory of the particular vehicles. A complete verification is supposed to serve for this algorithm

Algorithm 1 Privacy Trajectory building

Require: V_{id},RSU_{id}
Ensure: *RegisterSucess*, *key*
1: *Register success* = *False*;
2: **if** $V_{id} \in V_{pool}$ **then**
3: **return** Change V_{id}
4: **end if**
5: $key_v \leftarrow Blockchain$;
6: $V_{id} \leftarrow key_v$;
7: **if** $RSU_{id} \in RSU_{pool}$ **then**
8: **return** Change RSU_{id}
9: **end if**
10: $key_{RSU} \leftarrow Blockchain$;
11: $RSU_{id} \leftarrow key_{RSU}$;
12: $Pool \leftarrow Pool \cup V_{id} \cup RSU_{id}$;
13: *Register success* = *True*;
14: **return** *RegisterSuccess*, *key*

in Line 2. This method also includes the generation vehicle condition, and the time is appended to the block header whenever a new block is created. Again, this algorithm is commonly repeated. Contracting state Algorithm 2 with another algorithm would not decrease latency.

Algorithm 2 Vehicle Sign-Up with RSU

Require: $Time$, Position Pos, V_{id}, RSU_{id}
Ensure: Sign-Up Successful
1: Vehicle passes the RSU
2: RSU validates the vehicle identity V_{id}
3: **if** $V_{id} \in V_{pool}$ **then**
4: RSU generates a record $\{V_{id}, Time, Pos, RSU_{id}\}$
5: RSU stores the record $\{V_{id}, Time, Pos, RSU_{id}\}$ in the transaction database
6: **return** Sign-Up Successful
7: **else**
8: **return** Invalid V_{id}, Sign-Up Unsuccessful
9: **end if**

3.2.5.1 Vehicle Density

The proposed method integrates design, simulation, verification, deployment, and operation across systems. It facilitates safe and reliable collaboration and intractability between heterogeneous information systems and physical systems. The goal is to achieve

reliable, efficient, real-time perception and decision-making control for intelligent networked vehicles, thereby enhancing driving comfort.

$$\lambda \cdot \varepsilon_j = f\left(\frac{\lambda_j}{\delta_j}\right) + k$$

- k is the traffic flow control constant.

This method categorizes the existing research on measuring location privacy into two broad groups, based on distinct definitions of location privacy for different vehicle densities and the associated projected inference error. The concept of rural areas provides a significant conceptual basis for location privacy solutions that aim to maintain consistent traffic flow control. We acknowledge the varied density ranges and propose that mobility and speed cannot always be clearly distinguished from density in current traffic density studies. These studies provide users with high mobility in manipulating vehicle traffic indicators (speed, volume, and density).

The scenario can be summarized from three distinct perspectives.Moreover, it is both feasible and desirable to develop a location obfuscation mechanism that effectively combines these two unique privacy concepts. Our scheme is designed to be advantageous and practicable. Lastly, incorporating user-defined constraints enhances usability a key factor in privacy protection models and supports adaptive noise adjustment for distinguishability. It also meets mobile users' customized privacy/utility requirements, assuming that each driver reacts to stimuli from other vehicles ahead or behind in some manner.

3.2.6 Data Submission

As outlined in Algorithm 3, the most significant feature is the data processor equipped with a differential privacy mechanism. This allows us to treat the specific timestamp $Time_i$ as private. The rationale is to adhere to a fundamental blockchain rule: the timestamp of the last block must precede that of the next block. To maintain the utility or readability of the blockchain after applying the privacy mechanism, $Time$ is adjusted using the Dirichlet mechanism for the Query model. The final adjusted result is then recorded in the chain code. The complexity of ensuring high-level security requirements is notable; for example, RSU_{id} must be verified as part of the certificated RSU_{pool}. The input vehicle density D, the Dirichlet parameter P, and the output of Algorithm 3 are central to this algorithm. In case we consider the additional new RSUs, we would like to start the initialization step, which is similar to the Algorithm 1.

Algorithm 3 Data Submission Process

Require: Density D, Number of Vehicles N, Dirichlet parameter P, RSU_{id}
Ensure: Submit Successful

Ensure a sufficient number of vehicles pass the RSU
RSU computes the Dirichlet distribution Dir() based on P, D, and N
RSU augments Dir() to the timestamp of the record to generate $\{V_{id}, \text{Dir}(Time), D, N\}$

RSU sends the record $\{V_{id}, \text{Dir}(Time), D, N\}$ to the Smart Contract
if $RSU_{id} \in RSU_{pool}$ **then**
    Smart Contract commits the record to the blockchain
    RSU clears the record from the transaction database
    return Submit Successful
else
    return Invalid RSU_{id}, Submit Unsuccessful
end if

3.3 Security and Privacy Analysis

The privacy analysis is based on the analysis of the query part in the Dirichlet mechanism. Based on the linear query collection feature, it is necessary to acknowledge the qualification criteria of the privacy guarantees afforded via the Dirichlet mechanism. As mentioned in the preliminary [56], privacy guarantees would satisfy the requirement of the bounding ratios of the Dirichlet distributions. Moreover, a part of the function should include $gamma$ functions. assumption [21].

Assumption 1. *In* $\Delta_{n,U}^{(\eta,\bar{\eta})}$, $\eta > 0$, $\bar{\eta} > 0$, *and* $\eta + \bar{\eta} < \frac{1}{2}$.

Let d be a vector in $\mathbb{R}^n$. As mentioned by [22], the notation $d_{(i,j)}$ refers to the vector $(d_i, d_j)^T \in \mathbb{R}^2$, where $(\cdot)^T$ indicates the transpose. Additionally, $d_{(i,j)} \in \mathbb{R}^{n-2}$ represents the vector d with the i-th and j-th entries removed.

We use $\mathbb{D}[\cdot]$ to indicate the probability of an event, $\mathbb{E}[\cdot]$ for the expectation of a random variable, and Var[·] for its variance. The symbol $|\cdot|$ denotes the cardinality of a finite set, and $\|\cdot\|_1$ refers to the 1-norm of a vector.

Additionally, this part employed the gamma function $\Gamma(\cdot)$, the beta function $B(\cdot,\cdot)$, and the digamma function $\psi(\cdot)$.

$$\Gamma(x) = \int_0^\infty z^{x-1} \exp(-z) dz, \quad x \in \mathbb{R}_+$$
$$\text{beta}(a,b) = \int_0^1 t^{a-1}(1-t)^{b-1} dt = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}, \quad a,b \in \mathbb{R}_+$$
$$\psi^{(0)}(x) = \frac{d}{dx}\log\big(\Gamma(x)\big), \quad \psi^{(1)}(x) = \frac{d^2}{dx^2}\log\big(\Gamma(x)\big), \quad x \in \mathbb{R}_+$$

Assumption 2. For the Dirichlet mechanism by [22] $\mathscr{M}_T^{(k)}$, the parameter k satisfies

$$k \geq \max\left\{\frac{1}{\eta}, \frac{1}{1-\eta-\bar{\eta}}\right\}.$$

We later employ the texting option to fine-tune the compromise we have made between the Dirichlet mechanism's precision and privacy. After that, we prove that the Dirichlet mechanism is indeed secure.

Lemma 1. Let Assumptions 1 and 2 hold. Consider a set of indices W used to construct $\Delta_{n,U}^{(\eta,\bar{\eta})}$. Suppose p and q are any *b-adjacent* vectors in $\Delta_{n,U}^{(\eta,\bar{\eta})}$ with different i^{th} and j^{th} entries. Also base on a constant $k \in \mathbb{R}_+$, it holds that

$$\frac{\text{beta}(kq_i, kq_j)}{\text{beta}(kp_i, kp_j)} \leq \frac{\text{beta}(kq_i, k(1-\bar{\eta}-q_i))}{\text{beta}(kp_i, k(1-\bar{\eta}-p_i))}.$$

where the parameter $\gamma \in (0,1)$ defines the set Ω_1.

As previously stated, we present a simplified constraint on $epsilon$ that offers a direct dependency of $epsilon$ on other factors in the issue.

Ensuring strong privacy guarantees is a key objective of our blockchain-based location privacy-preserving scheme. While empirical validation is ideal for measuring privacy performance, this section provides a theoretical analysis of privacy protection levels and the trade-offs between privacy, utility, and system performance.

Privacy Budget and Location Obfuscation Our system employs a differential privacy mechanism to obfuscate vehicle location data before submission to the blockchain. The level of privacy protection is determined by the privacy budget ϵ. The probability of an adversary successfully inferring the original location x from the obfuscated location $\hat{x}$ is given by:

$$P(\hat{x} = x) \leq e^{-\epsilon} \tag{3.1}$$

This equation indicates that as ϵ decreases, the probability of successful inference drops exponentially, making it increasingly difficult for attackers to deduce the true location.

Privacy-Utility Trade-off A key challenge in privacy-preserving systems is the balance between privacy and utility. Higher privacy protection (lower ϵ) leads to greater distortion in location data, potentially affecting applications that rely on precise positioning. The theoretical relationship between privacy level and location accuracy is expressed as:

$$\mathscr{D}(\epsilon) = \frac{C}{\epsilon} \tag{3.2}$$

where $\mathscr{D}(\epsilon)$ represents the expected location distortion, and C is a system-dependent constant. This equation illustrates that reducing ϵ increases location distortion, impacting the precision of intelligent transportation applications.

Privacy Performance Estimation To estimate the expected privacy performance in real-world scenarios, we analyze the effect of different privacy budget values:

- When $\epsilon = 0.1$, the location distortion $\mathscr{D}(\epsilon)$ is large, ensuring strong privacy protection but reducing data accuracy.
- When $\epsilon = 1.0$, a moderate privacy level is achieved, balancing security and usability.
- When $\epsilon = 5.0$, the distortion is small, maintaining high location accuracy but providing weaker privacy guarantees.

These theoretical results suggest that optimal privacy settings depend on the specific requirements of the application. If security is the priority, a lower ϵ is preferable; if accuracy is critical, a higher ϵ may be selected.

Impact on Blockchain Performance The privacy-preserving transformations introduce computational overhead, affecting blockchain throughput and latency. The relationship between privacy budget and blockchain throughput $T(\epsilon)$ follows:

$$T(\epsilon) = T_0 e^{-\beta\epsilon} \tag{3.3}$$

Table 3.1: Summary of Hyperledger Fabric 2.1 Implementation

Component	Description
Tools	Docker 3.52.18, Docker Compose 1.29.2, Caliper 0.4.1
Network Configuration	CA node, administrator node, two client nodes, two peer nodes
Interface	Command-line for client nodes
Smart Contracts	Installed on users' computers
Privacy Features	Private collections, transient data in binary
Transaction Batching	Default block size of 100 transactions
Testing	Two peers per organization in three organizations

where T_0 represents the baseline throughput without privacy mechanisms, and β is a scaling factor reflecting the computational impact of privacy-preserving transformations. As privacy protection increases (lower ϵ), transaction processing slows due to additional noise computation and verification overhead.

Similarly, privacy protection impacts transaction processing latency, expressed as:

$$L(\epsilon) = L_0 + \gamma \cdot \mathscr{D}(\epsilon) \tag{3.4}$$

where $L(\epsilon)$ is the latency, L_0 is the base processing time, and γ represents the sensitivity of latency to privacy transformations. This equation suggests that higher privacy protection increases processing time, as the system must generate, verify, and store privacy-preserving transformations before committing transactions to the blockchain.

3.4 Performance analysis

3.4.1 Set up and Results

Hyperledger Fabric 2.1 uses Docker 3.52.18, Docker Compose 1.29.2, and Calliper 0.4.1. The network has a CA, administrator, client, and peer node. Client nodes interact via command-line and install smart contracts on users' machines. Private collections and binary transitory data improve data privacy in version 2.1. Initialization batches transactions into 100-transaction blocks. Testing involves two peers per isolated organisation in three organisations.This section analyses Calliper results on VANET-based applications to improve Intelligent Transportation Systems. A single organisation with one peer to three organisations with two peers are simulated in the study.

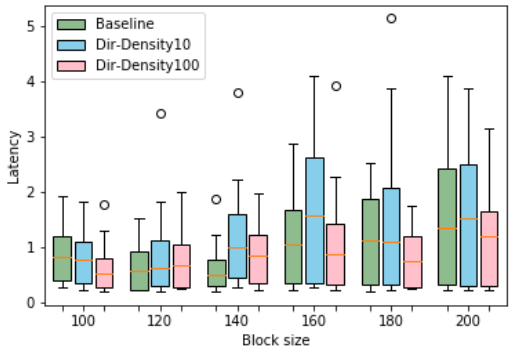

Figure 3.3: Latency distribution comparison with different blocksize in different vehicle density

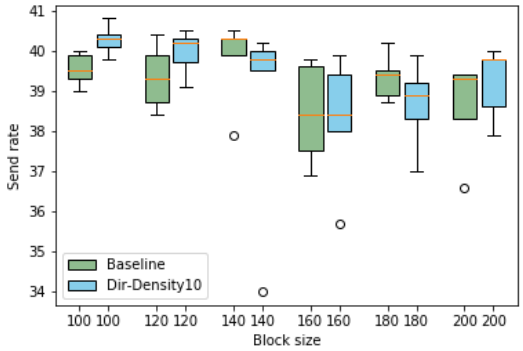

Figure 3.4: Send rate distribution comparison with different blocksize in different vehicle density

- The preparation period involves several tasks, such as initializing the entire blockchain network, reading the configuration file, deploying smart contracts, and initiating monitoring setup.
- The testing phase commences with the client leveraging the benchmark configuration. Subsequent to conducting the statistical results are typically returned.
- The "reporting" stage entails the analysis of statistical data and the generation of HTML reports.

3.4.2 Reporting and analysis

The data write throughput is substantially higher than the average volume and the federated chain's consensus structure will employ a Byzantine fault-tolerant mechanism that is both more efficient and less stressful on the environment [9] distributed. The provided approach has no significant influence on the reaction time or throughput of the blocks, and it preserves all of the federated chain's features. There are the results from mechanism based on the hyper ledger we designed, which we called the baseline result. As for the most existing proposed mechanism is based on the permissionless blockchain.

Experiments with fixed block sizes: The Figures 3.3, 3.4 and 3.5 represent the outcomes of the validation and commitment of anywhere from 100 to 200 transactions for a single run, with this process being repeated 10 times. Each block of 100 transactions contains a collection of previous transactions. The topic of latency will be covered first, followed by throughput. For both delay and throughput, the rate is measured in trans-

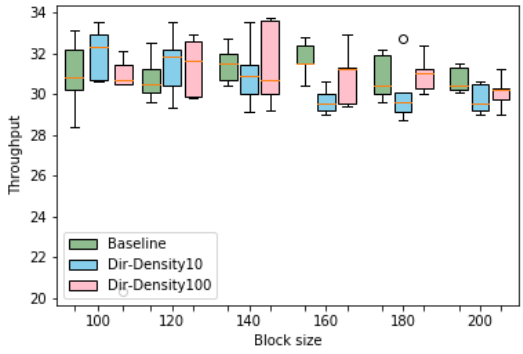

Figure 3.5: Throughput distribution comparison with different block size in different vehicle density

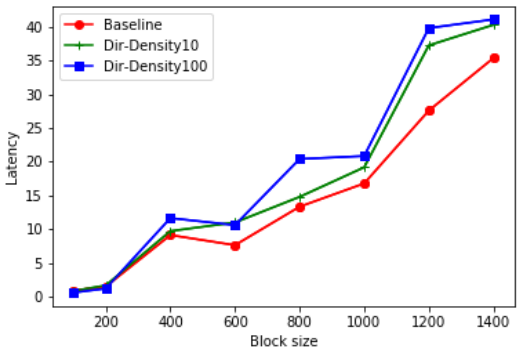

Figure 3.6: Latency comparison with different block size in different vehicle density

actions per second (TPS), and it varies depending on the size of the network. Due to batching, we display the latency for each individual transaction rather than the latency for each individual block.

The proposed framework's efficiency is reduced by blockchain latency, which may impose delay during the training phase. There is also involved in the throughput and send rate.

Data was separated into three pieces for box graphing. The first segment contains the median, while the second and third segments show the upper and lower half of the median for the baseline [81], D=10 D=100, with latency of 0.83, 0.79, and 0.70, respectively. The box plot is a classic way to show data distribution using five numbers: minimum, first quartile, median, third quartile, and maximum. Figures 3.3, 3.4, and 3.5 show the medians as segments within rectangles and the minimum and maximum values as boxes.

The line graph in Figure 3.6 shows the average block construction time for add-on vehicle density of 10 and 100, respectively. Statistics show average delay from 200 to 1400 and for each round of that block size. 10 rounds of transactions would be built in a large block. As block size rises, vehicle density will rise, hence each block's latency will primarily rise. Baseline time consumption shares the shortest delay of 28 ms per block at 1400 block size. Given the time efficiency of the proposed scheme, our results align with the self-imposed goal of increasing throughput without incurring additional latency. In fact, our enhancements have slightly reduced peer latency, accompanied by an upward trend in the transmission rate. While the pipelining in scenarios with a vehicle density

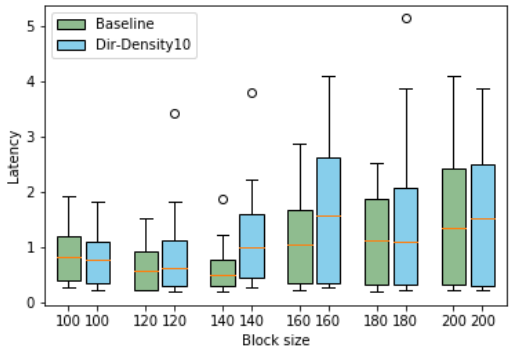

Figure 3.7: Latency distribution with different block size in vehicle density =10

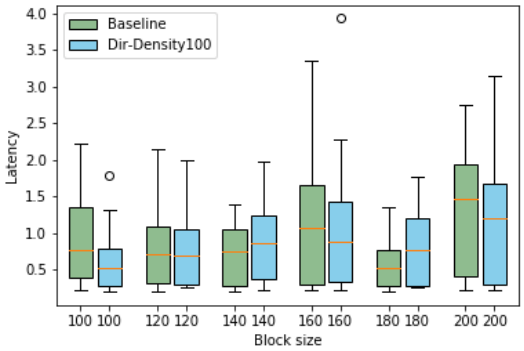

Figure 3.8: Latency distribution with different block size in vehicle density =100

equal to 10 introduces some delay, the other benefits observed more than compensate for this increase.

The throughput of the proposed scheme, measured in transactions per second, reflects the total load managed by the scheme each second. Figure 3.5 presents the throughput for block sizes ranging from 100 to 200, showcasing a stable trend with slight fluctuations within an acceptable range. As the block size increases, there is a gradual decline in the median throughput. Moreover, the reliability of transactions per second is less affected by changes in density. In summary, as the target block size grows, the throughput experiences a marginal decline. In private blockchains, these factors may be adjusted as a design choice. Conversely, a permissioned blockchain permits an approximate target inter-block duration. To assess the efficacy of our simulation against various transaction execution times on the blockchain, a broad spectrum of transaction inclusion times was tested. The outcomes, predicated on our hypothesis, are presented in Figures 3.7 and 3.8 below, depicted as box plots.

The box plots in Figure 3.10 effectively illustrate the dispersion characteristics of the dataset. The most efficient use of box plots is to compare across different block sizes and qualitative data of the privacy budget to obtain grouped box plots. Outliers are values that lie outside the typical range of the data set. The distribution exhibits a rightward skew, with outliers clustering around the value state transition when the vehicle density equals 10 for block sizes of 400.

By displaying the latency results of Fabric with different densities, Figures 3.7 and 3.8 are created. The grouped box plot of Dirichlet differential privacy, which, in comparison to the baseline experiment, seems to be considerably taller, may be compared with these.

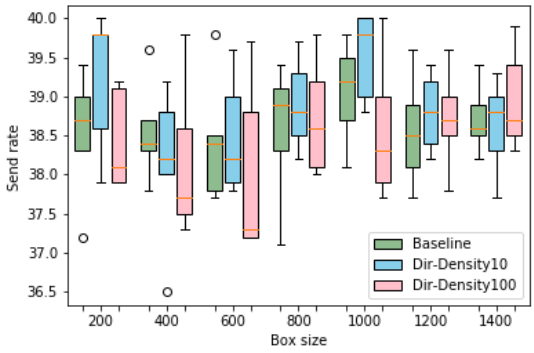

Figure 3.9: Send rate distribution with different block size in different privacy budget

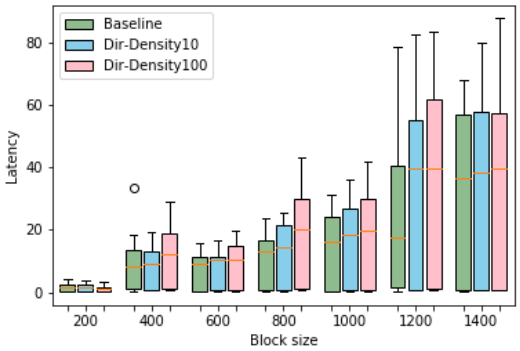

Figure 3.10: Latency results of Fabric with varying different block-size, with high volume transportation location update requirements such as in rush hour

The experiment may be distinguished from other similar experiments by the significant changes that can be seen throughout the box plots, which highlight these distinctions. As vehicle density increases, the difference in latency between the DP add-on and the baseline method exhibits fluctuations of approximately 15 percentage points. Considering that the block-created failure rate is 0.5 per cent, the solution provided in this study is for Fabric. The latency outcomes illustrated in Figure 3.10 are applicable to every block, factoring in the rise in vehicle density and the blockchain's vulnerability to potential attacks, as well as resource awareness. The network successfully processes nearly all submitted transactions, indicating that the performance results are not exclusively contingent upon our proposed scheme. Moreover, vehicle density has a minimal impact on the transmission rate, as depicted in Figure 3.9.

3.5 Summary

This chapter introduces a blockchain-based Dirichlet differential mechanism solution for intelligent transportation system privacy. When automobiles collide with roadside units, our system uses blockchain block timestamps to estimate geographic coordinates. Our trials show improved latency, send rate, and throughput. In addition, the Dirichlet mechanism handles vehicle density as a critical privacy concern using probabilistic mapping via Dirichlet distribution. Future research could examine blockchains in public or organisational settings. Regulation of the blockchain is necessary to provide strong security and privacy requirements for ordinary activities [79]. Most measurement tools

fall short of these strict requirements, indicating a critical development area.

CHAPTER 4

FEDERATED LEARNING WITH BLOCKCHAIN-ENHANCED MACHINE UNLEARNING

4.1 Introduction

Federated learning has found new synergies through the advancements in the Internet of Things (IoT), which have significantly transformed service computing, an approach to machine learning that allows collaborative model building among multiple clients [32].

However, according to recent regulations like the GDPR [59] and CCPA [8], users have the right to request the deletion of their data. Consequently, there emerged a new technology called machine unlearning, that integrate into federated learning to meet those regulations [80]. The need for machine unlearning stems from the imperative to systematically remove specific data from trained models in IoT environments. Currently, there are several methods to achieve machine unlearning in federated learning [7][5].

Unfortunately, existing machine unlearning techniques suffer from one critical issue: when they have deployed in servicing computing: the model owner cannot provide users with evidence that their data has truly been unlearned.

In other words, users must blindly believe that the server has executed the unlearning oracle to remove the impact of their data, with little opportunity to verify it [15]. This makes the entire unlearning process untrustworthy and may lead to concerns about privacy. An untrustworthy service providers may pretend to have performed the unlearning operation but actually take no action to avoid high computational costs or

utility decreasing [19].

Blockchain technology, renowned for its immutability, transparency, and robust security, does more than just establish a foundation of trust for digital transactions and record-keeping. It also extends this trust paradigm to machine unlearning[73]. This integration leverages the strengths of blockchain to address trust issues, offering several key advantages: First, blockchain guarantees that each step in the unlearning process is immutably recorded, ensuring a verifiable and transparent certification of data removal actions. This level of transparency is crucial for auditability, allowing stakeholders to easily trace and verify all unlearning actions, thereby assuring the integrity of the process. Second, blockchain's robust security framework is instrumental in preventing unauthorized alterations, a vital aspect for maintaining reliable certification. Finally, its decentralized nature spreads the trust mechanism across multiple nodes, effectively eliminating single points of failure and bolstering the system's resilience in certifying unlearning actions. The convergence of these benefits significantly bolsters the trustworthiness of the unlearning process in federated learning environments.

The blockchain layer functions as a decentralized ledger, providing an audit trail that confirms the accurate execution of unlearning requests. This design significantly increases the trustworthiness and integrity of the federated learning model. If we introduce a trustworthy machine unlearning framework that is based on the combination of blockchain and federated learning, we will have a certified machine unlearning technology.

However, there are several challenges when we integrate the blockchain with machine unlearning,

- **Certification:** Certification presents a complex issue, as it necessitates the system to authenticate an unlearning request in a decentralized environment.

- **Security and Privacy:** Balancing security and privacy is essential; although blockchain improves the transparency of the machine unlearning process, it potentially risks compromising the confidentiality of sensitive data. This is because every user in the blockchain network might have access to view the content.

- **Efficiency:** Efficiency remains a key concern. The integration of blockchain could potentially introduce latency or computational overhead, which needs to be carefully managed to preserve agility and responsiveness, especially in the context of IoT.

This chapter tackles these challenges by integrating of blockchain in federated learning guarantees an unalterable record of unlearning requests and actions, thereby enhancing certification, efficiency, and establishing a robust security and privacy framework. For certification, our framework employs smart contracts to automate and streamline the verification and validation of unlearning requests, ensuring their authenticity and accurate processing. Regarding security and privacy, we integrate differential privacy mechanisms into the smart contracts. This integration adds an extra layer of data protection, safeguarding confidentiality while maintaining transparent unlearning processes. Efficiency is enhanced by optimizing the storage of federated learning training process data. This adjustment minimizes computational overhead, ensuring that blockchain technology integration does not compromise the system's performance too much, especially in time-sensitive IoT applications.

Our contributions are summarized as follows:

- We proposed a novel framework that integrates blockchain technology with federated learning specifically for machine unlearning, providing a verifiable certification of the unlearning process in decentralized environments. This certification authenticates and validates each unlearning request, offering crucial proof within IoT contexts.
- We enhanced the security and privacy of machine unlearning in federated learning systems. Utilizing blockchain's transparent ledger and smart contract capabilities, it establishes a secure method to protect data security and privacy.
- We optimized the data management process in machine unlearning by combining a blockchain network with federated learning in an IoT scenario. This approach efficiently stores training process data in federated learning, ensuring system agility and responsiveness when an unlearning request is initiated in the blockchain network.

4.2 Problem Definition and System Model

4.2.1 Problem Definition

This research tackles the challenge of incorporating machine unlearning into Federated Learning (FL) systems by leveraging the advantages of Blockchain technology. The problem encompasses the following key components:

4.2.1.1 Federated Learning Environment

In a typical Federated Learning setup, a set of clients $C = \{C_1, C_2, \ldots, C_n\}$ collaboratively train a shared model M using their local datasets $D = \{D_1, D_2, \ldots, D_n\}$. This training occurs without the actual exchange of data, reflecting the privacy-preserving nature of Federated Learning. Each client C_i contributes to the global model by training a local model M_i on its dataset D_i, and these local models are aggregated to form the global model M. The challenge lies in coordinating this collaborative training process while ensuring data privacy and model efficacy.

4.2.1.2 Machine Unlearning Requirement

Machine unlearning, in this context, refers to the process of removing the impact of particular datasets D_{ui}, where $D_{ui} \subset D_i$ for a client C_i, from the trained global model M. This requirement arises for reasons such as data correction, compliance with legal requests for data removal, or ethical considerations. The key challenge is ensuring that the unlearning process is both effective and verifiable. It is crucial that once a dataset D_{ui} is unlearned, it impacts completely and demonstrates eliminations from the global model, a task complicated by the distributed nature of Federated Learning.

4.2.1.3 Blockchain Integration

The introduction of a decentralized ledger L into the federated learning system presents a novel solution to the challenges of machine unlearning. This ledger records and verifies all unlearning requests and actions, denoted as R_{ui} for dataset D_{ui}. By leveraging Blockchain's inherent characteristics of immutability and transparency, we can ensure that each step in the unlearning process is permanently recorded and openly verifiable. This integration aims to maintain the integrity of the unlearning process, making it tamper-proof and transparent to all participants in the Federated Learning system.

4.2.1.4 Trust and Verification

Ensuring trust and verifiability in the machine unlearning process is imperative. The model must be designed to allow transparent verification that, once a data point D_{ui} is unlearned, its impact is completely absent from the model M. This requirement raises significant challenges in creating mechanisms for such verification that do not compromise data privacy or the integrity of the federated learning process. The solution

must balance the need for transparency with the fundamental requirements of data security and privacy intrinsic to federated learning.

In the following section, we will elaborate on the system model that addresses these challenges, describing the integrated operation of federated learning, machine unlearning, and blockchain technologies within our proposed framework.

4.2.1.5 Machine Unlearning in Blockchain

The concept of machine unlearning requires that a participant's contributions be removed from a trained model, ensuring that no traces of their data persist. However, blockchain operates on immutability, meaning that any recorded transactions or updates cannot be physically deleted. This raises an apparent contradiction: How can blockchain-based federated learning achieve unlearning while maintaining an immutable ledger?

To resolve this, we define practical machine unlearning in blockchain as follows:

> "Machine unlearning in blockchain does not require physical deletion of past transactions but instead ensures that a participant's contribution is **mathematically, cryptographically, and structurally nullified**, rendering it irrecoverable in the context of model updates."

Our proposed approach achieves this via the following techniques:

- **Blockchain Metadata Retention, but Off-Chain Data Erasure:** The blockchain records only metadata about model updates. The actual user data and model checkpoints are stored off-chain, allowing participant contributions to be fully erased without altering the ledger.
- **LoRA-based Unlearning for Model Removal:** By leveraging Low-Rank Adaptation (LoRA), we ensure that removing a participant's influence is equivalent to training a new model without them, while blockchain maintains integrity.
- **Revocation of Training Weights:** If a user requests unlearning, their corresponding weight updates in the federated model are revoked mathematically, ensuring that their data ceases to influence predictions.
- **Zero-Knowledge Proof (ZKP) for Verification:** Instead of modifying blockchain data, we generate a ZKP proof demonstrating that a participant's data no longer influences the model, even though their previous transactions remain on the blockchain.

This framework ensures that our system remains fully compliant with unlearning regulations while preserving the security and integrity of blockchain-based federated learning.

4.2.2 System Model

Our system model innovatively integrates machine unlearning with blockchain technology, utilizing specialized clients with smart contracts, this research manages the unlearning process efficiently and transparently within a secure blockchain network.. This model includes a machine unlearning mechanism, two types of clients for training and unlearning, smart contracts for process automation, and a blockchain network for secure, immutable record-keeping.

4.2.2.1 Clients

The system employs two types of clients:

- **Training Clients:** These clients manage routine training processes, including local model training on datasets and contributing to the overall learning and updating of the global model.
- **Unlearning Clients:** Dedicated to handling unlearning requests, these clients identify the data points D_{ui} to be unlearned and submit these requests. They are pivotal in ensuring regulatory compliance and maintaining privacy standards.

4.2.2.2 Smart Contracts

Smart Contracts play a crucial role in automating the Machine Unlearning process. Triggered by unlearning requests from unlearning clients, these self-executing contracts on the Blockchain network activate the Machine Unlearning mechanism. They meticulously log each unlearning request and action, adhering to predefined rules and conditions, thus introducing a level of trust and automation to the system.

4.2.2.3 Blockchain Network

Secure, transparent record-keeping is built on a Blockchain network foundation. It chronicles all machine unlearning activities, encompassing requests from unlearning clients and the corresponding actions executed by Smart Contracts. The network's

immutable and transparent nature guarantees a tamper-proof and verifiable record, augmenting the security and integrity of the machine unlearning process.

In summary, our system model is a comprehensive amalgamation of machine unlearning, specialized clients, smart contract automation, and blockchain technology. It adeptly tackles the challenges of managing the data unlearning process in a transparent and secure manner, ensuring the integrity and compliance of learning models in diverse applications.

4.2.2.4 Machine Unlearning Mechanism

Central to our system is the machine unlearning machanism, tasked with the precise removal of designated data points D_{ui} from the trained model M. Initiated by smart contracts, this mechanism ensures the thorough and efficient execution of the unlearning process, preserving the integrity and performance of the model after data removal.

4.3 Proposed System

4.3.1 Overview

This part introduces an innovative system that integrates machine unlearning with blockchain technology, aiming to enhance the trustworthiness, transparency, and compliance of learning models in various applications. This system is centered around a machine unlearning mechanism, which is tasked with the precise and efficient removal of specific data points D_{ui} from the trained model M, while maintaining the model's integrity and performance after unlearning.

A key feature of our system's architecture is the strategic integration of blockchain technology. This technology serves as a secure, transparent, and immutable platform for documenting all activities related to machine unlearning, encompassing unlearning requests, actions implemented, and their outcomes. The use of blockchain ensures that every phase of the unlearning process is permanently and verifiably recorded, thereby significantly boosting the system's accountability.

Moreover, the system utilizes specialized clients for distinct operational roles. Training clients are responsible for the regular training and updating of the model, whereas unlearning clients exclusively manage the initiation of unlearning requests. These clients work in coordination with the blockchain's smart contracts, which automate and regulate the machine unlearning process according to predefined protocols, thereby enhancing the

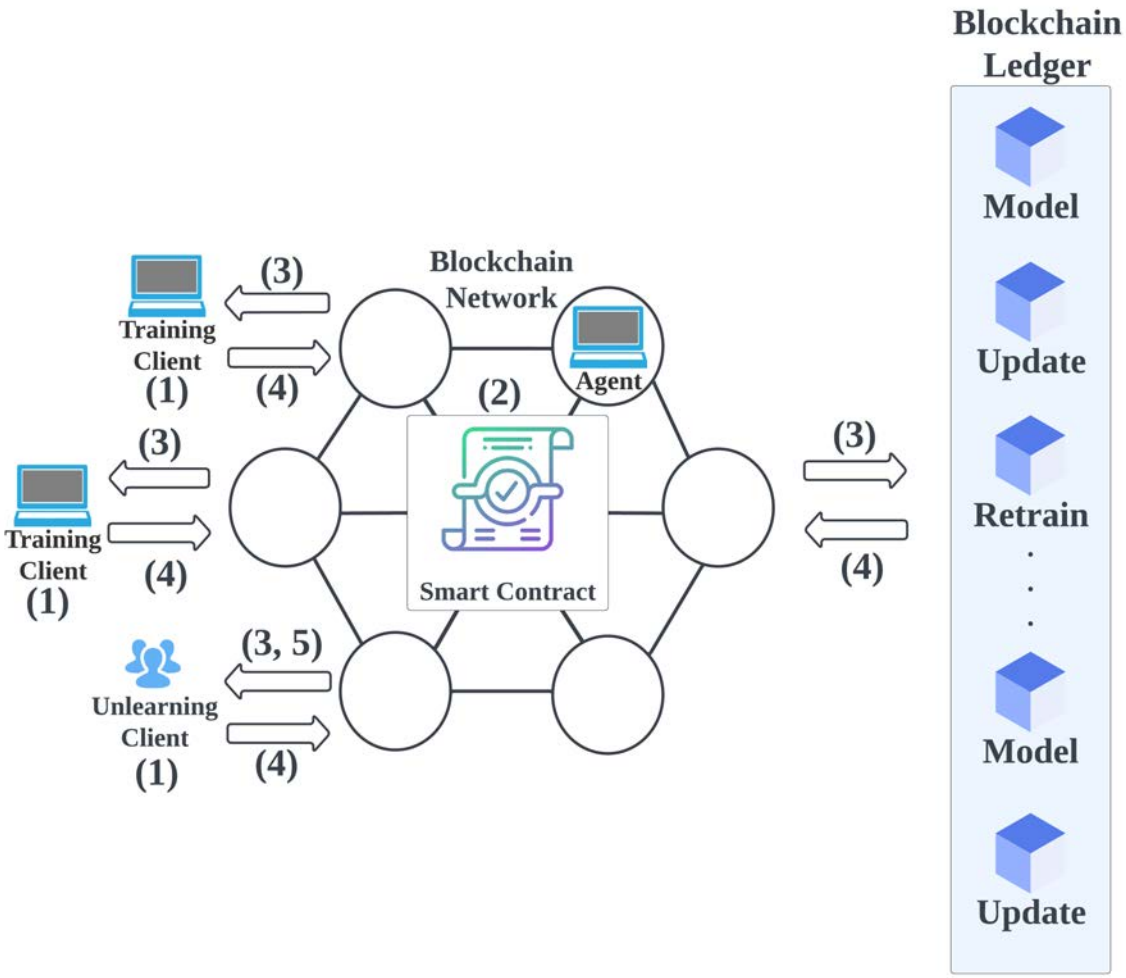

Figure 4.1: Overview and process of our proposed system. (1) Client and agent register. (2) Global model update. (3) Model updating process. (4) Model aggregation process. (5) Machine unlearning in blockchain network

process's consistency and dependability. The overview of our proposed system is shown in Figure 4.1.

4.3.2 Implementation of our designed system

4.3.2.1 Register

In this system, every *Client* and *Agent* is required to enroll in the blockchain network for participation. The registration process, outlined in Algorithm 4, is essential for integrating both *Client* and *Agent* into the network securely.

Algorithm 4 Client and Agent Register

Require: C_{id}, *Agent*
Ensure: *RegisterSucess, jwt*
1: *RegisterSucess* = *False*;
2: **if** $C_{id} \in U_{pool}$ **then**
3: **return** C_{id} already existed.
4: **end if**
5: **if** $A_{id} \in U_{pool}$ **then**
6: **return** A_{id} already existed.
7: **end if**
8: $(P_k, S_k) \leftarrow$*keyGenerator()*;
9: *jwt* = generateJWT(P_k, S_k);
10: C_{id}, *Agent* $\leftarrow$ *jwt* $\leftarrow$ SC;
11: $U_{\text{pool}} = U_{\text{pool}} \cup \{C_{\text{id}}, A_{\text{id}}\}$;
12: *RegisterSuccess* = *True*;
13: **return** *RegisterSucess, jwt*

In the proposed framework, the Algorithm 4 plays a pivotal role in the secure registration of clients and agents. Initially, the algorithm sets the registration success to false and checks whether the client (C_{id}) and agent (A_{id}) identifiers already exist in the user pool (U_{pool}). If an identifier is already present, the algorithm terminates, indicating duplication. For new identifiers, it employs a *keyGenerator* function to create a unique pair of keys (P_k and S_k), which are then used to generate a JSON Web Token (jwt). This jwt is crucial for secure information transmission and identity verification. Subsequently, the algorithm updates the user pool with the new identifiers, marking the successful registration of the entities. It concludes by returning the registration status and the generated jwt, ensuring a robust foundation for the secure operation of the system. This algorithmic approach effectively mitigates duplicate entries and maintains the integrity of the registration process, which is essential for the system's overall functionality.

4.3.2.2 Global Model Upload

Upon successful registration within the blockchain network, the *Agent* is enabled to upload the global model. Detailed in Algorithm 5, the Global Model Upload procedure facilitates the Agent in transferring a global model to the network. This algorithm requires inputs such as the Agent's *jwt* token and the global model ($Model_g$) intended for upload. Its outputs include an *UploadModel* success indicator and the global model (*Model*) that has been uploaded.

Algorithm 5 delineates the process for agents to upload models onto a blockchain

Algorithm 5 Global Model Upload

Require: A_{id},jwt,$Model$
Ensure: $UploadSuccess$, $Model_g$
1: $UploadSuccess$ = False
2: **if** *jwt token ineligibility* **then**
3: **return** Agent jwt token expired
4: **end if**
5: *Agent* sends $Model$ to SC
6: *SC* verifies and uploads the global model $Model$ to the blockchain network
7: $Model_g = Model$
8: $UploadSuccess$ = True
9: **return** $UploadSuccess, Model_g$

network. This procedure commences with the algorithm requiring the agent's identifier (A_{id}), a valid JSON Web Token (jwt), and the model intended for upload. Initially, the upload success is marked as false. The algorithm then verifies the jwt token's eligibility. On the other hand, while the token is expired or invalid, the process is halted with a notification of the jwt token's expiration. Upon successful verification, an agent transmits a model to a Smart Contract (SC), which validates and uploads it to a blockchain network. This model becomes the global model ($Model_g$). The algorithm then confirms the process completion by updating the upload success status. The algorithm finalizes the process by updating the upload success status and returns both this status and the global model, thus ensuring the secure and verified integration of models into the blockchain network.

4.3.2.3 Model Updating Process

Following the setup of the global model upload, configuring the epoch settings is essential. Algorithm 6, named "Model Updating Process," is designed to set up these parameters, specifically focusing on the required settings. The algorithm takes the epoch and batch size as input parameters and outputs a success indicator, *ModelUpdate*, along with the epoch parameter.

The process begins with setting the *ModelUpdate* flag to false. Clients then retrieve the global model ($Model_g$) for their training processes. The algorithm proceeds to establish the $epoch$ and $batchsize$ parameters in the Smart Contract (SC). The *epoch* setting will be adjusted to enhance system efficiency. During the training, for each epoch from 1 to n, clients upload their computed gradients to the SC. The algorithm also incorporates a check for the validity of the client's jwt token; if the token is found to be invalid, it returns a notification of the token's expiration and terminates. For each valid epoch,

Algorithm 6 Model Updating Process

Require: $epoch$, $batchsize$
Ensure: *gradient*
ModelUpdate = False;
Clients get the $Model_g$ and use for training process;
Set $epoch$ and $batchsize$ to SC
for $epoch$ = 1 to n **do**
    Client upload the *gradient* to SC;
    if *Client jwt token ineligibility* **then**
        return *Client jwt token* expired
    end if
    SC add DP to *gradient* to achieve *DP(gradient)*;
    SC publish the *DP(gradient)* in blockchain network;
end for
ModelUpdate = *True*;
return *ModelUpdate*

the SC adds *DP()* to received gradients and then publishes the DP(gradients) on the blockchain network. Once all epochs are processed, the *ModelUpdate* flag is true, it would show completion of the model updating process. This algorithm plays a vital role in ensuring the synchronization and consistency of training across clients.

4.3.2.4 Model Aggregation Process

The model aggregation process detailed in Algorithm 7, is a crucial component of the proposed system, focusing on the aggregation of model updates. It requires a JWT token and gradient inputs to perform model aggregation, indicated by *ModelAggregation*, and to update the global model ($Model_g$).

The procedure commences with setting the *ModelAggregation* flag to false. The agent, after retrieving gradients from the blockchain network. An expired token leads to the termination of the process with an indication of the agent's JWT expiration.

Once the it is verified as valid, then agent updates model using obtained gradients. This updated model is then sent to the Smart Contract (SC), which is responsible for uploading the model to the proposed framework. This algorithm then assigns this updated version to $Model_g$, signifying the new global model.

The process concludes with the algorithm setting the *ModelAggregation* flag to true, means successful aggregation. The output of this step algorithm 7 is the status of *ModelAggregation* and the updated global model ($Model_g$), playing a vital role in

Algorithm 7 Model Aggregation Process

Require: *jwt token*, $gradient$
Ensure: *ModelAggregation,* $Model_g$
1: *ModelAggregation* = False;
2: *Agent* get *gradient* for blockchain network;
3: **if** *jwt token ineligibility* **then**
4: **return** *Agent jwt token* expired
5: **end if**
6: $Model \leftarrow Model_g$;
7: Agent update the *Model* and send the new *Model* to SC;
8: SC upload the *Model* to blockchain network;
9: $Model_g \leftarrow Model$;
10: *ModelAggregation* = *True*;
11: **return** *ModelAggregation,* $Model_g$

integrating individual model updates into a cohesive global model and enhancing the system's overall learning efficacy.

4.3.2.5 Machine Unlearning in Blockchain Network

Algorithm 8 is designed to facilitate the removal of personal data from a trained model in a blockchain network. This algorithm requires the client's identifier (C_{id}) and ensures the successful execution of the unlearning process, denoted as *Unlearning*.

Algorithm 8 Machine Unlearning in Blockchain Network

Require: C_{id}
Ensure: *Unlearning*
1: *Unlearning* = False;
2: Client wants to do machine unlearning to unlearn personal data;
3: A machine unlearning request sent to SC;
4: **if** *jwt token ineligibility* **then**
5: **return** C_{id} *jwt token* expired
6: **end if**
7: SC find the first *epoch* before this client upload the *gradient*;
8: Agent update the *Model* according to SC's requirement and upload the new model *Model* to SC;
9: $Model \leftarrow Model_g$;
10: SC send the new model $Model_g$ to other clients;
11: Other clients continue the training process according to the new model $Model_g$;
12: **return** *Unlearning* = True;

The process begins by setting the *Unlearning* flag to false. The client, intending to

execute machine unlearning to remove personal data, sends a request to the Smart Contract (SC). A key step in this process is the verification of the client's JWT token. In other words, if the token is expired or invalid, the algorithm returns a message indicating the expiration of the C_{id}'s JWT token and halts the process.

Once the token's validity is confirmed, the SC identifies the first epoch before the client uploaded their gradient. Following this, the agent is tasked with updating the model based on SC's requirements. The updated model is then uploaded back to the SC. This updated model replaces the global model ($Model_g$), and the SC subsequently disseminates the new global model to other clients.

Other clients in the network continue their training process using this updated model, ensuring that the unlearned data is no longer part of the training process. The algorithm concludes by setting the *Unlearning* flag to true.

This algorithm is pivotal in maintaining the privacy and security of personal data within a blockchain-based federated learning environment, allowing for the dynamic removal of data from the trained model as required.

4.3.3 Case Study: Smart Healthcare Monitoring System Using Blockchain-Enhanced Federated Learning

4.3.3.1 Background

In the context of a smart healthcare monitoring system, an application of IoT, federated learning is employed to develop collaborative models across various devices. This system faces a critical need for machine unlearning, particularly when patients revoke consent or request data corrections.

4.3.3.2 Implementation of the System

The implementation of the system, in conjunction with our proposed method, is delineated as follows:

- **Registration:** Patients and healthcare providers, acting as clients and agents in the system, enroll through the blockchain network. This secure registration process, detailed in Algorithm 1, is vital for their participation and guarantees data integrity.
- **Global Model Upload:** Following registration, healthcare providers (agents) securely update model to the blockchain network, as described in proposed method.

This step ensures that all participating devices are synchronized with the most current and accurate model.

- **Model Updating Process:** Continuously collecting patients' health data necessitates regular model updates. Algorithm 3 defines the parameters like epoch and batch size for the training process, maintaining the model's relevance with the latest data. Additionally, Secure Computing (SC) incorporates Differential Privacy (DP) automatically, reinforcing the system's security and privacy.

- **Model Aggregation Process:** Individual model updates from diverse IoT devices are consolidated into an updated global model, as outlined in Algorithm 4. This aggregation is crucial for a unified and comprehensive understanding of patient health trends across the network.

- **Machine Unlearning in Blockchain Network:** Algorithm 5 enables the removal of a patient's personal data from the trained model when they opt to withdraw their data. This step is imperative for adhering to privacy requests and maintaining the system's trustworthiness.

4.3.3.3 Analysis

The integration of blockchain not only facilitated secure and verifiable data unlearning but also preserved the overall system's efficiency. The healthcare monitoring system adeptly adjusted to changes in patient data while safeguarding data privacy and maintaining system integrity.

4.3.3.4 Challenges and Solutions

The primary challenges involved optimizing the efficiency of the blockchain network and balancing security with privacy. These challenges were tackled by optimizing the federated learning training process's epoch aggregation settings and implementing Differential Privacy (DP) methods to secure the updating data.

4.3.3.5 Conclusion

This case study exemplifies the efficacy of integrating blockchain with federated learning in a practical IoT application. Our proposed framework significantly enhances data privacy, upholds the integrity of the learning model, and provides a verifiable, efficient

mechanism for machine unlearning, which is paramount in sensitive domains like healthcare monitoring.

4.4 Privacy and security analysis

4.4.1 Privacy analysis

Our innovative framework integrating machine unlearning with blockchain in a federated learning context significantly bolsters data privacy. This section delves into the privacy benefits and mechanisms of our system.

Data Localization in Federated Learning: A cornerstone of our framework is the localization of data on client devices. This approach inherently minimizes privacy risks commonly associated with centralized data storage. By keeping sensitive data on the client side and only sharing model updates, the framework upholds privacy by design.

Machine Unlearning for Privacy Enhancement: Machine unlearning is pivotal for privacy protection in our system. It allows for the precise and complete removal of data from the model upon user request or policy changes. This ensures that any data, once deemed unnecessary or sensitive, is effectively erased from the learning model, safeguarding user privacy.

Blockchain as a Catalyst for Privacy: Blockchain technology plays a crucial role in enhancing privacy in our framework. Its immutable ledger records all data transactions (including unlearning requests and actions) permanently. This transparency offers a robust audit trail, which is essential for accountability, yet it does not compromise data privacy, as the ledger only records transactional metadata, not the data itself.

Smart Contracts for Privacy Assurance: Our system employs smart contracts to automate and enforce privacy policies. These contracts guarantee that unlearning requests are executed efficiently and accurately, thereby maintaining the system's privacy integrity. Automation reduces the risk of human errors and biases, further strengthening privacy safeguards.

Addressing Privacy in IoT Contexts: The IoT environment, characterized by the vast and varied data from numerous devices, presents unique privacy challenges. Our framework is designed to manage these challenges by processing IoT-generated data in a manner that prioritizes user privacy. It ensures that any data removal or unlearning request is thoroughly and promptly implemented.

In summary, our proposed system markedly advances privacy protection in feder-

ated learning settings. By synergizing federated learning, machine unlearning, and blockchain, the framework adeptly navigates the complex landscape of data privacy, particularly in IoT scenarios. It sets a new standard for future developments in balancing sophisticated data processing with the paramount need to protect individual privacy.

4.4.2 Security analysis

In addition to enhancing privacy, our integrated system of machine unlearning with blockchain in a federated learning environment also significantly strengthens security. This section examines the key security aspects of our proposed framework.

Robust Data Security in Federated Learning: The federated learning aspect of our system ensures that data remains decentralized, greatly reducing the risk of large-scale data breaches common in centralized systems. By distributing the data across multiple nodes, the system inherently disperses security risks, making it more resilient to targeted attacks.

Immutable Record-Keeping with Blockchain: The implementation of blockchain provides an additional layer of security. Its immutable ledger ensures that every transaction, including the addition and removal of data, is permanently recorded. This immutability makes it virtually impossible to tamper with the data history, ensuring the integrity of the entire learning model.

Smart Contract-Driven Security Protocols: Our system leverages smart contracts to enforce security protocols automatically. These contracts are programmed to execute unlearning requests securely and ensure compliance with predefined security standards. This automation minimizes human intervention, thereby reducing the potential for errors and security lapses.

Enhanced Security through Machine Unlearning: Machine unlearning contributes to the system's security by ensuring that any data no longer needed or deemed sensitive can be removed promptly and completely. This not only protects against unauthorized access to outdated or irrelevant data but also reduces the 'attack surface' that hackers could exploit.

Security Challenges in IoT Environments: The diversity and scale of IoT networks present unique security challenges. Our framework addresses these by ensuring secure data handling and model training across various devices. The blockchain component adds a trust layer, verifying and recording each action taken within the network, thus safeguarding against malicious activities.

Resilience Against Data Tampering and Attacks: The combination of blockchain technology and federated learning creates a formidable barrier against data tampering and cyber-attacks. Blockchain's distributed nature makes it difficult for attackers to compromise the system, while federated learning limits the exposure of sensitive data.

In conclusion, our proposed framework presents a comprehensive approach to enhancing security in federated learning environments. By integrating machine unlearning and blockchain technology, it addresses critical security concerns, particularly relevant in the context of IoT. This framework not only ensures the secure handling of data but also builds trust among participants, paving the way for more secure and reliable data-driven applications.

4.5 Results and Analysis

4.5.1 MNIST Dataset Results

The MNIST dataset, comprising handwritten digits, served as our primary benchmark for evaluation. This dataset, known for its simplicity and clear learning tasks, was ideal for assessing the accuracy and efficiency of our machine unlearning process. We specifically focused on how unlearning requests affected the model's accuracy and training efficiency, observing the system's behavior under various volumes and frequencies of these requests.

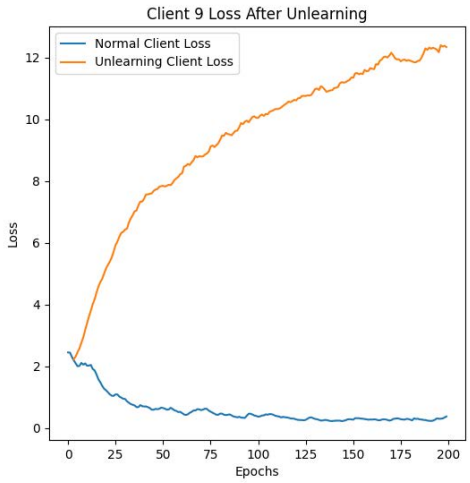

Figure 4.2: Specific Client loss comparison in 200 epochs

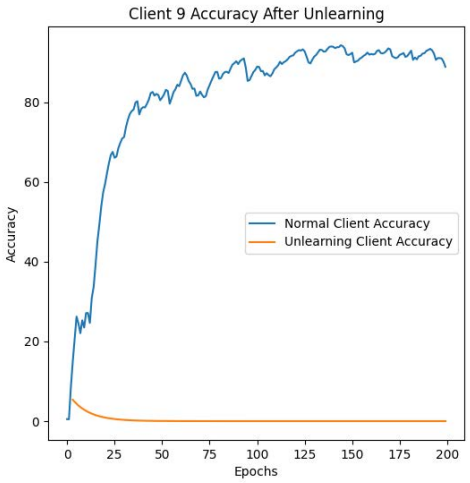

Figure 4.3: Specific Client accuracy comparison in 200 epochs

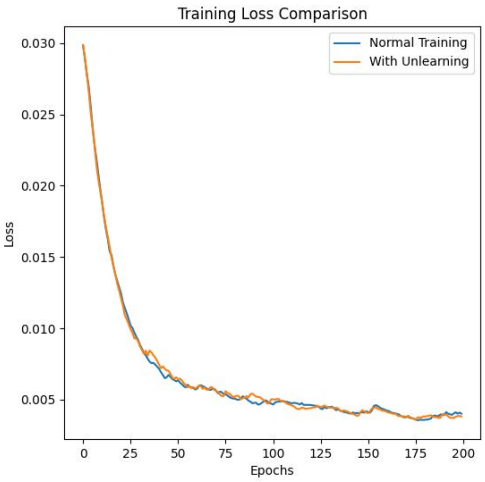

Figure 4.4: Loss comparison of normal learning and with unlearning in 200 epochs

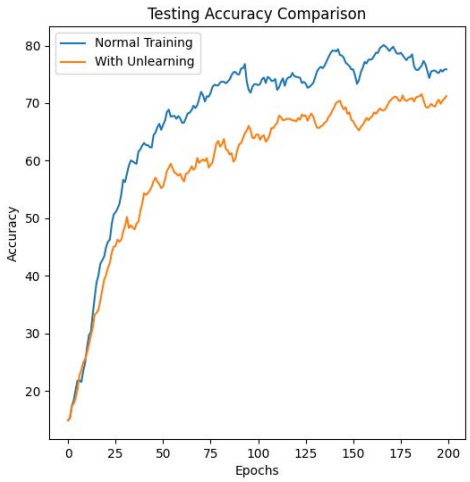

Figure 4.5: Testing accuracy comparison of normal learning and with unlearning in 200 epochs

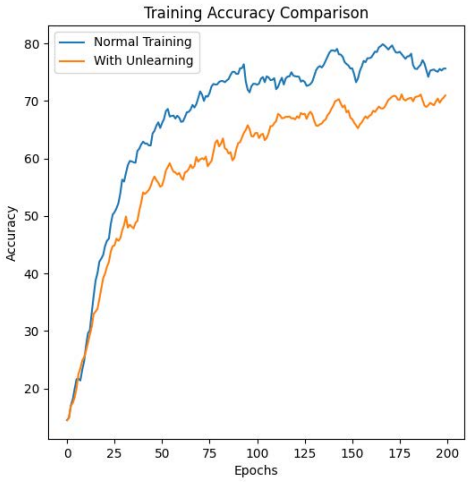

Figure 4.6: Training accuracy comparison of normal learning and with unlearning in 200 epochs

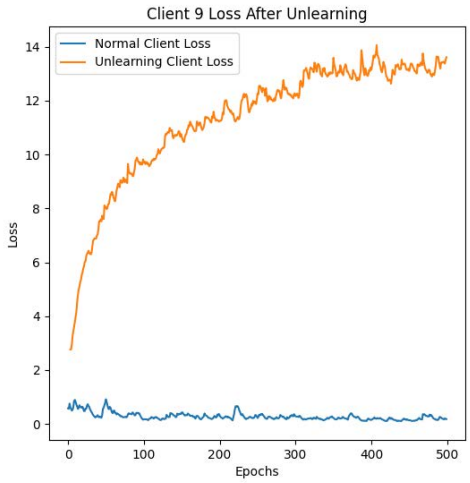

Figure 4.7: Specific Client loss comparison in 500 epochs

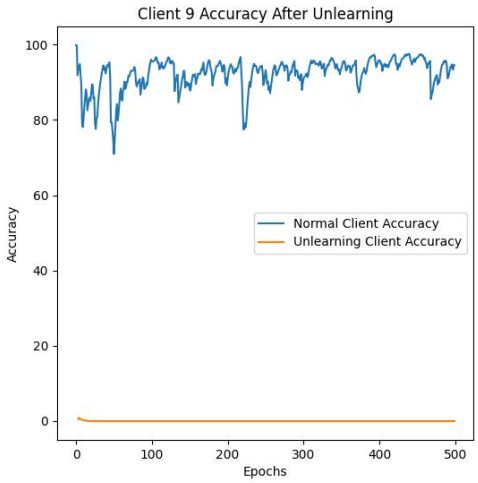

Figure 4.8: Specific Client accuracy comparison in 500 epochs

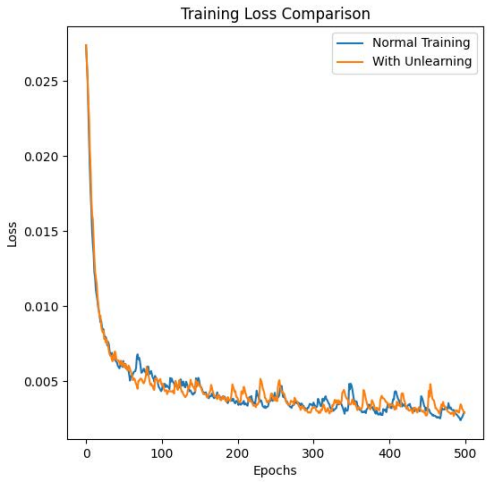

Figure 4.9: Loss comparison of normal learning and with unlearning in 500 epochs

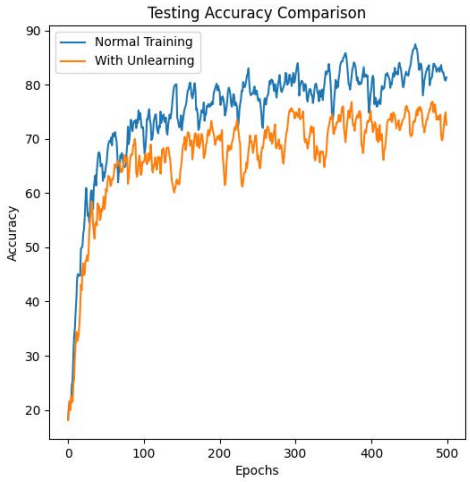

Figure 4.10: Testing accuracy comparison of normal learning and with unlearning in 500 epochs

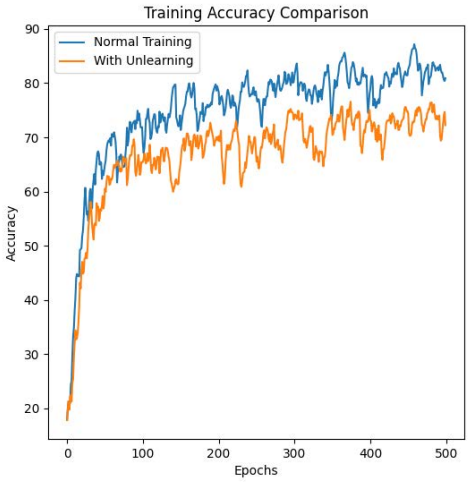

Figure 4.11: Training accuracy comparison of normal learning and with unlearning in 500 epochs

In evaluating the MNIST dataset's performance, we closely analyzed the model's response during and after the unlearning process. As illustrated in Figure 4.2, we compared loss trends between normal training and post-unlearning phases. During the training epochs, the loss trajectory for normal training consistently remained lower than that of the unlearning phase. This pattern began with a sharp decline in the early epochs, followed by stabilization over time. Notably, while there was a temporary increase in loss for the unlearning client, this eventually leveled off, approximating the normal client loss, albeit at a slightly higher value.

Figure 4.3 compares the accuracy during normal and unlearning phases for the same client. The accuracy in normal training quickly reached a high plateau, with minor fluctuations, indicative of a stable and effective learning process. In contrast, the accuracy for the client post-unlearning started at zero and remained constant, reflecting the targeted removal of class information from the model.

The overall training loss, depicted in Figure 4.4, highlighted the system's resilience. Both normal training and training with unlearning converged to a loss value close to zero, indicating that unlearning did not compromise the model's ability to learn from the remaining dataset. Similarly, Figures 4.5 and 4.6 displayed minimal difference in accuracy between normal training and training with unlearning. Both scenarios showed a sharp initial increase in accuracy, followed by closely aligned progression, suggesting that the unlearning process did not significantly affect the overall model accuracy. The most definitive proof of our system's unlearning accuracy is presented in Figure 4.12. Here, the accuracy for Class 0 post-unlearning was precisely zero, confirming the complete and effective removal of this class from the model's knowledge. For normal training, the average accuracy across all classes was 80.23%, while it slightly reduced to 71.35% post-unlearning. This decrease was mainly due to the targeted unlearning in Class 0, as the accuracies for other classes remained largely unaffected, demonstrating the specificity of the unlearning process.

These results underscore our system's ability to perform machine unlearning with high specificity and minimal impact on overall model performance. The precision of the unlearning process, along with the retained efficacy for the remaining classes, emphasizes the system's applicability in real-world machine learning scenarios where data privacy and the right to be forgotten are critical considerations.

Extending our evaluation to 500 epochs on the MNIST dataset, we derived insightful conclusions:

Figures 4.7 and 4.8, representing 'Client 9 Loss' and 'Client 9 Accuracy' post-unlearning,

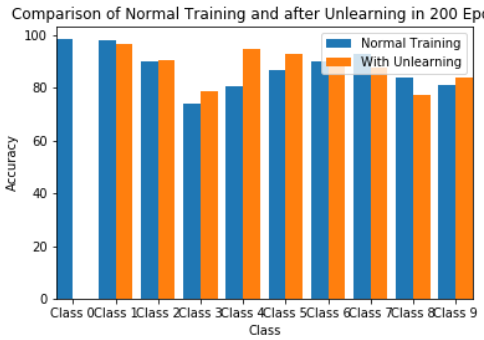

Figure 4.12: Class comparison in 200 epochs

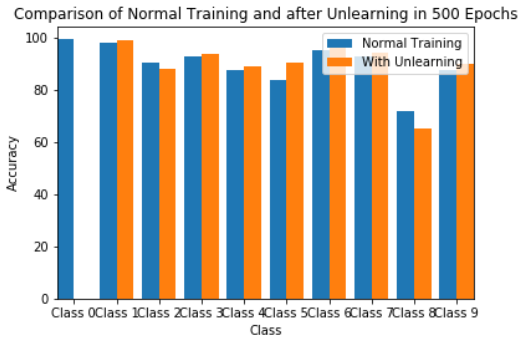

Figure 4.13: Class comparison in 500 epochs

demonstrate the success of the unlearning process. In Figure 4.7, the normal client loss remains low and stable, contrasting with the unlearning client loss, which, after initial stability, gradually increases. This divergence distinctly illustrates the impact of unlearning over extended training.

Figure 4.8 shows a significant difference in accuracy. While the normal client's accuracy consistently exceeds 80%, indicating stable model performance over time, the unlearning client's accuracy, initially parallel, plummets to near zero. This stark decline exemplifies the effective elimination of the learned data points.

From a broader view, Figures 4.9 and 4.10, showcasing 'Training Loss Comparison' and 'Testing Accuracy Comparison,' reveal a slight divergence between normal training and training with unlearning. Both figures indicate a marginally increased loss and reduced accuracy in training with unlearning, particularly in later stages. This suggests a cumulative effect of unlearning over time. Nonetheless, the overall loss remains low and the accuracy high, emphasizing the system's resilience.

Figure 4.11 reinforces the observation of sustained high accuracy levels throughout the epochs. The unlearning curve, while slightly lower in accuracy, aligns with the testing accuracy trends. The most evident unlearning effects are depicted in Figure 4.13. Normal training maintains a high accuracy of 87.11%, whereas post-unlearning accuracy drops to 77.74%. This decrease is most notable in Class 0 accuracy, which falls to zero, confirming the targeted and successful unlearning. The minor decrease in accuracy for other classes, likely due to the unlearning of Class 0, does not significantly impact the overall model performance.

In conclusion, these findings validate the precision of our unlearning process within a federated learning context. The system skillfully unlearns specific data points without

undermining the model's overall learning capacity, essential for maintaining privacy and compliance with data regulations. The slight overall accuracy reduction post-unlearning highlights the unlearning impact while affirming the system's capability to maintain robust performance across the remaining dataset.

4.5.2 CIFAR-10 Dataset Results

The CIFAR-10 dataset, comprising 60,000 color images across 10 classes, provided a more complex test environment. Utilizing this dataset, we evaluated the system's ability to handle intricate image data and assessed the scalability of our machine unlearning process. We focused on examining the model's resilience and adaptability to unlearning requests, while maintaining accuracy and learning efficiency in this intricate scenario.

Upon extending our experiments to include the CIFAR-10 dataset over 2000 epochs, we observed several key dynamics in model performance:

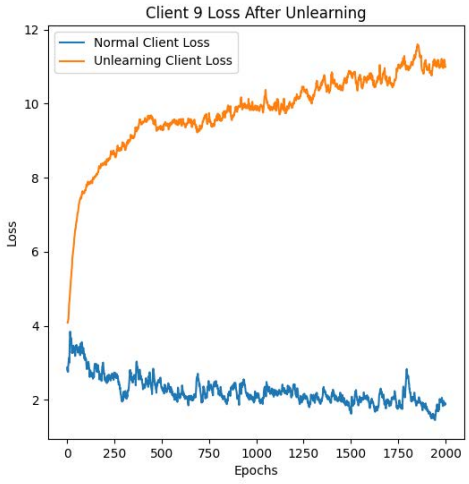

Figure 4.14: Specific Client loss comparison in 2000 epochs

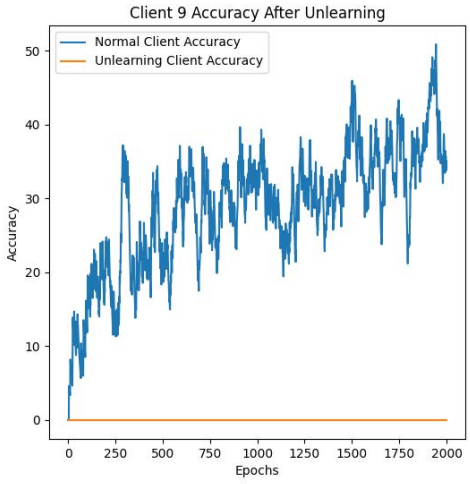

Figure 4.15: Specific Client accuracy comparison in 2000 epochs

Figure 4.14 illustrates the loss trajectory for the normal client, beginning with a rapid decline indicative of effective initial learning, then stabilizing at a lower value. In contrast, the unlearning client exhibited a moderate increase in loss over time. This pattern suggests that, despite a slight performance degradation, the unlearning process does not inhibit the client's ongoing learning capacity.

In Figure 4.15, a significant disparity in accuracy is evident. The normal client's accuracy fluctuates around 40%, reflecting the typical behaviour of a converging learning

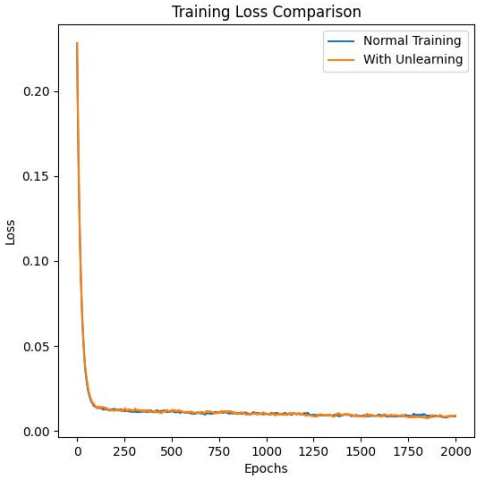

Figure 4.16: Loss comparison of normal learning and with unlearning in 2000 epochs

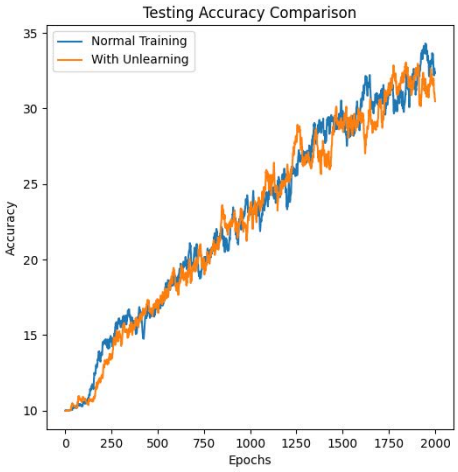

Figure 4.17: Testing accuracy comparison of normal learning and with unlearning in 2000 epochs

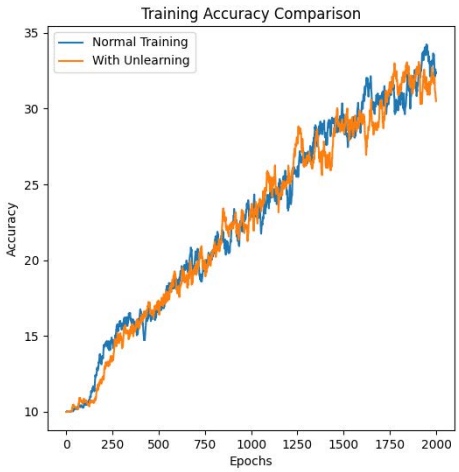

Figure 4.18: Training accuracy comparison of normal learning and with unlearning in 2000 epochs

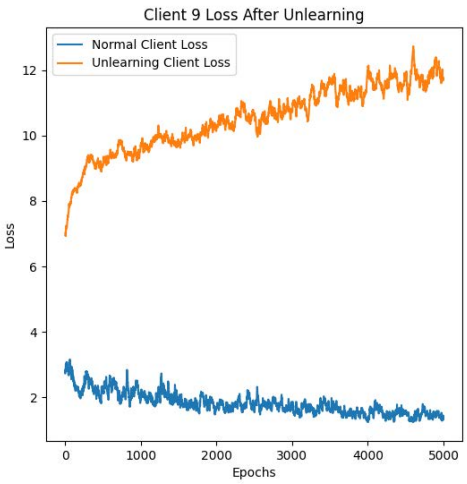

Figure 4.19: Specific Client loss comparison in 5000 epochs

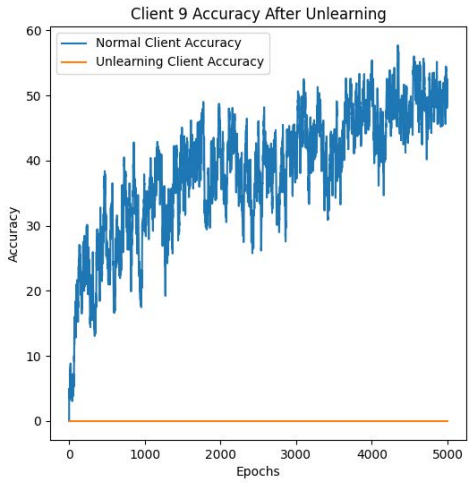

Figure 4.20: Specific Client accuracy comparison in 5000 epochs

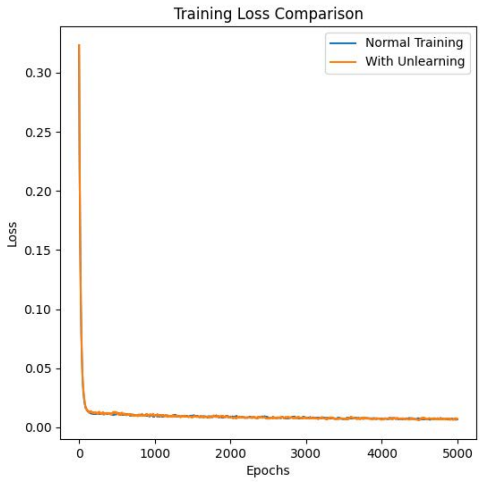

Figure 4.21: Loss comparison of normal learning and with unlearning in 5000 epochs

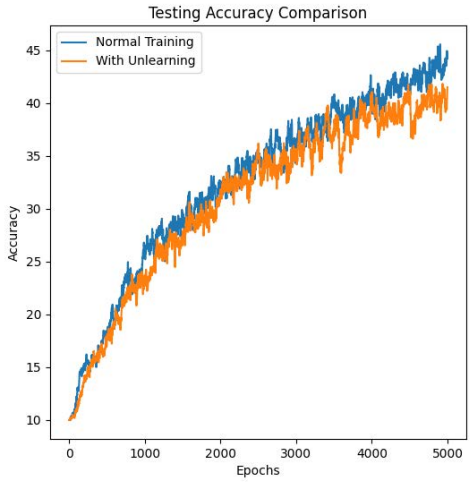

Figure 4.22: Testing accuracy comparison of normal learning and with unlearning in 5000 epochs

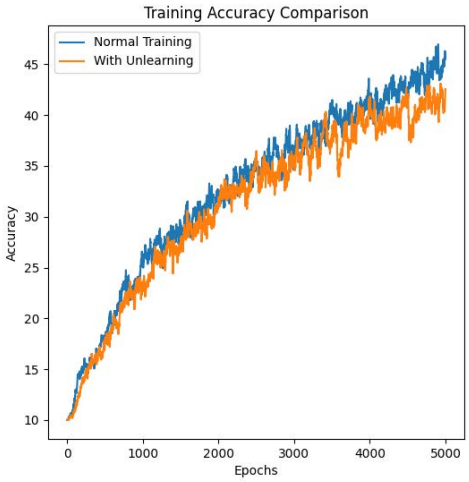

Figure 4.23: Training accuracy comparison of normal learning and with unlearning in 5000 epochs

model. Conversely, the unlearning client's accuracy quickly falls to nearly zero and remains consistently low, highlighting the unlearning mechanism's effectiveness.

Figure 4.16 shows that the system maintains its learning ability post-unlearning, with both normal training and training with unlearning converging to a minimal loss. Similarly, Figure 4.17 indicates that while accuracy in the unlearning scenario is slightly reduced, it tracks closely with normal training, implying robust overall model performance.

Figure 4.18 supports these observations, displaying a consistent gap between normal training and training with unlearning. Notably, the model trained with unlearning still achieves high accuracy, validating the unlearning process's precision.

The class-wise accuracy comparison in Figure 4.24 offers compelling evidence of the unlearning process's impact. Post-unlearning, the accuracy for Class 0 drops to zero, showing the model's complete inability to recognize the unlearned class, clearly evidencing the success of the unlearning process. The average accuracy across all classes slightly decreases from 34.12% to 33.21% post-unlearning, with the most significant reduction in the targeted class. This outcome illustrates the system's ability to selectively unlearn specific information while retaining general knowledge, an essential aspect for practical applications requiring selective data removal.

In conclusion, the evaluation using the CIFAR-10 dataset over an extended timeframe highlights the system's effectiveness in executing precise machine unlearning. The marginal overall accuracy decline post-unlearning underscores the unlearning process's targeted nature, ensuring that the model retains its learning capacity and maintains the integrity of its performance across the remaining classes.

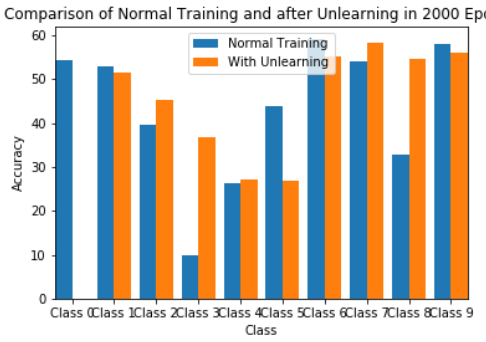

Figure 4.24: Class comparison in 2000 epochs

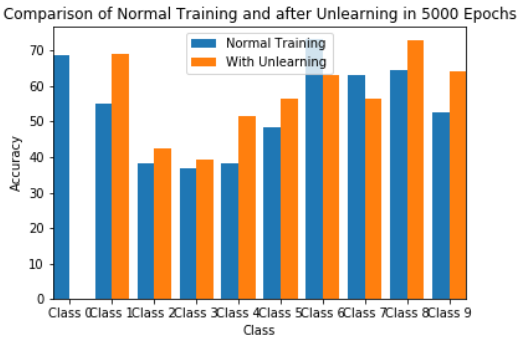

Figure 4.25: Class comparison in 5000 epochs

Extending our CIFAR-10 dataset analysis over 5000 epochs, we gained valuable

insights into the model's adaptability and the unlearning process's efficacy:

In Figure 4.19, we observed the loss patterns for the normal and unlearning clients. The normal client's loss showed a steep initial decline, signalling effective learning, and then stabilized at a low level, indicating a consistent learning pattern. Conversely, the unlearning client's loss followed a similar initial trend but gradually increased, eventually stabilizing at a higher level than the normal client. This pattern suggests that, although the unlearning client continues to learn, the unlearning process introduces a slight but noticeable alteration in its loss trajectory.

Figure 4.20 illustrates the accuracy of the normal client, which fluctuates within an acceptable range and peaks around 50%. For the unlearning client, the accuracy initially matches the normal client but then rapidly declines to zero. This dramatic drop demonstrates the successful elimination of the impact of the unlearned data from the model.

In Figure 4.21, the trajectories for both normal training and training with unlearning initially exhibit a sharp decline. However, as the epochs progress, the curves converge closely, suggesting that the unlearning process does not significantly impair the model's ability to learn from the remaining data.

Figure 4.22 shows the accuracy trends for normal training and training with unlearning. Both follow a parallel ascent, with the latter maintaining a slightly lower accuracy level. This indicates that the model continues to generalize effectively post-unlearning, albeit with a minor decrease in accuracy.

Figure 4.23 aligns with the testing accuracy trends, where training with unlearning shows a consistent but modest divergence from normal training. The slight decrease in accuracy after unlearning, though evident, does not significantly detract from the model's performance, highlighting the unlearning process's precision. The class-wise accuracy comparison in Figure 4.25 starkly demonstrates the unlearning impact. While normal training achieves an average accuracy of 45.92%, post-unlearning accuracy drops to 42.65%. Notably, the accuracy for Class 0 is zero post-unlearning, confirming the effective and specific execution of the unlearning. The other classes show only minor reductions in accuracy, consistent with the unlearning's targeted nature.

Overall, the extended analysis over 5000 epochs confirms the system's effectiveness in executing machine unlearning. The approach significantly reduces the model's accuracy for the specified unlearned class while minimally impacting the overall performance. This capability underscores our system's utility in scenarios where data privacy and regulatory compliance are crucial, facilitating selective unlearning as needed.

4.5.3 Blockchain Complexing Results

In our study, we thoroughly examined the blockchain component of our system, focusing on aspects such as scalability, transaction throughput, and latency due to blockchain integration in the federated learning process. Our objective was to assess the impact of these factors on the overall system performance, ensuring that blockchain integration does not compromise the efficiency or effectiveness of the machine unlearning process.

We utilized Hyperledger Fabric 2.0 to evaluate the impact of the blockchain network on our machine unlearning process's performance, particularly in IoT applications where additional computational overhead is a concern.

- **Blockchain Network Initialization**: The initial setup time for the blockchain network was approximately 35 seconds. This one-time overhead is deemed acceptable, especially considering the long-term benefits in federated learning applications where rapid deployment is essential.
- **Consensus Mechanism Overhead**: The time required for the consensus process varied based on the chosen consensus algorithm. After setting up the blockchain network, our consensus algorithm, requiring approval from all nodes, added around 3 seconds. This duration is reasonable and manageable within our IoT context involving federated learning.
- **Transaction Processing Efficiency**: The average processing time encompasses gradient aggregation and other related tasks, was 2 seconds. This efficiency in transaction handling by Hyperledger Fabric highlights its capability to manage additional blockchain-related tasks effectively.
- **Per-Epoch Time Cost**: During training, the duration per epoch, both for normal training and post-unlearning operations, remained consistent at 24-26 seconds, illustrating the system's stable performance irrespective of unlearning activities.

Table 4.1 presents a comparison of time costs between a standard federated learning cycle and our proposed blockchain-enhanced method. Initially, our method incurs a higher time cost due to setup and endorsement processes. However, this cost tends to normalize with increasing iterations, indicating promising scalability.

Overall, our results confirm that blockchain technology can be practically and scalably integrated into federated learning frameworks with unlearning capabilities. The additional time cost is balanced by the enhanced security and trustworthiness in the machine

Table 4.1: Time Cost Analysis for Federated Learning with and without Blockchain Integration over 1999 Iterations

Method	**t = 0**	**t = 9**	**t = 199**	**t = 1999**
Normal-Basic Federated Learning	26s	260s	5200s	26000s
Our Proposed System	66s	318s	5638s	28038s

unlearning process, highlighting the suitability of our approach for IoT applications where these features are paramount.

4.5.4 Analysis Conclusion

Drawing on the empirical results from the MNIST and CIFAR-10 datasets, along with the blockchain time complexity analysis, we can draw a cohesive conclusion:

Our system, which integrates machine unlearning into federated learning environments and supplements it with blockchain technology, has proven to be both effective and efficient. The evaluation using the MNIST dataset demonstrates that the unlearning process is executed with precision, having minimal impact on the overall model performance while ensuring the targeted data's removal. This finding is supported by the results from the CIFAR-10 dataset, showing that even in more complex scenarios, the system successfully unlearns specific classes without significantly affecting the accuracy of other classes.

In extended trials over thousands of epochs with the CIFAR-10 dataset, the system consistently exhibited its capability to execute machine unlearning precisely while maintaining high model accuracy. The slight reduction in accuracy post-unlearning highlights the targeted impact of the unlearning process, critical in scenarios prioritizing data privacy and the right to be forgotten.

From the perspective of blockchain complexity, our study utilized the Hyperledger Fabric 2.0 platform to evaluate the impact of blockchain integration on machine unlearning performance. Our analysis indicated that the initial time overhead is offset by the blockchain's contributions to security and trustworthiness. The blockchain's scalability is evident, as demonstrated by the normalization of time costs with increased iterations, reinforcing the system's suitability for long-term applications.

The blockchain time complexity results confirm that the blockchain network's overhead is manageable and does not significantly impact the federated learning process.

The initial setup time for the blockchain network and the latency due to the consensus mechanism are acceptable, particularly given the enhanced security and immutability benefits that blockchain brings.

In summary, our proposed system emerges as a scalable, secure, and efficient solution for federated learning, particularly in IoT scenarios. It effectively addresses the critical challenge of integrating machine unlearning without sacrificing learning performance or scalability. The inclusion of a blockchain layer augments trust and security, making our approach highly relevant for applications where these aspects are crucial. The comprehensive analyses across different datasets, coupled with an in-depth study of blockchain time complexity, collectively affirm the practicality and viability of our blockchain-enhanced federated learning system.

4.6 Summary

This work introduces an innovative method that integrates machine unlearning with blockchain technology in federated learning, specifically designed for IoT applications. Extensive experiments using the MNIST and CIFAR-10 datasets have shown the system's efficiency in executing machine unlearning with minimal effect on overall model performance. This underscores the effectiveness of our methodology in balancing precision in unlearning with maintaining overall model accuracy.

Additionally, the blockchain component, implemented using Hyperledger Fabric 2.0, has undergone a detailed time complexity analysis. The complexity analysis has highlighted the practicality of integrating blockchain into federated learning systems. Despite introducing a reasonable overhead, the incorporation of blockchain enhances the security and trustworthiness of the system, crucial for IoT applications.

Looking to the future, this chapter trajectory includes several ambitious goals: enhancing the efficiency of the blockchain component, extending our approach to more complex datasets, exploring advanced consensus mechanisms for blockchain, integrating differential privacy techniques for enhanced data security, probing into cross-chain interoperability to broaden the scope of our system, and evaluating the system's adaptability to ever-evolving data regulations. These endeavours aim to continuously refine and adapt our system, ensuring its applicability and relevance in the dynamic domain of secure, distributed machine learning in IoT and federated learning applications. Through sustained research and development efforts, we aspire to keep pace with the rapid advancements in this field and contribute to its growth and innovation.

CHAPTER 5

FEDERATED TRUSTCHAIN: BLOCKCHAIN-ENHANCED LLM TRAINING AND UNLEARNING

5.1 Introduction

The evolution of Large Language Models (LLMs) marks the beginning of a new era in artificial intelligence, significantly altering how we interact with and utilize machine learning [12, 31]. As these models progress, a significant challenge becomes apparent: by 2030, publicly available data sources are expected to be insufficient to support the continued growth and development of LLMs [66]. Therefore, the use of private data becomes crucial, not only to sustain development but also as an essential resource for LLMs to access.

With this demand, a significant challenge persists: data owners, aware of the value of LLMs, are hesitant to share their private data because of privacy concerns. At present, individuals have the option to download models and train them on their own private datasets. This method, however, leads to the development of isolated models. These models lack synergy and do not benefit from interconnected learning among various LLMs, underscoring the need for a more cohesive strategy to efficiently utilize private data.

Federated learning emerges as a prominent solution to address the pressing requirement for private data to enhance LLMs [17]. This method of collaborative machine learning enables the training of a model on multiple decentralized devices or servers,

each of which holds a portion of the entire dataset [91]. This approach guarantees that confidential information remains on the owner's device, eliminating the need to distribute or consolidate data, thus directly addressing privacy concerns.

However, merging federated learning with LLMs presents a series of new challenges. One major concern is the lack of transparency in the federated learning process when combined with LLMs. The decentralized nature of federated learning makes it difficult to track and verify the contributions of each participating model, as well as to ensure that the collective learning process is not negatively impacted by suboptimal or compromised models. Additionally, the need for effective unlearning mechanisms becomes crucial in this context, as data owners may wish to remove their data from the training process while minimizing the impact on other participants [11].

To address these challenges and enhance the transparency and accountability of federated learning in LLM training, we propose the integration of blockchain technology. Blockchain's immutable and distributed ledger provides a secure and transparent record of all transactions and interactions within the federated learning process [72]. By leveraging blockchain, we can create a tamper-proof record of each model's contributions, facilitating the identification and removal of problematic models without disrupting the overall learning process.

Furthermore, blockchain enables the implementation of effective unlearning mechanisms, ensuring that data owners can remove their data from the training process while maintaining the integrity of the collective model. Through these dedicated efforts, we introduce an innovative solution that utilizes blockchain technology's strengths to overcome the intricate challenges of training LLMs with private data within a federated learning framework. Our approach represents a substantial step forward in achieving a secure, efficient, and transparent methodology for integrating private data into LLM development.

In addressing the challenges previously outlined, this chapter offers three significant contributions, each targeting a key aspect of merging federated learning with LLMs via blockchain technology:

- We present a blockchain-based architecture meticulously documenting every facet of the federated learning training process. This architecture is crucial for facilitating effective unlearning, as it provides a detailed and unchangeable record of all training actions, ensuring transparency and verifiability at every step.

- We introduce an unlearning function within this blockchain environment. This

feature is designed to seamlessly integrate with the federated learning mechanism, enabling the targeted removal of specific models or data while preserving the integrity of the wider learning system. Its deployment is vital for upholding the federated learning framework's integrity and effectiveness, allowing it to dynamically respond to changing data privacy requirements.

- Our approach strengthens the accountability and verification process by methodically recording unlearning actions on the blockchain. This procedure is essential for evaluating the unlearning process's success.

5.2 Problem Definition and System Model

5.2.1 Problem Definition

The integration of Large Language Models (LLMs) with federated learning, supported by a blockchain framework, introduces distinct aims that require a precise problem definition. These aims arise from the complexities of managing private data, ensuring model integrity, and implementing efficient unlearning processes. We formalize these aims as follows:

1. **Data Privacy and Model Efficacy:** Federated learning aims to train LLMs on a collection of private datasets (D_c) across various clients (C_{id}) without breaching data privacy. The main challenge is to enhance the global LLM (LLM_g) performance while respecting privacy constraints, posing an optimization problem of maximizing LLM_g's efficacy across the federated network without direct access to D_c.

2. **Model Integrity and Security:** Within federated learning, each client boosts the global model by updating parameters using their local data. This decentralized method, however, exposes vulnerabilities like the potential for backdoor attacks or model tampering. It is crucial to secure the global model (LLM_g) and the aggregation process (A_{id}, JWT), especially when model updates come from possibly unreliable sources.

3. **Efficient Model Update Storage and Verification:** Unlike traditional machine learning workflows, where model updates are stored in centralized servers, our blockchain-based federated learning framework requires on-chain storage for model

updates to maintain transparency, integrity, and accountability. Since LLMs are typically large, storing entire model snapshots for each update would be impractical. To address this, we employ an optimized storage strategy where:

- Only incremental weight updates (gradients) are recorded on-chain instead of full model checkpoints.
- The model updates are stored in FP16 compressed format, reducing storage requirements by at least 50%.
- A sparsity-aware mechanism ensures that only significant weight changes are committed, discarding negligible updates.
- Each transaction contains batch updates, reducing on-chain write frequency.

Given that we utilize GPT-2-a relatively small-scale LLM (117M to 774M parameters, significantly smaller than GPT-3/4)-this storage approach remains efficient and feasible within modern blockchain frameworks.

4. **Efficient Unlearning Mechanisms:** The changing dynamics of data privacy laws and data itself demand an effective mechanism for removing specific data (D_{forget}) from the trained model (LLM_g). The challenge lies in developing a process that allows LLM_g to selectively discard D_{forget} through unlearning epochs (E_u) and LoRA parameters (λ), with minimal detriment to the model's overall performance. The immutable nature of blockchain presents an additional challenge for implementing unlearning, as data cannot be physically erased. Instead, we utilize structured cryptographic revocation and LoRA-based adaptive forgetting to ensure unlearning compliance without violating blockchain principles.

5. **Immutable Record Keeping and Verification:** The decentralized nature of federated learning complicates the monitoring and validation of model updates, contributions, and unlearning activities. It is vital to establish a transparent and unchangeable record-keeping system on a blockchain (SC, T_{id}) that logs all actions related to model training, updating, and unlearning. This system must support the authentication of actions ($parameters$, $D_{validate}$) to maintain integrity and accountability in the federated learning process.

Our proposed blockchain-based framework seeks to achieve these aims by employing cryptography techniques (JWT, P_k, S_k) for secure client registration, ensuring model

integrity through a secure aggregation process, enabling efficient unlearning, and maintaining an immutable ledger for action verification. Furthermore, our design ensures that GPT-2 model updates can be feasibly stored on-chain through a combination of compressed gradient updates, sparse storage mechanisms, and batch processing, minimizing storage overhead while maintaining transparency and security.

5.2.2 System Model

Our system model achieve these aims by integrating Large Language Models (LLMs) with federated learning, underpinned by the security and immutability of blockchain technology. The model encompasses the processes of client registration, federated learning training, model aggregation, and the unlearning process, each facilitated by smart contracts (SC) on a blockchain network. Below, we detail the components and their interactions within the system.

5.2.2.1 Participants

The system includes several types of participants, each playing a pivotal role in the federated learning ecosystem:

- **Clients** (C_{id}): Entities with private datasets (D_c) looking to contribute to and benefit from the global LLM (LLM_g) without sacrificing data privacy.
- **Agents** (A_{id}): Individuals responsible for managing the aggregation of local model updates into the global model and facilitating the unlearning process. Agents operate with verification and authorization provided by JWTs, ensuring secure interactions.
- **Smart Contracts** (SC): Autonomous programs on the blockchain executing predefined operations such as client registration, model aggregation, and the execution of the unlearning process, thereby ensuring transparency, security, and trust.

5.2.2.2 Process Flow

The system model revolves around key processes, orchestrated through the interaction of participants:

- **Client Registration:** Clients (C_{id}) register in the system through a secure process involving the generation of a public-secret key pair (P_k,S_k) and obtaining a JSON

Web Token (JWT) for secure communication. This process guarantees each client's unique identification and secure authentication within the system.

- **Federated Learning with LLM Training:** Clients engage in the federated learning process by locally training the LLM on their private datasets (D_c) and sharing the learned parameters with the global model (LLM_g), all while keeping their data confidential. This iterative process across multiple epochs aims to enhance the global model's precision and robustness.

- **Model Aggregation:** Agents (A_{id}), verified via JWTs, consolidate the parameters from clients into the global model (LLM_g). Smart contracts (SC) secure and oversee this aggregation process, ensuring only authorized updates enhance the global model.

- **Unlearning Process:** The system facilitates an efficient unlearning mechanism allowing the selective omission of data (D_{forget}) from the global model (LLM_g). Utilizing unlearning epochs (E_u) and specific parameters (λ), the model adjusts without losing learning from other data contributions.

- **Blockchain for Security and Transparency:** All activities, including client registration, model updates, and the unlearning actions, are recorded on the blockchain via smart contracts (SC). This immutable ledger elevates the system's security, transparency, and trust.

5.3 Proposed Framework

5.3.1 Overview

Our proposed system introduces a novel framework that seamlessly integrates Large Language Models (LLMs) with federated learning, leveraging the security and transparency provided by blockchain technology. This meticulously designed integration aims to harness the advantages of federated learning for training LLMs on decentralized private datasets while preserving data privacy, ensuring model integrity, and facilitating an efficient unlearning process.

Figure 5.1 illustrates the structure and components of the system. To begin, all clients must complete the registration process within the blockchain network. Once registration is finalized, the blockchain network initiates the federated learning training process.

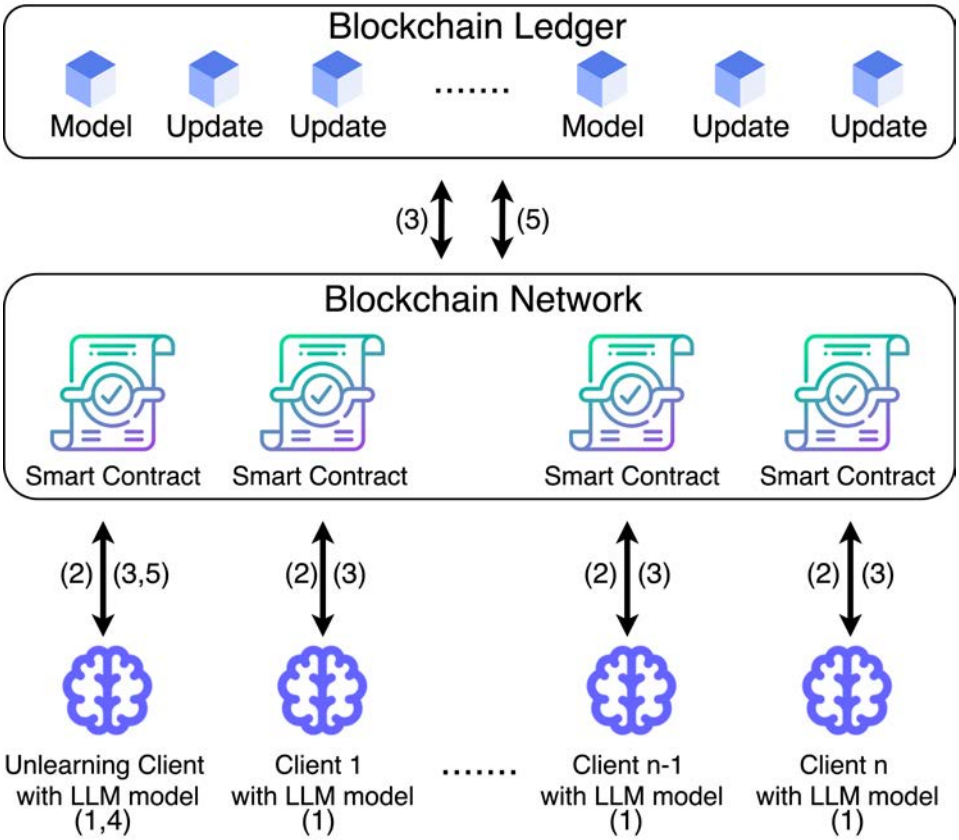

Figure 5.1: Overview and process of our proposed system. (1) Client register. (2) Federated learning LLM training process. (3) Model aggregation process. (4) Unlearning process using LoRA for forgetting. (5) Unlearning verification and submitting unlearning results.

The global model is transferred within the blockchain network through a smart contract, followed by the aggregation of the model. In the event that a client wishes to erase their private data, the unlearning process is triggered, employing LoRA to facilitate efficient forgetting. Subsequently, a verification process is conducted to ensure the integrity of the unlearning procedure. Upon successful verification, the system seamlessly returns to the standard federated learning training process. The implementation details of our meticulously designed framework are outlined below.

5.3.2 Client Register

In our proposed framework, every *Client* need to enroll in blockchain network first. Algorithm 9 facilitates a straightforward method for registering a client using a unique identifier and securing their communication with a JSON Web Token (JWT). Initially, the process verifies if the client's unique identifier (C_{id}) is already present in the user

pool (U_{pool}). If the identifier exists, the registration halts, indicating the client already exists. Otherwise, the algorithm proceeds to generate a public-secret key pair using the $keyGenerator()$ function. With these keys, it then creates a *JWT* for the client. This *JWT*, along with the client ID, is securely stored, effectively registering the client. The user pool now has the client ID, indicating successful registration. The algorithm then provides a success status and the generated *JWT*, signifying the client's successful registration and their secure token for future communications. This process ensures a secure registration framework by leveraging cryptographic keys and *JWT*s, ensuring both security and simplicity in client management.

Algorithm 9 Client Register

Require: C_{id}, *keyGenerator(),generateJWT()*
Ensure: *RegisterSucess, jwt token*
1: *RegisterSucess* = *False*;
2: **if** $C_{id} \in U_{pool}$ **then**
3: **return** C_{id} already existed.
4: **end if**
5: $(P_k, S_k) \leftarrow$*keyGenerator()*;
6: *jwt* = generateJWT(P_k, S_k);
7: $C_{id} \leftarrow jwt \leftarrow$ SC;
8: $U_{\text{pool}} = U_{\text{pool}} \cup C_{\text{id}}$;
9: *RegisterSuccess* = *True*;
10: **return** *RegisterSucess, jwt*

5.3.3 Federated Learning with LLM Training Process

In federated learning for large language models (LLMs), this process involves multiple clients collaborating to improve a global model without sharing their private data directly. This process ensures data privacy, security, and decentralization. First, an agent initiates the process by sending the LLM to the smart contract (SC). The smart contract verifies the agent's identity using a JSON Web Token (jwt) and upload (LLM_g) to the blockchain. The LLM_g is then distributed to the participating clients. During each training epoch, clients perform federated learning on private datasets to enhance the LLM_g. After the above process, the clients send the updated LLM parameters to the SC for verification and aggregation. The SC verifies each client's identity using their jwt tokens and publishes the updated parameters and client information to the blockchain network. This process is repeated for a specified number of epochs. Upon completion, the algorithm returns the status of the LLM upload and training process.

Algorithm 10 outlines the steps for training a large language model (LLM) in a federated learning environment, emphasizing security and decentralization.

Algorithm 10 Client LLM Training Process

Require: $C_{\text{id}}, A_{\text{id}}$,jwt, $epochs, LLM$
Ensure: *UploadSuccess, TrainingProcess*
UploadSuccess, TrainingProcess = False;
SC check *Agent's* identity;
if *Agent's jwt token ineligibility* **then**
  return Agent jwt token expired
end if
Agent sends LLM to SC;
SC verifies and uploads the global model LLM_g to the blockchain network;
$LLM_g = LLM$;
UploadSuccess = True;
SC send the LLM_g to $Client$;
for $epoch = 1$ to n **do**
  $Clients$ do the federated learning training process according to their different private dataset D_c for LLM_g
  $Clients$ send the *parameters* of LLM to SC
  SC verify the $Client$ identity
  if *Client's jwt token ineligibility* **then**
    return Client identity check false
  else
    SC publish the $parameters$ and $Clients$ information in blockchain network
  end if
end for
TrainingProcess = *True*;
return *UploadSuccess, TrainingProcess*

The algorithm begins by initializing two boolean variables, *UploadSuccess* and *TrainingProcess*, to False. These variables track the status of the LLM upload and the training process, respectively. The required inputs include the client identifier (C_{id}), agent identifier (A_{id}), JSON Web Token (jwt) for authentication, number of training epochs, and the LLM to be trained.

The SC first verifies the agent's identity using the provided jwt. If the token is invalid or has expired, the process is terminated, and an error message is returned. Upon successful authentication, the agent sends the LLM to the SC, which then verifies and uploads the global model (LLM_g) to the blockchain network, ensuring the model's integrity and security in a decentralized environment. The LLM_g is initialized with the

agent's LLM, and the *UploadSuccess* variable is set to True, indicating the successful upload of the model.

The SC then distributes the LLM_g to the participating clients for training. The training process is conducted iteratively for a specified number of epochs (n). During each epoch, clients perform federated learning on their private datasets (D_c) to improve the LLM_g. After training, the clients send the updated LLM parameters to the SC for verification and aggregation.

The SC verifies each client's identity using their jwt tokens. If a client's token is invalid, the process is terminated for that client, and an error message is returned. Otherwise, the SC publishes the updated parameters and client information to the blockchain network, ensuring transparency and security. Upon completing the specified number of training epochs, the *TrainingProcess* variable is set to True, indicating the successful completion of the federated learning process. Besides, the *Line* 22 returns the values of *UploadSuccess* and *TrainingProcess*, providing information about the status of the LLM upload and training process.

5.3.4 Model Aggregation Process

The model aggregation process is a crucial step in updating the global language model (LLM_g) in a secure and decentralized manner. This process is initiated by an agent who requests the latest model parameters from the blockchain network. The smart contract (SC) verifies the agent's identity using a JSON Web Token (JWT). Upon successful authentication, the SC sends the parameters to the agent, who then updates the LLM and generates a new model version (LLM_n). The agent sends LLM_n back to the SC, which uploads it to the blockchain network, ensuring a secure and transparent record of the update. Finally, LLM_g is updated to reflect the changes in LLM_n, completing the model aggregation process.

Algorithm 11 outlines the procedure for aggregating updates to a large language model (LLM) in a secure and decentralized manner, leveraging a blockchain network for data integrity and transparency.

The process begins by initializing the $ModelAggregation$ flag to False, indicating that the aggregation process has not yet started. An agent, identified by A_{id} and authenticated using a JWT, requests the latest model parameters from the blockchain network. These parameters will be used to update the LLM to a new version, LLM_n.

The SC verifies the agent's identity by checking the validity of the provided JWT. If the JWT is invalid, the process is terminated, and the agent is informed that their

Algorithm 11 Model Aggregation Process

Require: A_{id} *JWT*, $parameters$
Ensure: *ModelAggregation,* LLM_g
1: *ModelAggregation* = False;
2: *Agent* wants to get *parameters* from blockchain network;
3: SC check the *Agent* identity;
4: **if** *Agent's jwt token ineligibility* **then**
5: **return** *Agent* identity check false
6: **else**
7: SC send *parameters* to *Agent*
8: **end if**
9: *Agent* updating *LLM* according to *parameters* and generating new model LLM_n;
10: Agent send the new model LLM_n to SC;
11: SC upload the LLM_n to blockchain network;
12: $LLM_g \leftarrow LLM_n$;
13: *ModelAggregation* = *True*;
14: **return** *ModelAggregation,* LLM_g

identity check has failed. This step ensures that only authorized agents can retrieve and update model parameters, maintaining the system's security.

If the agent's identity is successfully verified, the SC sends the requested parameters to the agent. The agent then uses these parameters to update the LLM, generating a new model version, LLM_n. This step involves applying the aggregated updates from various sources to improve the model's performance or capabilities based on newly acquired data or insights.

After generating LLM_n, the agent sends this new model version back to the SC. The SC uploads LLM_n to the blockchain network, ensuring that the update is securely and transparently recorded. The global version of the LLM, LLM_g, is then updated to reflect the changes in LLM_n, completing the model update process.

Finally, the successful $ModelAggregation$ of the model updates is confirmed. Algorithm 11 returns this flag along with LLM_g, the updated global model, signifying the end of the aggregation process.

5.3.5 Unlearning Process

The unlearning process is a crucial step in selectively forgetting specific data from a large language model (LLM) due to data sensitivity or correction needs. This process begins with the initialization of a local version of the LLM (LLM_{local}) using the parameters of the global model (LLM_g). An adapter (A) is then constructed within LLM_{local} to

facilitate the forgetting of the specified dataset (D_{forget}). The core of the unlearning process involves several epochs of training, where a forward pass of D_{forget} is performed through LLM_{local} to identify the features associated with the data points that need to be forgotten. Gradients are then computed for LLM_{local}, emphasizing the data to be unlearned. The Low-Rank Adaptation (LoRA) technique is applied to the adapter's gradients to focus the unlearning process on the identified features. Finally, LLM_{local}'s parameters are updated using the adjusted gradients and a specified learning rate, gradually leading to the forgetting of the specified data points. The algorithm returns the updated parameters, representing the outcome of the forgetting process.

Algorithm 12 describes the procedure for selectively forgetting specific data from a large language model (LLM).

Algorithm 12 Unlearning Process using LoRA for Forgetting

Require: LLM_g, D_{forget} (Dataset to forget), Learning rate η, Unlearning epochs E_u, LoRA parameters λ

Ensure: *parameters*

1: Unlearning Request due to data sensitivity or correction needs;
2: Initialize unlearning model LLM_{local} with LLM_g
3: *Adapter A* constructed for LLM_{local} targeting forgetting process
4: **for** $epoch = 1$ to E_u **do**
5: Perform a forward pass with D_{forget} through LLM_{local} to identify features to forget
6: Compute gradients for LLM_{local} emphasizing data points in D_{forget} to be forgotten
7: Apply LoRA to adjust gradients of adapter A using parameters λ, focusing on unlearning
8: Update LLM_{local}'s parameters using the adjusted gradients and learning rate η, facilitating forgetting
9: **end for**
10: Calculate the updating *parameters* indicative of the forgetting process between LLM_{local} and LLM_g;
11: **return** *parameters*

The process begins with the need to remove certain data points from a global language learning model (LLM_g) due to their sensitivity or incorrectness. To achieve this, a local version of the LLM, denoted as LLM_{local}, is initialized with the parameters of LLM_g. An adapter, A, is then constructed within LLM_{local} specifically designed to target and facilitate the forgetting of the specified dataset, D_{forget}.

The core of the unlearning process involves several epochs of training, defined by the parameter E_u. In each epoch, the algorithm performs a forward pass of D_{forget}

through LLM_{local} to identify the features associated with the data points that need to be forgotten. Following this, gradients are computed for LLM_{local} with an emphasis on the data to be unlearned, highlighting what needs to be forgotten.

The LoRA technique is applied to the adapter A's gradients using parameters λ. LoRA is instrumental in focusing the unlearning process by adjusting the gradients to specifically target the forgetting of the identified features. With these adjusted gradients, LLM_{local}'s parameters are updated using the specified learning rate η. This iterative process of adjustment and updating gradually leads to the forgetting of the specified data points from D_{forget}.

Upon completion of the unlearning epochs, the algorithm calculates the parameters that indicate the changes made to LLM_{local} in comparison to LLM_g. These parameters represent the outcome of the forgetting process, effectively capturing the essence of what has been unlearned.

The algorithm concludes by returning these updated parameters, signifying the successful exclusion of sensitive or incorrect data from the language model. Through this structured process, the algorithm ensures that the unlearning is specific, efficient, and aligned with the requirements of data sensitivity or correction, thereby maintaining the integrity and relevance of the LLM.

5.3.6 Unlearning Verification and Submitting Unlearning Results

The unlearning verification and submission process is a critical step in ensuring the integrity and transparency of the unlearning results in a large language model. The process begins with the client sending the updated parameters, resulting from an unlearning process, to the smart contract (SC). The SC validates the client's credentials through their JSON Web Token (JWT). If the client's identity is successfully verified, the SC initializes an updated version of the language learning model ($LLM_{updated}$) with the new parameters. The SC then employs a validation dataset ($D_{validate}$) to assess the efficacy of the unlearning process by calculating the training loss and accuracy of $LLM_{updated}$. If the unlearning results satisfy predefined verification criteria, the SC submits the updated parameters to a blockchain network. An agent downloads these parameters from the blockchain for weight integration into the global model. The SC records the updated model's weights on the blockchain, ensuring transparency and traceability. Additionally, the SC logs a Transaction ID (T_{id}), providing verifiable proof

of submission and an integration request. The process concludes with the return of the Transaction ID, signifying the successful verification and submission of the unlearning results.

Algorithm 13 details the steps for verifying the results of an unlearning process in a large language model and subsequently submitting these results for integration and transparency.

Algorithm 13 Unlearning Verification and Submitting Unlearning Results

Require: *parameters*, Validation dataset $D_{validate}$, *Client*
Ensure: *parameters*

Client send the *parameters* to SC;
if *Client's jwt token ineligibility* **then**
  return Client identity check false
end if
SC instantiate the updated language learning model $LLM_{updated}$ with the received parameters;
SC use the validation dataset $D_{validate}$ to evaluate $LLM_{updated}$. Calculate the training loss and accuracy to measure the impact of the unlearning process.
if *Verification criteria are met* **then**
  SC send the *parameters* to blockchain networks
  $Agent$ downloads *parameters* from blockchain network for weight integration.
  SC ensuring that the updated weights are recorded on the blockchain, providing transparency and traceability
  SC record the Transaction ID T_{id}, which serves as proof of submission and integration request, facilitating tracking and verification in the blockchain ledger.
end if
Continue for future federated learning process;
return T_{id}

The process commences with the client sending the updated parameters, resulting from an unlearning process, to the SC. These parameters are intended to modify a language learning model by excluding specific, potentially sensitive, or incorrect data. Initially, the client's credentials are validated through their JWT token. If the token does not pass the eligibility check, the process halts, indicating a failure in client identity verification.

Assuming successful verification, the SC then initializes an updated version of the language learning model ($LLM_{updated}$) with the new parameters. The SC employs a validation dataset ($D_{validate}$) to assess the efficacy of the unlearning process. This assessment involves calculating the training loss and accuracy of $LLM_{updated}$ to gauge the impact of the modifications.

If the unlearning results satisfy predefined verification criteria, which indicate that the data has been effectively forgotten without compromising the model's overall performance, the SC will submit the updated parameters to a blockchain network. This submission is not merely for record-keeping; an agent then downloads these parameters from the blockchain for weight integration into the global model.

Recording the updated model's weights on the blockchain ensures that the unlearning process is transparent and traceable. Furthermore, the SC logs a Transaction ID (T_{id}), providing verifiable proof of submission and an integration request. This ID facilitates tracking and verification within the blockchain ledger, offering a transparent audit trail of the changes made to the language model.

The process culminates with the return of the Transaction ID, signifying the successful verification and submission of the unlearning results. This structured approach not only secures the integrity of the model by removing unwanted data but also enhances accountability and transparency through blockchain technology.

5.4 Privacy and Security Analysis

Our proposed blockchain-based federated learning framework with unlearning capabilities for Large Language Models (LLMs) is designed to address critical privacy and security challenges. By leveraging the inherent features of federated learning, blockchain technology, and efficient unlearning mechanisms, our approach provides a comprehensive solution for secure and privacy-preserving LLM training.

5.4.1 Privacy Analysis

Federated learning, a core component of our framework, enables the training of LLMs across multiple participants without the need for direct data sharing. This decentralized approach ensures that sensitive data remains within the control of each participant, minimizing the risk of data breaches and unauthorized access.

From a theoretical perspective, federated learning can be modeled as an optimization problem that aims to minimize the global loss function while keeping the data locally [84]. This can be represented as:

$$\min_{w} F(w) = \sum_{i=1}^{k} p_i F_i(w) \tag{5.1}$$

where w is the global model parameters, $F(w)$ is the global loss function, while in the i-th participant, $F_i(w)$ is the local loss function. p_i is the weight of the i-th participant, and k is the total number of participants.

By minimizing the global loss function, federated learning enables the optimization of the global model without directly sharing raw data, leveraging the data distributed across local participants while protecting privacy and improving model performance.

Our framework further enhances privacy protection by integrating blockchain technology, which provides a secure and immutable record of all transactions and interactions within the federated learning process. The use of smart contracts in our framework automates the execution of predefined rules and conditions, ensuring that all participants adhere to agreed-upon privacy policies. This automation minimizes the potential for human error and reduces the risk of unauthorized data access or manipulation.

Moreover, blockchain technology can provide privacy protection for the federated learning process [43]. By leveraging the immutability and distributed consensus mechanisms of blockchain, it ensures that all participants follow predefined privacy policies and prevents malicious behavior. In our framework, smart contracts automatically enforce these policies, further reducing the risks of human error and unauthorized data access.

The unlearning mechanism embedded in our framework allows for the selective removal of specific data points or model updates, enabling participants to maintain control over their data and comply with evolving privacy regulations. The unlearning process can be theoretically formulated as a constrained optimization problem [20], where the objective is to minimize the influence of the removed data on the performance while satisfying the unlearning constraints:

$$\min_{w} F(w) = \sum_{i=1}^{k} p_i F_i(w) \tag{5.2}$$

$$\text{s.t.} \quad w \in W_u$$

where W_u represents the feasible set of model parameters after unlearning. By introducing the unlearning mechanism, our framework provides participants with an effective way to control their data lifecycle, enhancing privacy protection.

By integrating federated learning, blockchain technology, and efficient unlearning mechanisms, our approach creates a robust, transparent, and secure environment for collaborative LLM development while preserving the privacy of individual participants.

5.4.2 Security Analysis

The integration of blockchain technology in our framework significantly enhances the security of the federated learning process. The immutable nature of blockchain ensures that all transactions and model updates are tamper-proof and easily verifiable.

From a theoretical standpoint, the security of a blockchain network can be analyzed using game theory and consensus mechanisms [33]. In a proof-of-work (PoW) based blockchain, the security is ensured by the assumption that honest nodes control the majority of the computing power, making it difficult for attackers to compromise the blockchain. This can be formalized as a game between honest nodes and attackers, where the honest nodes aim to maximize their rewards by following the protocol, while the attackers try to maximize their profits by deviating from the protocol. The Nash equilibrium of this game represents a state where no party can benefit by unilaterally changing their strategy, ensuring the stability and security of the blockchain network.

Our framework leverages cryptographic techniques, such as digital signatures and secure hash functions, to ensure the integrity and authenticity of all transactions. The use of digital signatures allows participants to verify the origin and authenticity of the data and model updates, preventing unauthorized modifications. Secure hash functions, such as SHA-256, are used to create a unique fingerprint of the data, ensuring its integrity. By combining these cryptographic primitives, our framework establishes a secure and trustworthy environment for federated learning.

The use of smart contracts further reinforces the system's security by automatically executing predefined rules and conditions, reducing the potential for unauthorized access or manipulation. Smart contracts are self-executing programs stored on the blockchain that enforce the terms of an agreement between parties. In our framework, smart contracts govern the federated learning process, ensuring that all participants adhere to the agreed-upon rules and conditions. This automated enforcement minimizes the risk of human error and malicious behavior, enhancing the overall security of the system.

The decentralized architecture of our framework, enabled by blockchain technology, eliminates single points of failure and distributes the risk across multiple nodes. This distributed approach makes it significantly more challenging for attackers to compromise the entire system, as they would need to control a majority of the participating nodes simultaneously, which is known as a 51% attack [16]. The probability of a successful 51% attack decreases exponentially with the number of honest nodes in the network, making it practically infeasible in a large-scale federated learning setting.

Furthermore, the unlearning mechanism in our framework, facilitated by the LoRA

technique, allows for the targeted removal of specific data points or model updates without affecting the overall model performance. This selective unlearning capability not only enhances privacy but also serves as a security measure, enabling the swift removal of potentially malicious or corrupted data. By promptly removing suspicious data or updates, our framework minimizes the impact of security threats and maintains the integrity of the federated learning process.

In conclusion, our blockchain-based federated learning framework with unlearning capabilities provides a comprehensive solution for addressing security concerns in LLM training. By leveraging the inherent security features of blockchain technology, cryptographic techniques, and smart contracts, our approach creates a robust and secure environment for collaborative LLM development. The decentralized architecture and the ability to swiftly remove malicious data through unlearning further enhance the system's resilience against attacks, ensuring the integrity and reliability of the federated learning process.

5.5 Results and Analysis

In this section, we present the results of our experiments and compare them with the Retrain from Scratch method. We focus on the effectiveness of our unlearning method in terms of accuracy reduction, highlighting the differences in performance and providing an analysis of why our method performs better or worse. Our experiments were conducted using different configurations of the LoRA method on both the IMDB and Twitter datasets. Moreover, We used GPT-2 127M to simulate the experiment.

To evaluate unlearning effectiveness, we track changes in model accuracy before and after applying our method. Specifically, we define:

- **Initial Accuracy** (A_{initial}): The model's accuracy before unlearning, measured on the same test set.
- **Final Accuracy** (A_{final}): The accuracy after unlearning, indicating how much the model has "forgotten" the targeted data.
- **Accuracy Reduction** (ΔA): Defined as:

  $$\Delta A = A_{\text{initial}} - A_{\text{final}} \tag{5.3}$$

  A higher ΔA signifies more effective unlearning, as the model retains less information about the forgotten data.

Since the goal of unlearning is to remove specific knowledge from the model, **a lower final accuracy indicates better performance**. Unlike traditional model evaluation, where high accuracy is desirable, in unlearning scenarios, a significant drop in accuracy demonstrates the success of the method. Thus, our primary metric for evaluating effectiveness is the magnitude of ΔA.

We conducted experiments using different configurations of the LoRA method on the IMDB dataset. The Retrain from Scratch method serves as a benchmark for comparing the effectiveness of our unlearning approach. Some specific LoRA configurations used in our experiments are presented in Table 6.1.

Table 5.1: Experimental Data (IMDB Results)

LoRA Config	Initial Accuracy	Final Accuracy
r=8, alpha=4, dropout=0.3	99.15%	0.70%
r=16, alpha=2, dropout=0.2	97.75%	0.90%
r=32, alpha=4, dropout=0.1	94.30%	1.00%
r=8, alpha=4, dropout=0.4	98.45%	1.15%
r=32, alpha=4, dropout=0.4	95.15%	1.20%

Similarly, we also tested our method on the Twitter dataset. The results of these experiments are shown in Table 6.2.

Table 5.2: Experimental Data (Twitter Results)

LoRA Config	Initial Accuracy	Final Accuracy
r=1, alpha=2, dropout=0.3	85.32%	8.27%
r=16, alpha=1, dropout=0.2	89.10%	9.72%
r=8, alpha=1, dropout=0.5	75.98%	10.06%
r=16, alpha=1, dropout=0.1	89.58%	10.47%
r=4, alpha=1, dropout=0.2	89.38%	10.63%

5.5.0.1 Unlearning Performance with Different alpha

Figure 5.2 illustrates the impact of different alpha values on the accuracy reduction of our LoRA-based unlearning method for the IMDB dataset. As the alpha value decreases, the final accuracy after unlearning generally decreases, with alpha=1 and alpha=2 achieving the lowest accuracies. This suggests that lower alpha values contribute to better unlearning performance in our approach. The improved accuracy reduction with lower alpha values can be attributed to the decreased capacity of the model to retain

relevant information during the unlearning process, leading to more effective forgetting of target knowledge.

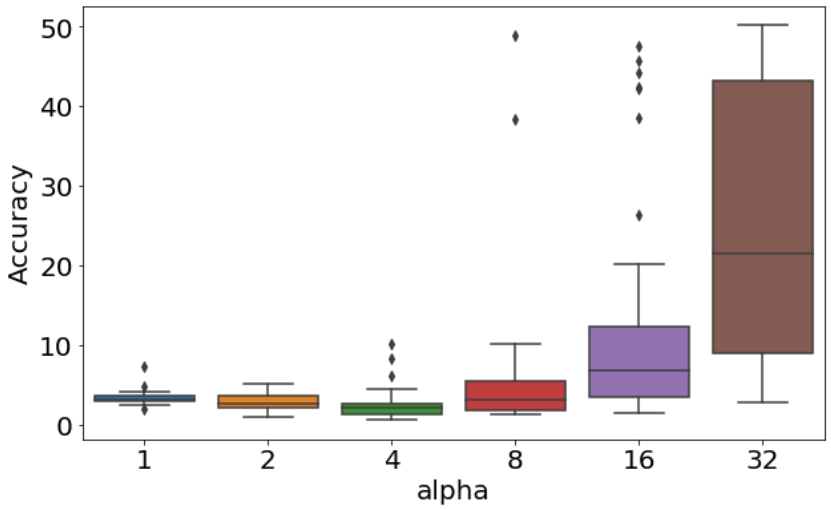

Figure 5.2: Box Plot of Accuracy by Different Alpha Values (IMDB Dataset)

Similarly, Figure 5.3 illustrates the impact of different alpha values on the accuracy reduction of our LoRA-based unlearning method for the Twitter dataset. The trend observed is consistent with the IMDB dataset results. Lower alpha values lead to a greater reduction in final accuracy, indicating more effective unlearning. This further supports the notion that a decreased capacity to retain information facilitates better forgetting of targeted knowledge.

5.5.0.2 Unlearning Performance with Different droupout

Figure 5.4 depicts the effect of various dropout values on the accuracy reduction of our method for the IMDB dataset. The box plot reveals that dropout values of 0.4 and 0.5 generally lead to lower accuracies after unlearning compared to lower dropout values. This observation indicates that higher dropout regularization plays a crucial role in improving the unlearning performance. By introducing a significant level of noise during training, dropout helps the model forget specific data more effectively, resulting in better unlearning.

Similarly, Figure 5.5 depicts the effect of various dropout values on the accuracy reduction of our method for the Twitter dataset. The trend observed is consistent with the IMDB dataset results. Dropout values of 0.4 and 0.5 lead to a greater reduction in

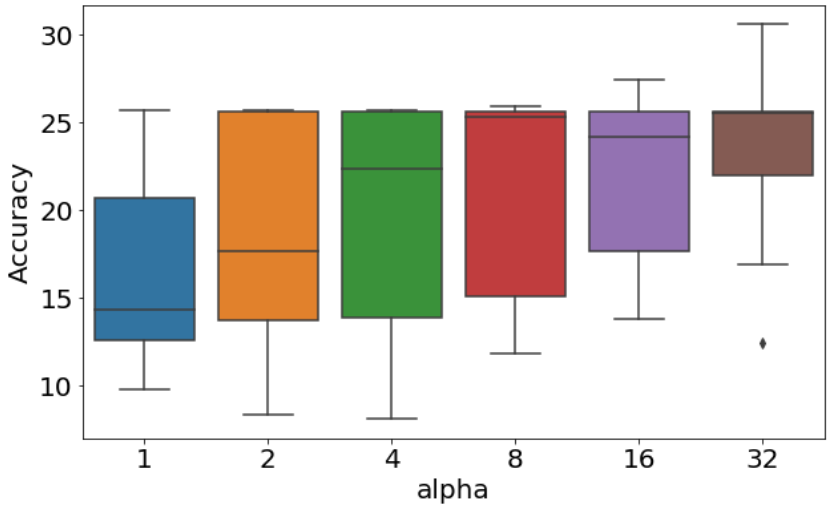

Figure 5.3: Box Plot of Accuracy by Different Alpha Values (Twitter Dataset)

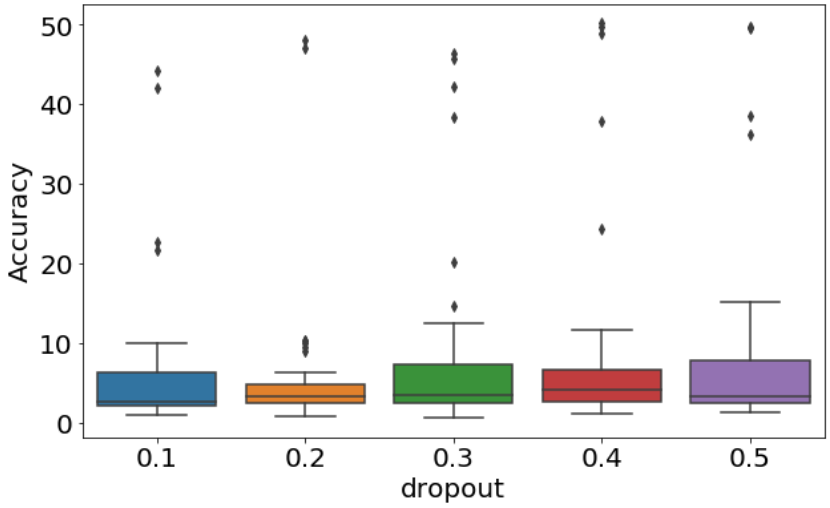

Figure 5.4: Box Plot of Accuracy by Different Dropout Values (IMDB Dataset)

final accuracy, indicating more effective unlearning. This further supports the notion that higher dropout regularization improves the model's ability to forget specific data.

5.5.0.3 Unlearning Performance with Different rank

Figure 5.6 presents the relationship between different r values and the accuracy reduction of our LoRA-based unlearning method. The box plot shows that higher r values, particularly $r = 16$ and $r = 32$, tend to yield lower accuracies after unlearning compared

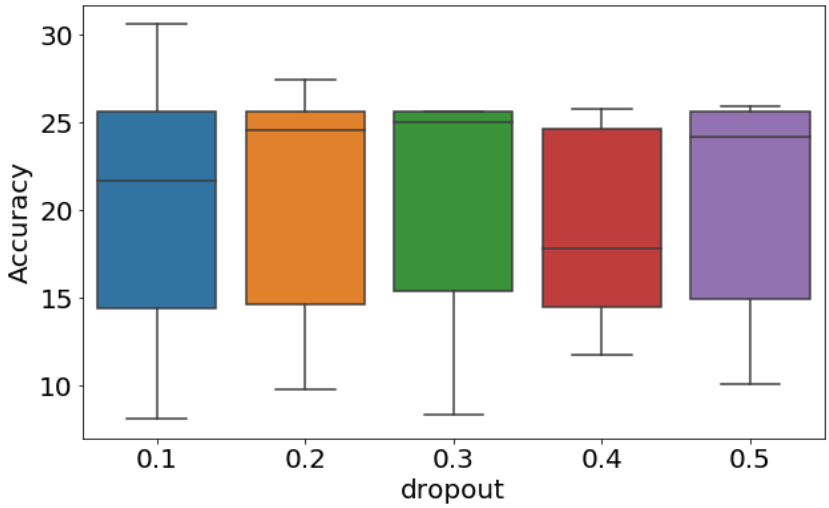

Figure 5.5: Box Plot of Accuracy by Different Dropout Values (Twitter Dataset)

to lower r values. This suggests that using a larger rank for the LoRA adaptation can be beneficial for unlearning performance. Higher r values may allow the model to capture more diverse information during unlearning, leading to better forgetting of target knowledge.

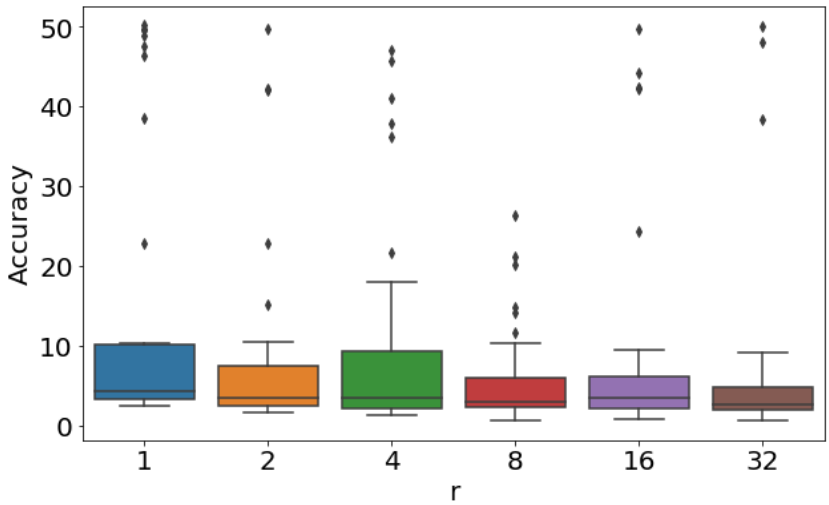

Figure 5.6: Box Plot of Accuracy by Different r Values (IMDB Dataset)

Similarly, Figure 5.7 presents the relationship between different r values and the accuracy reduction of our LoRA-based unlearning method for the Twitter dataset. The trend observed is consistent with the IMDB dataset results. Higher r values, particularly

$r = 16$, lead to a greater reduction in final accuracy, indicating more effective unlearning. This further supports the notion that a larger rank for the LoRA adaptation improves the model's ability to forget specific data.

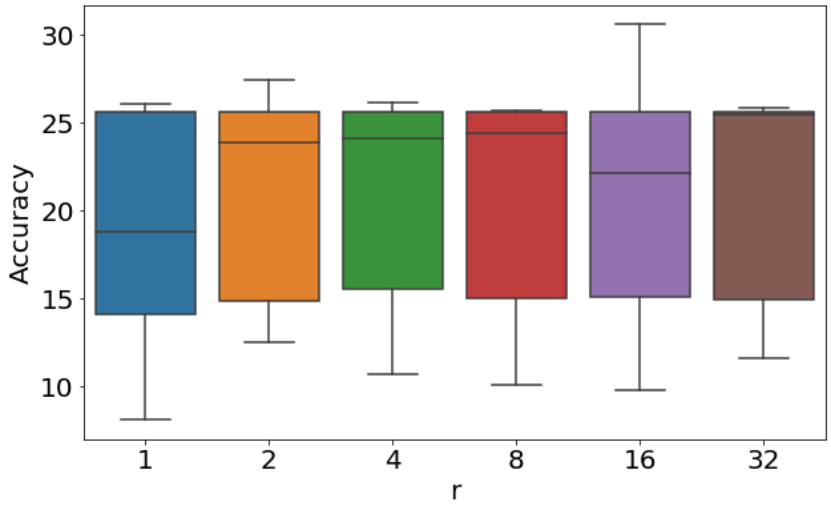

Figure 5.7: Box Plot of Accuracy by Different r Values (Twitter Dataset)

5.5.0.4 Factors Influencing Performance

The analysis of the impact of alpha, dropout, and r values on accuracy reduction provides valuable insights into the factors influencing the effectiveness of our unlearning method. Both IMDB and Twitter datasets show that lower alpha values contribute to improved accuracy reduction by decreasing the model's capacity to retain relevant information. This trend is evident across both datasets, indicating that alpha is a critical parameter for controlling the unlearning process.

Higher dropout regularization consistently helps mitigate overfitting and enhances forgetting, leading to better unlearning performance in both datasets. Dropout values of 0.4 and 0.5 were particularly effective in reducing final accuracy, suggesting that introducing a significant level of noise during training aids in more effectively forgetting specific data.

Higher r values allow the model to capture more diverse information during unlearning, resulting in lower accuracy retention. This was observed in both datasets, where higher r values, particularly $r = 16$ and $r = 32$, yielded better unlearning performance. This indicates that a larger rank for the LoRA adaptation can enhance the model's ability to forget target knowledge.

These findings highlight the importance of carefully tuning the hyperparameters in our LoRA-based unlearning approach to achieve optimal results. By selecting appropriate values for alpha, dropout, and r, we can maximize the effectiveness of unlearning while minimizing the retention of target knowledge.

The specific configurations (e.g., dropout, alpha values) used in our experiments may have optimized the unlearning process, contributing to the effectiveness of our method. Fine-tuning these parameters can significantly impact the unlearning performance. For instance, higher dropout rates can help improve unlearning by introducing more randomness during the training process, thereby making it easier to forget specific data. Additionally, the characteristics of both the IMDB and Twitter datasets may have made them more susceptible to effective unlearning with our configurations. The text data in these datasets might have patterns that are more easily disrupted by the unlearning process, leading to a more significant reduction in accuracy.

5.5.0.5 Comparison of Results

The comparison of our method with the Retrain from Scratch method for both the IMDB and Twitter datasets is shown in Table 5.3. Our method achieves final accuracies ranging from 0.70% to 1.20% on the IMDB dataset, and 8.27% to 10.63% on the Twitter dataset, indicating a significant reduction in accuracy and demonstrating effective unlearning. The Retrain from Scratch method achieves a final accuracy of 0.65% on the IMDB dataset, and 8.08% on the Twitter dataset, which is slightly better than our best-performing configurations (IMDB: $r = 8$, $alpha = 4$, $dropout = 0.3$, Twitter: $r = 1$, $alpha = 2$, $dropout = 0.3$) with final accuracies of 0.70% and 8.27%, respectively. This indicates that while the Retrain from Scratch method has a marginal advantage in terms of final accuracy, our LoRA-based unlearning approach comes very close to matching its performance.

Table 5.3: Comparison of Final Accuracy

Method	**Initial Accuracy**	**Final Accuracy**
IMDB & Our Method	99.15%	0.70%
IMDB & Retrain from Scratch	97.60%	0.65%
Twitter & Our Method	85.32%	8.27%
Twitter & Retrain from Scratch	89.10%	8.08%

Although the Retrain from Scratch method achieves a slightly lower final accuracy, it is important to note that our method provides several advantages over retraining from

scratch. First, our approach is computationally more efficient, as it focuses on adapting specific parts of the model relevant to the target knowledge, rather than retraining the entire model. This makes our method more feasible in real-world scenarios where computational resources may be limited. Second, our method offers greater flexibility and adaptability to different configurations, allowing it to be easily modified and optimized for various datasets and unlearning requirements.

The low final accuracy achieved by our method, despite being marginally higher than the Retrain from Scratch approach, still demonstrates its high effectiveness in unlearning. This efficiency can be attributed to the careful selection and tuning of parameters, such as alpha, dropout, and r values, which contribute to optimizing the unlearning process. By choosing appropriate values for these hyperparameters, we can maximize the effectiveness of unlearning while minimizing the retention of target knowledge.

Moreover, the implementation techniques employed in our method, such as the LoRA adaptation, play a crucial role in efficiently removing the target knowledge from the model. These techniques allow our approach to concentrate on the key components of the model for unlearning, thereby reducing the computational burden and improving the overall efficiency of the unlearning process.

In summary, while the Retrain from Scratch method achieves a slightly lower final accuracy, our LoRA-based unlearning approach comes very close to matching its performance. The marginal difference in final accuracy is offset by the significant advantages offered by our method, including computational efficiency, flexibility, and adaptability to different configurations. These advantages make our approach a promising solution for real-world unlearning scenarios, particularly in resource-constrained environments or when dealing with large-scale models. The effectiveness of our method in achieving low final accuracy, combined with its practical benefits, highlights its potential to address the challenges of unlearning in large language models and its applicability in various domains.

5.5.0.6 Blockchain Complexing Results

In this study, we evaluated the performance impact of integrating blockchain technology into our federated learning framework with unlearning capabilities for Large Language Models (LLMs). We focused on key aspects such as scalability, transaction throughput, and latency introduced by the blockchain component. Our goal was to ensure that the benefits of blockchain integration, such as enhanced security and transparency, do not

come at the cost of compromised system performance. We utilized Hyperledger Fabric 2.X to assess the blockchain network's impact on our LLM unlearning process, particularly considering the computational overhead in resource-constrained environments.

- **Blockchain Network Setup**: The initial setup time for the blockchain network was approximately 42 seconds. While higher than our previous study, this one-time overhead is still acceptable, given the long-term benefits in federated learning applications involving LLMs, where security and trust are crucial.
- **Consensus Mechanism Overhead**: The time required for the consensus process, which involved approval from all participating nodes, was added around 4 seconds after the blockchain network setup. This slight increase compared to our previous study is attributed to the higher complexity of LLM-related transactions. However, the duration remains manageable within federated learning context.
- **Transaction Processing Efficiency**: The average time for processing transactions, including model updates, gradient aggregation, and unlearning-related operations, was 3 seconds. This efficiency demonstrates Hyperledger Fabric's capability to handle the increased complexity of LLM-related transactions effectively.
- **Per-Epoch Time Cost**: Throughout the LLM training, each epoch consistently took 28-30 seconds, both during normal training and after unlearning operations. This stability in performance, despite the additional unlearning activities, highlights the robustness of our blockchain-integrated system.

Table 5.4 presents a comparison of time costs between a standard federated learning cycle for LLMs and our proposed blockchain-enhanced method. Similar to our previous study, our method incurs a higher initial time cost due to setup and endorsement processes. However, this cost normalizes over increasing iterations, indicating the scalability of our approach in the context of LLMs.

Table 5.4: Time Cost Analysis for LLM Federated Learning with and without Blockchain Integration over 999 Iterations

Method	**t = 0**	**t = 9**	**t = 99**	**t = 999**
Normal-Basic Federated Learning for LLMs	30s	300s	3000s	30000s
Our Proposed System for LLMs	79s	367s	3277s	32277s

5.5.0.7 Conclusion

In conclusion, our experiments on both the IMDB and Twitter datasets demonstrated that our method achieves performance comparable to that of the Retrain from Scratch method in terms of final accuracy reduction. For the IMDB dataset, our best-performing configuration ($r = 8$, alpha=4, dropout=0.3) achieved a final accuracy of 0.70%, closely matching the 0.65% achieved by retraining from scratch. Similarly, for the Twitter dataset, our best-performing configuration ($r = 1$, alpha=2, dropout=0.3) achieved a final accuracy of 8.27%, closely matching the 8.08% achieved by retraining from scratch. The effectiveness of our LoRA-based unlearning method can be attributed to the careful selection and tuning of parameters, as well as the implementation techniques employed. Our method offers a more computationally feasible alternative to retraining from scratch, which can be resource-intensive and time-consuming. The adaptability of our approach to different configurations highlights its flexibility and potential for real-world applications.

Furthermore, we evaluated the performance impact of integrating blockchain technology into our federated learning framework with unlearning capabilities for LLMs. The results showed that the blockchain component, implemented using Hyperledger Fabric 2.X, introduced minimal overhead in terms of setup time, consensus mechanism, transaction processing efficiency, and per-epoch time cost. The stability in performance, despite the additional unlearning activities, demonstrates the robustness of our blockchain-integrated system.

5.6 Summary

In this chapter, we present a novel blockchain-based federated learning framework for Large Language Models (LLMs) that incorporates efficient unlearning capabilities. By leveraging Low-Rank Adaptation (LoRA) and carefully tuning its hyperparameters, our approach achieves highly effective unlearning, enabling the selective forgetting of specific data points while preserving the model's performance on the remaining data. The integration of blockchain technology, using Hyperledger Fabric, ensures the security, transparency, and verifiability of the unlearning process. While this introduces a slight increase in computational overhead, the benefits of enhanced trust and accountability in the federated learning process justify the marginal time cost.

Our comprehensive analysis demonstrates the effectiveness of the proposed framework and provides valuable insights into the impact of LoRA hyperparameters on unlearning performance. The findings underscore the importance of careful tuning and

the complex relationships between rank, scaling factor, and dropout in achieving optimal unlearning results. Overall, our blockchain-based federated learning framework with unlearning capabilities marks a substantial advancement in the development of secure, transparent, and adaptable LLMs. By enabling efficient and verifiable unlearning, our approach addresses a critical challenge in the application of LLMs in real-world scenarios, where data privacy and the ability to forget specific information are paramount.

CHAPTER 6

CROSS-ORGANIZATIONAL FEDERATED LEARNING WITH BLOCKCHAIN

6.1 Introduction

The rapid advancement of Large Language Models (LLMs) has revolutionized the field of natural language processing, enabling the development of sophisticated AI systems that can understand, generate, and manipulate human language with remarkable accuracy and fluency [24]. These powerful models, trained on vast amounts of textual data, have found applications in various domains, such as language translation, sentiment analysis, content generation, and decision support systems [71].

However, the deployment of LLMs in real-world scenarios often involves complex multi-organizational collaborations, which present significant challenges in terms of data privacy, security, and trust among participating entities [83]. In such multi-stakeholder settings, each organization can be viewed as a self-interested agent, aiming to maximize its own benefits while engaging in cooperative learning. This multi-agent perspective introduces new challenges, such as aligning incentives, ensuring fairness, and managing conflicts among participating organizations.

Federated learning has emerged as a promising solution to enable collaborative model training without direct data exchange [91]. Nevertheless, applying federated learning to LLMs introduces new challenges, such as ensuring the efficiency and scalability of the training process, preventing data leakage, and facilitating model interpretability and

accountability [75]. Moreover, organizations must contend with evolving data privacy regulations and the need for effective data governance mechanisms, including the ability to selectively remove specific data points or model contributions [42].

To address these challenges, we propose a novel hybrid blockchain-based federated learning framework with multi-agent interactions and unlearning capabilities for LLMs in cross-organizational settings. Our framework leverages the advantages of both public and private blockchains to create a safe, transparent, more efficient environment for multi-agent LLM training. The public blockchain establishes a decentralized and immutable ledger for recording model updates and transactions, fostering trust and accountability among participating organizations. Simultaneously, private blockchains enable organizations to securely share their proprietary data and model updates within their respective consortia, ensuring that sensitive information remains protected while allowing for collaborative learning.

Furthermore, we introduce a multi-agent system where each participating organization is represented by an intelligent agent. These agents employ Q-learning, a reinforcement learning technique, to make optimal decisions built on the current system state and the rewards received from their actions. This decentralized decision-making approach enables organizations to maintain granular control over their data and model contributions while collaborating towards a common goal.

In addition, we integrate an efficient unlearning mechanism based on the Low-Rank Adaptation (LoRA) technique [26], which allows for the selective removal of specific data points or model contributions without compromising the overall performance of the trained LLMs. This unlearning approach minimizes computational overhead and the impact on model accuracy while providing verifiable guarantees of data removal to satisfy regulatory requirements and maintain user trust.

To demonstrate the practical applicability of our framework, we present two case studies focusing on cross-organizational collaborations in the education and healthcare sectors. These case studies illustrate how our framework can be leveraged to develop LLMs for personalized learning, research assistance, medical decision support, and patient care while preserving data privacy, intellectual property rights, and regulatory compliance.

The main contributions of this chapter are:

- A novel hybrid blockchain-based federated learning framework with multi-agent interactions and unlearning capabilities for secure, transparent, and efficient collaborative LLM training in cross-organizational settings.

- A multi-agent system with Q-learning for decentralized decision-making, enabling organizations to maintain control over their data and model contributions while fostering collaboration.
- An efficient unlearning mechanism based on LoRA for selective data removal, ensuring compliance with data privacy regulations and maintaining user trust.
- Extensive experimental evaluations and case studies demonstrating the effectiveness and practical applicability of our framework in real-world scenarios.

6.2 Multi-Agent Systems and Q-Learning

Multi-agent systems (MAS) focus on the interactions of autonomous agents within a shared environment [77]. In MAS, each agent operates independently, making decisions based on local information and its own objectives. These agents interact with each other and their surroundings to optimize their strategies, often through cooperation, competition, or negotiation.

Q-learning is a reinforcement learning algorithm that enables agents to improve decision-making by learning from past interactions [76]. In a multi-agent Q-learning setting, each agent maintains a Q-table that records expected cumulative rewards (Q-values) for taking a specific action in a given state. The Q-values are updated iteratively using the following equation:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)] \tag{6.1}$$

where:

- s and a represent the current state and action.
- s' is the next state resulting from action a.
- r is the immediate reward received for taking action a in state s.
- α is the learning rate, determining how much new information overrides existing knowledge.
- γ is the discount factor, controlling the importance of future rewards.

Q-Learning in Federated Learning In our proposed framework, we employ multi-agent Q-learning to model the decision-making process of participating organizations in a federated learning environment. Each organization is treated as an agent, aiming to maximize its utility while contributing effectively to the collaborative training of a large language model (LLM). The Q-learning mechanism helps organizations determine the optimal level of participation by assessing the trade-offs between resource contribution, privacy concerns, and model performance.

State, Action, and Reward Definitions To apply Q-learning effectively in this setting, we define the state space, action space, and reward function as follows:

- **State** (s): Each organization‚Äôs local environment is represented by:
  - The current model accuracy on its local dataset.
  - The available computational resources (e.g., GPU capacity, memory usage).
  - The level of data sharing permitted within privacy constraints.
- **Action** (a): At each training round, an organization chooses one of the following:
  - The amount of local data to contribute to training ($\mathscr{D}_{\text{train}}$).
  - The learning rate (η) to use for updating model weights.
  - Whether to participate in unlearning (removing certain contributions upon request).
- **Reward** (r): After executing an action, an agent receives a reward based on:
  - **Model improvement:** A positive reward is assigned if the global model accuracy increases.
  - **Efficiency:** Efficient resource usage earns higher rewards.
  - **Compliance:** Successfully fulfilling unlearning requests provides additional incentives.

Decision Optimization Using Q-Learning Each agent updates its Q-values after each training round, gradually learning the most effective strategies over time. The policy (π) dictating action selection follows an epsilon-greedy exploration strategy:

$$\pi(a|s) = \begin{cases} \arg\max_a Q(s,a), & \text{with probability } (1-\epsilon) \\ \text{random action}, & \text{with probability } \epsilon \end{cases} \tag{6.2}$$

where the parameter ϵ ensures that the agent occasionally explores new strategies rather than always exploiting known high-reward actions.

Q-Learning in Blockchain-Integrated Federated Learning By integrating Q-learning into our federated learning framework, we achieve the following benefits:

- **Autonomous Optimization**: Organizations dynamically adjust their participation strategies based on observed outcomes.
- **Privacy-Aware Decision Making**: Agents balance data contribution with privacy constraints.
- **Incentive Alignment**: The reward structure encourages honest participation and discourages free-riding behavior.

This integration ensures that the federated learning process remains efficient, secure, and adaptive, allowing organizations to collaborate effectively while maintaining autonomy in decision-making.

6.3 Problem Definition and System Model

6.3.1 Problem Definition

The integration of Large Language Models (LLMs) with federated learning in cross-organizational collaborations presents several challenges related to data privacy, security, and efficiency. The main problems that need to be addressed are:

1. **Data Privacy and Security**: Organizations are often hesitant to share their sensitive data due to privacy concerns and the risk of data breaches. In a federated learning setting, it is crucial to ensure that each organization's data remains secure and confidential while still allowing for collaborative model training.
2. **Incentive Misalignment**: In a multi-agent setting, each organization has its own objectives and interests, which may not always align with the global goal of

the federated learning process. This can lead to strategic behavior, such as free-riding or withholding valuable data, which can hinder the overall performance and fairness of the collaboration.

3. **Efficiency and Scalability**: Training large language models can be computationally intensive, especially when dealing with multiple organizations and large datasets. Ensuring the efficiency and scalability of the federated learning process in a multi-agent setting is a significant challenge.

4. **Unlearning and Compliance**: As data privacy regulations evolve and organizations' data management requirements change, there is a growing need for efficient unlearning mechanisms. Unlearning allows organizations to selectively remove specific data or model contributions from the federated learning model while minimizing the impact on the overall model performance and ensuring compliance with legal and ethical standards.

5. **Transparency and Accountability**: Maintaining transparency and accountability in a multi-agent federated learning setting is crucial for building trust among participating organizations. Ensuring that all agents' contributions are properly recorded, and the learning process is verifiable is essential for preventing malicious behavior and resolving disputes.

To address these challenges, we propose a hybrid blockchain-based multi-agent framework that leverages the strengths of both public and private blockchains while incorporating multi-agent interactions and unlearning capabilities to align incentives, ensure compliance, and optimize agents' strategies in the federated learning process.

6.3.2 System Model

Our system model consists of four main components: Agents, Public Blockchain, Private Blockchain, and Multi-Agent Interactions. These components work together to facilitate secure, efficient, and incentive-aligned federated learning for LLMs in cross-organizational collaborations.

- **Agents**: Agents represent the participating organizations in the federated learning process. Each agent has its own local dataset, computational resources, and objectives. Agents aim to maximize their individual utilities while contributing to the collaborative learning process. The utility of an agent can be defined as a

function of its local model performance, the rewards received from the system, and the costs incurred during the learning process.

- **Public Blockchain**: The public blockchain serves as a transparent and immutable ledger for recording the global model updates and the overall progress of the federated learning process. It ensures its integrity and accountability of the collaboration by providing a tamper-proof record of all transactions and contributions made by the agents. The framework utilizes the public blockchain, which also hosts smart contracts that govern the incentive mechanism and the overall rules of the federated learning process, to ensure integrity and protect against poisoning attacks without significantly impacting the overall performance of the learning process.
- **Private Blockchain**: Each agent maintains its own private blockchain to securely store and manage its local dataset, model updates, and other sensitive information. The private blockchain ensures data privacy and confidentiality by restricting access to authorized parties within the organization. It also facilitates efficient local model training and secure communication with the public blockchain for global model updates.
- **Multi-Agent Interactions**: Agents interact with each other and with the blockchain network to optimize their strategies and contribute to the federated learning process. These interactions include:
  - *Local Model Training*: Agents perform local model training on their private datasets using their computational resources and Q-learning strategies to optimize their contributions.
  - *Model Update Sharing*: Agents share their local model updates with the private blockchain, which aggregates the updates and communicates with the public blockchain for global model aggregation.
  - *Unlearning Requests*: Agents can request the removal of data level points or model contributions from the federated learning model through the unlearning process, which leverages LoRA for efficient and targeted data removal.
  - *Reward Distribution*: Agents receive rewards based on their contributions to the federated learning process, as determined by the incentive mechanism implemented through smart contracts on the public blockchain.

The interaction between these components can be summarized as follows:

1. Agents register on the public blockchain and establish their private blockchains.
2. The global model is initialized on the public blockchain, and the federated learning process begins.
3. Each agent performs local model training on its private dataset using Q-learning strategies to optimize its contributions.
4. Agents share their local model updates with their private blockchains, which aggregate the updates and communicate with the public blockchain for global model aggregation.
5. If an agent requests the removal of specific data or model contributions, the unlearning process is triggered, leveraging LoRA for efficient and targeted data removal.
6. The public blockchain aggregates the model updates from the private blockchains and updates the global model according to the federated learning algorithm.
7. The incentive mechanism distributes rewards to the agents based on their contributions to the federated learning process.
8. The process repeats from step 3 until the global model converges or a predetermined stopping criterion is met.

By combining the hybrid blockchain architecture with multi-agent interactions and unlearning capabilities, our system model provides a secure, transparent, and incentive-aligned framework for federated learning with LLMs in cross-organizational collaborations. The Q-learning strategies employed by the agents enable them to optimize their contributions based on the rewards and penalties defined by the incentive mechanism, promoting honest participation and efficient resource allocation in the federated learning process.

6.4 Proposed Framework

6.4.1 Overview

Our proposed framework introduces a novel hybrid blockchain architecture that seamlessly integrates public and private blockchains to facilitate secure and efficient cross-organizational collaboration using Large Language Models (LLMs). The framework

ensures transparency, traceability, and data privacy protection while enabling the effective sharing and utilization of data across multiple organizations. The key components of our framework include client registration, global model upload, private blockchain establishment, federated learning training process, private blockchain aggregation, unlearning process using LoRA for forgetting, unlearning verification and submitting unlearning results, and public blockchain aggregation.

In our framework, we introduce a multi-agent system where each participating organization is represented by an agent. These agents are responsible for managing the local training process, contributing to the global model, and interacting with the blockchain network. The agents employ Q-learning, a reinforcement learning technique, to make optimal decisions based on the current state of the system and the rewards received for their actions.

Figure 6.1 illustrates the procedure of our proposed federated learning system incorporating public and private blockchains along with multi-agent interactions, addressing the massive computing challenges in future IoE scenarios. During registration, clients specify their affiliated organization or company. Upon completing registration, the agent uploads the global model to the public chain. Companies with numerous clients then establish private chains to enhance the efficiency of the federated learning training process. The training epoch for federated learning is specified within the smart contract. When the training process in the private chain reaches the designated epoch, model aggregation begins within the private chain.

Once the private model, LLM_p, is prepared, it is sent to the public chain for further aggregation. If an organization requests the removal of their data or model updates, the unlearning process is initiated using LoRA to facilitate forgetting. The results of the unlearning process are verified and then submitted to the public chain. After completing the model aggregation and unlearning verification processes, the smart contract updates the final global model, LLM_f, which the agent can subsequently obtain. The following sections provide a detailed explanation of each component within our proposed framework.

6.4.2 Client and Agent Register

The client and agent registration process is the initial step in our framework. Each participating organization must register as a client to join the collaborative network, while agents are responsible for managing the local training process and interacting with the blockchain network. During registration, both clients and agents provide their

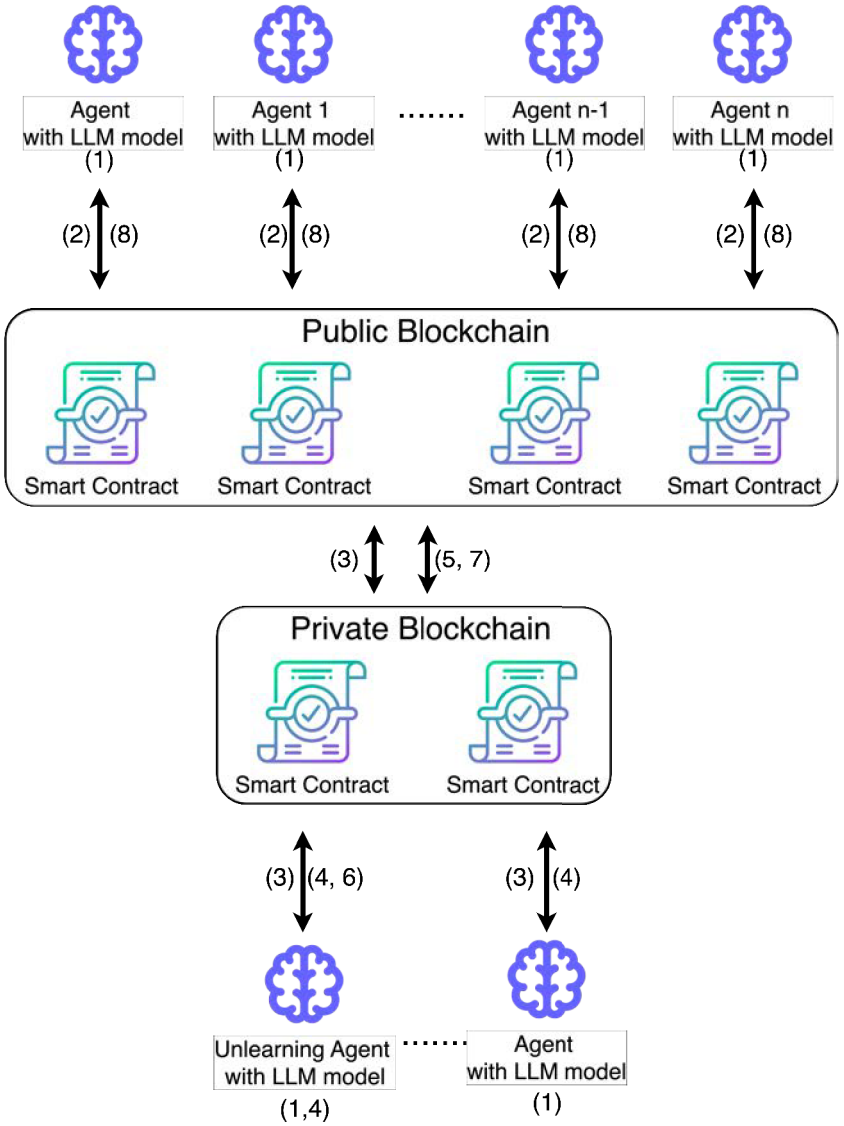

Figure 6.1: Overview and process of our proposed system. (1) Client register. (2) Global model upload. (3) Private blockchain establish. (4) Federated learning training process. (5) Private blockchain aggregation. (6) Unlearning process using LoRA. (7) Unlearning verification and submitting. (8) Public blockchain aggregation.

unique identifiers and establish secure communication channels using cryptographic techniques such as public-key cryptography.

Algorithm 14 Client and Agent Register

Require: $E_{name}, Role, Org$
Ensure: $RegSuccess, jwt$
$RegSuccess$ = False;
Check Org;
if $E_{name} \in E_{pool}$ **then**
  return E_{name} already existed.
end if
$P_k, S_k \leftarrow keyGen()$;
$jwt \leftarrow P_k, S_k$;
$E_{name} \leftarrow jwt$;
if $Role$ is $Client$ **then**
  $Pool_c \leftarrow E_{pool} \cup ID_{ci}$;
else if $Role$ is $Agent$ **then**
  $Pool_a \leftarrow E_{pool} \cup ID_{ai}$;
end if
$RegSuccess$ = True;
return $RegSuccess, jwt$

The algorithm begins by checking if the entity's unique identifier (E_{name}) is already present in the entity pool (E_{pool}). If the identifier exists, the registration halts, indicating that the entity already exists. The algorithm also verifies the organization (Org) associated with the entity.

If the entity is new, the algorithm proceeds to generate a public-secret key pair using the $keyGen()$ function. With these keys, it then creates a JSON Web Token (JWT) for the entity. This JWT, along with the entity ID, is securely stored, effectively registering the entity.

Finally, the $RegSuccess$ indicator is set to true, and both $RegSuccess$ and the generated jwt are returned, signifying the entity's successful registration and their secure token for future communications.

By incorporating both client and agent registration in this process, the algorithm ensures that all participating entities are properly authenticated and authorized to engage in the federated learning process while maintaining the security and integrity of the system.

6.4.3 Global Model Upload

After successful registration, the agent could update the global model to this framework public chain, ensuring that all participating organizations have access to the initial model for collaborative training.

Algorithm 15 Global Model Upload

Require: jwt, LLM_g
Ensure: $UploadSuccess, LLM$
1: $UploadSuccess$ = False;
2: **if** jwt is invalid **then**
3: **return** jwt expired
4: **end if**
5: $LLM \leftarrow LLM_g$;
6: $UploadSuccess$ = True;
7: **return** $UploadSuccess, LLM$

The algorithm starts by verifying the validity of the agent's JWT. If the token is invalid or has expired, the process is terminated, and an error message is returned. Upon successful authentication, the global LLM_g is uploaded to the public-sector blockchain as LLM. The process is straightforward and includes necessary security checks to maintain the integrity of the system.

6.4.4 Private Blockchain Establish

The private blockchain serves as a secure and tamper-proof ledger for storing and managing the organization's sensitive data and model updates. It ensures data privacy by restricting access to authorized parties within the organization.

Algorithm 16 Private Blockchain Establish

Require: jwt, LLM_g
Ensure: $EstablishSuccess, LLM_p$
1: $EstablishSuccess$ = False;
2: **if** jwt is invalid **then**
3: **return** jwt expired
4: **end if**
5: $LLM_p \leftarrow LLM_g$;
6: $EstablishSuccess$ = True;
7: **return** $EstablishSuccess, LLM_p$

The algorithm verifies the validity of the organization's JWT. If the token is invalid or has expired, the process is terminated, and an error message is returned. Upon successful authentication, the LLM_g is uploaded to the private chain as LLM_p.

This part of the framework ensures that organizations with a large number of clients can establish their private blockchains to securely store and manage their sensitive data and model updates. The process includes necessary security checks to maintain the privacy and integrity of the organization's data while allowing them to participate in the federated learning process.

6.4.5 Multi-Agent Federated Learning Process on Private Chain

This process on the private chain enables organizations with a large number of clients to collaboratively train the LLM without directly sharing their sensitive data. Each agent within the organization participates in the training process by leveraging its local data and computational resources. The agents train the model locally and share only the model updates with the organization's private chain. This approach ensures data privacy while benefiting from the collective knowledge of all agents within the organization.

Algorithm 17 Multi-Agent Federated Learning Process on Private Chain

Require: $Private_{epoch}, LLM_p$
Ensure: $TrainSuccess$
$TrainSuccess$ = False;
for $epoch = 1$ to $Private_{epoch}$ **do**
  for each agent A_i in the organization **do**
    A_i receives LLM_p from the private chain
    A_i trains LLM_p using local data and Q-learning strategy
    A_i sends updated model $LLM_{p,i}$ to the private chain
  end for
  Aggregate $LLM_{p,i}$ from all agents to update LLM_p on the private chain
end for
$TrainSuccess$ = True;
return $TrainSuccess$

The algorithm begins by setting the $TrainSuccess$ indicator to false. It then iterates for the specified number of $Private_{epoch}$. Within each epoch, the algorithm loops through each agent A_i in the organization. Each agent receives the current LLM_p from the private chain, trains the model using its local data and Q-learning strategy, and sends the updated model $LLM_{p,i}$ back to the private chain. The Q-learning strategy enables agents to make optimal decisions based on the current state of the system and the

rewards received for their actions, such as contributing high-quality data or model updates.

After all agents have completed their training for the current epoch, the algorithm aggregates the updated models $LLM_{p,i}$ from all agents to update the LLM_p on the private chain.

This modified version of the Training emphasizes the multi-agent approach, where each agent within the organization contributes to the collaborative learning process using Q-learning strategies. The training occurs on the private chain, allowing organizations with a large number of clients to maintain data privacy and security while benefiting from the collective knowledge of their agents.

6.4.6 Private Blockchain Aggregation

During this step, each agent's model updates are securely stored and aggregated within their respective organization's private blockchain. The private blockchain aggregation mechanism ensures the integrity and traceability of the model updates. It allows each organization to maintain a transparent record of their agents' contributions to the collaborative model while preserving the confidentiality of their sensitive data.

Algorithm 18 Private Blockchain Aggregation

Require: $jwt, Private_{epoch}, LLM_p, Agg_p$
Ensure: $AggSuccess$
Agg_p = False;
SC realizes that the private chain has reached $Private_{epoch}$;
if N_p reaches $Private_{epoch}$ **then**
  SC aggregates LLM_p from all agents;
  $LLM \leftarrow LLM_p$;
  if jwt is invalid **then**
    return jwt expired
  end if
end if
SC sends LLM to $PublicChain$;
Agg_p = True;
return Agg_p, LLM

When the number of epochs N_p reaches the $Private_{epoch}$ setting, the smart contract (SC) aggregates the model updates from all agents on the private chain to obtain LLM_p. After verifying the JWT and preparing the aggregated model, it is sent for further

aggregation. The private chain aggregation indicator, Agg_p, is set to true, and both Agg_p and the model LLM are returned.

6.4.7 Unlearning Process Using LoRA for Forgetting

The unlearning process enables the selective removal of specific data or model contributions from the federated learning model. When an organization requests to remove their data or model updates, the unlearning process is triggered. We employ Low-Rank Adaptation (LoRA) technology to efficiently forget the specified data without compromising the overall model performance. The unlearning process ensures data privacy and compliance with regulatory requirements.

Algorithm 19 Unlearning Process using LoRA for Forgetting

Require: LLM_g, D_{forget}, Learning rate η, Unlearning epochs E_u, LoRA parameters λ
Ensure: $params$

Unlearning Request due to data sensitivity or correction needs;
Initialize unlearning model LLM_{local} with LLM_g;
Adapter A constructed for LLM_{local} targeting forgetting process;
for $epoch = 1$ to E_u **do**
    Forward pass with D_{forget} through LLM_{local} to identify features to forget;
    Compute gradients for LLM_{local} emphasizing data points in D_{forget} to be forgotten;
    Apply LoRA to adjust gradients of adapter A using parameters λ, focusing on unlearning;
    Update LLM_{local}'s parameters using the adjusted gradients and learning rate η, facilitating forgetting;
end for
Calculate the updating $params$ indicative of the forgetting process between LLM_{local} and LLM_g;
return $params$

The process begins by initializing a local version of the LLM, denoted as LLM_{local}, with the parameters of LLM_g. An adapter, A, is constructed within LLM_{local} specifically designed to target and facilitate the forgetting of the specified dataset, D_{forget}.

The core of the unlearning process involves several epochs of training, defined by the parameter E_u. In each epoch, a forward pass of D_{forget} through LLM_{local} is performed to identify the features associated with the data points that need to be forgotten. Gradients are computed for LLM_{local} with an emphasis on the data to be unlearned. The LoRA technique is applied to the adapter A's gradients using parameters λ to focus the unlearning process. With the adjusted gradients, LLM_{local}'s parameters are updated

using the specified learning rate η. This iterative process gradually leads to the forgetting of the specified data points from D_{forget}.

Upon completion of the unlearning epochs, the algorithm calculates the parameters $params$ that indicate the changes made to LLM_{local} in comparison to LLM_g. These parameters represent the outcome of the forgetting process, effectively capturing the essence of what has been unlearned. The algorithm concludes by returning these updated parameters.

6.4.8 Unlearning Verification and Submitting Unlearning Results

The unlearning verification and submission process ensures the integrity and transparency of the unlearning results in the federated learning model. The process involves the agent sending the updated parameters, resulting from the unlearning process, to the SC. The SC validates the agent's credentials and evaluates the unlearning results using a validation dataset. If the unlearning results satisfy the verification criteria, the SC submits the updated parameters to the blockchain network.

Algorithm 20 Unlearning Verification and Submitting Unlearning Results

Require: $params$, Validation dataset D_{val}, Agent
Ensure: T_{id}

Agent sends $params$ to SC;
if Agent's jwt is invalid **then**
    return Agent identity check failed
end if
SC instantiates updated LLM $LLM_{updated}$ with received $params$;
SC uses D_{val} to evaluate $LLM_{updated}$. Calculates training loss and accuracy to measure unlearning impact.
if Verification criteria are met **then**
    SC sends $params$ to blockchain network;
    Agents download $params$ from blockchain for weight integration;
    SC ensures updated weights are recorded on blockchain for transparency and traceability;
    SC records Transaction ID T_{id} as proof of submission and integration request;
end if
Continue for future federated learning process;
return T_{id}

The algorithm starts with the agent sending the updated parameters $params$ to the SC. The agent's credentials are validated through their JWT. If the token is invalid,

the process halts, indicating a failure in agent identity verification. Upon successful verification, the SC initializes an updated version of the LLM ($LLM_{updated}$) with the new parameters. The SC employs a validation dataset (D_{val}) to assess the efficacy of the unlearning process by calculating the training loss and accuracy.

If the unlearning results satisfy the predefined verification criteria, the SC submits the updated parameters $params$ to the blockchain network. Agents download these parameters from the blockchain for weight integration into the global model. The SC ensures that the updated weights are recorded on the blockchain, providing transparency and traceability. Additionally, the SC logs a Transaction ID (T_{id}), serving as proof of submission and an integration request.

The process concludes with the return of the Transaction ID, signifying the successful verification and submission of the unlearning results.

6.4.9 Public Blockchain Aggregation

The public blockchain aggregation component facilitates the secure and transparent aggregation of the model updates from all participating organizations. The central server collects the model updates from each organization's private blockchain and aggregates them using secure aggregation techniques. The aggregated model updates are then stored in the public sector among proposed framework with transparency. This immutable record of the collaborative learning process enhances trust among the participating organizations.

Algorithm 21 Public Blockchain Aggregation

Require: LLM, jwt, Agg_g
Ensure: $epoch$, Agg_p
1: Agg_g = False;
2: **if** N_g reaches $epoch$ **then**
3: SC aggregates LLM from all organizations;
4: $LLM_g \leftarrow LLM$;
5: **if** jwt is invalid **then**
6: **return** jwt expired
7: **end if**
8: **end if**
9: Agg_g = True;
10: **return** Agg_g, LLM_g

This completes the detailed explanation of our proposed framework, which leverages a hybrid blockchain architecture to facilitate secure and efficient cross-organizational

collaboration using Large Language Models (LLMs) while ensuring data privacy, transparency, and traceability.

6.4.10 Case Studies

To demonstrate the practical applicability and versatility of our proposed framework, we present two case studies that highlight its potential in real-world scenarios. These case studies illustrate how our blockchain-based federated learning framework with unlearning capabilities and multi-agent interactions can be leveraged to address the unique challenges faced by different industries, such as education and healthcare, when collaborating on LLM development.

6.4.10.1 Case Study 1: Education University Alliance

In the first case study, we consider an alliance of universities collaborating to develop an LLM for educational purposes. The LLM aims to assist students, faculty, and researchers by providing personalized learning experiences, intelligent tutoring, and advanced research assistance. Each university possesses a wealth of educational data, including course materials, student interactions, and research publications. However, sharing this data directly among the universities raises concerns about data privacy, intellectual property rights, and the potential misuse of sensitive information.

Implementation of the System To realize the education university alliance case study, we adopt our proposed blockchain-based federated learning framework with multi-agent interactions. First, each participating university registers as a client in the system, as shown in Algorithm 1. Then, one university is selected as the agent to upload the initial global LLM model to the public blockchain, as demonstrated in Algorithm 2. Universities with large amounts of data establish their own private blockchains to ensure the privacy and security of their sensitive data (Algorithm 3).

On the private blockchain, each university is represented by an agent that trains the LLM using its local data and Q-learning strategies to make optimal decisions (Algorithm 5). After multiple rounds of training, the aggregated LLM is shared on the private blockchain and sent to the public blockchain to aggregate the global model (Algorithm 6).

If a university needs to remove specific data points or model contributions, the unlearning process is triggered (Algorithm 4). The LoRA technique is used to selectively remove data without significance effect on the final accuracy of the LLM. The unlearning

results are verified and submitted to the public blockchain (Algorithm 5), ensuring the integration and transparency of the removal request.

Analysis By utilizing our framework, the education university alliance can leverage the collective knowledge and expertise of multiple institutions while preserving data privacy and intellectual property rights. The resulting educational LLM can offer advanced learning experiences and research support to students, faculty, and researchers across the participating universities. The private blockchain ensures that each university's sensitive data remains protected, while the public blockchain facilitates secure collaboration among the different institutions. The multi-agent approach allows each university to make optimal local decision-making model and Q-learning strategies, imporving the accurancy of the LLM. The LoRA-driven unlearning mechanism allows universities to effectively remove specific data as needed while maintaining the overall performance of the LLM.

Challenges and Solutions One major challenge in implementing the education university alliance case study is coordinating data sharing and model updates among the different universities. Each university may have varying data formats, privacy requirements, and technical infrastructures. To address this challenge, our framework provides a standardized interface for data sharing and model aggregation, streamlining the collaboration process across different institutions.

Another challenge is ensuring that the unlearning process complies with each university's data retention policies and regulatory requirements. Our framework offers a flexible and verifiable approach to manage diverse data retention needs by using LoRA for fine-grained data removal and verifying the unlearning results on the blockchain.

Conclusion The education university alliance case study demonstrates the potential application of our blockchain-based federated learning framework with multi-agent interactions in the education domain. By allowing universities to collaborate while protecting data privacy and intellectual property rights, our framework paves the way for developing LLMs for personalized learning, intelligent tutoring, and advanced research. The multi-agent approach enables universities to make optimal decisions based on their local data and learning strategies, while the unlearning mechanism provided by the framework enables universities to manage their data retention policies while ensuring compliance and transparency.

6.4.10.2 Case Study 2: Cross-Hospital Collaboration in Healthcare

The second case study focuses on the collaboration among different hospitals within a healthcare system to develop an LLM for medical decision support and patient care. The LLM aims to assist healthcare professionals by providing evidence-based recommendations, analyzing patient data, and facilitating knowledge sharing among hospitals. Each hospital has a vast repository of medical records, including patient histories, diagnostic images, and treatment outcomes. However, the sensitive nature of this data and the strict regulations governing healthcare information pose significant challenges for collaboration.

Implementation of the System To realize the cross-hospital collaboration case study, we adopt a similar approach to the education university alliance case study. Each hospital registers as a client in the system, while one hospital is selected as the agent to upload the initial global LLM to the public blockchain. Hospitals with large amounts of patient data establish their own private blockchains to ensure data privacy and security.

On the private blockchain, each hospital is represented by an agent that trains the LLM using its local patient data and Q-learning strategies to make optimal decisions. The agents collaborate to improve the LLM while maintaining data privacy. The aggregated LLM is shared on the private blockchain. If specific patient data or model contributions need to be removed, hospitals can trigger the unlearning process, using LoRA to selectively remove data without affecting the overall performance of the LLM.

Analysis By adopting our framework, the cross-hospital collaboration in healthcare can leverage the collective knowledge and expertise of multiple institutions to develop a powerful medical LLM. The resulting LLM can assist healthcare professionals in making informed decisions, improving patient outcomes, and advancing medical research while maintaining the highest standards of data privacy and regulatory compliance. The private blockchain ensures that each hospital's sensitive patient data remains protected, while the public blockchain facilitates secure collaboration among the hospitals. The multi-agent approach allows hospitals to make optimal decisions based on their local data and Q-learning strategies. The LoRA-driven unlearning mechanism allows hospitals to effectively remove specific data as needed while preserving the integrity of the medical LLM.

Challenges and Solutions One major challenge in implementing the cross-hospital collaboration case study is ensuring compliance with strict healthcare regulations, such as HIPAA. Our framework addresses this challenge by using private blockchains to isolate sensitive patient data and leveraging secure aggregation protocols to share model updates among hospitals. The blockchain technology also provides an immutable audit trail of data access and sharing activities, ensuring compliance.

Another challenge is managing the different data retention policies and patient consent requirements across hospitals. Our framework offers a flexible and verifiable approach to handle these variations by using LoRA for fine-grained data removal and verifying the unlearning results on the blockchain. This allows hospitals to customize the unlearning process based on their specific data management requirements.

Conclusion The cross-hospital collaboration case study in healthcare demonstrates the potential application of our blockchain-based federated learning framework with multi-agent interactions in the medical domain. By allowing hospitals to collaborate while maintaining patient privacy and regulatory compliance, our framework paves the way for developing LLMs for medical decision support, patient care, and medical research. The multi-agent approach enables hospitals to make optimal decisions based on their local data and learning strategies, while the unlearning mechanism provided by the framework enables hospitals to manage their data retention policies while ensuring compliance and accountability. These two case studies showcase the wide-ranging applicability of our blockchain-based federated learning framework with multi-agent interactions in real-world scenarios. By addressing the unique challenges faced by different industries, our framework provides a viable path for the responsible development and deployment of LLMs in critical domains such as healthcare and education.

6.5 Privacy and Security Analysis

6.5.1 Privacy Analysis

Proposed framework with multi-agent interactions and unlearning capabilities for Large Language Models (LLMs) is meticulously designed to address the critical privacy challenges associated with collaborative learning in cross-organizational settings. By synergistically integrating the inherent privacy-preserving features of federated learning, the immutability and transparency of blockchain technology, and the efficient data re-

moval mechanisms of unlearning, our approach offers a holistic solution for secure and privacy-centric LLM training.

At its core, federated learning enables the distributed training of LLMs across multiple participants without necessitating sensitive data and its the direct exchange [88]. This decentralized paradigm ensures that each participant maintains control over their proprietary data, significantly mitigating the risks of data breaches and unauthorized access. Mathematically, federated learning can be formulated as an optimization problem that seeks to minimize the global objective function while keeping the data localized:

$$\min_{\theta} \mathscr{L}(\theta) = \sum_{i=1}^{N} \frac{n_i}{n} \mathscr{L}_i(\theta) \tag{6.3}$$

where θ represents the model parameters, $\mathscr{L}(\theta)$ denotes the global objective function, where $\mathscr{L}_i(\theta)$ is the local objective function of the i-th participant, n_i signifies the number of data samples held by the i-th participant, N is the total number of participants, and n is the total number of data samples across all participants.

By optimizing the global objective function in this manner, federated learning facilitates the collaborative enhancement of the LLM without exposing raw data, effectively leveraging the distributed data across participants while safeguarding privacy and boosting model performance.

The introduction of multi-agent interactions and Q-learning strategies in our framework further enhances privacy protection by enabling participating organizations to make intelligent decisions about data sharing and model contributions based on their individual privacy constraints and objectives. Each agent autonomously learns the optimal strategies for engaging in the federated learning process, considering factors such as data sensitivity, regulatory compliance, and the potential benefits and risks of collaboration. This decentralized decision-making approach empowers organizations to maintain granular control over their data and reduces the reliance on centralized control mechanisms that may introduce additional privacy vulnerabilities.

Moreover, the integration of blockchain technology in our framework provides an immutable and transparent ledger of all interactions and transactions within the federated learning process, ensuring the integrity and accountability of the collaborative learning process. This automation minimizes the potential for human error and mitigates the risk of unauthorized data access or manipulation.

The unlearning mechanism embedded within our framework empowers participants to selectively remove specific data points or model updates, granting them fine-grained control over their data lifecycle and facilitating compliance with evolving privacy regula-

tions. Theoretically, the unlearning process can be modeled as a constrained optimization problem, wherein the objective is to minimize the impact of the removed data on the model's performance while satisfying the unlearning constraints:

$$\min_{\theta} \mathscr{L}(\theta) = \sum_{i=1}^{N} \frac{n_i}{n} \mathscr{L}_i(\theta) \quad \text{s.t.} \quad \theta \in \Theta_u \tag{6.4}$$

where Θ_u represents the feasible set of model parameters after unlearning. The goal is to identify the optimal model parameters while adhering to the unlearning constraints. By incorporating this unlearning mechanism, our framework provides participants with a powerful tool to manage their data lifecycle and maintain model performance.

6.5.2 Security Analysis

The integration of blockchain technology and multi-agent interactions in our federated learning framework. The immutable nature of blockchain ensures that all model updates and transactions are tamper-proof and easily verifiable, providing a robust defense against malicious actors attempting to manipulate the learning process [55].

From a theoretical perspective, the security of a blockchain network can be analyzed through the lens of game theory and consensus mechanisms. In a proof-of-stake (PoS) based blockchain, network security is maintained by requiring participants to stake a portion of their assets as collateral [3]. This staking mechanism incentivizes participants to act honestly, as any malicious behavior would result in the loss of their staked assets. The security of the network can be modeled as a game between honest and malicious participants, where honest participants aim to maximize their rewards by following the protocol, while malicious participants seek to maximize their gains by deviating from the protocol. The Nash equilibrium of this game represents a state in which no participant can benefit by unilaterally altering their strategy, ensuring the stability and security of the blockchain network.

The multi-agent approach introduced in our framework adds an extra layer of security by enabling participating organizations to independently assess the credibility and trustworthiness of other agents based on their past behavior and contributions. Agents can learn to identify and isolate malicious or free-riding participants, minimizing their impact on the collaborative learning process. This decentralized trust mechanism complements the security features of the blockchain, creating a more resilient and adaptive system that can effectively respond to evolving security threats.

Furthermore, the Q-learning strategies employed by the agents allow them to dynamically adapt their behavior based on the observed security state of the system. Agents can

learn to take proactive measures, such as increasing the frequency of model validations or adjusting the staking requirements, to integrate of the federated learning process in the face of potential attacks. This adaptive security approach enables the system to remain robust and responsive even in the presence of sophisticated adversaries.

Our framework also leverages advanced cryptographic primitives, such as threshold signatures and zero-knowledge proofs, to ensure the integrity and confidentiality of all transactions. Threshold signatures allow for the distributed generation and verification of signatures, eliminating single points of failure and enhancing the resilience of the system against attacks. Zero-knowledge proofs enable participants to validate the correctness of computations without revealing the underlying data [67], preserving privacy while maintaining trust in the federated learning process. By combining these cryptographic techniques with the security features of blockchain and multi-agent interactions, our framework establishes a secure and trustworthy environment for collaborative LLM development.

The utilization of smart contracts further strengthens the security of the system by automating the execution of predefined rules and conditions, minimizing the potential for unauthorized access or manipulation. In our framework, smart contracts govern the federated learning process, enforcing participant adherence to agreed-upon security protocols and facilitating the secure aggregation of model updates. This automated enforcement reduces the risk of human error and malicious behavior, bolstering the overall security of the system.

The decentralized architecture of our framework, enabled by the hybrid blockchain design, eliminates single points of failure and distributes risk across multiple nodes. This distributed approach significantly increases the difficulty for attackers to compromise the entire system, as they would need to control a substantial portion of the participating nodes simultaneously. The probability of a successful attack decreases exponentially with the number of honest nodes in the network, making it practically infeasible in a large-scale, cross-organizational federated learning setting.

In conclusion, our hybrid blockchain-based federated learning framework with multi-agent interactions and unlearning capabilities offers a comprehensive solution for addressing security concerns in cross-organizational LLM training. By harnessing the inherent security features of blockchain technology, multi-agent interactions, and advanced cryptographic techniques, our approach creates a resilient and secure environment for collaborative LLM development. The adaptive security measures enabled by Q-learning strategies and the decentralized trust mechanism further fortify the system's defenses

against evolving security threats, ensuring the integrity and reliability of the federated learning process in complex, multi-stakeholder settings.

6.6 Performance Evaluation

In this section, we present a comprehensive evaluation of our proposed multi-agent federated learning framework augmented with blockchain technology. We focus specifically on how different LoRA configurations impact the efficacy of the unlearning process when applied to a GPT-2 model. Our primary objective is to investigate how various settings within LoRA influence the model's ability to selectively erase data, a critical feature for maintaining data privacy and complying with dynamic regulatory requirements. We measure the effectiveness of each configuration by examining changes in model accuracy, providing a clear metric for performance comparison across different configurations. This thorough evaluation not only demonstrates the practical implications of our approach but also helps identify the optimal LoRA settings that enhance unlearning capabilities without compromising the overall accuracy of the GPT-2 model.

6.6.1 Experimental Setup

To validate the performance of our innovative multi-agent blockchain-enhanced federated learning system with unlearning capabilities, we conducted a series of experiments. These experiments focused on how different LoRA configurations affect unlearning performance. Our main goal was to measure the system's ability to selectively forget certain data points while maintaining overall model accuracy within a collaborative multi-agent environment.

Datasets: For our experiments, we utilized two distinct datasets: the IMDB dataset for sentiment analysis and a dataset of tweets from Twitter. These datasets were selected based on several important criteria:

- **Relevance to LLM Applications:** The IMDB dataset is a standard benchmark for sentiment analysis, making it ideal for evaluating the performance of our framework in a well-known context. The Twitter dataset, on the other hand, provides real-world text data applicable to various NLP tasks such as sentiment analysis, topic modeling, and text classification.
- **Size and Complexity:** The IMDB dataset consists of 50,000 movie reviews, offering a substantial but manageable volume for federated learning experiments.

The Twitter dataset includes a large number of tweets, which helps assess the scalability and efficiency of our framework.

- **Diversity of Content:** The IMDB dataset contains structured, domain-specific reviews, while the Twitter dataset includes a wide range of topics, opinions, and writing styles. This diversity allows us to test the robustness and adaptability of our framework.
- **Sensitive Information:** Tweets often contain personal or sensitive information that users might wish to erase. This makes the Twitter dataset particularly suitable for evaluating our unlearning mechanism's effectiveness in removing specific data points while preserving overall model performance.

Evaluation Metrics: Accuracy was chosen as the primary metric for evaluating our experiments. It provides a straightforward measure of the proportion of correctly classified reviews post-unlearning. By comparing accuracy before and after unlearning, we can gauge the framework's effectiveness in forgetting specific data points while retaining overall model performance.

Experimental Comparisons: Given the unique combination of federated learning, blockchain technology, and unlearning capabilities in our approach, there are no direct counterparts in the current literature for comparison. Our framework is pioneering in addressing selective unlearning within a federated learning setup for LLMs, enhanced with blockchain to ensure privacy and security.

For comprehensive evaluation, we varied LoRA hyperparameters such as rank and scaling factor to identify the configurations that best balance unlearning effectiveness and model accuracy. The multi-agent system configuration in our experiments involved each agent (representing an organization) training locally on its data and collaboratively updating the global model via blockchain to ensure transparency and accountability.

Hardware and Software Environment: The experiments were run on a system with an Intel Xeon 6238R processor, 64GB RAM, and an NVIDIA A6000 GPU, using software environments like Ubuntu 20.04, Visual Studio Code, Hyperledger Fabric, FATE, and machine learning frameworks such as PyTorch and TensorFlow.

6.6.2 Results and Discussion

This section details the outcomes of our experiments, comparing the efficacy of our method with the Retrain from Scratch technique. We focus on the variations in per-

formance due to different LoRA configurations and analyze the reasons behind any performance differences.

IMDB Dataset Results: We experimented with various LoRA settings on the IMDB dataset. The Retrain from Scratch method served as a benchmark to evaluate our unlearning approach. Table 6.1 presents the initial and final accuracies for different LoRA configurations.

Table 6.1: Results on IMDB Dataset

LoRA Config	Initial Accuracy	Final Accuracy
r=32, alpha=2, dropout=0.1	97.10%	0.95%
r=16, alpha=8, dropout=0.1	95.95%	1.00%
r=16, alpha=4, dropout=0.5	98.00%	1.10%
r=16, alpha=4, dropout=0.2	98.70%	1.20%
r=2, alpha=16, dropout=0.1	84.41%	1.25%

Twitter Dataset Results: Similarly, we conducted experiments on the Twitter dataset to assess our method's performance. Table 6.2 shows the results of these experiments.

Table 6.2: Results on Twitter Dataset

LoRA Config	Initial Accuracy	Final Accuracy
r=16, alpha=8, dropout=0.2	83.98%	7.93%
r=16, alpha=1, dropout=0.1	84.68%	8.33%
r=2, alpha=1, dropout=0.3	74.42%	8.41%
r=16, alpha=2, dropout=0.4	81.33%	8.42%
r=8, alpha=1, dropout=0.1	91.00%	9.04%

6.6.2.1 Impact of Rank on Unlearning

Figures 6.2 and 6.3 illustrate how different ranks (r) affect accuracy reduction for the Twitter and IMDB datasets, respectively. Higher ranks, such as $r = 16$ and $r = 32$, generally result in more significant accuracy reductions post-unlearning, indicating better unlearning performance.

6.6.2.2 Impact of Alpha on Unlearning

Figures 6.4 and 6.5 show the effects of different alpha values on accuracy reduction for the Twitter and IMDB datasets. Lower alpha values generally lead to more effective unlearning, as evidenced by greater accuracy reductions.

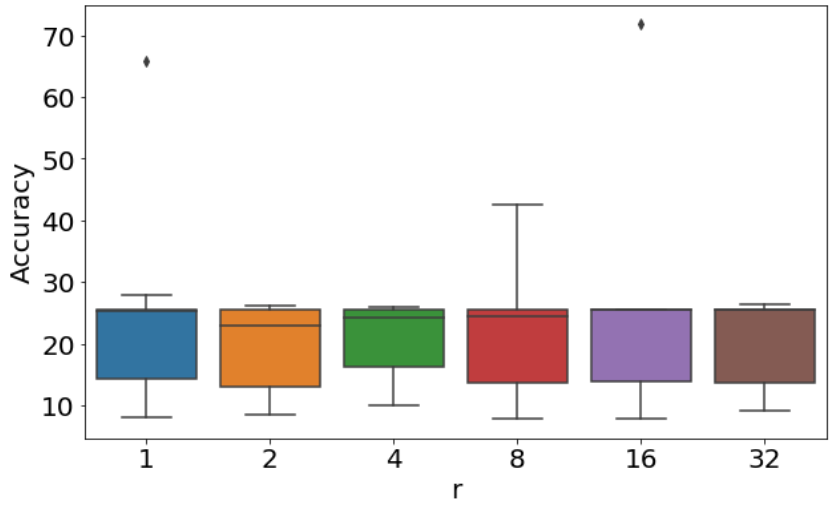

Figure 6.2: Impact of Different r Values on Accuracy (Twitter)

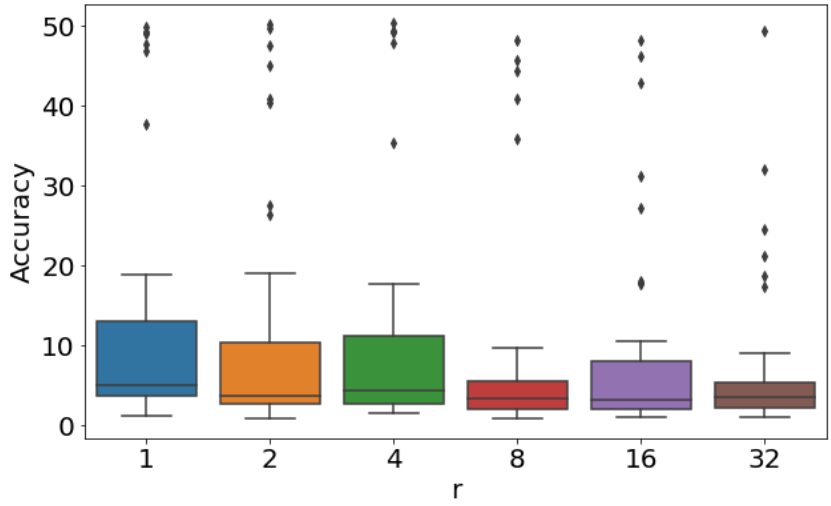

Figure 6.3: Impact of Different r Values on Accuracy (IMDB)

6.6.2.3 Impact of Dropout on Unlearning

Figures 6.6 and 6.7 depict how different dropout values influence accuracy reduction. Higher dropout rates (0.4 and 0.5) tend to enhance unlearning performance by introducing more noise during training, thus facilitating better forgetting of specific data.

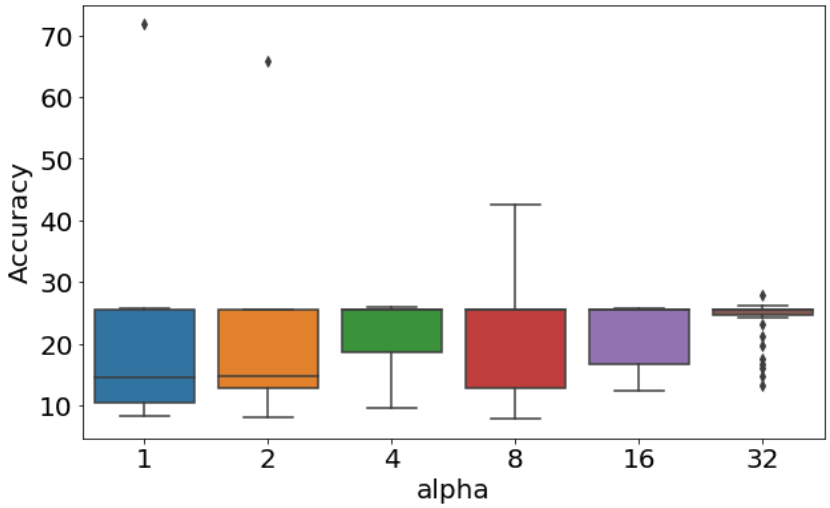

Figure 6.4: Impact of Different Alpha Values on Accuracy (Twitter)

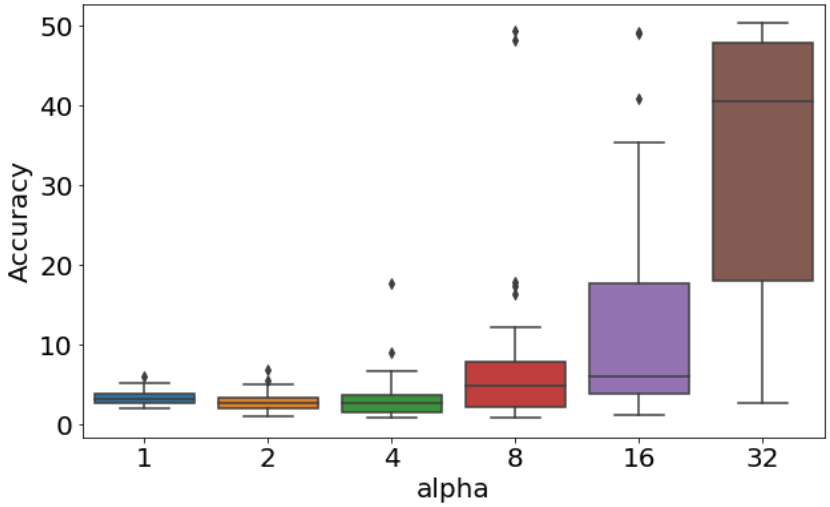

Figure 6.5: Impact of Different Alpha Values on Accuracy (IMDB)

6.6.2.4 Performance Influencing Factors

Analyzing the impact of alpha, dropout, and r values reveals important insights into our unlearning method's performance. Both datasets show that lower alpha values and higher dropout rates contribute significantly to improved unlearning by reducing the model's capacity to retain information.

Moreover, higher r values allow the model to capture more diverse information, facilitating better unlearning. These findings underscore the need to carefully tune

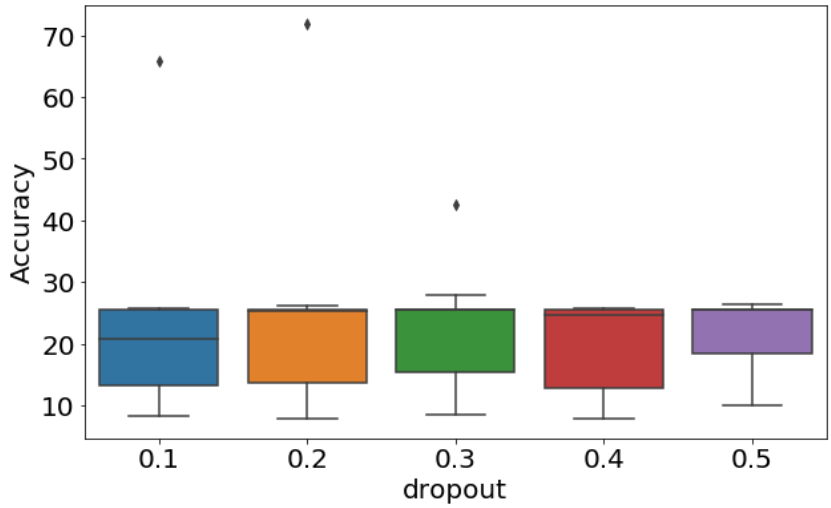

Figure 6.6: Impact of Different Dropout Values on Accuracy (Twitter)

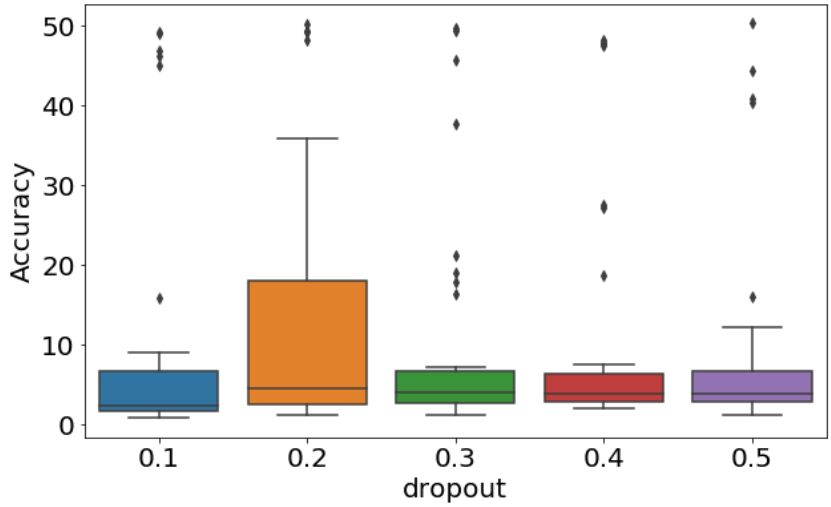

Figure 6.7: Impact of Different Dropout Values on Accuracy (IMDB)

hyperparameters in our LoRA-based unlearning approach to maximize effectiveness.

6.6.2.5 Comparison with Retrain from Scratch

Table 6.3 compares our method with the Retrain from Scratch technique for both datasets. Our approach achieves similar final accuracies, demonstrating effective unlearning while being computationally more efficient.

Table 6.3: Final Accuracy Comparison

Method	Initial Accuracy	Final Accuracy
Twitter & Our Method	83.98%	7.93%
Twitter & Retrain from Scratch	87.53%	7.84%
IMDB & Our Method	97.10%	0.95%
IMDB & Retrain from Scratch	95.60%	0.85%

6.6.2.6 Blockchain Integration and Performance

Our study also evaluates the performance implications of integrating a hybrid blockchain structure, combining public and private blockchains, into our federated learning framework with unlearning capabilities. We focused on scalability, transaction throughput, and latency.

- **Setup Time:** Initial setup of the hybrid blockchain network took approximately 48 seconds. This is slightly higher than using only a public blockchain but acceptable considering the long-term benefits in security and privacy.
- **Consensus Overhead:** The consensus process added around 6 seconds due to the additional coordination required between public and private blockchains. This increase is manageable within the federated learning context.
- **Transaction Processing:** Average transaction processing time, including model updates and data sharing, was 4 seconds, demonstrating the hybrid blockchain's efficiency.
- **Per-Epoch Duration:** Training duration per epoch remained consistent at 30-32 seconds, even with additional unlearning activities and blockchain coordination, highlighting the system's robustness.

Table 6.4 compares time costs across different federated learning cycles, showing that while the hybrid blockchain method has a slightly higher initial time cost, it normalizes over iterations, indicating scalability.

The hybrid blockchain architecture provides additional data privacy benefits by using private blockchains for sensitive information sharing while maintaining transparency on the public blockchain. This approach balances the need for transparency and accountability with privacy requirements.

Overall, our results demonstrate that integrating a hybrid blockchain into our federated learning framework introduces minimal overhead while ensuring robust and

Table 6.4: Time Cost Analysis for LLM Federated Learning with and without Blockchain Integration

Method	**t = 0**	**t = 9**	**t = 99**	**t = 999**
Normal Federated Learning for LLMs	30s	300s	3000s	30000s
Public Blockchain-Enhanced Method for LLMs	79s	367s	3277s	32277s
Hybrid Blockchain-Enhanced Method for LLMs	84s	378s	3384s	33384s

scalable performance. This makes our system a promising solution for secure, transparent, and privacy-preserving federated learning with unlearning capabilities for LLMs.

6.6.2.7 Conclusion

Our experiments on the IMDB and Twitter datasets confirm that our method achieves performance levels comparable to the Retrain from Scratch technique in terms of accuracy reduction. The success of our LoRA-based unlearning approach is due to carefully selected and tuned parameters and specific implementation techniques. Our method offers a computationally viable alternative to retraining from scratch, which can be resource-intensive and time-consuming.

Additionally, we evaluated the impact of incorporating blockchain technology into our federated learning framework. The results show that the added blockchain components introduce negligible overhead in terms of setup, consensus, transaction processing, and per-epoch time costs. The system's consistent performance, despite additional unlearning activities, highlights its resilience and scalability.

The differences in unlearning results between the IMDB and Twitter datasets can be attributed to both intrinsic dataset characteristics and the use of a private blockchain in Chapter 6. Unlike in Chapter 5, where a traditional federated learning setup was used, the private blockchain introduces additional security and verification layers that ensure stricter model update validation. This affects the learning and unlearning dynamics in the following ways:

- **Stronger Verification of Model Updates:** The private chain ensures that only authenticated, consensus-approved updates are applied, reducing unintended variations in model behaviour. This can lead to more consistent unlearning effects across training nodes.

- **Deterministic Update Propagation:** Due to the controlled network environment, the propagation of weight updates and unlearning requests follows a structured

sequence, unlike a fully decentralized setup where update delays and asynchrony can introduce variations.

- **Reduced External Noise:** Since only verified participants contribute to training, external fluctuations caused by adversarial updates or irregular participation are minimized. This is particularly relevant in datasets like Twitter, where the data itself is inherently more diverse and noisy.

By balancing performance, privacy, and computational efficiency, our multi-agent blockchain-integrated federated learning framework with unlearning capabilities presents a robust solution for secure and effective LLM training in diverse applications. The controlled nature of the private blockchain further enhances reliability, ensuring that unlearning mechanisms function as expected without interference from external network conditions or unverified model updates.

6.7 Summary

This chapter introduces an innovative hybrid blockchain-based multi-agent federated learning framework for training Large Language Models (LLMs) in cross-organizational collaborations, with data unlearning capabilities. Our framework leverages the strengths of both public and private blockchains to create a secure, transparent, and efficient collaborative environment while incorporating multi-agent interactions and efficient data unlearning mechanisms.

Through extensive experiments on IMDB and Twitter datasets, we demonstrate the superior performance of our framework in terms of data privacy protection, collaboration efficiency improvement, and targeted data forgetting. The carefully tuned LoRA hyperparameters enable our approach to efficiently remove target data while maintaining the model's performance on the remaining data. The multi-agent system enhances collaboration efficiency through interactions and knowledge sharing among agents. Furthermore, the hybrid blockchain architecture introduces minimal computational overhead and time cost, highlighting the scalability and robustness of our system.

Compared to existing methods, our framework exhibits significant advantages in terms of computational efficiency, versatility, and adaptability. It provides a secure, transparent, and efficient solution for federated learning of LLMs in cross-organizational settings. Our framework has the potential to drive innovative applications, particularly in scenarios where data privacy and selective data forgetting are of paramount importance.

CHAPTER 7

DISCUSSION, FUTURE WORK AND CONCLUSION

7.1 Discussion

The research presented in this thesis explores the innovation and insights of blockchain technology with privacy protection, machine unlearning, and federated learning to address the challenges of secure and trustworthy collaborative learning across various domains. The proposed approaches demonstrate the potential of blockchain in enhancing privacy, security, and trust, particularly in intelligent transportation systems, large language model training, and cross-organizational collaboration.

The novel scheme for dynamic location privacy-preserving using blockchain in intelligent transportation systems highlights the effectiveness of integrating blockchain with privacy-preserving techniques. This scheme ensures the protection of sensitive location data while enabling efficient and secure communication, offering enhanced security and verifiability compared to traditional methods.

The trustworthy approach for integrating blockchain with machine unlearning in federated learning environments enhances the efficiency, scalability, and verifiability of machine unlearning techniques. By leveraging the immutable and transparent nature of blockchain, this approach creates a secure and auditable record of data removal processes, ensuring verifiable and efficient data removal.

Federated TrustChain, a blockchain-enhanced framework for LLM training and unlearning in federated learning, addresses the challenges of data privacy and secure collaboration in LLM training. By creating a transparent and verifiable framework for

model updates and data removal, Federated TrustChain ensures secure and auditable collaborative model development while maintaining data privacy.

The blockchain-based solution for enabling secure and efficient cross-organizational collaboration in federated learning proposes a decentralized and trustworthy framework. This solution ensures data confidentiality, integrity, and privacy across multiple organizations, facilitating secure data sharing and model updates among participating entities.

Comprehensive experimental evaluations and case studies validate the practical applicability and performance of the proposed blockchain-based privacy protection and machine unlearning techniques. The results provide valuable insights into the strengths and limitations of the proposed approaches, guiding future research and development efforts in the field of secure and trustworthy collaborative learning.

Overall, this research contributes to the advancement of privacy-preserving technologies, machine unlearning, and secure collaborative learning frameworks. The proposed approaches demonstrate the potential of blockchain in creating secure, verifiable, and trustworthy collaborative learning environments.

7.2 Limitations

While this research makes significant contributions, several limitations should be acknowledged. The proposed blockchain-based frameworks may face challenges in scalability when dealing with large-scale datasets and a high number of participating entities. As the size of the collaborative learning network grows, the computational and communication overhead associated with blockchain operations could impact system efficiency, necessitating further research to optimize scalability. Additionally, the regulatory compliance aspect is another limitation. The approaches focus on technical aspects, but ensuring compliance with diverse regulatory frameworks across different jurisdictions is a challenge. Future research should explore the integration of regulatory compliance mechanisms into the proposed frameworks to facilitate their adoption in real-world applications.

Interoperability with existing systems and technologies also poses a challenge. The seamless integration and compatibility with legacy systems and the ability to exchange data and models across different blockchain platforms are important considerations for practical deployment. Further research is needed to address these interoperability aspects. User adoption and usability are critical for the success of the proposed frame-

works. Factors such as ease of use, trust in the technology, and perceived benefits play a crucial role in user engagement. The usability aspects, including user interface design and user experience, need to be carefully considered to facilitate widespread adoption. Lastly, the integration of blockchain technology introduces additional computational and communication overhead. Consensus mechanisms, cryptographic operations, and data replication associated with blockchain may impact the overall performance, especially in resource-constrained environments. Further optimization techniques and performance enhancements are required to ensure efficiency and scalability.

7.3 Future Work

The research presented in this thesis opens several avenues for future work in the field of secure and trustworthy collaborative learning. Future research should focus on developing advanced techniques to improve the scalability of blockchain-based frameworks. This could involve investigating more efficient consensus mechanisms, sharding techniques, and off-chain computation approaches to handle large-scale datasets and a high number of participating entities. Additionally, while the proposed approaches integrate privacy protection mechanisms such as differential privacy, there is scope for further exploration of advanced privacy-preserving techniques. Future research could investigate the integration of homomorphic encryption, secure multi-agent LLM, and zero-knowledge proofs to enhance the privacy guarantees of the proposed frameworks.

To facilitate the adoption of the proposed approaches in real-world applications, future work should also focus on developing comprehensive regulatory compliance frameworks. Collaborating with legal experts and policymakers to understand the specific requirements and guidelines of different jurisdictions and incorporating them into the proposed frameworks will be essential. Furthermore, future research should explore the development of interoperability standards and protocols for blockchain-based collaborative learning frameworks. Defining common data models, APIs, and communication protocols will promote the widespread adoption of the proposed approaches.

To promote user adoption and engagement, future work should emphasize user-centric design and usability. Conducting user studies to understand user requirements, preferences, and concerns will help inform the design of intuitive interfaces that enhance user experience and facilitate active participation. Performance optimization is another critical area for future research. Investigating advanced techniques to minimize the computational and communication overhead introduced by blockchain integration will ensure

the proposed approaches operate effectively in resource-constrained environments.

By pursuing these future research directions, we can further advance the field of secure and trustworthy collaborative learning and develop innovative solutions that address the evolving challenges of data confidentiality, protection, and reliability in decentralized collaborative learning environments.

7.4 Conclusion

This thesis presents a comprehensive exploration of integrating blockchain technology with privacy protection, machine unlearning, and federated learning to address the challenges of secure and trustworthy collaborative learning. The proposed approaches demonstrate the potential of blockchain in enhancing privacy, security, and trust, particularly in intelligent transportation systems, large language model training, and cross-organizational collaboration.

The novel contributions of this thesis include a dynamic location privacy-preserving scheme using blockchain in intelligent transportation systems, a trustworthy approach for integrating blockchain with machine unlearning in federated learning environments, Federated TrustChain-a blockchain-enhanced framework for LLM training and unlearning, and a blockchain-based solution for enabling secure and efficient cross-organizational collaboration in federated learning. These contributions showcase the effectiveness of blockchain in creating secure, verifiable, and privacy-preserving collaborative learning frameworks.

The extensive experimental evaluations and case studies conducted in this thesis validate the practical applicability and performance of the proposed approaches in various domains and real-world scenarios. The results provide valuable insights into the strengths and limitations of the proposed frameworks and techniques, guiding future research and development efforts in the field of secure and trustworthy collaborative learning.

While the research presented in this thesis makes significant strides towards secure and trustworthy collaborative learning, there are several limitations and challenges that need to be addressed. Future work should focus on enhancing the scalability, interoperability, user adoption, and performance aspects of the proposed approaches to enable their widespread adoption and practical implementation in various collaborative learning domains.

The insights gained from this research contribute to the advancement of privacy-

preserving technologies, machine unlearning, and secure collaborative learning frameworks. The proposed approaches and solutions pave the way for more responsible and ethical use of data and AI technologies, ensuring the protection of user privacy, the integrity of collaborative learning processes, and the establishment of trust among participating entities.

As the world becomes increasingly data-driven and interconnected, the need for secure and trustworthy collaborative learning frameworks becomes more critical than ever. The research presented in this thesis lays the foundation for the development of innovative solutions that harness the power of blockchain technology to address the challenges of privacy, security, and trust in collaborative learning environments. By continuing to explore the synergies between blockchain, privacy protection, machine unlearning, and federated learning, we can unlock the full potential of collaborative learning and drive the responsible and ethical advancement of AI technologies for the benefit of society.

APPENDIX

A.1 Notations of This Thesis

Table A.1: Notations

Notations	Description
$\mathscr{M}_D^{(k)}$	The Dirichlet mechanism applied to a dataset D with privacy parameter k
$\mathbb{P}[\mathscr{M}_D^{(k)}(p) = x]$	The probability density function (PDF) of the Dirichlet mechanism output x
k	The concentration parameter of the Dirichlet mechanism
$\mathrm{B}(kp)$	The multivariate Beta function normalizing the Dirichlet distribution.
$\Gamma(\cdot)$	The Gamma function, which generalizes the factorial function to continuous values
Δk	The privacy sensitivity parameter
V_{id}	The vehicle ID
RSU_{id}	The roadside Unit ID
key_v	The token for vehicle (V_i)
key_{RSU}	The token for RSU
$Pool$	The pool of registered vehicle and RSU IDs
$Time$	The timestamp
Pos	The position
λ	The traffic flow
ε_j	The traffic density at location (j)

Table A.1: Notations

Notations	Description
λ_j	The traffic volume at location (j)
δ_j	The average speed at location (j)
k	The traffic flow control constant
D	The vehicle density
N	The number of vehicles
P	The dirichlet parameter
Dir()	The dirichlet distribution
D	The Dataset
$\|D_i\|$	The number of samples in D_i
$\|D\|$	The total number of samples across all clients
$\mathscr{L}_i$	The loss function computed on the i^{th} client's dataset
$\theta^{(t)}$	The global model parameters before the update in round t
$\theta^{(t+1)}$	The global model parameters after the update in round t
η	The learning rate
$\mathscr{M}$	Model of machine unlearning
$\mathscr{M}_{\text{new}}$	The updated model after retraining
C	The client of blockchain network
jwt	A jwt token which used to verified user's identity
P_k	The public key of the user
S_k	The private key of the user
ϵ	The parameter of differential privacy
δ	The parameter of differential privacy
C_{id}	The ID of Client
$Agent$	The agent of blockchain network
A_{id}	The ID of Agent
U_{pool}	The pool of User
SC	The smart contract of the blockchain network
$Model_g$	The global model
$epoch$	The epoch setting of federated learning
$batchsize$	The batch-size setting of federated learning
LLM_g	The global large language model
LLM_n	The new version of large language model

Table A.1: Notations

Notations	Description
D_{forget}	The dataset to forget
η	The learning rate of unlearning
E_u	The epoch of unlearning
λ	The parameter of LoRA fine-tuning
LLM_{local}	The local large language model
$LLM_{updated}$	The updated large language model
T_{id}	The ID of block transaction
E_{name}	The unique identifier of entity
Org	The organisation of entity
$Role$	The role of entity
N_p	The number of epochs in private chain
N_g	The number of epochs in public chain

APPENDIX B

APPENDIX

B.1 Security Summary Across Different Scenarios

Throughout this work, we have explored different security challenges across various scenarios, each requiring distinct security mechanisms based on its operational context. This section provides a summary of why these requirements differ and the motivations behind them.

Vehicular Networks and Intelligent Transportation (Section 3.3.2) In vehicular networks, the primary concern is secure communication between vehicles (V2I) and roadside units (RSUs). The system must prevent unauthorized access to sensitive location data while ensuring that only authenticated entities participate in the network. The main security mechanisms involve authentication (certificate authorities for RSUs), encryption protocols for message confidentiality, and secure data aggregation to prevent tracking.

Federated Learning and Unlearning (Section 4.2.1) In federated learning, security requirements shift toward ensuring the integrity and privacy of model updates. Since data remains decentralized, the key challenge is preventing poisoning attacks while maintaining trust in the unlearning process. The introduction of blockchain-based logging ensures that unlearning requests are auditable, while cryptographic verification methods

confirm that unwanted data has been successfully forgotten without modifying blockchain immutability.

Federated TrustChain for LLM Training (Section 5.2.1) When training large language models (LLMs) within federated environments, security risks arise from adversarial manipulation of model updates. Ensuring model integrity is critical, as malicious clients could introduce backdoors or poisoned gradients. To address this, secure aggregation mechanisms validate model updates, while unlearning operations use LoRA-based adaptive forgetting to remove specific knowledge effectively. Unlike traditional FL, where updates are directly aggregated, here we ensure that unlearning requests do not disrupt the overall model performance.

Cross-Organizational Federated Learning (Section 6.3.1) When multiple independent organizations participate in federated learning, security concerns extend beyond model integrity to trust management and fair contribution tracking. Some participants may attempt free-riding (benefiting from the model without contributing meaningful updates). To counter this, we incorporate multi-agent trust mechanisms that detect abnormal behaviors and enforce reputation-based incentives to align participants‚Äô motivations.

Why Do These Security Models Differ? Each of these scenarios demands different security approaches because of variations in:

- **Threat Models:** Vehicular networks deal with real-time tracking threats, while federated learning is more concerned with model poisoning and unlearning verification.
- **Data Privacy Concerns:** FL focuses on user-level privacy (e.g., preventing inference attacks on model updates), while vehicular networks emphasize location privacy.
- **Verification Mechanisms:** Blockchain plays a key role in FL by ensuring auditability and model update integrity, whereas in vehicular networks, real-time authentication is prioritized.

By designing security models tailored to each scenario, we ensure that our system remains robust and practical across different applications.

BIBLIOGRAPHY

[1] H. N. ABISHU, A. M. SEID, Y. H. YACOB, T. AYALL, G. SUN, AND G. LIU, *Consensus mechanism for blockchain-enabled vehicle-to-vehicle energy trading in the internet of electric vehicles*, IEEE Transactions on Vehicular Technology, 71 (2021), pp. 946–960.

[2] S. AGGARWAL AND N. KUMAR, *A consortium blockchain-based energy trading for demand response management in vehicle-to-grid*, IEEE Transactions on Vehicular Technology, 70 (2021), pp. 9480–9494.

[3] N. A. AKBAR, A. MUNEER, N. ELHAKIM, AND S. M. FATI, *Distributed hybrid double-spending attack prevention mechanism for proof-of-work and proof-of-stake blockchain consensuses*, Future Internet, 13 (2021), p. 285.

[4] T. ALLADI, V. CHAMOLA, N. SAHU, V. VENKATESH, A. GOYAL, AND M. GUIZANI, *A comprehensive survey on the applications of blockchain for securing vehicular networks*, IEEE Communications Surveys & Tutorials, 24 (2022), pp. 1212–1239.

[5] T. BAUMHAUER, P. SCHÖTTLE, AND M. ZEPPELZAUER, *Machine unlearning: Linear filtration for logit-based classifiers*, Machine Learning, 111 (2022), pp. 3203–3226.

[6] M. A. BOUCHIHA, Q. TELNOFF, S. BAKKALI, R. CHAMPAGNAT, M. RABAH, M. COUSTATY, AND Y. GHAMRI-DOUDANE, *Llmchain: Blockchain-based reputation system for sharing and evaluating large language models*, arXiv preprint arXiv:2404.13236, (2024).

[7] L. BOURTOULE, V. CHANDRASEKARAN, C. A. CHOQUETTE-CHOO, H. JIA, A. TRAVERS, B. ZHANG, D. LIE, AND N. PAPERNOT, *Machine unlearning*, in 2021 IEEE Symposium on Security and Privacy (SP), IEEE, 2021, pp. 141–159.

[8] P. BUKATY, *The california consumer privacy act (ccpa): An implementation guide*, IT Governance Ltd, 2019.

[9] A. BUZACHIS, B. FILOCAMO, M. FAZIO, J. A. RUIZ, M. Á. SOTELO, AND M. VILLARI, *Distributed priority based management of road intersections using blockchain*, in 2019 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2019, pp. 1159–1164.

[10] A. CABALLERO HINOJOSA, *Exploring the power of large language models: News intention detection using adaptive learning prompting*, (2023).

[11] W. CHANG, T. ZHU, H. XU, W. LIU, AND W. ZHOU, *Class machine unlearning for complex data via concepts inference and data poisoning*, arXiv preprint arXiv:2405.15662, (2024).

[12] Y. CHANG, X. WANG, J. WANG, Y. WU, L. YANG, K. ZHU, H. CHEN, X. YI, C. WANG, Y. WANG, ET AL., *A survey on evaluation of large language models*, ACM Transactions on Intelligent Systems and Technology, (2023).

[13] C. CHEN, X. FENG, J. ZHOU, J. YIN, AND X. ZHENG, *Federated large language model: A position paper*, arXiv preprint arXiv:2307.08925, (2023).

[14] C. DWORK AND G. N. ROTHBLUM, *Concentrated differential privacy*, CoRR, abs/1603.01887 (2016).

[15] T. EISENHOFER, D. RIEPEL, V. CHANDRASEKARAN, E. GHOSH, O. OHRIMENKO, AND N. PAPERNOT, *Verifiable and provably secure machine unlearning*, arXiv preprint arXiv:2210.09126, (2022).

[16] I. EYAL AND E. G. SIRER, *Majority is not enough: Bitcoin mining is vulnerable*, Communications of the ACM, 61 (2018), pp. 95–102.

[17] T. FAN, Y. KANG, G. MA, W. CHEN, W. WEI, L. FAN, AND Q. YANG, *Fate-llm: A industrial grade federated learning framework for large language models*, arXiv preprint arXiv:2310.10049, (2023).

[18] U. FAROOQ AND N. JAVAID, *Blockchain based decentralized vehicular communication system and smart payment method*, Management, 6, p. 8.

[19] A. GHORBANI AND J. ZOU, *Data shapley: Equitable valuation of data for machine learning*, in International conference on machine learning, PMLR, 2019, pp. 2242–2251.

[20] A. GINART, M. GUAN, G. VALIANT, AND J. Y. ZOU, *Making ai forget you: Data deletion in machine learning*, Advances in neural information processing systems, 32 (2019).

[21] P. GOHARI, B. WU, M. HALE, AND U. TOPCU, *The dirichlet mechanism for differential privacy on the unit simplex*, in 2020 American Control Conference (ACC), IEEE, 2020, pp. 1253–1258.

[22] P. GOHARI, B. WU, C. HAWKINS, M. HALE, AND U. TOPCU, *Differential privacy on the unit simplex via the dirichlet mechanism*, IEEE Transactions on Information Forensics and Security, 16 (2021), pp. 2326–2340.

[23] S. GUPTA, Y. HUANG, Z. ZHONG, T. GAO, K. LI, AND D. CHEN, *Recovering private text in federated learning of language models*, Advances in neural information processing systems, 35 (2022), pp. 8130–8143.

[24] M. U. HADI, R. QURESHI, A. SHAH, M. IRFAN, A. ZAFAR, M. B. SHAIKH, N. AKHTAR, J. WU, S. MIRJALILI, ET AL., *Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects*, Authorea Preprints, (2023).

[25] ——, *A survey on large language models: Applications, challenges, limitations, and practical usage*, Authorea Preprints, (2023).

[26] E. J. HU, Y. SHEN, P. WALLIS, Z. ALLEN-ZHU, Y. LI, S. WANG, L. WANG, AND W. CHEN, *Lora: Low-rank adaptation of large language models*, arXiv preprint arXiv:2106.09685, (2021).

[27] J. Y. HUANG, W. ZHOU, F. WANG, F. MORSTATTER, S. ZHANG, H. POON, AND M. CHEN, *Offset unlearning for large language models*, arXiv preprint arXiv:2404.11045, (2024).

[28] F. JIANG, L. DONG, S. TU, Y. PENG, K. WANG, K. YANG, C. PAN, AND D. NIYATO, *Personalized wireless federated learning for large language models*, arXiv preprint arXiv:2404.13238, (2024).

[29] J. JIANG, X. LIU, AND C. FAN, *Low-parameter federated learning with large language models*, arXiv preprint arXiv:2307.13896, (2023).

[30] M. I. JORDAN AND T. M. MITCHELL, *Machine learning: Trends, perspectives, and prospects*, Science, 349 (2015), pp. 255–260.

[31] E. KASNECI, K. SESSLER, S. KÜCHEMANN, M. BANNERT, D. DEMENTIEVA, F. FISCHER, U. GASSER, G. GROH, S. GÜNNEMANN, E. HÜLLERMEIER, ET AL., *Chatgpt for good? on opportunities and challenges of large language models for education*, Learning and individual differences, 103 (2023), p. 102274.

[32] L. U. KHAN, W. SAAD, Z. HAN, E. HOSSAIN, AND C. S. HONG, *Federated learning for internet of things: Recent advances, taxonomy, and open challenges*, IEEE Communications Surveys & Tutorials, 23 (2021), pp. 1759–1799.

[33] A. KIAYIAS, A. RUSSELL, B. DAVID, AND R. OLIYNYKOV, *Ouroboros: A provably secure proof-of-stake blockchain protocol*, in Annual international cryptology conference, Springer, 2017, pp. 357–388.

[34] W. KUANG, B. QIAN, Z. LI, D. CHEN, D. GAO, X. PAN, Y. XIE, Y. LI, B. DING, AND J. ZHOU, *Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning*, arXiv preprint arXiv:2309.00363, (2023).

[35] Q. LI, Z. WEN, Z. WU, S. HU, N. WANG, Y. LI, X. LIU, AND B. HE, *A survey on federated learning systems: Vision, hype and reality for data privacy and protection*, IEEE Transactions on Knowledge and Data Engineering, 35 (2021), pp. 3347–3366.

[36] Y. LI, C. CHEN, N. LIU, H. HUANG, Z. ZHENG, AND Q. YAN, *A blockchain-based decentralized federated learning framework with committee consensus*, IEEE Network, 35 (2020), pp. 234–241.

[37] L. LIANG, H. YE, AND G. Y. LI, *Toward intelligent vehicular networks: A machine learning framework*, IEEE Internet of Things Journal, 6 (2019), pp. 124–135.

[38] R. LIANG, B. LI, AND X. SONG, *Blockchain-based privacy preserving trust management model in vanet*, in International Conference on Advanced Data Mining and Applications, Springer, 2020, pp. 465–479.

[39] G. LIU, X. MA, Y. YANG, C. WANG, AND J. LIU, *Federaser: Enabling efficient client-level data removal from federated learning models*, in 2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS), IEEE, 2021, pp. 1–10.

[40] Y. LIU, Z. XIONG, Q. HU, D. NIYATO, J. ZHANG, C. MIAO, C. LEUNG, AND Z. TIAN, *Vrepchain: A decentralized and privacy-preserving reputation system for so-*

cial internet of vehicles based on blockchain, IEEE Transactions on Vehicular Technology, (2022).

[41] Y. LIU, L. XU, X. YUAN, C. WANG, AND B. LI, *The right to be forgotten in federated learning: An efficient realization with rapid retraining*, in IEEE INFOCOM 2022-IEEE Conference on Computer Communications, IEEE, 2022, pp. 1749–1758.

[42] Z. LIU, Y. JIANG, J. SHEN, M. PENG, K.-Y. LAM, AND X. YUAN, *A survey on federated unlearning: Challenges, methods, and future directions*, arXiv preprint arXiv:2310.20448, (2023).

[43] Y. LU, X. HUANG, Y. DAI, S. MAHARJAN, AND Y. ZHANG, *Blockchain and federated learning for privacy-preserved data sharing in industrial iot*, IEEE Transactions on Industrial Informatics, 16 (2019), pp. 4177–4186.

[44] H. LUO, J. LUO, AND A. V. VASILAKOS, *Bc4llm: Trusted artificial intelligence when blockchain meets large language models*, arXiv preprint arXiv:2310.06278, (2023).

[45] L. LYU, H. YU, X. MA, C. CHEN, L. SUN, J. ZHAO, Q. YANG, AND S. Y. PHILIP, *Privacy and robustness in federated learning: Attacks and defenses*, IEEE transactions on neural networks and learning systems, (2022).

[46] Z. MA, T. ZHANG, X. LIU, X. LI, AND K. REN, *Real-time privacy-preserving data release over vehicle trajectory*, IEEE transactions on vehicular technology, 68 (2019), pp. 8091–8102.

[47] J. G. M. MBOMA, K. LUSALA, M. MATALATALA, O. T. TSHIPATA, P. S. NZAKUNA, AND D. T. KAZUMBA, *Integrating llm with blockchain and ipfs to enhance academic diploma integrity*, in 2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA), IEEE, 2024, pp. 1–6.

[48] B. MIN, H. ROSS, E. SULEM, A. P. B. VEYSEH, T. H. NGUYEN, O. SAINZ, E. AGIRRE, I. HEINTZ, AND D. ROTH, *Recent advances in natural language processing via large pre-trained language models: A survey*, ACM Computing Surveys, 56 (2023), pp. 1–40.

[49] N. O. NAWARI AND S. RAVINDRAN, *Blockchain and the built environment: Potentials and limitations*, Journal of Building Engineering, 25 (2019), p. 100832.

[50] C. T. NGUYEN, Y. LIU, H. DU, D. T. HOANG, D. NIYATO, D. N. NGUYEN, AND S. MAO, *Generative ai-enabled blockchain networks: Fundamentals, applications, and case study*, arXiv preprint arXiv:2401.15625, (2024).

[51] D. C. NGUYEN, M. DING, Q.-V. PHAM, P. N. PATHIRANA, L. B. LE, A. SENEVIRATNE, J. LI, D. NIYATO, AND H. V. POOR, *Federated learning meets blockchain in edge computing: Opportunities and challenges*, IEEE Internet of Things Journal, 8 (2021), pp. 12806–12825.

[52] T. T. NGUYEN, T. T. HUYNH, P. L. NGUYEN, A. W.-C. LIEW, H. YIN, AND Q. V. H. NGUYEN, *A survey of machine unlearning*, arXiv preprint arXiv:2209.02299, (2022).

[53] T. PENG, J. LIU, J. CHEN, AND G. WANG, *A privacy-preserving crowdsensing system with muti-blockchain*, in 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2020, pp. 1944–1949.

[54] S. R. POKHREL AND J. CHOI, *Federated learning with blockchain for autonomous vehicles: Analysis and design challenges*, IEEE Transactions on Communications, 68 (2020), pp. 4734–4746.

[55] E. POLITOU, F. CASINO, E. ALEPIS, AND C. PATSAKIS, *Blockchain mutability: Challenges and proposed solutions*, IEEE Transactions on Emerging Topics in Computing, 9 (2019), pp. 1972–1986.

[56] P. PRINCE AND S. LOVESUM, *Privacy enforced access control model for secured data handling in cloud-based pervasive health care system*, SN Computer Science, 1 (2020), pp. 1–8.

[57] Y. QIAN, Y. MA, J. CHEN, D. WU, D. TIAN, AND K. HWANG, *Optimal location privacy preserving and service quality guaranteed task allocation in vehicle-based crowdsensing networks*, IEEE Transactions on Intelligent Transportation Systems, 22 (2021), pp. 4367–4375.

[58] D. QIU, Y. WANG, T. ZHANG, M. SUN, AND G. STRBAC, *Hybrid multi-agent reinforcement learning for electric vehicle resilience control towards a low-carbon transition*, IEEE Transactions on Industrial Informatics, (2022).

[59] G. D. P. REGULATION, *General data protection regulation (gdpr)*, Intersoft Consulting, Accessed in October, 24 (2018).

[60] J. H. RO, S. BHOJANAPALLI, Z. XU, Y. ZHANG, AND A. T. SURESH, *Efficient language model architectures for differentially private federated learning*, arXiv preprint arXiv:2403.08100, (2024).

[61] M. SHAYAN, C. FUNG, C. J. YOON, AND I. BESCHASTNIKH, *Biscotti: A blockchain system for private and secure federated learning*, IEEE Transactions on Parallel and Distributed Systems, 32 (2020), pp. 1513–1525.

[62] N. SI, H. ZHANG, H. CHANG, W. ZHANG, D. QU, AND W. ZHANG, *Knowledge unlearning for llms: Tasks, methods, and challenges*, arXiv preprint arXiv:2311.15766, (2023).

[63] P. K. SINGH, R. SINGH, S. K. NANDI, K. Z. GHAFOOR, D. B. RAWAT, AND S. NANDI, *Blockchain-based adaptive trust management in internet of vehicles using smart contract*, IEEE Transactions on Intelligent Transportation Systems, 22 (2020), pp. 3616–3630.

[64] C. R. STORCK AND F. DUARTE-FIGUEIREDO, *A survey of 5g technology evolution, standards, and infrastructure associated with vehicle-to-everything communications by internet of vehicles*, IEEE Access, 8 (2020), pp. 117593–117614.

[65] A. SUNYAEV AND A. SUNYAEV, *Distributed ledger technology*, Internet computing: Principles of distributed systems and emerging internet-based technologies, (2020), pp. 265–299.

[66] P. VILLALOBOS, J. SEVILLA, L. HEIM, T. BESIROGLU, M. HOBBHAHN, AND A. HO, *Will we run out of data? an analysis of the limits of scaling datasets in machine learning*, arXiv preprint arXiv:2211.04325, (2022).

[67] Z. WAN, Y. ZHOU, AND K. REN, *zk-authfeed: Protecting data feed to smart contracts with authenticated zero knowledge proof*, IEEE Transactions on Dependable and Secure Computing, 20 (2022), pp. 1335–1347.

[68] F. WANG, B. LI, AND B. LI, *Federated unlearning and its privacy threats*, IEEE Network, (2023).

[69] J. WANG, S. GUO, X. XIE, AND H. QI, *Federated unlearning via class-discriminative pruning*, in Proceedings of the ACM Web Conference 2022, 2022, pp. 622–632.

[70] L. WANG, T. CHEN, W. YUAN, X. ZENG, K.-F. WONG, AND H. YIN, *Kga: A general machine unlearning framework based on knowledge gap alignment*, arXiv preprint arXiv:2305.06535, (2023).

[71] L. WANG, C. MA, X. FENG, Z. ZHANG, H. YANG, J. ZHANG, Z. CHEN, J. TANG, X. CHEN, Y. LIN, ET AL., *A survey on large language model based autonomous agents*, Frontiers of Computer Science, 18 (2024), pp. 1–26.

[72] M. WANG, T. ZHU, T. ZHANG, J. ZHANG, S. YU, AND W. ZHOU, *Security and privacy in 6g networks: New areas and new challenges*, Digital Communications and Networks, 6 (2020), pp. 281–291.

[73] M. WANG, T. ZHU, X. ZUO, M. YANG, S. YU, AND W. ZHOU, *Differentially private crowdsourcing with the public and private blockchain*, IEEE Internet of Things Journal, (2023).

[74] M. WANG, T. ZHU, X. ZUO, D. YE, S. YU, AND W. ZHOU, *Blockchain-based gradient inversion and poisoning defense for federated learning*, IEEE Internet of Things Journal, (2023).

[75] ——, *Public and private blockchain infusion: A novel approach to federated learning*, IEEE Internet of Things Journal, (2024).

[76] C. J. WATKINS AND P. DAYAN, *Q-learning*, Machine learning, 8 (1992), pp. 279–292.

[77] M. WOOLDRIDGE, *An introduction to multiagent systems*, John Wiley & Sons, 2009.

[78] L. WU, S. GUO, J. WANG, Z. HONG, J. ZHANG, AND Y. DING, *Federated unlearning: Guarantee the right of clients to forget*, IEEE Network, 36 (2022), pp. 129–135.

[79] H. XU, T. ZHU, L. ZHANG, W. ZHOU, AND P. S. YU, *Machine unlearning: A survey*, ACM Comput. Surv., 56 (2023).

[80] H. XU, T. ZHU, L. ZHANG, W. ZHOU, AND P. S. YU, *Machine unlearning: A survey*, ACM Computing Surveys, 56 (2023), pp. 1–36.

[81] X. XU, G. SUN, L. LUO, H. CAO, H. YU, AND A. V. VASILAKOS, *Latency performance modeling and analysis for hyperledger fabric blockchain network*, Information Processing & Management, 58 (2021), p. 102436.

[82] J. YANG, H. JIN, R. TANG, X. HAN, Q. FENG, H. JIANG, S. ZHONG, B. YIN, AND X. HU, *Harnessing the power of llms in practice: A survey on chatgpt and beyond*, ACM Transactions on Knowledge Discovery from Data, (2023).

[83] ——, *Harnessing the power of llms in practice: A survey on chatgpt and beyond*, ACM Transactions on Knowledge Discovery from Data, 18 (2024), pp. 1–32.

[84] Q. YANG, Y. LIU, T. CHEN, AND Y. TONG, *Federated machine learning: Concept and applications*, ACM Transactions on Intelligent Systems and Technology (TIST), 10 (2019), pp. 1–19.

[85] Y. YAO, X. XU, AND Y. LIU, *Large language model unlearning*, arXiv preprint arXiv:2310.10683, (2023).

[86] D. YE, T. ZHU, Z. CHENG, W. ZHOU, AND S. Y. PHILIP, *Differential advising in multiagent reinforcement learning*, IEEE transactions on cybernetics, 52 (2020), pp. 5508–5521.

[87] D. YE, T. ZHU, C. ZHU, W. ZHOU, AND S. Y. PHILIP, *Model-based self-advising for multi-agent learning*, IEEE transactions on neural networks and learning systems, (2022).

[88] F. YIN, Z. LIN, Q. KONG, Y. XU, D. LI, S. THEODORIDIS, AND S. R. CUI, *Fedloc: Federated learning framework for data-driven cooperative localization and location data processing*, IEEE Open Journal of Signal Processing, 1 (2020), pp. 187–215.

[89] C. YU, S. JEOUNG, A. KASI, P. YU, AND H. JI, *Unlearning bias in language models by partitioning gradients*, in Findings of the Association for Computational Linguistics: ACL 2023, 2023, pp. 6032–6048.

[90] L. ZHANG, T. ZHU, F. K. HUSSAIN, D. YE, AND W. ZHOU, *A game-theoretic method for defending against advanced persistent threats in cyber systems*, IEEE Transactions on Information Forensics and Security, 18 (2022), pp. 1349–1364.

[91] L. ZHANG, T. ZHU, H. ZHANG, P. XIONG, AND W. ZHOU, *Fedrecovery: Differentially private machine unlearning for federated learning frameworks*, IEEE Transactions on Information Forensics and Security, (2023).

[92] W. ZHAO, Y. DU, N. D. LANE, S. CHEN, AND Y. WANG, *Enhancing data quality in federated fine-tuning of large language models*, in ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models.

[93] Z. ZHENG, S. XIE, H.-N. DAI, X. CHEN, AND H. WANG, *Blockchain challenges and opportunities: A survey*, International journal of web and grid services, 14 (2018), pp. 352–375.

[94] S. ZHOU, C. LIU, D. YE, T. ZHU, W. ZHOU, AND P. S. YU, *Adversarial attacks and defenses in deep learning: From a perspective of cybersecurity*, ACM Computing Surveys, 55 (2022), pp. 1–39.

[95] S. ZOU, J. XI, H. WANG, AND G. XU, *Crowdblps: A blockchain-based location-privacy-preserving mobile crowdsensing system*, IEEE Transactions on Industrial Informatics, 16 (2019), pp. 4206–4218.

[96] X. ZUO, M. WANG, T. ZHU, L. ZHANG, S. YU, AND W. ZHOU, *Federated learning with blockchain-enhanced machine unlearning: A trustworthy approach*, arXiv preprint arXiv:2405.20776, (2024).