

Federated Learning Framework with Differential Privacy over Homomorphic Vector Encryption for Data-Sensitive Applications

Manu Narula

manunarula1996@gmail.com

Delhi Technological University

Jasraj Meena

Jawaharlal Nehru University

Dinesh Vishwakarma

Delhi Technological University

Research Article

Keywords: Federated Learning, Differential Privacy, Security, Homomorphic Encryption, Data-Sensitive Applications

Posted Date: August 26th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10801503/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

Federated Learning Framework with Differential Privacy over Homomorphic Vector Encryption for Data-Sensitive Applications

Manu Narula^{1*}, Jasraj Meena^{2†} and Dinesh Vishwakarma^{1†}

^{1*}Delhi Technological University, Delhi, India.

²Jawaharlal Nehru University, Delhi, India.

*Corresponding author(s). E-mail(s): manunarula1996@gmail.com;

Contributing authors: jasrajengg@gmail.com;

dvishwakarma@gmail.com;

[†]These authors contributed equally to this work.

Abstract

Federated Learning (FL) has become a popular paradigm in recent years, attributed to its concept of insight sharing for ensuring data privacy. It has been found to be of extensive utility in applications that regularly deal with confidential data and has been widely used in the fields of healthcare and the Internet of Medical Things (IoMT). However, an equal amount of research is conducted to reverse-engineer the used datasets from the shared insights, and a native FL implementation alone is not suitable for these sensitive applications. We propose a lightweight secure FL framework incorporating both Differential Privacy and Homomorphic Encryption at the vector level to minimize the efficiency of reverse-engineering algorithms over the transmitted insights in IoMT and other resource-constrained networks. Through experiments on multiple image classification datasets and comparison with secure federated learning baselines, the proposed framework demonstrates the feasibility of combining differential privacy with encrypted aggregation while quantifying the resulting predictive and cryptographic overhead.

Keywords: Federated Learning, Differential Privacy, Security, Homomorphic Encryption, Data-Sensitive Applications

1 Introduction

Artificial Intelligence (AI) has gradually become an integral part of everyday life, influencing a wide range of sectors. Almost every industry is now utilizing AI in some form, whether in planning, operations, or decision-making processes—either explicitly or behind the scenes [1]. For effective functioning, AI systems typically require access to large datasets for training purposes. However, many times, this data could be highly sensitive and could potentially reveal personal or confidential user information when analyzed. Uncontrolled exposure of such data poses serious privacy risks, including the threat of cybercrime. Due to these concerns, Federated Learning (FL) has become a presumable privacy-preserving solution [2]. FL allows collaborative model training across decentralized devices and servers that helps maintain the local data. FL enables only model updates—such as parameters or gradients—to be shared across all nodes, rather than exchange of raw data, which safeguard individual datasets [3].

1.1 Federated Learning

McMahan et al. first introduced Federated Learning (FL) in 2016 [3], proposing an innovating paradigm where model parameters are exchanged in a distributed learning environment, rather than raw data. This idea significantly enhanced data security and gave data owners greater autonomy over what information is shared and to what extent. FL has since been adopted across a range of Internet of Things (IoT) domains, including the Internet of Medical Things (IoMT) [4], Industrial IoT (IIoT) [5, 6], and vehicular communication networks [7]. FL systems are broadly categorized into two types: Horizontal (or homogeneous) FL [8, 9] and Vertical (or heterogeneous) FL [10, 11].

Figure 1a illustrates a typical FL system with homogeneous data scenario, while Figure 1b shows the heterogeneous case. In both scenarios, a group of client devices (e.g., smartphones, tablets, laptops) communicates with a centralized cloud or edge server. Each device performs a local model training using its private dataset. In the heterogeneous configuration, a host entity may be responsible for managing the variations in the feature space across each client. After local training, updated model parameters are sent to the server, where they are aggregated and refine the global model. This iterative process continues for multiple rounds, with a randomly selected subset of devices participating each time to ensure data diversity and fairness throughout the training cycles.

1.2 Data-Sensitive Applications

A Data-Sensitive Application (DSA) refers to a system that routinely processes confidential information, which includes personal identifiers or digital footprints [12, 13], medical records [14], and financial details [15]. The rapid addition of Artificial Intelligence (AI) into everyday services has led to a significant rise in the number of such applications. While DSAs enhance the convenience and improve quality of life, they also add the risk of exposing private information. Federated Learning (FL) offers a promising solution by enabling AI capabilities in DSAs without directly sharing sensitive data, thereby reducing the likelihood of data breaches.

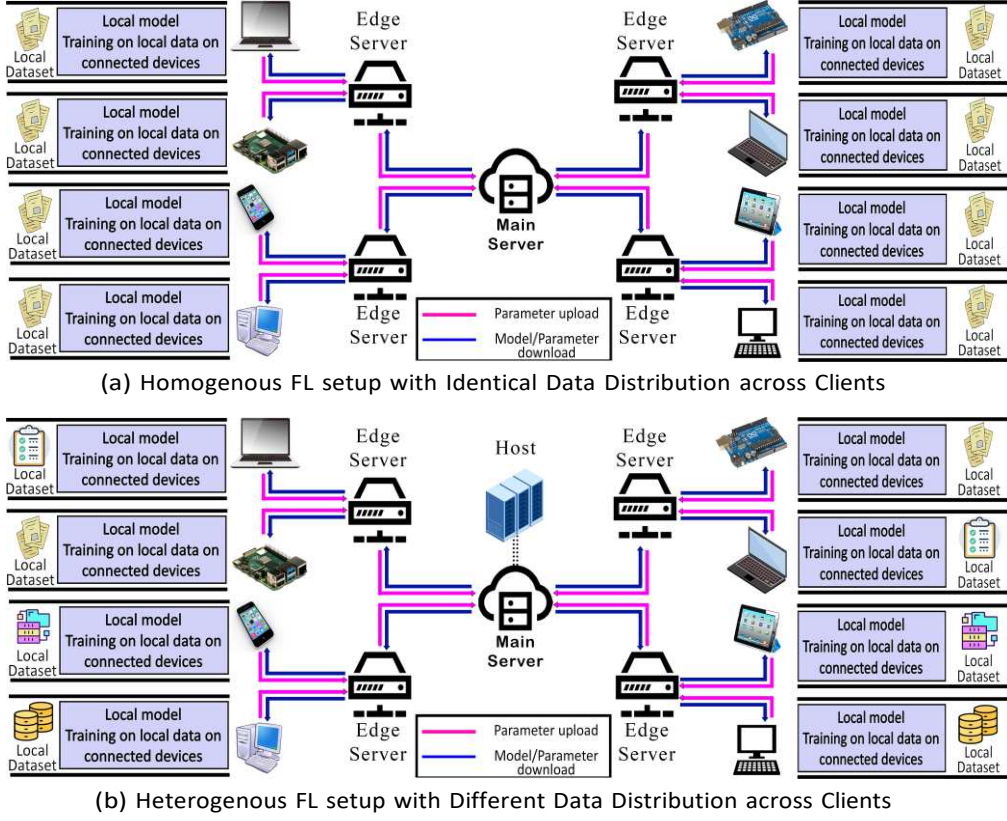


Fig. 1: Architecture of Generic FL.

A range of techniques have recently emerged that aim to enhance both the resource efficiency and security of Federated Learning (FL) systems. Among them, quantization is commonly used to reduce communication overhead and protect sensitive data, although often at the expense of model accuracy [4, 16]. To address this trade-off, certain algorithms have been developed to recover the performance lost due to quantization. For example, Li et al. [17] proposed a highly effective method that extends Incentive-Aware FL by incorporating workload-awareness, which improves the overall training accuracy under quantized settings. Building upon this, Narula et al. [18] introduced a hybrid framework that integrates quantization with workload-aware FL. This new approach successfully reduces the impact of quantization loss and also preserves data privacy.

1.3 Differential Privacy

Differential Privacy (DP) provides a formal privacy guarantee, by injecting carefully calibrated random noise it bounds the change in output distributions resulting from the inclusion or exclusion of any individual data record. This mechanism limits the

influence of any single participant, thus preventing adversaries from inferring individual membership during data analysis or model training. Local Differential Privacy (LDP) is a mechanism where prior to sharing any data or model updates, noise is applied to each individual client or edge device, this leads to high variance and degraded utility, particularly when local datasets are small. Distributed Differential Privacy (DDP) effectively mitigates this limitation by enabling collaborative privacy preservation across multiple distributed nodes, allowing the aggregation of perturbed information to achieve a superior privacy–utility trade-off while maintaining rigorous DP guarantees. Consequently, if accurate global model construction is required without compromising individual privacy, then DDP is well-suited for FL and edge intelligence scenarios [19].

1.4 Homomorphic Encryption

Homomorphic Encryption (HE) is a cryptographic technique which enables arithmetic operations to be performed directly on encrypted data, in such a way that decrypting the computed result yields the same output as if the operations had been performed on the plaintext. In federated learning, healthcare analytics, and edge AI systems, HE, allows clients, edge devices or medical institutions to securely transmit encrypted features, model updates, or inference requests to untrusted servers while ensuring that sensitive information—such as patient health records or proprietary data—remains confidential throughout the computation process. HE, support operations on encrypted gradients or parameters, thus preventing information leakage during collaborative model training and aggregation, making it particularly suitable for privacy-critical healthcare applications and distributed intelligence at the network edge. Although Fully Homomorphic Encryption (FHE) provides strong security guarantees, its substantial computational and communication overhead often limits a practical deployment; therefore, real-world systems typically rely on optimized or leveled HE schemes which balance privacy protection and efficiency in resource-constrained federated and edge learning environments [20].

1.5 Major Issues, Challenges, and Contributions

While reviewing the current literature and advanced strategies for secure FL, especially in the context of resource-constrained Data-Sensitive Applications (DSAs), we identified the following critical challenges:

- **Accuracy Trade-off:** Most of the existing security mechanisms for constrained environments rely on quantization, which inevitably leads to decreased model accuracy proportional to the degree of quantization applied, thereby limiting how aggressively it can be used.
- **Incomplete Recovery:** Although several auxiliary algorithms aim to mitigate the loss in performance, none of them have been able to fully restore the original accuracy. This highlights the need for alternative security solutions which ensure both privacy and high model performance in DSA environments.

We propose a hybrid Federated framework with Differential Privacy over Homomorphic Vector Encryption for Data-Sensitive Applications (Fed-DHVE) which will replace quantization as the base security measure and will provide a stable and private environment for DSA implementations.

We summarize the main contributions of this paper as follows:

- We introduce a hybrid framework that combines Differential Privacy (DP) with homomorphic encryption that reduces statistical leakage and protects the confidentiality of client update vectors in sensitive Federated Learning (FL) scenarios.
- To further reinforce security, the framework integrates Homomorphic Encryption (HE) during parameter transmission, ensuring confidentiality even in the presence of potential eavesdroppers.
- We further evaluate the framework on multiple classification datasets and report the resulting predictive utility and encryption/decryption overhead, while explicitly identifying the additional privacy-accounting and attack evaluations required for a complete security validation.

1.6 Assumptions for Scalable Real-World Implementation of Fed-DHVE

The experiment use a simulated federation of 1000 devices which represents a resource-constrained deployment. The setup is intended to examine the computational behavior of the framework under a relatively large client population while remaining within the available host resources. Present study does not provide sufficient experimental evidence to claim scalability beyond the simulated population; therefore, performance at substantially larger network sizes is treated as a future evaluation requirement.

The remainder of the paper is organized as follows: Section 2 reviews the relevant literature and prior work in the domain. Section 3 presents the proposed Fed-DHVE framework and its algorithms. Section 4 describes the experimental setup and results. Finally, Section 5 concludes the study and outlines possible directions for future research.

2 Related Work

2.1 Differential Privacy in Federated Learning

Studies have shown that inference attacks can still leak sensitive information via gradients and model updates in FL. By injecting calibrated noise to local or aggregated models, differentially private stochastic gradient descent was extended by authors in [21] to the federated setting, providing formal client-level or record-level privacy guarantees. Many studies have explored Local Differential Privacy, a technique in which noise is added directly at the client level, offering strong privacy protection. Special forms of DP, such as shuffled and distributed DP, improve the privacy–utility trade-off by leveraging secure shuffles or multi-client collaboration, making DP-based FL more practical for large-scale deployments cite HDVEr49.

2.2 Homomorphic Encryption in Federated Learning

Confidentiality of model updates during the aggregation and communication process is important, and it is addressed by a cryptographic solution, Homomorphic Encryption (HE), in FL. HE enables arithmetic operations on encrypted gradients and parameters, preventing an unreliable server or cloud service provider from accessing the plaintext updates. Initially, HE-based FL systems used additive HE schemes that focused on linear and logistic regression, which demonstrated the feasibility of encrypted aggregation; later, these ideas were extended to deep learning, which employed leveled or somewhat homomorphic encryption to reduce computational overhead. Even though HE offers strong confidentiality guarantees, its high computation and communication costs remain a major challenge, particularly in resource-constrained edge and mobile environments [22].

2.3 Hybrid Differential Privacy and Homomorphic Encryption Approaches

Recognizing that DP and HE address complementary threat models, recent research has investigated hybrid frameworks that combine both techniques in federated learning. In such systems, DP is typically applied at the client side to limit information leakage from individual updates, while HE is used to encrypt the noisy updates during transmission and aggregation, protecting against an honest-but-curious server. These hybrid approaches provide stronger end-to-end privacy guarantees than using either technique alone. However, balancing privacy strength, model utility, and system efficiency remains an open research problem, motivating ongoing work on adaptive noise injection, lightweight encryption schemes, and efficient secure aggregation protocols[23].

2.4 Threat Model and Security Objectives

We consider a federated setting in which the central server is honest-but-curious: it follows the prescribed training and aggregation protocol but may attempt to infer sensitive information from individual client updates. We additionally consider an external eavesdropper capable of intercepting client–server communications. Client datasets remain local and are never directly transmitted to the server or other clients. The framework is designed to protect individual update confidentiality during transmission and encrypted aggregation, while differential privacy limits statistical leakage from the perturbed learning signal. Fed-DHVE does not address arbitrary model-poisoning, backdoor, Sybil, or fully malicious-client attacks; these threats are outside the scope of the present study.

Accordingly, the security objectives are: (i) prevent an honest-but-curious server or network observer from directly accessing an individual client’s plaintext update through the encryption layer, and (ii) provide a quantifiable differential-privacy guarantee for the protected learning signal. These objectives are complementary: homomorphic encryption protects ciphertext confidentiality during communication and aggregation, whereas differential privacy provides a formal privacy guarantee for the released noisy update/model signal. The manuscript therefore distinguishes

confidentiality claims from DP claims and does not treat encryption alone as a differential-privacy mechanism.

3 Proposed Framework (Fed-DHVE)

This section describes in detail, our hybrid approach combining the security of differential privacy and homomorphic encryption. A detailed view of the complete framework, including all its sub-functions, is shown in Figure 2.

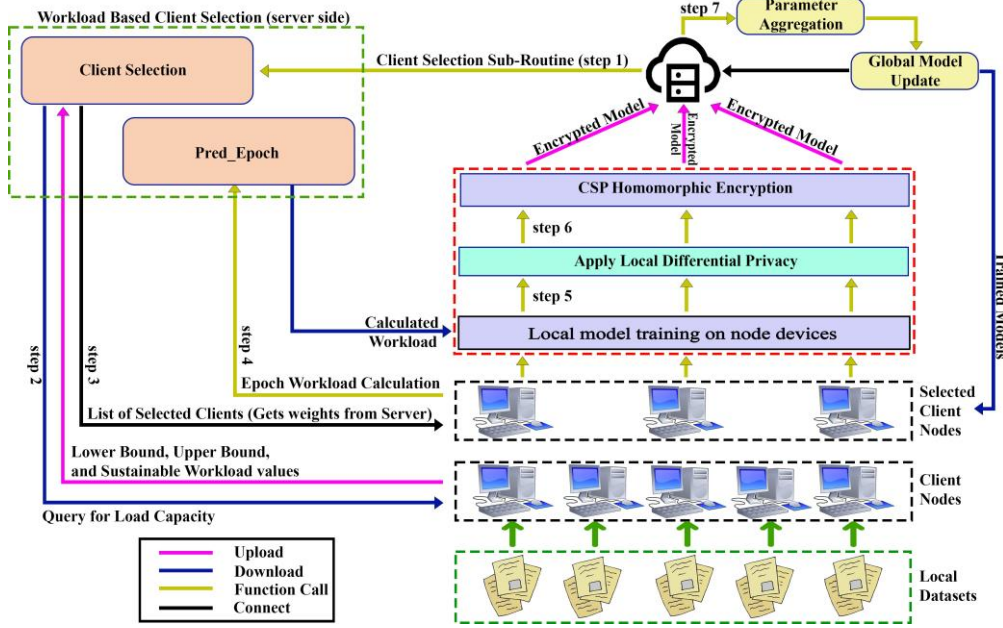


Fig. 2: Proposed Architecture of Fed-DHVE.

At the start of each round, the server selects a subset of clients according to workload capacity and past participation information. The current global model and the public parameters required by the CSP-based encryption scheme are sent to the selected clients. Each selected client performs local training on its private data, clips the resulting gradient, adds Gaussian noise for differential privacy, and encrypts the noisy update before transmission. The server aggregates the encrypted updates without accessing the individual plaintext updates, decrypts the aggregate, and updates the global model. This sequence is formalized in Algorithm 3.

3.1 Standard FL algorithm

Introduced by Google [8], Federated Learning (FL), enables collaborative model training without sharing raw data. In each training round a subset of clients, called the federation, are randomly selected and receive the current global model from the server.

Clients update the model locally using their private data and return the modified parameters. The server aggregates these updates to refine the global model. This process is repeated over multiple rounds with different client subsets. Furthermore, these enhancements will replace direct parameter sharing with gradient updates, ensuring that sensitive data remain local to the device.

3.2 Workload-Aware Client Selection

To reduce the likelihood of highly loaded or slow devices dominate the communication round, this proposed framework incorporates workload-aware client selection. For each client i , let c_i denote its normalized available computational capacity, p_i denote a normalized historical participation or training-performance score, and q_i denote a normalized workload or estimated completion-cost indicator. A generic selection score can be written as

$$S_i = \alpha c_i + \beta p_i - \gamma q_i, \quad (1)$$

where α , β , and γ are non-negative weighting coefficients. Eligible clients are ranked by the server according to S_i and selects the top K clients subject to the participation constraints of the round. The exact values of these coefficients and the workload measurements must be reported with the implementation used to reproduce the experiments.

3.3 Differential Privacy in Federated Learning

DP is a mathematical framework that protects individual data by ensuring that the inclusion or exclusion of a single record has minimal impact on the output of a computation. This is achieved by adding calibrated noise to queries or model updates, controlled by a privacy parameter ϵ . DP provides formal, quantifiable privacy guarantees, enabling secure analysis and learning from sensitive data without revealing personal information.

Algorithm 1, T denotes the total number of federated rounds, and S_t is the randomly selected subset of K clients participating in round t (line 1-3). Each client computes its local gradient $g_i = \nabla L(w_t; D_i)$ (line 5), clips it to a maximum norm C (line 6), and adds Gaussian noise $\tilde{g}_i = g_i + N(0, \sigma^2 C^2 I)$ to ensure differential privacy (line 7). The server aggregates the noisy updates to produce $w_{t+1} = w_t - \eta \frac{1}{K} \sum_{i \in S_t} \tilde{g}_i$ (lines 8-9). This process repeats until the global model converges while protecting client-level privacy. In a complete DP evaluation, the privacy guarantee should be reported as (ϵ, δ) -DP using the chosen clipping bound, noise multiplier, client sampling rate, number of communication rounds, and the privacy accountant used to obtain the final privacy budget.

3.4 Privacy Accounting and Reporting Requirements

The Gaussian mechanism in Algorithm 1 specifies the perturbation mechanism but does not, by itself, determine the final privacy budget over multiple communication rounds. For a complete privacy evaluation, the implementation must account for client sampling, the clipping bound C , the noise multiplier σ , the number of local updates, and the total number of rounds T . The final guarantee should be reported in the

Algorithm 1: Differentially Private Federated Learning

Input: Number of rounds T , initial global model w_0 , total clients N , subset size K , learning rate η , gradient clipping norm C , noise scale σ

Output: Trained global model w_T

```
1 for  $t = 1$  to  $T$  do
2   Randomly select a subset of clients  $S_t \subseteq \{1, 2, \dots, N\}$  of size  $K$ ;
3   Server sends  $w_t$  to each client  $i \in S_t$ ;
4   for each client  $i \in S_t$  in parallel do
5     Compute local gradient:  $g_i = \nabla L(w_t; D_i)$ ;
6     Clip gradient to bound sensitivity:  $g_i \leftarrow \frac{g_i}{\max(1, \|g_i\|_2 / C)}$ ;
7     Add Gaussian noise for differential privacy:  $\tilde{g}_i \leftarrow g_i + \mathbf{N}(0, \sigma^2 C^2 I)$ ;
8     Send noisy gradient  $\tilde{g}_i$  to the server;
9   Server aggregates updates:  $w_{t+1} \leftarrow w_t - \eta \cdot \frac{1}{K} \sum_{i \in S_t} \tilde{g}_i$ 
```

standard form (ϵ, δ) -DP using a stated privacy accountant, such as Rényi Differential Privacy (RDP) or an equivalent composition method. In particular, if $q = K/N$ is the client sampling rate, the accountant should compose the per-round Gaussian mechanism over all T rounds and convert the resulting RDP orders into an (ϵ, δ) guarantee. No numerical ϵ value is asserted here unless it is calculated from the exact parameters of the executed experiment. This distinction is important because reporting only C and σ is insufficient to establish an end-to-end privacy budget.

For reproducibility, the final experimental version of this work should therefore report $N, K, T, C, \sigma, \delta$, the number of local steps/epochs, the sampling strategy, and the accountant implementation together with the resulting ϵ .

3.5 Conjugate Search Problem - Homomorphic Encryption

The Conjugate Search Problem (CSP) is a computational hardness assumption from non-abelian group theory. Given a non-commutative group G and elements $a, b \in G$, the CSP asks for an element $x \in G$ such that

$$b = x^{-1}ax, \quad (2)$$

assuming such an x exists. The difficulty of solving this problem forms the security assumption used by the CSP-based encryption component considered in this work.

In the proposed framework, the CSP-based construction is used to support homomorphic processing of encrypted update vectors. Algorithm 2 describes the encryption and aggregation workflow. Each selected client computes its local gradient and encrypts the update before transmission. The server aggregates ciphertexts without decrypting individual client updates; only the aggregate is decrypted for the global-model update. In Fed-DHVE, differential privacy is applied before this encryption step, as specified in Algorithm 3.

For cryptographic reproducibility, a submission version must specify the concrete group representation, key-generation procedure, encoding of real-valued vectors, ciphertext representation, homomorphic aggregation operation, decryption procedure, security parameter, key size, and ciphertext expansion of the implemented CSP-based construction. The present manuscript uses E_{CSP} and D_{CSP} as abstractions for these operations and therefore limits its empirical claim to the reported encryption/decryption timing rather than asserting a new cryptographic security proof.

Algorithm 2: CSP-based Homomorphic Encryption in Federated Learning

Input: Number of rounds T , initial global model w_0 , total clients N , subset size K , learning rate η , CSP public parameters (G, a, b) , encryption E_{CSP} , decryption D_{CSP}

Output: Trained global model w_T

```

1 for  $t = 1$  to  $T$  do
2   Randomly select a subset of clients  $S_t \subseteq \{1, 2, \dots, N\}$  of size  $K$ ;
3   Server sends  $w_t$  and CSP public parameters  $(G, a, b)$  to clients in  $S_t$ ;
4   for each client  $i \in S_t$  in parallel do
5     Compute local gradient:  $g_i = \nabla L(w_t; D_i)$ ;
6     Encrypt gradient:  $\tilde{g}_i = E_{CSP}(g_i)$ ;
7     Send encrypted gradient  $\tilde{g}_i$  to server;
8   Aggregate encrypted gradients:  $\tilde{G}_t = \text{Aggregate}(\tilde{g}_i, i \in S_t)$ ;
9   Decrypt aggregated gradient:  $G_t = D_{CSP}(\tilde{G}_t)$ ;
10  Update global model:  $w_{t+1} = w_t - \eta \cdot G_t$ ;
```

3.6 Fed-DHVE

Combining the paradigm of differential privacy, homomorphic encryption, and workload-aware client selection, we propose a Federated Learning Framework with Differential Privacy over Homomorphic Vector Encryption (Fed-DHVE), shown in Algorithm 3.

T denotes the total number of federated rounds, and S_t is the subset of K clients selected for round t . Each client $i \in S_t$ computes its local gradient $g_i = \nabla L(w_t; D_i)$, clips it to a maximum norm C , and adds Gaussian noise for differential privacy: $\hat{g}_i = g_i + \mathbf{N}(0, \sigma^2 C^2 I)$. The noisy gradient is then encrypted using CSP-based homomorphic encryption $\tilde{g}_i = E_{CSP}(\hat{g}_i)$ before being sent to the server. The server performs homomorphic aggregation on the encrypted gradients $\tilde{G}_t = \text{Aggregate}(\tilde{g}_i, i \in S_t)$, and the aggregated result is decrypted $G_t = D_{CSP}(\tilde{G}_t)$ to update the global model $w_{t+1} = w_t - \eta \cdot G_t$. This hybrid approach ensures that individual gradients remain encrypted during aggregation and are differentially obfuscated, providing strong privacy protection for federated learning applications.

Algorithm 3: Fed-DHVE

Input: Number of rounds T , initial global model w_0 , total clients N , subset size K , learning rate η , gradient clipping norm C , DP noise scale σ , CSP public parameters (G, a, b) , encryption E_{CSP} , decryption D_{CSP}

Output: Trained global model w_T

```
1 for  $t = 1$  to  $T$  do
2   Select a subset of clients based on their past performance,
    $S_t \subseteq \{1, 2, \dots, N\}$  of size  $K$ ;
3   Server sends  $w_t$  and CSP public parameters  $(G, a, b)$  to clients in  $S_t$ ;
4   for each client  $i \in S_t$  in parallel do
5     Compute local gradient:  $g_i = \nabla L(w_t; D_i)$ ;
6     Clip gradient:  $g_i \leftarrow \frac{g_i}{\max(1, \|g_i\|_2 / C)}$ ;
7     Add DP noise:  $\hat{g}_i = g_i + \mathbf{N}(0, \sigma^2 C^2 I)$ ;
8     Encrypt noisy gradient:  $\tilde{g}_i = E_{CSP}(\hat{g}_i)$ ;
9     Send encrypted noisy gradient  $\tilde{g}_i$  to server;
10  Aggregate encrypted updates:  $\tilde{G}_t = \text{Aggregate}(\tilde{g}_i, i \in S_t)$ ;
11  Decrypt aggregated gradient:  $G_t = D_{CSP}(\tilde{G}_t)$ ;
12  Update global model:  $w_{t+1} = w_t - \eta \cdot G_t$ ;
```

4 Experimental Results and Analysis

This section describes the baseline methods, experimental environment, datasets, and the measured predictive and cryptographic performance of Fed-DHVE.

4.1 Baseline Models

We compare Fed-DHVE with Standard FL, Fed-MSQE with 5-bit parameter quantization, VerifyNet, Turbo-Aggregate, and the earlier Fed-DLC framework. Standard FL provides the non-secure reference point. Fed-MSQE represents a quantization-based approach, while VerifyNet and Turbo-Aggregate provide alternative secure or efficient aggregation mechanisms. Fed-DLC is included as a lightweight encryption baseline for resource-constrained federated learning. The comparison focuses on predictive performance and the reported encryption/decryption overhead.

4.2 Experimental Setup

Our simulations are carried out on a system running Ubuntu 20.04 LTS equipped with an Intel (R) Xeon(R) Silver 4114 CPU @ 2.20GHz, 128GB DDR4 RAM, and two NVIDIA Titan RTX GPUs with 24GB VRAM each. We implemented our models in Python 3.10.13 with CUDA 11.8 [24], TensorFlow 2.10 [25], and NumPy 1.24 [26] over Jupyter Notebook in a Conda virtual environment. A federation of 1000 simulated devices is used simultaneously to represent a resource-constrained environment on the host machine. The present manuscript does not claim experimental validation beyond this simulated population; larger-scale behavior is reserved for future evaluation.

4.3 Dataset Description

We evaluate Fed-DHVE against the selected baselines over four openly available image classification datasets with different image modalities and dataset sizes. The dataset characteristics used in the experiments are summarized in Table ??.

4.4 Federated Training Protocol

The reported experiments use the same datasets and evaluation protocol as the supplied experimental results. To make the study reproducible, the final submission should explicitly report the number of communication rounds, number of selected clients per round, local epochs or local optimization steps, batch size, optimizer, learning rate, gradient clipping bound, DP noise multiplier, client partitioning strategy, and random seeds. These parameters are directly affecting both convergence and the cumulative privacy budget. Because the supplied manuscript does not provide all values, this section deliberately does not invent missing hyperparameters.

4.5 Results

Table 2 summarizes the reported predictive performance of Fed-DHVE and the comparison methods across the four datasets. Fed-DHVE obtains accuracies of 0.9783, 0.9397, 0.9681, and 0.9492 on MNIST, Fashion-MNIST, EMNIST, and CIFAR-10, respectively. The results show that adding privacy protection introduces a utility cost relative to Standard FL, which is expected when Gaussian noise is applied to client updates. The appropriate interpretation is therefore a privacy–utility trade-off rather than an expectation of identical accuracy to Standard FL.

Table 2: Performance comparison of Fed-DHVE with secure federated learning baselines.

Method	MNIST	Fashion-MNIST	EMNIST	CIFAR-10	Security mechanism	Encryption time	Decryption time
Standard FL	0.9843	0.9537	0.9761	0.9552	None	–	–
Fed-MSQE (5-bit) [31]	0.9537	0.9016	0.9750	0.8890	Quantization	1.3 s	–
VerifyNet [32]	0.9791	0.9580	0.9785	0.9476	Authentication	0.8 s	–
Turbo-Aggregate [33]	0.9823	0.9525	0.9770	0.9554	Secure aggregation	–	–
Fed-DLC	0.9850	0.9526	0.9764	0.9554	Encryption	1.3 s	1.3 s
Fed-DHVE	0.9783	0.9397	0.9681	0.9492	DP + HE	2.04 s	0.72 s

Across the four datasets, Fed-DHVE trades a reduction in predictive accuracy for stronger privacy protection. The reported Fed-DHVE accuracy is below Standard FL on all four datasets, consistent with the expected utility cost of differential privacy. At the same time, the framework combines two complementary protection mechanisms: DP limits statistical leakage from the learning signal, while HE prevents direct access to individual encrypted updates during aggregation. The measured average encryption and decryption times are 2.04 s and 0.72 s, respectively.

The present experimental evidence reports aggregate predictive and timing metrics. Accuracy and cryptographic timing alone do not establish resistance to privacy attacks. A stronger validation should therefore include an explicit (ϵ, δ) privacy budget and direct inference or gradient-reconstruction experiments. These are identified as

validation requirements rather than claimed as completed experiments in the present study.

4.6 Ablation and Security Evaluation Protocol

A complete journal evaluation should isolate the contribution of each component through at least the following configurations: (i) Standard FL, (ii) FL with DP only, (iii) FL with HE only, (iv) FL with DP and HE, and (v) the complete Fed-DHVE configuration including workload-aware client selection. The configurations should be compared using accuracy, precision, recall, F1-score, convergence behavior, communication cost, training time, and encryption/decryption time. In addition, gradient-inversion or reconstruction attacks and membership-inference attacks should be evaluated where applicable. The present supplied results do not contain these ablation or attack measurements, so numerical values are intentionally not fabricated here.

5 Conclusion and Future Scope

This paper presented Fed-DHVE, a federated learning framework that combines client-side differential privacy with CSP-based homomorphic encryption and workload-aware client selection for data-sensitive applications. The framework clips client gradients, adds Gaussian noise, encrypts the resulting update vectors, performs aggregation over ciphertexts, and updates the global model from the decrypted aggregate. This design addresses two complementary privacy concerns: statistical leakage from learning updates and direct exposure of individual updates during communication and aggregation.

The reported experiments over MNIST, Fashion-MNIST, EMNIST, and CIFAR-10 show that Fed-DHVE achieves accuracies of 0.9783, 0.9397, 0.9681, and 0.9492, respectively. The measured average encryption and decryption times are 2.04 s and 0.72 s. Compared with Standard FL, the reduction in predictive performance reflects the expected privacy–utility trade-off introduced by differential privacy. The results therefore support the feasibility of combining DP and encrypted aggregation, but they do not by themselves establish resistance to specific inference attacks.

The next stage of evaluation should quantify the end-to-end privacy budget using an explicit (ϵ, δ) accountant and the exact training parameters, followed by direct privacy-attack experiments such as gradient inversion and membership inference. The CSP-based homomorphic encryption component should also be documented and evaluated in greater cryptographic detail, including the concrete construction, security parameter, key sizes, ciphertext expansion, and computational complexity. Finally, the workload-aware client-selection mechanism should be evaluated through an ablation study and larger heterogeneous client populations should be considered to establish practical scalability.

Competing Interest. The author(s) declare that they have no conflict of interest regarding the publication of this paper.

References

- [1] Ahmed, I., Jeon, G., Piccialli, F.: From artificial intelligence to explainable artificial intelligence in industry 4.0: a survey on what, how, and where. *IEEE Transactions on Industrial Informatics* **18**(8), 5031–5042 (2022) <https://doi.org/10.1109/TII.2022.3146552> . <https://doi.org/10.1109/TII.2022.3146552>
- [2] Narula, M., Meena, J., Vishwakarma, D.K.: A comprehensive review on federated learning for data-sensitive application: Open issues & challenges. *Engineering Applications of Artificial Intelligence* **133**, 108128 (2024) <https://doi.org/10.1016/j.engappai.2024.108128> . <https://doi.org/10.1016/j.engappai.2024.108128>
- [3] McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: *Artificial Intelligence and Statistics*, pp. 1273–1282 (2017). PMLR
- [4] Zhiang, X., Yijia, G., Chinmay, C., Qiaozhi, H., Shengbo, C., Keping, Y.: A simple federated learning-based scheme for security enhancement over internet of medical things. *IEEE Journal of Biomedical and Health Informatics*. **10** (2023) <https://doi.org/10.1109/JBHI.2022.3187471> . <https://doi.org/10.1109/JBHI.2022.3187471>
- [5] Wang, X., Garg, S., Lin, H., Hu, J., Kaddoum, G., Piran, M.J., Hossain, M.S.: Toward accurate anomaly detection in industrial internet of things using hierarchical federated learning. *IEEE Internet of Things Journal* **9**(10), 7110–7119 (2021) <https://doi.org/10.1109/JIOT.2021.3074382> . <https://doi.org/10.1109/JIOT.2021.3074382>
- [6] Wang, X., Hu, J., Lin, H., Garg, S., Kaddoum, G., Piran, M.J., Hossain, M.S.: Qos and privacy-aware routing for 5g-enabled industrial internet of things: A federated reinforcement learning approach. *IEEE Transactions on Industrial Informatics* **18**(6), 4189–4197 (2021) <https://doi.org/10.1109/TII.2021.3124848> . <https://doi.org/10.1109/TII.2021.3124848>
- [7] Khan, I.A., Moustafa, N., Pi, D., Haider, W., Li, B., Jolfaei, A.: An enhanced multi-stage deep learning framework for detecting malicious activities from autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems* **23**(12), 25469–25478 (2021) <https://doi.org/10.1109/TITS.2021.3105834> . <https://doi.org/10.1109/TITS.2021.3105834>
- [8] Liu, D., Bai, L., Yu, T., Zhang, A.: Towards method of horizontal federated learning: A survey. In: *2022 8th International Conference on Big Data and Information Analytics (bigdia)*, pp. 259–266 (2022). <https://doi.org/10.1109/BigDIA56350.2022.9874186> . IEEE. <https://doi.org/10.1109/BigDIA56350.2022.9874186>
- [9] Huang, W., Li, T., Wang, D., Du, S., Zhang, J., Huang, T.: Fairness and accuracy in horizontal federated learning. *Information Sciences* **589**, 170–185 (2022) <https://doi.org/10.1016/j.ins.2022.05.045>

[//doi.org/10.1016/j.ins.2021.12.102](https://doi.org/10.1016/j.ins.2021.12.102) . <https://doi.org/10.1016/j.ins.2021.12.102>

- [10] Wei, K., Li, J., Ma, C., Ding, M., Wei, S., Wu, F., Chen, G., Ranbaduge, T.: Vertical federated learning: Challenges, methodologies and experiments. arXiv preprint arXiv:2202.04309 (2022)
- [11] Yang, K., Song, Z., Zhang, Y., Zhou, Y., Sun, X., Wang, J.: Model optimization method based on vertical federated learning. In: 2021 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–5 (2021). <https://doi.org/10.1109/ISCAS51556.2021.9401521> . IEEE. <https://doi.org/10.1109/ISCAS51556.2021.9401521>
- [12] Lin, H., Kaur, K., Wang, X., Kaddoum, G., Hu, J., Hassan, M.M.: Privacy-aware access control in iot-enabled healthcare: A federated deep learning approach. IEEE Internet of Things Journal **10**(4), 2893–2902 (2021) <https://doi.org/10.1109/JIOT.2021.3112686> . <https://doi.org/10.1109/JIOT.2021.3112686>
- [13] Zhou, X., Liang, W., Ma, J., Yan, Z., Kevin, I., Wang, K.: 2d federated learning for personalized human activity recognition in cyber-physical systems. IEEE Transactions on Network Science and Engineering **9**(6), 3934–3944 (2022) <https://doi.org/10.1109/TNSE.2022.3144699> . <https://doi.org/10.1109/TNSE.2022.3144699>
- [14] Sabri, B., Fethi, J., Neethu, V., Shabir, M., Haithum, E., Michel, M.: Federated uncertainty-aware learning for distributed hospital ehr data. <http://arxiv.org/abs/1910.12191> (2019)
- [15] Schreyer, M., Sattarov, T., Borth, D.: Federated and privacy-preserving learning of accounting data in financial statement audits. In: Proceedings of the Third ACM International Conference on AI in Finance, pp. 105–113 (2022). <https://doi.org/10.1145/3533271.3561674> . <https://doi.org/10.1145/3533271.3561674>
- [16] Kerkouche, R., Acs, G., Castelluccia, C., Genevès, P.: Privacy-preserving and bandwidth-efficient federated learning: An application to in-hospital mortality prediction. In: Proceedings of the Conference on Health, Inference, and Learning, pp. 25–35 (2021). <https://doi.org/10.1145/3450439.3451859> . <https://doi.org/10.1145/3450439.3451859>
- [17] Li, L., Liu, D., Duan, M., Zhang, Y., Ren, A., Chen, X., Tan, Y., Wang, C.: Federated learning with workload-aware client scheduling in heterogeneous systems. Neural Networks **154**, 560–573 (2022) <https://doi.org/10.1016/j.neunet.2022.07.030> . <https://doi.org/10.1016/j.neunet.2022.07.030>
- [18] Narula, M., Meena, J., Vishwakarma, D.K.: Federated workload-aware quantized framework for secure learning in data-sensitive applications. Future Generation Computer Systems **168**, 107772 (2025) <https://doi.org/10.1016/j.future.2025.107772> . <https://doi.org/10.1016/j.future.2025.107772>

- [19] Cheu, A., Smith, A., Ullman, J., Zeber, D.: Distributed differential privacy via shuffling. In: Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I, pp. 375–403. Springer, ??? (2019). https://doi.org/10.1007/978-3-030-17653-2_13 . https://doi.org/10.1007/978-3-030-17653-2_13
- [20] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998) <https://doi.org/10.1109/5.726791> . <https://doi.org/10.1109/5.726791>
- [21] Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308–318 (2016) <https://doi.org/10.1145/2976749.2978318> . <https://doi.org/10.1145/2976749.2978318>
- [22] Phong, L.T., Aono, Y., Hayashi, T., Wang, L., Moriai, S.: Privacy-preserving deep learning via additive homomorphic encryption. IEEE Transactions on Information Forensics and Security **13**(5), 1333–1345 (2018) <https://doi.org/10.1109/TIFS.2017.2787987> . <https://doi.org/10.1109/TIFS.2017.2787987>
- [23] Truex, S., Liu, L., Gursoy, M.E., Yu, L., Wei, W.: Privacy-preserving federated learning with differential privacy and homomorphic encryption. Proceedings of the 28th ACM International Conference on Information and Knowledge Management, 2455–2458 (2019) <https://doi.org/10.1145/3357384.3358120> . <https://doi.org/10.1145/3357384.3358120>
- [24] Dehal, R.S., Munjal, C., Ansari, A.A., Kushwaha, A.S.: Gpu computing revolution: Cuda. In: 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), pp. 197–201 (2018). <https://doi.org/10.1109/ICACCCN.2018.8748495> . <https://doi.org/10.1109/ICACCCN.2018.8748495>
- [25] Imambi, S., Prakash, K.B., Kanagachidambaresan, G.R.: Pytorch. In: Programming with TensorFlow: Solution for Edge Computing Applications, pp. 87–104 (2021). https://doi.org/10.1007/978-3-030-57077-4_10 . https://doi.org/10.1007/978-3-030-57077-4_10
- [26] Oliphant, T.E., *et al.*: Guide to Numpy vol. 1. Trelgol Publishing USA, ??? (2006). <http://www.trelgol.com>
- [27] Ziller, A., Trask, A., Lopardo, A., Szymkow, B., Wagner, B., Bluemke, E., Nounahon, J.-M., Passerat-Palmbach, J., Prakash, K., Rose, N., *et al.*: Pysyft: A library for easy federated learning. Federated learning systems: Towards next-generation AI, 111–139 (2021) https://doi.org/10.1007/978-3-030-70604-3_5 . https://doi.org/10.1007/978-3-030-70604-3_5

- [28] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86**(11), 2278–2324 (1998) <https://doi.org/10.1109/5.726791> . <https://doi.org/10.1109/5.726791>
- [29] Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017). <http://arxiv.org/abs/1708.07747>
- [30] EMNIST (Extended MNIST) — Kaggle. Accessed on June 07, 2023. <https://www.kaggle.com/datasets/crawford/emnist>
- [31] Xu, Z., Guo, Y., Chakraborty, C., Hua, Q., Chen, S., Yu, K.: A simple federated learning-based scheme for security enhancement over internet of medical things. *IEEE Journal of Biomedical and Health Informatics* (2022) <https://doi.org/10.1109/JBHI.2022.3187471> . <https://doi.org/10.1109/JBHI.2022.3187471>
- [32] Xu, G., Li, H., Liu, S., Yang, K., Lin, X.: Verifynet: Secure and verifiable federated learning. *IEEE Transactions on Information Forensics and Security* **15**, 911–926 (2019)
- [33] So, J., Güler, B., Avestimehr, A.S.: Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *IEEE Journal on Selected Areas in Information Theory* **2**(1), 479–489 (2021)