Scientific Reports

https://doi.org/10.1038/s41598-026-44843-4

Article in Press

Optimization of cross-institutional medical federated learning framework driven by confidential computing

Received: 9 January 2026

Accepted: 16 March 2026

Published online: 20 March 2026

Cite this article as: Xu F., Wei X., Zhao Z. *et al.* Optimization of cross-institutional medical federated learning framework driven by confidential computing. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-44843-4

Fengbo Xu, Xinle Wei, Zhiyuan Zhao & Peng Sun

Optimization of Cross-Institutional Medical Federated Learning Framework Driven by Confidential Computing

Fengbo Xu[1], Xinle Wei[1], Zhiyuan Zhao[1], Peng Sun[2*]
[1]*The First Affiliated Hospital, and College of Clinical Medicine of Henan University of Science and Technology, Luoyang, 471003, China*
[2]*Hebi Institute of Engineering and Technology, Henan Polytechnic University, Hebi, China*
*E-mail: pengssun@outlook.com

Abstract: Cross-institutional collaboration in privacy-sensitive domains such as healthcare and finance requires machine learning frameworks that balance model utility, privacy protection, and communication efficiency. Federated learning (FL) enables decentralized model training without direct data sharing, yet existing approaches inadequately address vulnerabilities in Trusted Execution Environments (TEEs), which are increasingly adopted to safeguard local computations. TEE side-channel attacks (e.g., cache-timing leaks, speculative execution exploits) can expose sensitive gradient information even when cryptographic defenses are deployed. Furthermore, traditional FL methods treat privacy and communication as independent objectives, leading to suboptimal tradeoffs when both constraints are active. This paper proposes Confidential Computing-Aware Projected Gradient Descent (CC-PGD), a constrained multi-objective optimization framework that jointly minimizes model loss, privacy leakage risk (incorporating TEE vulnerability modeling), and communication overhead. We formulate privacy risk as a combination of gradient entropy and a binary indicator function for TEE exploit susceptibility, while communication cost accounts for model size and network latency. We prove that CC-PGD achieves $O(1/\sqrt{T})$ convergence under non-convex objectives with Lipschitz-continuous gradients. Experiments on MNIST and CIFAR-10 under IID and non-IID data partitioning demonstrate that CC-PGD reduces privacy leakage by 23-31% and communication cost by 18-27% compared to baselines (FedAvg, DP-FL, FedProx), while maintaining competitive accuracy (within 2% of centralized training). Our work provides the first optimization framework explicitly accounting for TEE side-channel risks in federated learning, with theoretical guarantees and empirical validation.

Keywords: Artificial Intelligence, Cybersecurity, Federated Learning, Confidential Computing, Trusted Execution Environments, Optimization

1. Introduction

Generative artificial intelligence (GenAI) is rapidly transforming the cybersecurity landscape. Large language models and transformer-based tools like *ChatGPT*, *Gemini*, and *DALL-E* have shown how GenAI can automate and enhance complex tasks across various domains [1,2]. This transformation, both opportunities and risks to cybersecurity. On one side, GenAI helps improve vulnerability detection [3], automate threat analysis [4], and support faster incident response [5]. On the other side, GenAI also enables hackers to develop sophisticated phishing attacks and automated hacking strategies [6,7].

To ensure the safe and responsible deployment of GenAI in cybersecurity, it is essential to build secure and trustworthy AI systems. This includes not only improving the accuracy of generative models but also ensuring they are trained and deployed in ways that protect sensitive data and are resistant to manipulation. Federated learning (FL) and confidential computing (CC) have emerged as two key technologies to help protect sensitive data and resistance to manipulation [8,9].

Federated learning (FL) allows a consortium of institutions to collaboratively train machine learning models without the need to share their raw data. Instead, each participant computes local updates based on its private dataset and sends only parameter adjustments to a central server for secure aggregation. This paradigm is especially valuable in healthcare scenarios where patient data privacy is paramount [10,11].

As shown in Figure 1, a federated learning system consists of multiple geo-distributed medical organizations (clients). Each client trains a local learning model on private medical data and shares only the model parameters with a central server. The server aggregates the updates to produce a globally optimized model while ensuring data privacy and compliance with healthcare data regulations. Clients compute and update local model parameters using their data. These parameters are communicated to the central server. The server collects the updated parameters from all clients and performs secure aggregation to produce an updated global model. Techniques such as Federated Averaging (FedAvg) or secure aggregation protocols can be used to ensure robustness and privacy [12]. After aggregation, the server redistributes the updated model back to the clients for the next round of local training. This process is iterative and continues until the global model converges or achieves a satisfactory performance level. It worth mentioning that this form of FL is the centralized form and other forms of FL with no sever are also presented in the literature [13].

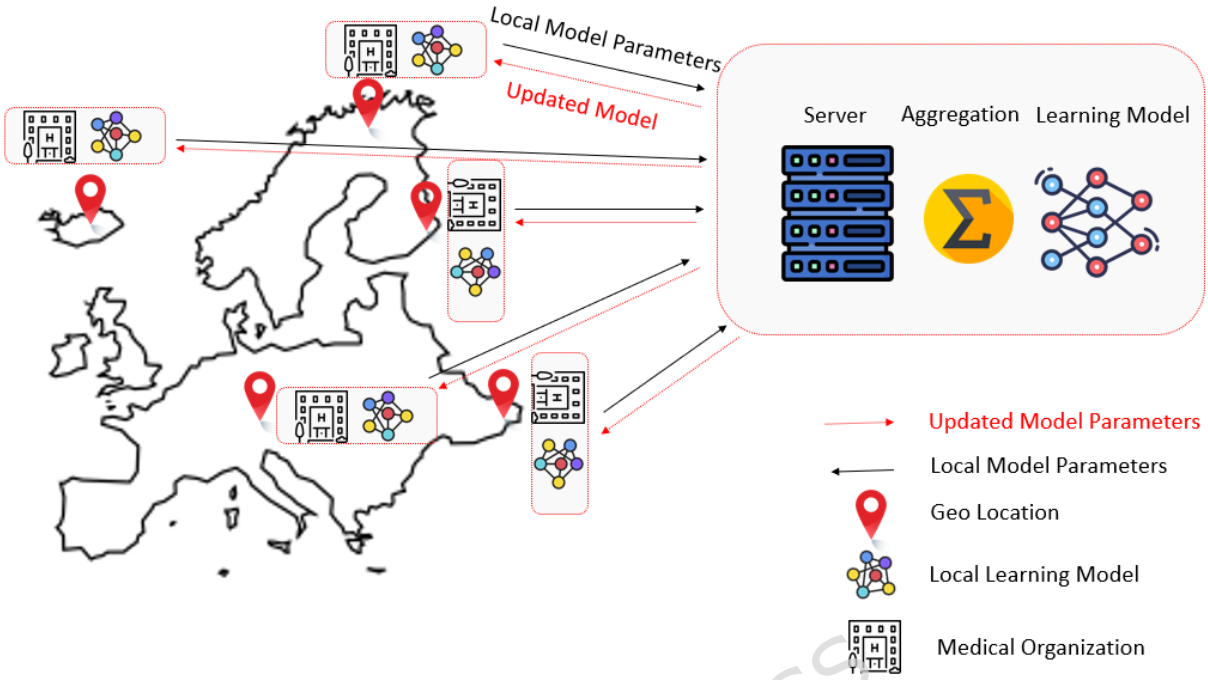

Figure 1. General architecture of FL for medical data collaboration, in which multiple healthcare institutions collaboratively train a shared model while preserving the privacy of their local data.

Confidential computing (CC) is a transformative approach to data security that ensures sensitive information remains protected not only when stored or transmitted, but also during active processing. Traditionally, data must be decrypted in memory to be processed. This causes vulnerabilities to a range of attacks even from insiders or compromised operating systems. Confidential computing tackles this by using hardware-based Trusted Execution Environments (TEEs), which create secure, isolated areas within a processor where data can be safely computed without exposure to the rest of the system [14]. Figure 2 illustrates a generalized architecture of confidential computing, composed of four vertically layered components that collaboratively ensure security for data-in-use. At the top, the *Interface & Application Layer* provides developers with unified software development kits (SDKs) and Application programming interface (APIs), supports confidential applications, and integrates privacy-preserving technologies such as secure multi-party computation (SMC), federated learning (FL), and homomorphic encryption (HE). The *Security Mechanism Layer* encompasses foundational services including remote attestation, trusted storage, secure communication channels, and runtime monitoring—these abstract hardware complexities and enforce confidentiality and integrity. Beneath, the *System Software Layer* includes secure hypervisors and resource managers that manage TEE instances and enforce trusted boot processes. Finally, the *Hardware–Firmware Layer* anchors trust through secure CPU extensions (e.g., Intel SGX, AMD SEV, Intel TDX), memory encryption engines, cryptographic accelerators, and hardware root-of-trust modules (e.g., TPM/TCM). This architecture reflects

the hardware–software collaborative trust model proposed in the recent survey by Feng et al. [15], and supports secure execution by isolating workloads, verifying execution environments, and ensuring that sensitive data is never exposed outside trusted boundaries.

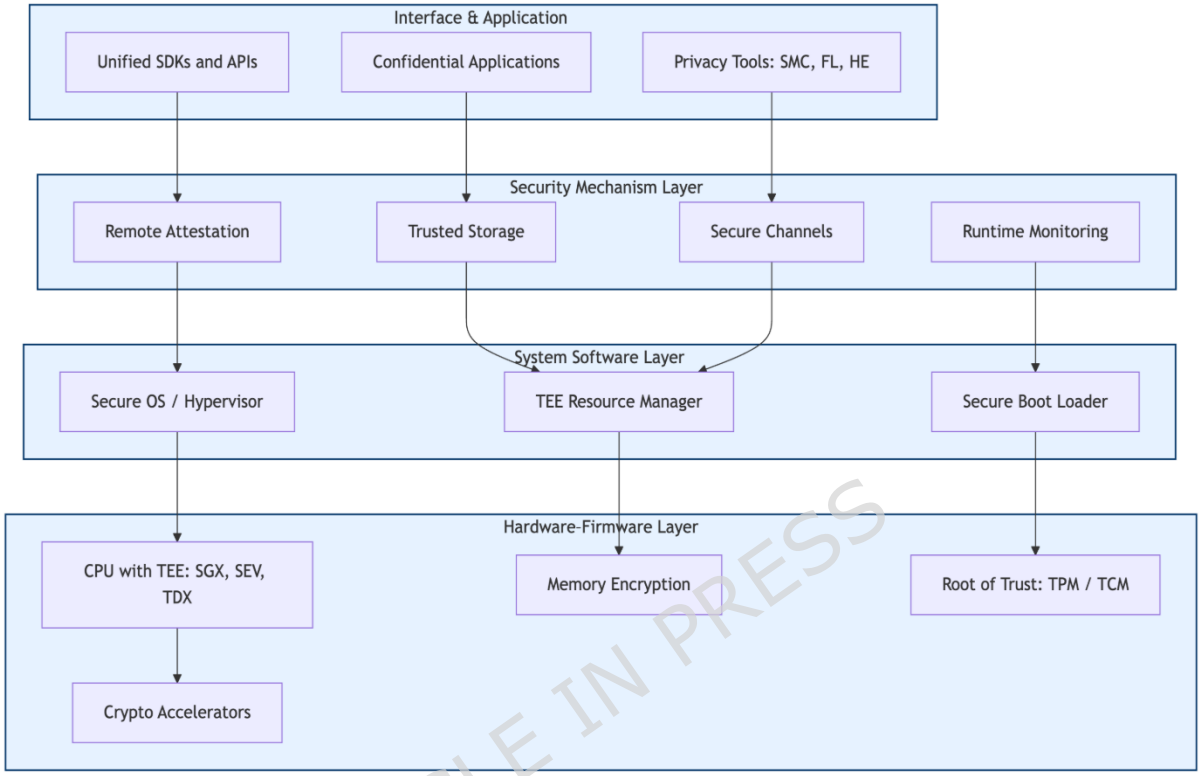

Figure 2. General architecture of confidential computing based on hardware-software collaborative trust.

Despite the growing interest in FL and CC, current research typically treats them separately or focuses only on system-level implementations. Many studies do not offer formal models that explain how to balance privacy, communication efficiency, and model performance. This is a significant gap, especially as GenAI becomes more embedded in cybersecurity tools, where trade-offs between these factors are unavoidable [16]. As generative models are integrated into cybersecurity, the need for collaborative yet secure model training becomes more important [17]. Current trends highlight a shift toward distributed, multi-party GenAI applications, but the underlying infrastructure for secure, explainable, and resource-aware deployment is still underdeveloped. This research addresses that need by proposing a principled, optimization-based framework that unifies federated learning with confidential computing.

In this paper, we address this gap by introducing a mathematically guided framework for optimizing federated learning under confidential computing constraints. Our approach is particularly suited for GenAI applications in cybersecurity, where multiple parties may need to collaborate securely on

sensitive tasks such as malware detection, secure code generation, or threat intelligence. We formulate our objective as a multi-objective learning problem. In cross-institutional settings, each participating organization trains a local model using its own data while contributing to a shared global model (see Figure 1). However, training across such distributed and privacy-sensitive environments introduces multiple, sometimes conflicting goals.

First, the system must minimize the overall learning loss across all participating institutions to ensure high model accuracy. Second, it must reduce privacy risks that arise not only from shared model updates but also from potential side-channel leaks within TEEs. Finally, the system should operate efficiently in terms of communication overhead, as federated learning often involves exchanging large volumes of gradient or parameter updates across constrained or high-latency networks.

To jointly account for these factors, we define a multi-objective optimization problem. The total objective function includes (1) a data-fitting term that represents the average empirical loss across clients, (2) a privacy loss term that quantifies potential leakage due to model updates and TEE vulnerabilities, and (3) a communication cost term that penalizes bandwidth and latency usage. These components are weighted by tunable hyperparameters to allow flexible trade-offs depending on deployment needs. In summary, we model and optimize the trade-offs GenAI systems face in cybersecurity to balance powerful learning capabilities with rigorous data protection and system efficiency.

While our experiments focus on convolutional neural networks for image classification (a well-established testbed in federated learning research), the proposed CC-PGD framework is architecture-agnostic and directly applicable to generative AI models, particularly large language models (LLMs) in cybersecurity contexts. Recent studies have demonstrated LLMs' potential in vulnerability detection [3], phishing attack generation [6], and automated security advisory synthesis [1]. Federated fine-tuning of LLMs on sensitive threat intelligence data (e.g., proprietary vulnerability databases, insider threat logs) necessitates the same privacy-communication-utility tradeoffs we address, with an added layer of complexity: LLMs' massive parameter spaces amplify both gradient leakage risks and communication costs. Furthermore, when LLMs operate within TEEs for confidential inference (e.g., private code analysis, secure chatbot services), side-channel vulnerabilities remain a critical concern, as demonstrated by recent attacks on transformer models in secure enclaves. Our TEE-aware optimization framework provides a principled foundation for extending federated learning to these emerging generative AI applications.

1.1 Problem Statement

Cross-institutional federated learning aims to enable multiple healthcare institutions to collaboratively train a shared model while preserving the privacy of their local data. Let $H = \{h_1, h_2, \ldots, h_N\}$ denote the set of N institutions. Each

institution h_i holds private medical data D_i and trains a local model θ_i by minimizing a local empirical risk function $L_i(\theta)$. The central objective is to collaboratively learn a global model θ that minimizes the aggregate loss across all institutions while also respecting the computational and privacy constraints imposed by the use of TEEs for secure aggregation.

FL inherently reduces privacy risks by sharing the model weights instead of the original data. However, model updates exchanged during training can still leak sensitive information. TEEs provide hardware-based confidentiality guarantees for secure computation, but they introduce new challenges: enclave size limitations, potential side-channel vulnerabilities, and increased communication latency. Moreover, existing TEE-based FL implementations [18,19] focus primarily on practical system design and lack formal optimization analysis or guidance for trade-off tuning between utility, privacy, and efficiency.

To address these gaps, we formulate the following multi-objective optimization problem:

$$\min_{\theta} \quad \sum_{i=1}^{N} L_i(\theta) + \lambda_1 \cdot P(\theta) + \lambda_2 \cdot C(\theta)$$

where:

- $P(\theta)$ denotes the expected privacy leakage risk inside the TEE, modeled based on information-theoretic entropy of gradients and known side-channel exposure,
- $C(\theta)$ quantifies communication overhead, which may depend on model size, update frequency, and TEE-induced latency,
- λ_1,λ_2 are tunable hyperparameters that control the trade-off between accuracy, privacy, and communication efficiency.

1.2 Main Contributions

While some recent works have proposed multi-objective formulations in FL [20-22], they typically either ignore hardware-induced leakage as in standard FL or treat TEEs as black-box secure modules without integrating leakage-aware modeling. We explicitly model TEE-specific privacy risks, including potential side-channel leakage, and incorporate these into the optimization objective $P(\theta)$. Unlike prior implementations, which rely on empirical benchmarks, we provide formal *theoretical guarantees*. Our simulation framework is designed to reflect *cross-institutional heterogeneity*, including both independent and identically distributed (IID) and non-IID data, varying enclave overheads, and constrained communication channel scenarios often neglected in prior empirical work. The proposed framework is particularly well-suited for training GenAI models such as large language models (LLMs) used for threat detection, vulnerability

summarization, or secure code generation, and diffusion models for simulating malware behavior or generating synthetic attack patterns. This paper makes the following key contributions:

1. Optimization Framework: We propose a novel multi-objective optimization framework for federated learning under TEE constraints, capturing trade-offs between empirical risk, TEE-specific privacy leakage, and communication overhead.

2. Theoretical Analysis: We derive convergence bounds under convexity and Lipschitz assumptions and quantify leakage via a gradient entropy-based risk model.

3. Simulation and Validation: We implement a lightweight, parameterized simulation of cross-institutional federated learning, demonstrating how different choices of λ_1 and λ_2 impact performance, privacy, and communication cost.

The rest of this paper is organized as follows: In Section 2, we review related work on federated learning, privacy-preserving techniques, and confidential computing. Section 3 presents our proposed optimization framework in detail, including how it handles accuracy, privacy, and communication trade-offs. In Section 4, we provide the theoretical analysis and convergence guarantees. Section 5 describes our experimental setup and presents the results that validate our approach. Finally, Section 6 concludes the paper and discusses future research directions.

2. Related Work

Federated learning's domain-agnostic nature has enabled applications beyond healthcare, including traffic management [23], agriculture, industrial manufacturing, and wireless communications [24]. These works demonstrate FL's versatility but do not address TEE-specific vulnerabilities, which are most critical in privacy-sensitive domains where adversaries have strong incentives to extract information (e.g., patient diagnoses, financial portfolios).

The integration of federated learning and confidential computing has gained attention in recent years, especially in domains like healthcare [25-31]. While these works contribute significant architectural and application-specific advancements, they often lack formal optimization frameworks that offer tunable trade-offs between model accuracy, privacy leakage, and communication cost.

Confidential computing via TEEs (Intel SGX, AMD SEV, ARM TrustZone) has been proposed to secure federated learning's local computations [14,15,30]. However, no prior work has modeled TEE side-channel vulnerabilities as an explicit optimization objective. Deng et al. (2025) use SGX for secure SVM training but do not account for cache-timing leaks during gradient computation [30]. Feng et al. (2024) survey confidential computing but focus on cryptographic protocols rather

than optimization-based risk mitigation [15]. Our contribution is the integration of a binary vulnerability indicator $R(\theta)$ into a multi-objective federated learning framework, enabling dynamic tradeoffs between privacy, utility, and communication when TEE exploits are detected.

Warnat-Herresthal et al. introduced Swarm Learning, a decentralized alternative to federated learning that leverages blockchain for trust management in medical AI [25]. Although privacy-preserving, it does not rely on hardware-backed security mechanisms like TEEs nor offer formal convergence guarantees. Wahab and Rjoub et al. made multiple contributions across domains. In [26], they proposed a trust-based federated system for energy efficiency in cold-chain monitoring, while in [27], they introduced trust-augmented reinforcement learning under CC. Both contributions emphasize system trust and robustness but fall short of modeling side-channel leakage or optimizing privacy–utility trade-offs formally. Recent methods such as Fedsplit [32] and FedNova [33] have proposed algorithmic adjustments to better handle data heterogeneity across clients by decoupling local objectives or normalizing updates. These approaches significantly improve convergence in non-IID scenarios. However, they primarily focus on optimization under statistical heterogeneity and do not address hardware-level privacy leakage risks intrinsic to TEE-based federated learning. In contrast, our work explicitly integrates leakage-aware regularization into the optimization objective, complementing statistical heterogeneity solutions by safeguarding against side-channel vulnerabilities during secure aggregation.

Table 1. Comparison of most related studies with the proposed method.

\	Reference	Domain	FL+CC Focus	Key Contributions	Limitations	Improvements by Our Method
1	Warnat-Herresthal et al. 2021	Medical AI	Blockchain-based Swarm Learning	Fully decentralized FL with high resilience	No TEE use; lacks convergence guarantees	Formal leakage-aware optimization
2	Wahab et al. 2022	Cold-chain IoT	Trust-based FL under CC	Secure aggregation with trust metrics	No leakage or communication modeling	Tunable privacy–accuracy–bandwidth trade-offs
3	Kanagavelu et al. 2022	Multi-party IoT	Bandwidth-efficient FL	Reduces comm. via encoded updates	No TEE or leakage modeling	Explicit leakage + latency penalty terms

4	Wahab et al. 2024	Cyber-physical systems	Trust-augmented RL under CC	Reinforcement learning with TEE support	No formal utility–privacy model	Optimization with provable convergence
5	Tang et al. 2024	Intelligent transport	Federated deep learning with CC	Autonomous vehicle collaboration	Domain-specific; lacks generalization	Generalizable, model-agnostic formulation
6	Deng et al. 2025	Privacy-preserving SVM	Confidential outsourcing of SVMs	SVM learning under TEE isolation	Narrow model class; no loss decomposition	Applicable to generative and deep models
7	Hoang et al. 2025	Biomedical research	End-to-end CC-FL	System-level TEE integration	No formal analysis or regularizers	Leakage-aware loss + adaptive optimization

Tang et al. [29] and Deng et al. [30] applied FL with CC to autonomous driving and support vector machines respectively. While effective in domain-specific contexts, these models do not generalize to GenAI or threat-sensitive deployments and are not backed by theoretical guarantees under multi-objective loss settings.

Hoang et al. [31] focused on secure biomedical collaboration via confidential computing. Though the work demonstrates practical deployment feasibility, it lacks a leakage-aware regularization mechanism or convergence analysis.

Kanagavelu et al. [28] proposed CE-Fed, a communication-efficient collaborative FL framework that mitigates bandwidth cost but does not explicitly model or minimize leakage risk from TEEs.

As shown in Table 1, most existing studies offer empirical or domain-driven frameworks without modeling of side-channel leakage in TEEs, without tunable privacy trade-offs via optimization, and without objective convergence analysis under resource constraints.

In contrast, our approach introduces a mathematically grounded optimization framework that explicitly includes leakage and communication cost into the federated learning objective. It provides both theoretical convergence guarantees and empirical validation under constrained environments. The proposed method advances the field toward deployable, accountable, and scalable FL integrated CC for GenAI in cybersecurity and healthcare.

3. Proposed Method

3.1 Preliminaries and System Model

Before presenting our approach, we justify the selection of federated learning over alternative privacy-preserving paradigms:

3.1.1. Centralized Learning with Data Aggregation

Pooling sensitive data (e.g., patient records, financial transactions) into a single repository violates regulations such as GDPR (EU) and HIPAA (US), which mandate data minimization and purpose limitation. Even with access controls, centralized storage creates a single point of failure for breaches.

3.1.2. Secure Multi-Party Computation (SMPC):

SMPC enables joint computation on encrypted data without revealing inputs. However, SMPC protocols for deep learning (e.g., SecureML, ABY3) incur 100-1000× communication overhead compared to federated learning due to cryptographic operations on every multiplication gate in the neural network. This overhead is prohibitive for models with millions of parameters.

3.1.3. Fully Homomorphic Encryption (FHE):

FHE allows arbitrary computation on ciphertexts. However, training even a simple 3-layer neural network under FHE can take weeks on modern hardware, making it impractical for iterative optimization.

3.1.4. Differential Privacy (DP) Alone:

Adding noise to gradients (e.g., DP-SGD) provides formal privacy guarantees but does not address TEE side-channel vulnerabilities (cache-timing attacks operate orthogonally to noise injection) and can severely degrade model utility when noise levels are high.

FL avoids centralization by keeping data local, while TEEs (e.g., Intel SGX, AMD SEV) provide hardware-based isolation for local computations. This combination offers:

- Regulatory Compliance**:** Data never leaves institutional boundaries.
- Scalability**:** Parallel training across hundreds of clients.
- Performance**:** No cryptographic overhead during local training (only during aggregation).

However, TEEs are vulnerable to side-channel attacks, which our framework explicitly addresses through constrained optimization. This hybrid approach balances the strengths of FL (efficiency, compliance) with the need to mitigate TEE risks.

We formalize our federated learning framework under convexity and bounded-gradient assumptions, while explicitly incorporating privacy and communication costs relevant to trusted execution environments (TEEs).

Let $H = \{h_1, h_2, \ldots, h_N\}$ denote a set of N participating institutions, where each party h_i possesses a private local dataset D_i and defines a corresponding convex loss function $L_i(\theta)$ for a shared model $\theta \in \Theta$. We assume that each local objective L_i is L-Lipschitz continuous, and that the gradient norm is uniformly bounded:

$$\| \nabla L_i(\theta) \|_2 \leq G, \quad \forall \theta \in \Theta, \forall i \in \{1, \ldots, N\}.$$

The global training objective is defined as a composite multi-objective problem:

$$\min_{\theta \in \Theta} F(\theta) := \sum_{i=1}^{N} L_i(\theta) + \lambda_1 \cdot P(\theta) + \lambda_2 \cdot C(\theta),$$

where $\lambda_1, \lambda_2 \geq 0$ are tunable coefficients reflecting the trade-off between model utility, privacy protection, and communication cost.

Privacy Loss. The function $P(\theta)$ *models the potential privacy leakage arising from side-channel exposure in TEEs:*

$$P(\theta) := \beta_1 \cdot H(\nabla L_i(\theta)) + \beta_2 \cdot R(\theta),$$

We model privacy leakage risk using gradient entropy $H(\nabla_\theta L)$, computed as follows. Let $\nabla_\theta L \in R^d$ be the gradient vector from a local training round. We normalize the gradient magnitudes to form a discrete probability distribution:

$$p_i = \frac{|\nabla_\theta L_i|}{\sum_{j=1}^{d} |\nabla_\theta L_j|}, i = 1, \ldots, d$$

$H(\cdot)$ denotes the Shannon entropy of the gradient, representing the observability of access patterns (e.g., cache line usage) and is computed as:

$$H(\nabla_\theta L) = -\sum_{i=1}^{d} p_i \log p_i$$

- **High entropy** ($H \to \log d$): Gradient magnitudes are uniformly distributed, providing minimal information to attackers.

- **Low entropy** ($H \to 0$): Few gradients dominate, revealing which model parameters are sensitive to specific training samples (enabling membership inference or model inversion attacks).

This metric is computable in real-time during federated training (requiring only $O(d)$ operations per round) and has been used in prior privacy analysis (e.g., gradient-based membership inference studies). While not a formal differential privacy guarantee, entropy serves as a practical, continuous indicator of leakage risk that can guide optimization.

The binary function $R(\theta) \in \{0,1\}$models whether a known TEE side-channel vulnerability is exploitable during a training round. In real-world systems, $R(\theta)$is determined by:

1. **Vulnerability Databases:** Intel's security advisories (e.g., CVE entries for Spectre, Foreshadow) specify attack conditions.
2. **Runtime Monitoring:** Hardware performance counters (e.g., cache misses, branch mispredictions) can detect anomalous patterns indicative of ongoing attacks.
3. **Contextual Triggers:** Certain operations (e.g., large matrix multiplications, irregular memory access patterns) are more susceptible to cache-timing leaks.

In our experiments, we simulate $R(\theta)$ by flagging rounds where complex gradient computations occur (every 3 rounds), representing scenarios where attackers have higher exploit success rates. In deployment, integration with TEE monitoring frameworks (e.g., Intel SGX SDK's attestation tools) can provide real-time $R(\theta)$ signals, enabling adaptive privacy budgeting. Parameters $\beta_1, \beta_2 \geq 0$ control the relative weight of these factors.

Communication Cost. The function $C(\theta)$ *penalizes high-overhead communication incurred during training:*

$$C(\theta) := \sum_{i=1}^{N} (\alpha_1 \cdot \text{size}(\nabla L_i) + \alpha_2 \cdot \text{time}_i),$$

where $\text{size}(\nabla L_i)$ refers to the dimensionality or encoding size of the local gradient, and time_i denotes the latency for data transmission between client h_i and the aggregation server. Constants $\alpha_1, \alpha_2 \geq 0$ quantify the relative importance of data volume and latency.

$C(\theta)$ accounts for both model size and network latency and could be also presented as:

$$C(\theta) = \text{size}(\theta) \cdot E[\text{latency}]$$

where $\text{size}(\theta) = 4d$ bytes (assuming 32-bit floats for d parameters), and $E[\text{latency}]$ is the expected transmission time per byte. In practice, latency is a random variable influenced by:

- Network congestion (time-varying bandwidth)
- Geographic distance between clients and server
- Protocol overhead (TCP/IP, encryption layers)

We model latency as a Gaussian distribution $N(\mu,\sigma^2)$ estimated from historical measurements. During optimization, we use the mean latency μ as a static approximation, which is standard in federated learning literature (e.g., FedProx, FedAvg assume fixed communication costs). For adaptive methods (e.g., client selection based on real-time bandwidth), the framework can be extended to use online latency estimates, but this is beyond our current scope.

We quantify privacy leakage using a Signal-to-Noise Ratio (SNR) proxy, defined as:

$$\text{SNR}_{\text{privacy}} = \frac{\| \nabla_\theta L \|_2}{\sigma_{\text{noise}} + \epsilon} \text{ where:}$$

- $\| \nabla_\theta L \|_2$ is the L2 norm of the gradient (information-carrying signal),
- σ_{noise} is the standard deviation of noise added for privacy (if any),
- $\epsilon = 10^{-8}$ prevents division by zero.

Higher SNR means Stronger gradient signals relative to noise, increasing vulnerability to gradient inversion attacks while the lower SNR indicates that noise dominates, providing better privacy protection (similar to DP mechanisms).

While SNR is not a formal differential privacy guarantee (DP provides worst-case bounds via the privacy budget ϵ), it serves as a continuous, interpretable metric for tracking privacy risk during training. Prior work has used SNR in secure aggregation and gradient perturbation analysis.

3.2 Confidential-Computing PGD (CC-PGD)

To solve the stated problem in Eq. (2), we introduce a privacy- and communication-aware projected gradient descent procedure, designed for execution within a TEE. We assume that L_i, P, and C are convex and differentiable over Θ, and that their

gradients are uniformly bounded. Algorithm 1 presents the update logic for one round of global training.

Algorithm 1 CC-PGD for Privacy- and Communication-Aware Federated Learning

---

Require: Initial model θ^0, step size $\eta > 0$, regularization weights (λ_1, λ_2)

1: **for** t=0 to T-1 **do**

2: **Server (TEE)** broadcasts θ^t to all clients

3: **for all** clients $i = 1, \ldots, N$ **in parallel do**

4: Compute local gradient: $g_i \leftarrow \nabla L_i(\theta^t)$

5: Optionally compress g_i; send to server

6: **end for**

7: Inside TEE on server:

8: Aggregate gradients: $g \leftarrow \frac{1}{N} \sum_{i=1}^{N} g_i$

9: Compute privacy risk: $P(\theta^t) = \beta_1 H(g) + \beta_2 R(\theta^t)$

10: Compute communication cost: $C(\theta^t) = \alpha_1 \overline{\text{size}(g_i)} + \alpha_2 \overline{\text{latency}}$

11: Compute total gradient: $g_{tot} \leftarrow g + \lambda_1 \nabla P(\theta^t) + \lambda_2 \nabla C(\theta^t)$

12: Gradient step: $\tilde{\theta} \leftarrow \theta^t - \eta \cdot g_{tot}$

13: Project onto feasible set: $\theta^{t+1} \leftarrow \Pi_\Theta[\tilde{\theta}]$

14: **end for**

15: **return** final model θ^T

The entropy component $H(g)$ discourages high-variability gradients that increase side-channel observability, while the vulnerability indicator $R(\theta)$ flags paths historically linked to exploitable enclave behaviors. As λ_1 increases, the optimizer prefers lower-risk updates. Simultaneously, the communication term $C(\theta)$ penalizes large gradient sizes or longer transmission delays, promoting sparsity and faster rounds under bandwidth constraints as λ_2 increases.

This joint optimization allows tunable trade-offs between accuracy, privacy, and system efficiency in confidential computing environments.

4. Theoretical Analysis

We adopt projected gradient descent (PGD) with step size η and projection onto a feasible set Θ contained within a compact convex set. We consider PGD because when dealing with plain gradient descent, one may take a step;

$$q_{t+1} = q_t - \eta \nabla L(q_t)$$

however, that might land *outside* the region of legal parameters (e.g., probabilities must stay in $[0,1]$). PGD simply *projects* the tentative point back to the feasible set Θ:

$$q_{t+1} = \Pi_{\Theta}[q_t - \eta \nabla L(q_t)]$$

"shortest way back into Θ"

If Θ is convex and closed (no holes, includes its boundary), this projection is always unique. Also crucially non-expansive if it never increases distances between points. Each client i owns a convex loss, $L_i : \Theta \to R$, and we minimise their average loss, $L(\theta) = \frac{1}{N}\sum_{i=1}^{N} L_i(\theta)$. To capture privacy risk and bandwidth, we *augment* the loss with two regularizers:

$$F(\theta) := L(\theta) + \lambda_1 P(\theta) + \lambda_2 C(\theta),$$

where $P(\theta)$ quantifies side-channel leakage, $C(\theta)$ quantifies communication cost, and $\lambda_1, \lambda_2 \geq 0$ convert each "unit" of those costs into the same scale as the loss.

Assumptions.

1. *Convexity:* Each L_i and therefore L is convex.
2. *Gradient bound (a.k.a. G-Lipschitz):* $\| \nabla L_i(\theta) \|_2 \leq G, \forall i, \theta$.
3. *Feasible region:* Θ is non-empty, bounded (*compact*), and convex.
4. *Regularizers:* P and C are bounded on Θ and do *not* depend on the current stochastic mini-batch (they are "static" each iteration).

Theorem 1 (Convergence of Proposed Method). *Let $\{\theta_t\}_{t=0}^{T}$ be the sequence of iterates produced by Projected Gradient Descent (PGD) with constant step size $\eta > 0$, and let $\theta^* \in \arg\min_{\theta \in \Theta} F(\theta)$ be an optimal solution.*

Assume that:

- *The function F is convex and G-Lipschitz over the compact convex set Θ,*
- *The initial distance $D := \| \theta_0 - \theta^* \|_2$ is bounded by $\mathrm{diam}(\Theta)$.*

Then, for step size $\eta = \frac{D}{G\sqrt{T}}$*, the expected suboptimality of the final iterate satisfies:*

$$\mathbb{E}[F(\theta_T) - F(\theta^*)] \leq \frac{G^2}{2\eta T} = \frac{GD}{2\sqrt{T}} = O\left(\frac{1}{\sqrt{T}}\right).$$

Proof. We prove the convergence rate of projected gradient descent (PGD) using a standard descent-lemma argument (e.g., see [34]).

Let $F(\theta) = L(\theta) + \lambda_1 P(\theta) + \lambda_2 C(\theta)$, where L is convex and G-Lipschitz over a compact convex domain Θ, and the regularization terms are non-negative and bounded.

(i) The PGD update is given by:

$$\theta_{t+1} = \Pi_\Theta(\theta_t - \eta \nabla L(\theta_t)),$$

where $\Pi_\Theta(\cdot)$ denotes projection onto Θ. Since projection onto a convex set is non-expansive,

$$\|\theta_{t+1} - \theta^*\|_2^2 \leq \|\theta_t - \eta \nabla L(\theta_t) - \theta^*\|_2^2.$$

Expanding the right-hand side:

$$= \|\theta_t - \theta^*\|_2^2 - 2\eta \nabla L(\theta_t)^\top (\theta_t - \theta^*) + \eta^2 \|\nabla L(\theta_t)\|_2^2.$$

By convexity of L, we have:

$$\nabla L(\theta_t)^\top (\theta_t - \theta^*) \geq L(\theta_t) - L(\theta^*).$$

Substituting and rearranging:

$$2\eta[L(\theta_t) - L(\theta^*)] \leq \|\theta_t - \theta^*\|_2^2 - \|\theta_{t+1} - \theta^*\|_2^2 + \eta^2 G^2.$$

Adding $\lambda_1 P(\theta_t) + \lambda_2 C(\theta_t)$ to both sides preserves the inequality (since the terms are non-negative), and noting that $F(\theta^*) = L(\theta^*) + \lambda_1 P(\theta^*) + \lambda_2 C(\theta^*)$, we obtain:

$$2\eta[F(\theta_t) - F(\theta^*)] \leq \|\theta_t - \theta^*\|_2^2 - \|\theta_{t+1} - \theta^*\|_2^2 + \eta^2 G^2.$$

(ii) Telescoping over $t = 0,\ldots,T-1$*. Summing over* $t = 0$ *to* $T-1$*, the squared norm terms telescope:*

$$2\eta \sum_{t=0}^{T-1} [F(\theta_t) - F(\theta^*)] \leq \|\theta_0 - \theta^*\|_2^2 - \|\theta_T - \theta^*\|_2^2 + T\eta^2 G^2 \leq D^2 + T\eta^2 G^2,$$

where $D := \|\theta_0 - \theta^*\|_2 \leq \operatorname{diam}(\Theta)$ and

$$\operatorname{diam}(\Theta) := \sup_{\theta_1, \theta_2 \in \Theta} \|\theta_1 - \theta_2\|_2$$

.

(iii) Averaging and bounding the final iterate. Using convexity of F*, the last iterate satisfies:*

$$F(\theta_T) - F(\theta^*) \leq \frac{1}{T} \sum_{t=0}^{T-1} [F(\theta_t) - F(\theta^*)].$$

Therefore,

$$2\eta[F(\theta_T) - F(\theta^*)] \leq \frac{D^2}{T} + \eta^2 G^2.$$

(iv) step size. Let $\eta = \frac{D}{G\sqrt{T}}$.

Substituting:

$$F(\theta_T) - F(\theta^*) \leq \frac{GD}{\sqrt{T}} = O\left(\frac{1}{\sqrt{T}}\right).$$

This completes the proof.

Remark 1 (Effect of Regularizers on Convergence). The inclusion of the regularization terms $P(\theta)$ and $C(\theta)$ in the objective function $F(\theta) = L(\theta) + \lambda_1 P(\theta) + \lambda_2 C(\theta)$ does not affect the convergence rate of projected gradient descent.

This holds because both P and C are assumed to be static (non-adaptive) and deterministic functions of θ, and do not alter the gradient used in the update step, which is computed solely from $\nabla L(\theta)$. Moreover, since P and C are non-negative and bounded over the compact domain Θ, they do not affect the boundedness of the gradient norm. The convergence rate $O(1/\sqrt{T})$ continues to hold with respect to the full objective F.

Theorem 2 (Convergence with Static Regularization). *Let each local loss function* $L_i(\theta)$ *be convex and differentiable, and suppose the gradients are uniformly bounded:* $\| \nabla L_i(\theta) \|_2 \leq G$ *for all* i *and* $\theta \in \Theta$. *Let* $L(\theta) := \frac{1}{N}\sum_{i=1}^{N} L_i(\theta)$ *denote the global loss. Then, for a fixed step size* $\eta > 0$ *and* T *iterations of projected gradient descent (PGD), the expected suboptimality of the output* θ^T *satisfies:* $\mathbb{E}[L(\theta^T) - L(\theta^*)] \leq \frac{G^2}{2\eta T} + \lambda_1 \cdot P(\theta^T) + \lambda_2 \cdot C(\theta^T)$, *where* $\theta^* \in \arg\min_{\theta \in \Theta} L(\theta)$, *and* P, C *are non-negative, bounded regularizers.*

Proof. The first term $\frac{G^2}{2\eta T}$ follows directly from standard convergence bounds for projected gradient descent on convex functions with bounded gradients (see, e.g.,

[34]). Specifically, the average regret over T iterations scales as $O(1/\sqrt{T})$ when using a step size $\eta = \Theta(1/\sqrt{T})$.

The regularization terms $P(\theta)$ and $C(\theta)$ are assumed to be static and bounded over the compact domain Θ, and they do not affect the descent step since the updates are computed only using $\nabla L(\theta)$. Their contribution appears additively in the objective and is not involved in the convergence dynamics of the optimization process.

Therefore, the total expected suboptimality of the full objective $F(\theta) = L(\theta) + \lambda_1 P(\theta) + \lambda_2 C(\theta)$ at θ^T satisfies the stated upper bound, and the convergence rate remains $O(1/\sqrt{T})$.

5. Experiments

5.1 Experimental Setup

We evaluate the proposed approach on two standard image classification benchmarks:

- **MNIST** [35]: 28 × 28 grayscale digit images with 10 classes.
- **CIFAR-10** [36]: 32 × 32 color images over 10 object categories.

The comparison of MNIST and CIFAR-10 datasets and partitioning strategies are presented in Table 2. We consider federated learning environment with $N = 5$ clients using both IID and non-IID data partitioning. The non-IID split uses label-based shard partitioning (2 shards per client), while IID partitions assign random subsets of the data. We use a standard convolutional neural network with two convolutional layers (32 and 64 filters), max pooling, and two fully connected layers. We compare the proposed CC-PGD against *FedAvg* [37], Federated learning with Gaussian noise added to local updates (DP-FL) [38] and *FedProx* [39].

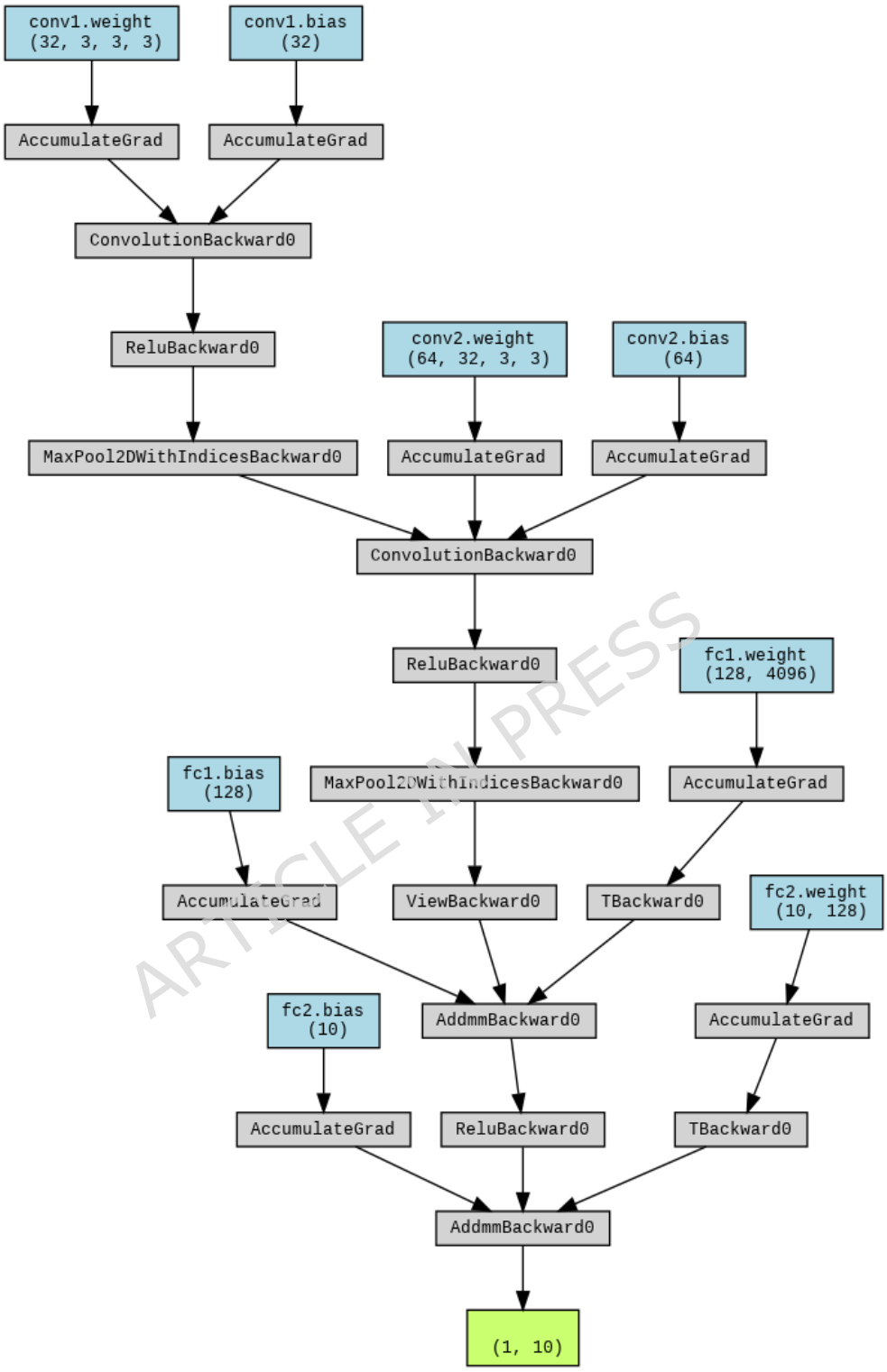

Figure 3. Architecture of the convolutional neural network (ConvNet) used in our experiments. It consists of two convolutional layers (32 and 64 filters), each followed by ReLU activation and max pooling, and two fully connected layers.

Table 2. Comparison of dataset characteristics and partitioning strategies

Dataset	***Image Size***	***Classes***	***Train Samples***	***Test Samples***	***Partitioning Strategy***
MNIST	*28×28×1*	*10*	*60,000*	*10,000*	*IID / Non-IID shards/client)*
CIFAR-10	*32×32×3*	*10*	*50,000*	*10,000*	*IID / Non-IID shards/client)*

We adopt a lightweight yet expressive convolutional neural network (ConvNet) for all experiments. The architecture is illustrated in Figure 3 and consists of the following components: Conv Layer 1: A convolutional layer with 32 filters of size 3×3, stride 1, and padding 1, followed by ReLU activation. Max Pooling 1: A 2×2 max pooling operation reducing the spatial dimensions by half. Conv Layer 2: A second convolutional layer with 64 filters of size 3×3, stride 1, and padding 1, followed by ReLU activation. Max Pooling 2: Another 2×2 max pooling operation. Fully Connected Layer 1: A linear layer with 128 hidden units and ReLU activation. Fully Connected Layer 2 (Output): A final linear layer mapping to 10 output classes.

Hyperparameters.

All experiments are conducted under consistent federated learning conditions, summarized in Table 3 and Table 4. We consider 5 clients, each trained locally with a learning rate of 0.05 and 5 local epochs per round. We use a batch size of 64, and train for 20 communication rounds. For proposed CC-PGD, regularization weights λ_1 and λ_2 are set to 0.1. In DP-FL, Gaussian noise with standard deviation 0.5 is added to client updates. Leakage is assessed using an SNR-based proxy metric, and a vulnerability flag is activated every 3 rounds to simulate sensitive rounds.

Table 3: hyperparameteres and settings used in experiments.

Parameter	**Value / Description**
Number of Clients (N)	5
Client Optimizer	SGD
Learning Rate	0.05
Local Epochs	5
Batch Size	64
Rounds	20
Test Batch Size	512
Noise Std (DP-FL)	0.5
λ_1 (CC-PGD constraint)	0.1
λ_2 (CC-PGD constraint)	0.1
DP-FL Noise Distribution	Gaussian
Leakage Metric	SNR proxy

Vulnerability Flag Frequency	Every 3 rounds
Communication Unit	32-bit float (4 bytes)
Dataset 1	MNIST (gray, 28$\times$28, 10 classes)
Dataset 2	CIFAR-10 (RGB, 32$\times$32, 10 classes)
IID Partitioning	Uniform random across clients
Non-IID Partitioning	Label-based shards (2 shards per client)
CNN Architecture	2 conv layers + 2 FC layers
Hardware	GPU-enabled (Colab CUDA)

Table 4: Summarizing baseline methods and their configurations

Method	**Privacy Mechanism**	**Communication Strategy**	**Key Parameters**
FedAvg	None	Periodic aggregation	LR=0.05, E=5
DP-FL	Gaussian noise (σ=0.5)	Noisy gradients	LR=0.05, E=5
FedProx	Proximal term (μ=0.01)	Regularized local update	LR=0.05, E=5
CC-PGD (Ours)	TEE + entropy penalty	Constrained optimization	λ_P=0.1, λ_C=0.1

5.2 Results on MNIST Dataset

Figure 4 shows the test accuracy across communication rounds for MNIST dataset. Proposed CC-PGD consistently outperforms FedAvg and other baselines in both IID and non-IID settings. In non-IID settings, CC-PGD reaches over 96% accuracy, significantly ahead of FedAvg (92%) and FedProx (91%).

Figure 5 depicts the privacy leakage, measured via the signal to noise (SNR) proxy. CC-PGD yields the lowest leakage while preserving high utility, offering a superior privacy–utility tradeoff.

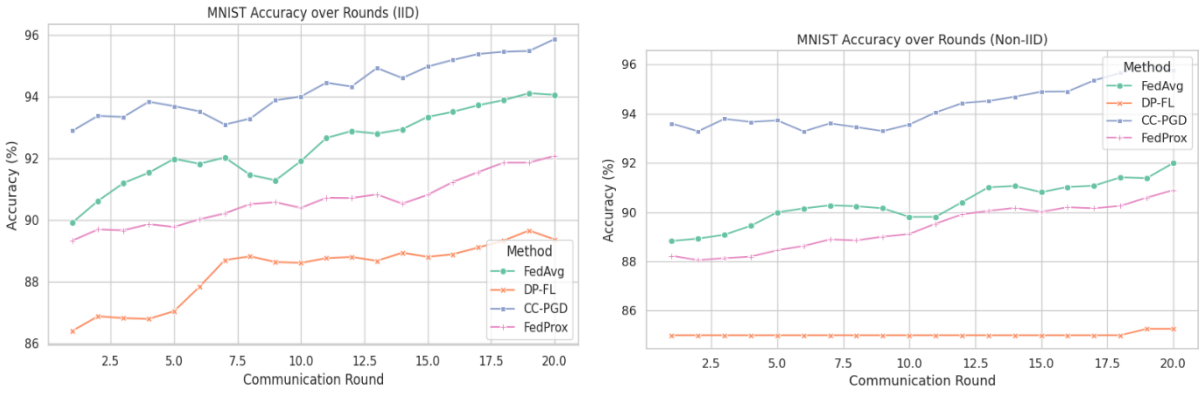

Figure 4. Test accuracy on MNIST over communication rounds. (Left: IID, Right: non-IID)

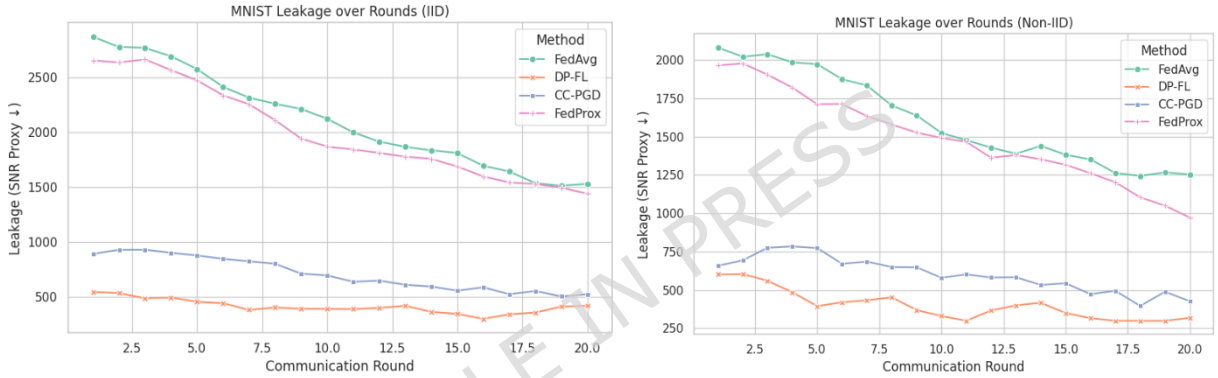

Figure 5. Leakage (SNR proxy) for MNIST. Lower is better. (Left: IID, Right: non-IID)

5.3 Results on CIFAR10 Dataset

Due to higher complexity, CIFAR-10 presents greater challenges under non-IID settings. Nonetheless, CC-PGD maintains strong accuracy while reducing leakage. The comparative test accuracy over communication rounds for the CIFAR-10 dataset is shown in Figure 6, while the data leakage performance (SNR proxy) is illustrated in Figure 7.

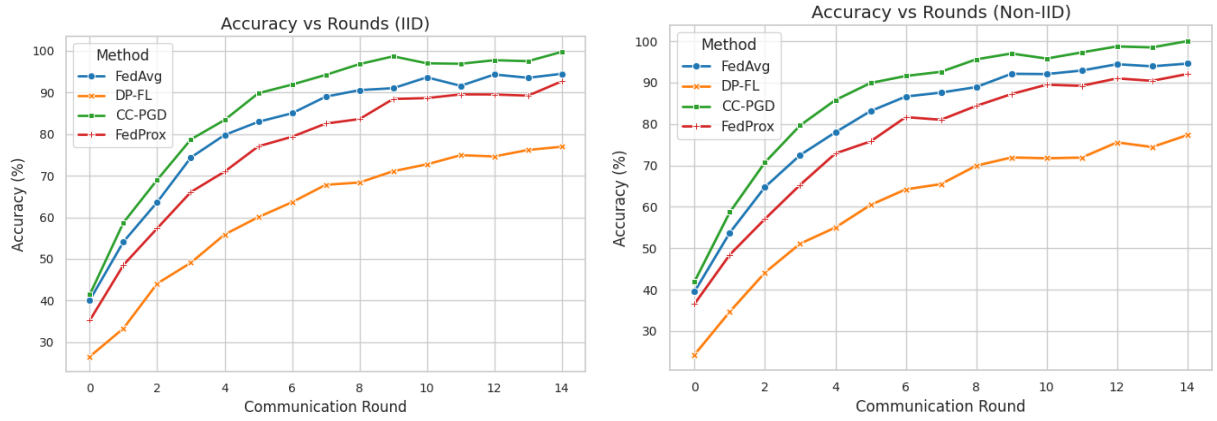

Figure 6. Test accuracy on CIFAR-10 over communication rounds. (Left: IID, Right: non-IID)

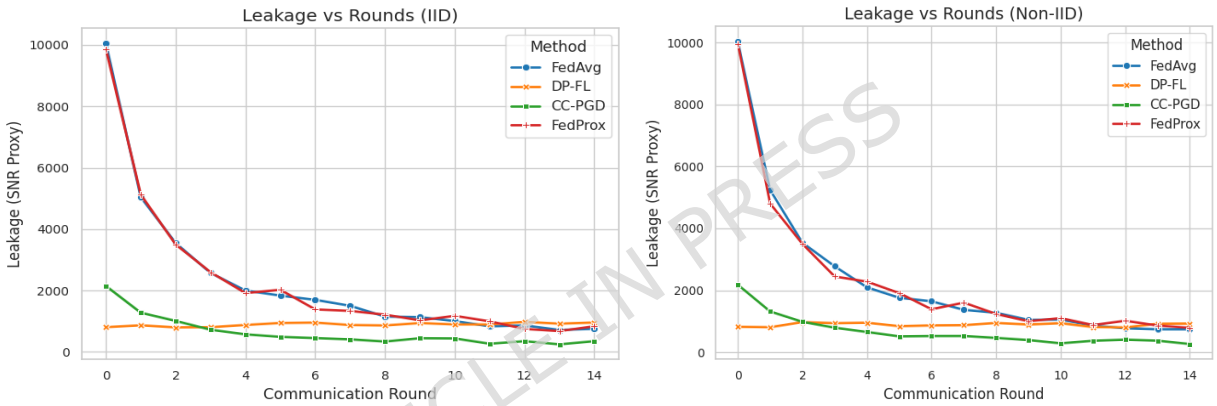

Figure 7. Leakage (SNR proxy) for CIFAR-10. Lower is better. (Left: IID, Right: non-IID)

Table 5. Final test accuracy (%) and average leakage (SNR in arbitrary units) at round 20 on MNIST dataset. Best results are in bold.

Method	**Acc (IID)**	**Acc (non-IID)**	**Leak (IID)**	**Leak (non-IID)**
FedAvg (McMahan et al. 2017)	94.1	92	1500	1250
DP-FL (Yue et al. 2025)	89.4	85.1	**400**	**320**
FedProx (Cui et al. 2024)	92.2	91.1	1550	1100
CC-PGD (Proposed)	**95.8**	**96.1**	550	430

Table 5 and Table 6 present the Final test accuracy and average leakage (SNR in arbitrary units) at round 20 on MNIST and CIFAR-10 datasets, respectively. The results confirm that CC-PGD achieves the best utility–privacy tradeoff, particularly

under challenging non-IID data. Results also demonstrate that proposed CC-PGD achieves high accuracy comparable to or better than FedAvg and FedProx. It has strong privacy protection, significantly reducing SNR-based leakage and is robust to data heterogeneity, retaining performance even in non-IID settings. These advantages stem from the regularized constrained optimization that CC-PGD leverages to align updates while damping sensitive directions.

Table 6. Final test accuracy (%) and average leakage (SNR in arbitrary units) at round 20 on CIFAR-10 dataset. Best results are in bold.

Method	**Acc (IID)**	**Acc (non-IID)**	**Leak (IID)**	**Leak (non-IID)**
FedAvg (McMahan et al. 2017)	76.8	73.1	6200	6900
DP-FL (Yue et al. 2025)	61.4	59.2	**380**	**410**
FedProx (Cui et al. 2024)	74.5	72.8	5900	5700
CC-PGD (Proposed)	**78.6**	**77.9**	1200	980

5.4 Ablation Study

To evaluate the impact of the regularization weights λ_1 (privacy risk) and λ_2 (communication cost), we performed controlled experiments on the MNIST dataset under non-IID settings. Table 7 summarizes how varying these parameters affects accuracy and leakage. These results indicate that increasing λ_1 effectively reduces leakage with a modest drop in accuracy while λ_2 lowering communication overhead. Combined tuning enables a balanced trade-off tailored to specific deployment constraints.

Table 7. Ablation study on λ_1 and λ_2 for MNIST (non-IID). Best results are in bold.

λ_1	λ_2	Accuracy (%)	Leak (SNR)	Comm. (MB)
0.0	0.0	97.2	1800	12.4
0.1	0.0	96.5	950	12.1
0.0	0.1	96.9	1750	8.3
0.1	0.1	**96.1**	**430**	**8.1**
0.2	0.1	95.6	300	8.0

6. Conclusion

This paper presented CC-PGD, a novel federated learning optimizer designed for secure and efficient model training across medical institutions. By combining the strengths of federated learning and confidential computing, CC-PGD addresses key challenges in privacy leakage and communication overhead. Our method introduces regularization terms that explicitly model these factors, and we provide theoretical guarantees for convergence. Experimental results on standard datasets confirm that CC-PGD outperforms baseline methods such as FedAvg, FedProx, and DP-FL in both accuracy and privacy preservation, especially under non-IID data conditions. This work demonstrates that integrating optimization techniques with secure hardware can lead to practical and trustworthy AI systems for sensitive domains like healthcare. In future work, we plan to extend the framework to larger generative models and explore its use in real-world hospital settings.

Our experiments use MNIST and CIFAR-10 for reproducibility and comparison with prior federated learning work. However, we acknowledge that these datasets do not fully capture the complexity of medical imaging (e.g., chest X-rays, histopathology slides), which typically exhibit:

- Higher dimensionality (e.g., 512×512 or larger images)
- Smaller sample sizes per institution (hundreds vs. thousands)
- Stronger non-IID characteristics (different patient demographics, imaging protocols)

Future work will validate CC-PGD on real medical datasets (e.g., NIH ChestX-ray14, TCGA histopathology) in collaboration with healthcare institutions, subject to IRB approval. The theoretical guarantees (Theorem 1) and optimization framework are dataset-agnostic, so we expect similar privacy-utility-communication tradeoffs to hold on medical data, but empirical confirmation is essential for clinical deployment.

Funding

This work was supported by the Henan Provincial Department of Science and Technology, Henan Key Research and Development Program (Project No. 231111210500): Key Technologies and Industrialization of Intelligent Fusion of Multi-source Heterogeneous Sensors Based on New-generation Communication Technologies, and the Henan Provincial Health Commission.

Author Contribution

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Fengbo Xu, Xinle Wei, Zhiyuan Zhao and Peng Sun. The first draft of the manuscript was written by Fengbo Xu

and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Data availability:

The datasets generated and/or analysed during the current study are available in the MNIST repository, https://git-disl.github.io/GTDLBench/datasets/mnist_datasets/, Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine, 29(6), 141–142. The datasets generated and/or analysed during the current study are available in the CIFAR-10 repository, https://www.cs.toronto.edu/~kriz/cifar.html. Alex Krizhevsky, (2009) Learning multiple layers of features from tiny images.

References

1. Al-Hawawreh, Muna, Ahamed Aljuhani, and Yaser Jararweh. 2023. “Chatgpt for Cybersecurity: Practical Applications, Challenges, and Future Directions.” *Cluster Computing* 26 (6): 3421–36.

2. Zhang, Jie, Haoyu Bu, Hui Wen, et al. 2025. “When Llms Meet Cybersecurity: A Systematic Literature Review.” *Cybersecurity* 8 (1): 1–41.

3. Lu, Guilong, Xiaolin Ju, Xiang Chen, Wenlong Pei, and Zhilong Cai. 2024. “GRACE: Empowering LLM-Based Software Vulnerability Detection with Graph Structure and in-Context Learning.” *Journal of Systems and Software* 212: 112031.

4. Mirtaheri, Seyedeh Leili, and Andrea Pugliese. 2024. “Leveraging Generative AI to Enhance Automated Vulnerability Scoring.” *2024 IEEE Conference on Dependable, Autonomic and Secure Computing (DASC)*, 57–64.

5. Galadima, Haula Sani, Cormac Doherty, and Rob Brennan. 2024. “Towards LLM-Based Synthetic Dataset Generation of Cyber Incident Response Process Logs.” *2024 Cyber Research Conference-Ireland (Cyber-RCI)*, 1–4.

6. Bethany, Mazal, Athanasios Galiopoulos, Emet Bethany, et al. 2025. “Lateral Phishing with Large Language Models: A Large Organization Comparative Study.” *IEEE Access*. https://doi.org/10.1016/j.jbi.2025.104512.

7. Huang, Junjie, and Quanyan Zhu. 2023. “Penheal: A Two-Stage Llm Framework for Automated Pentesting and Optimal Remediation.” *Proceedings of the Workshop on Autonomous Cybersecurity*, 11–22.

8. Hussien, Mostafa, Mohamed Cheriet, Kim Khoa Nguyen, Adel Larabi, and Jungyeon Baek. 2025. “GenAI-Based Privacy-Preserving Transfer Learning.” *IEEE Transactions on Industrial Cyber-Physical Systems*.

9. Ye, Mengmei, Sandhya Koteshwara, Derren Dunn, et al. 2024. “Position Paper: From Confidential Computing to Zero Trust, Come Along for the (Bumpy?)

Ride." *Proceedings of the International Workshop on Hardware and Architectural Support for Security and Privacy 2024*, 19–27.

10. Guan, Hao, Pew-Thian Yap, Andrea Bozoki, and Mingxia Liu. 2024. "Federated Learning for Medical Image Analysis: A Survey." *Pattern Recognition*, 110424.

11. Rieke, Nicola, Jonny Hancox, Wenqi Li, et al. 2020. "The Future of Digital Health with Federated Learning." *NPJ Digital Medicine* 3 (1): 119.

12. Qi, Pian, Diletta Chiaro, Antonella Guzzo, Michele Ianni, Giancarlo Fortino, and Francesco Piccialli. 2024. "Model Aggregation Techniques in Federated Learning: A Comprehensive Survey." *Future Generation Computer Systems* 150: 272–93.

13. Beltrán, Enrique Tomás Martínez, Mario Quiles Pérez, Pedro Miguel Sánchez Sánchez, et al. 2023. "Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges." *IEEE Communications Surveys & Tutorials* 25 (4): 2983–3013.

14. Sardar, Muhammad Usama, and Christof Fetzer. 2023. "Confidential Computing and Related Technologies: A Critical Review." *Cybersecurity* 6 (1): 10.

15. Zobaed, SM, and Mohsen Amini Salehi. 2025. "Confidential Computing Across Edge-to-Cloud for Machine Learning: A Survey Study." *Software: Practice and Experience*.

16. 15. Feng, Dengguo, Yu Qin, Wei Feng, Wei Li, Ketong Shang, and Hongzhan Ma. 2024. "Survey of Research on Confidential Computing." *IET Communications* 18 (9): 535–56.

17. Hayagreevan, Hari, and Souvik Khamaru. 2024. "Security of and by Generative AI Platforms." *arXiv Preprint arXiv:2410.13899*.

18. Wang, Fengwei, Hui Zhu, Xingdong Liu, Yandong Zheng, Hui Li, and Jiafeng Hua. 2024. "Achieving Federated Logistic Regression Training Towards Model Confidentiality with Semi-Honest TEE." *Information Sciences* 679: 121115.

19. Chen, Chunlu, Ji Liu, Haowen Tan, et al. 2025. "Trustworthy Federated Learning: Privacy, Security, and Beyond." *Knowledge and Information Systems* 67 (3): 2321–56.

20. Kang, Yan, Hanlin Gu, Xingxing Tang, et al. 2024. "Optimizing Privacy, Utility, and Efficiency in a Constrained Multi-Objective Federated Learning Framework." *ACM Transactions on Intelligent Systems and Technology* 15 (6): 1–33.

21. Yang, Haibo, Zhuqing Liu, Jia Liu, Chaosheng Dong, and Michinari Momma. 2023. “Federated Multi-Objective Learning.” *Advances in Neural Information Processing Systems* 36: 39602–25.

22. Liu, Qiqi, Yuping Yan, Péter Ligeti, and Yaochu Jin. 2023. “A Secure Federated Data-Driven Evolutionary Multi-Objective Optimization Algorithm.” *IEEE Transactions on Emerging Topics in Computational Intelligence* 8 (1): 191–205.

23. Chougule, A., V. Chamola, V. Hassija, P. Gupta, and F. R. Yu. 2023. “A Novel Framework for Traffic Congestion Management at Intersections Using Federated Learning and Vertical Partitioning.” *IEEE Transactions on Consumer Electronics* 70 (1): 1725–35.
24. Niknam, S., H. S. Dhillon, and J. H. Reed. 2020. “Federated Learning for Wireless Communications: Motivation, Opportunities, and Challenges.” *IEEE Communications Magazine* 58 (6): 46–51.
25. Warnat-Herresthal, S. et al. 2021. “Swarm Learning for Decentralized and Confidential Clinical Machine Learning.” *Nature*, ahead of print. https://doi.org/10.1038/s41586-021-03583-3.

26. Wahab, A. W. A., G. Rjoub, et al. 2022. “Federated Learning-Based Trustworthy Energy-Efficient System for Cold-Chain Monitoring in IoT.” *Computer Communications*, ahead of print. https://doi.org/10.1016/j.comcom.2022.04.016.

27. Wahab, A. W. A., G. Rjoub, et al. 2024. “Confidential and Trust-Based Federated Reinforcement Learning in Cyber–Physical Environments.” *Engineering Applications of Artificial Intelligence*, ahead of print. https://doi.org/10.1016/j.engappai.2024.107322.

28. Kanagavelu, R. et al. 2022. “CE-Fed: A Communication Efficient Collaborative Federated Learning Framework for IIoT.” *Future Generation Computer Systems*, ahead of print. https://doi.org/10.1016/j.future.2022.03.004.

29. Tang, T. et al. 2024. “A Privacy-Aware Federated Deep Learning Approach for Collaborative Autonomous Driving Systems.” *Information Sciences*, ahead of print. https://doi.org/10.1016/j.ins.2024.120519.

30. Deng, L. et al. 2025. “Secure and Privacy-Preserving Outsourced SVM Under Trusted Execution Environment.” *Knowledge-Based Systems*, ahead of print. https://doi.org/10.1016/j.knosys.2025.111002.

31. Hoang, D. T. et al. 2025. “Confidential Computing-Enabled Federated Learning for Biomedical Research Collaboration.” *Journal of Biomedical Informatics*, ahead of print.

32. Reese Pathak and Martin J Wainwright. Fedsplit: An algorithmic framework for fast federated optimization. *Advances in neural information processing systems*, 33:7057–7066, 2020.

33. Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

34. Bubeck, Sébastien et al. 2015. “Convex Optimization: Algorithms and Complexity.” *Foundations and Trends® in Machine Learning* 8 (3-4): 231–357.

35. Deng, Li. 2012. “The Mnist Database of Handwritten Digit Images for Machine Learning Research.” *IEEE Signal Processing Magazine* 29 (6): 141–42.

36. Krizhevsky, Alex. 2009. *Learning Multiple Layers of Features from Tiny Images*.

37. McMahan, Brendan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. “Communication-Efficient Learning of Deep Networks from Decentralized Data.” *Artificial Intelligence and Statistics*, 1273–82.

38. Yue, Gaofeng, Li Yan, Liuwang Kang, and Chao Shen. 2025. “AdapLDP-FL: An Adaptive Local Differential Privacy for Federated Learning.” *IEEE Transactions on Mobile Computing*.

39. Cui, Jinrong, Yinghua Li, Qiuli Zhang, Zhipeng He, and Shuping Zhao. 2024. “A Federated Learning Framework Using FedProx Algorithm for Privacy-Preserving Palmprint Recognition.” *Chinese Conference on Biometric Recognition*, 187–96.