Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Privacy Preservation in Data-Hungry Deep Learning: A Comprehensive Review of Attacks and Techniques

Omar Nassim Adel Benyamina [1*], Zohra Slama [2]
[1,2]*Department of Computer Science, Djillali Liabes University, Sidi bel abbes, Algeria*
[1,2]*EEDIS Laboratory, Sidi bel abbes, Algeria*
** Corresponding Author: benyamina.adel98@gmail.com, adel.benyamina@univ-sba.dz*

ARTICLE INFO

Received: 26 Aug 2025

Revised: 09 Oct 2025

Accepted: 02 Nov 2025

ABSTRACT

Deep learning (DL) has achieved remarkable success across various domains, including healthcare, finance, and natural language processing; however, its reliance on sensitive data poses significant privacy risks. Privacy-preserving deep learning (PPDL) has therefore emerged as a critical research direction, integrating cryptographic techniques, statistical privacy mechanisms, and distributed training paradigms. This survey reviews state-of-the-art privacy-preserving deep learning (PPDL) techniques centered on homomorphic encryption (HE), secure multi-party computation (SMPC) (and hybrid protocols), differential privacy (DP), and secure enclaves (SE/TEEs). We also position federated learning (FL) as an orchestration paradigm that composes these techniques at scale. We systematically analyze their efficacy, privacy, and efficiency trade-offs, and map common attack vectors—such as reconstruction, inversion, membership inference, poisoning, and hardware-level side channels—to representative defenses. Bibliometric analysis using VOSviewer further highlights the thematic structure of the field, with strong clusters around cryptography, differential privacy, and system-level optimization. Our findings reveal that no single paradigm suffices in practice: while HE and SMPC provide strong confidentiality, they incur high costs; DP enables formal guarantees at the expense of accuracy; and FL reduces raw-data exposure but introduces novel vulnerabilities. We conclude that hybrid, layered strategies combining DP, cryptography, and robust aggregation are the most promising path toward scalable, trustworthy PPDL for real-world deployment.

Keywords: Privacy-preserving deep learning, Homomorphic encryption, Secure multi-party computation, Differential privacy, Federated learning, Adversarial attacks, Bibliometric analysis.

INTRODUCTION

The massive collection of data in recent years has raised significant challenges for privacy preservation. On the one hand, privacy is increasingly recognized as a fundamental right of end users and customers; on the other hand, it poses a constraint on the utilization of data for analytics and artificial intelligence (AI). This tension is particularly acute in machine learning (ML) and deep learning (DL), which require large volumes of data for training. Protecting this data from leakage or misuse has become a central concern. In parallel, the computational and communication costs of training large-scale models have surged, further complicating the design of privacy-preserving systems. Users fear that sensitive information could be exposed, while companies are concerned about protecting the confidentiality of their proprietary DL models. If compromised, adversaries may impersonate customers or reverse-engineer model behaviors, undermining trust. Classical privacy-preserving approaches have emerged to mitigate these risks, including anonymization, cryptographic methods, and differential privacy (DP).

To contextualize privacy-preserving deep learning (PPDL), it is essential to first map out the types of attacks that exploit DL models and data. Notable threats include re-identification, reconstruction, model inversion, and membership inference [1]. A clear understanding of these attack vectors enables systematic evaluation of PPDL

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

techniques, guides the management of data leakage risks, and informs the design of mechanisms that safeguard data, models, and results simultaneously.

More recently, the evolution of PPDL has shifted toward distributed and generative paradigms. Federated learning (FL) enables collaborative model training without direct data sharing, but introduces novel vulnerabilities such as model poisoning and gradient leakage [2]. At the same time, large language models (LLMs) and other generative architectures pose new privacy risks, notably unintended memorization of training data and inference-based extraction attacks [3]. Addressing these challenges requires integrating privacy-preserving techniques into both distributed training frameworks and modern generative models.

Article Outline. The remainder of this article is organized as follows

Section II introduces and classifies traditional privacy-preserving techniques, presenting a taxonomy and a comparative view of different neural network architectures. Section III reviews the main classes of attacks that threaten privacy in ML/DL, highlighting their targets and required access assumptions. Section IV details the methodology adopted for this survey, including research questions, analysis attributes, and evaluation criteria; it also incorporates a bibliometric visualization generated with VOSviewer. Section V presents the survey results, structured around the three global metrics: efficacy, privacy, and efficiency. Section VI discusses the most influential works, compares their strengths and weaknesses, and provides answers to the research questions. Finally, Section VII concludes by summarizing the key findings and outlining promising directions for future research.

PRIVACY-PRESERVING TECHNIQUES

We classify classical privacy-preserving methods into four categories **Figure 1**: group-based anonymity, cryptographic techniques (e.g., HE, SMPC), differential privacy (DP), and secure enclaves (SE/TEEs). Although homomorphic encryption [4], functional encryption (proposed by [5], formalized by [6]), and secure multi-party computation techniques make it possible to perform computations on encrypted data without revealing the original plaintext, we need to preserve the confidentiality of sensitive personal data, such as medical and health data. The first step in preserving this confidentiality is to use data anonymization techniques to mask this sensitive personal data.

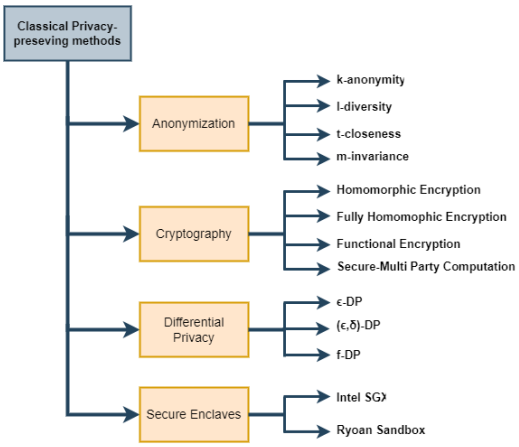

Figure 1. taxonomy of classical pp methods

Both FE and HE enable computation over encrypted inputs, but they differ in what the evaluator and the decryptor learn. In *functional encryption* (FE), a holder of a function-specific secret key sk_f can decrypt a ciphertext $ct = Enc(m)$ to obtain the plaintext value $f(m)$—and nothing else about m. In *homomorphic encryption* (HE), the evaluator transforms $Enc(m)$) into $Enc(f(m))$ using public evaluation keys; only the data owner with the decryption key can later recover $f(m)$. FE typically requires a trusted authority to issue function keys (one per authorized function), whereas HE does not require such a function-key issuer for evaluation.

Secure Multi-party Computation is a cryptographic protocol that distributes computing among several parties without allowing any of them to access the data of others. In 1986, [7] introduced two-party secure computing and the Garbled Circuit (GC) which requires a constant number of communication rounds. Compared to HE and FE

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

schemes, in the Secure-MPC, the parties jointly calculate a function on their inputs using a protocol, instead of a single party. Information concerning the parties' confidentiality shall not be exposed throughout the process.

Differential privacy is a mathematical framework [8] proposed by [9] as a strong standard for ensuring data privacy. Sometimes, details of this data are stored by AI models when they are trained and they may "leak" them later. DP measures this leakage and reduces the possibility of it happening by adding noise to data. Extensions include federated DP, where noise is added in distributed settings to enhance scalability [10].

Secure enclaves proposed by [11] (also called trusted execution environments, TEEs), refer to a computing environment that isolates code and data from the operating system. They use hardware isolation or isolate an entire virtual machine by placing the hypervisor in the Trusted Computing Base (TCB).

A- Deep Learning for Privacy-Preserving

After learning about the traditional approach, we will understand how its evolution leads to a new method called privacy-preserving deep learning. It combines the traditional approach with the emerging field of deep learning. The deep learning prediction algorithm, also known as a model, is designed as a layered architecture with an input layer and an output layer. There may be one or more hidden layers between the input and output layers; the more hidden layers there are, the more accurate the DL model is. However, one should beware of the problem of overfitting; to learn so much that we cannot generalize.

Different neural architectures are commonly applied to PPDL tasks. **Table 1** compares representative deep neural networks (DNN) [12], convolutional neural networks (CNN), recurrent neural networks (RNN), generative adversarial networks (GAN), and the more recent Transformers and Large Language Models (LLMs), highlighting their typical data domains and structural characteristics. These architectures have been widely adopted for privacy-preserving applications such as medical imaging, biometric authentication, language modeling, and multimodal tasks.

It is also worth noting that some studies adopt a *privacy-aware* rather than formally privacy-preserving approach. For instance, Benyamina and Slama [13] highlight the role of **feature selection** in limiting the exposure of sensitive attributes. By selecting only the most informative features, their model achieved 85.76% accuracy on the UCI Adult dataset while reducing the risk of revealing private information such as gender or race. However, such heuristic strategies cannot replace formal PPDL guarantees, as they do not provide protection against reconstruction or inference attacks.

TABLE 1. COMPARATIVE TABLE OF DIFFERENT NEURAL NETWORKS

	DNN	CNN	RNN	GAN	LLMS
Type of data	Tabular Textual	Image Video	Sequential (time, text, audio)	Image Video Audio	Code Vision Speach
Parameters sharing	No	Yes	Yes	Yes	Yes
Fixed length data	Yes	Yes	No	Yes	Flexible
Recurrent connections	No	No	Yes	No	no
Spatial relationships	No	Yes	Limited Short-term	Yes	yes

PRIVACY ISSUES IN MACHINE LEARNING

We first outline the main categories of attacks that threaten the privacy of personal or sensitive data in ML/DL systems, and the vulnerability points where they occur.

Conceptually, three components are involved: (i) the ***raw data*** *holder*, which provides inputs; (ii) the

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

computation part, which executes the ML/DL tasks; and (iii) the ***result part***, which outputs predictions or stores trained models. If these three parts operate as a single unit, confidentiality is implicitly preserved. However, once separated, multiple attack vectors emerge.

TABLE 2. EXISTING THREATS

Type of Threats	Limitations	Access
Reconstruction Attacks	Do not use ML models that store explicit feature vectors (e.g., SVM, kNN).	White-box
Model Inversion Attacks	Restrict outputs to black-box access; Round confidence values or release only predicted lables.	White-box Black-box
Membership Inference Attacks	Limit outputs to class lables; DP can mitigate the attack.	Black-box

Previous works have highlighted three dominant classes of threats. ***Reconstruction attacks*** [14-17] attempt to extract raw data from feature vectors or gradients, requiring access to internal representations. **Model inversion attacks** [15] infer representative inputs from model outputs, especially when confidence scores are exposed. **Membership inference attacks** [18, 19] determine whether a given record was included in the training set, typically through repeated model queries.

Table 2 provides a concise taxonomy of these threats, their typical limitations, and the type of access (white-box or black-box) required by the adversary. This high-level view motivates the need for privacy-preserving deep learning (PPDL) techniques. A more detailed mapping of attacks to defenses, with representative citations, is provided later in **Table 8** (**Section 5.F**).

METHODOLOGY

This section outlines the methodology adopted for this survey, describing how research on privacy-preserving deep learning (PPDL) was collected, analyzed, and synthesized. The main objective is to identify existing PPDL techniques that enable secure training of deep learning models on sensitive data. Protecting data alone is insufficient, since trained models may leak information, allowing adversaries to reconstruct similar or even identical records if the input distribution is known. Therefore, ensuring the confidentiality of both ***data*** and ***models*** is essential and complementary.

This survey further aims to (i) characterize the most frequent privacy attacks, (ii) construct a taxonomy of PPDL techniques, and (iii) identify which approaches preserve privacy while maintaining utility.

A- Research Questions

To guide the review, we formulated the following research questions:

- **RQ.1** What attacks can compromise the privacy of private data in machine/deep learning?
- **RQ.2** Can we quantify and control the rate of data leakage?
- **RQ.3** Which privacy-preserving methods are most promising to reduce model and data vulnerability to attacks?
- **RQ.4** How have PPDL techniques evolved, particularly for generative models and distributed networks?

B- Systematic Corpus Selection (PRISMA 2020)

We followed the PRISMA 2020 [69] recommendations to identify, screen, and include studies on privacy-preserving deep learning (PPDL). The aim was to gather peer-reviewed works using formal methods (HE, SMPC, DP, secure enclaves) and, in a second step, add recent papers (2023–2025) discovered through manual/citation searches.

1. ***Sources and time windows.:*** For the *main search* (2013–2022) we queried four databases: SpringerLink, Wiley Online Library, ACM Digital Library, and ScienceDirect. In the PRISMA figure, the small

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

bucket "Others (IEEE/arXiv..)" is listed to reflect records obtained from commonly used scholarly registers (IEEE Xplore) and preprint registers (arXiv). For the *update search* (2023–2025), we used manual website checks and citation chaining to capture very recent, high-impact studies.

2. **Search terms.** We used combinations of:

(Deep OR Machine OR Federated OR Distributed) AND Learning AND ("Privacy preserving" OR Private OR Privacy OR "Data privacy") in the *advanced search interfaces* of the previously listed databases.

3. **Screening and eligibility.** Screening was performed in two consecutive phases: a preliminary metadata filter followed by full-text eligibility assessment. During the ***screening phase***, duplicates were removed and basic quality filters were applied. Records were excluded if they lacked full-text access, were written in a language other than English, or—within the 2013–2022 tranche—had zero citations, ensuring the selection of peer-recognized and mature studies. Titles and abstracts were then screened to remove works outside the Computer Science domain or not directly addressing privacy-preserving deep learning (PPDL). During the **eligibility phase**, full-text articles were examined in detail against the defined inclusion and exclusion criteria.

The inclusion criteria required that a paper:

- Proposes or implements a privacy-preserving method "*applied to deep learning"*;
- Evaluates the method on real or benchmark datasets;
- Is published in a peer-reviewed journal or conference, or indexed in a reputable research register (IEEE Xplore, arXiv, SpringerLink, ACM, Wiley);
- Clearly reports methodological and empirical contributions in English.

Exclusion criteria eliminated:

- Duplicated or out-of-scope works (outside 2013–2022 for the main search);
- Articles not in Computer Science or not research-oriented (editorials, reviews, abstracts);
- Non-English publications;
- Survey and systematic literature review (SLR) papers;
- Works where privacy was not the main objective;
- Studies classified as *privacy-preserving methods based on deep learning (PPMBDL)*, i.e., papers that use DL itself as a privacy mechanism (e.g., autoencoders, GAN-based anonymization) rather than applying privacy mechanisms "to" DL models.

For the 2023–2025 update, the same eligibility rules were applied except for the citation requirement, since recent papers may not yet have accrued citations. Articles published prior to 2023 were excluded as out of range.

4. **Included studies.** From the main search, *25* primary studies (2013–2022) met all criteria. The update search added *11* recent papers (2023–2025). The final corpus therefore contains *36* studies.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

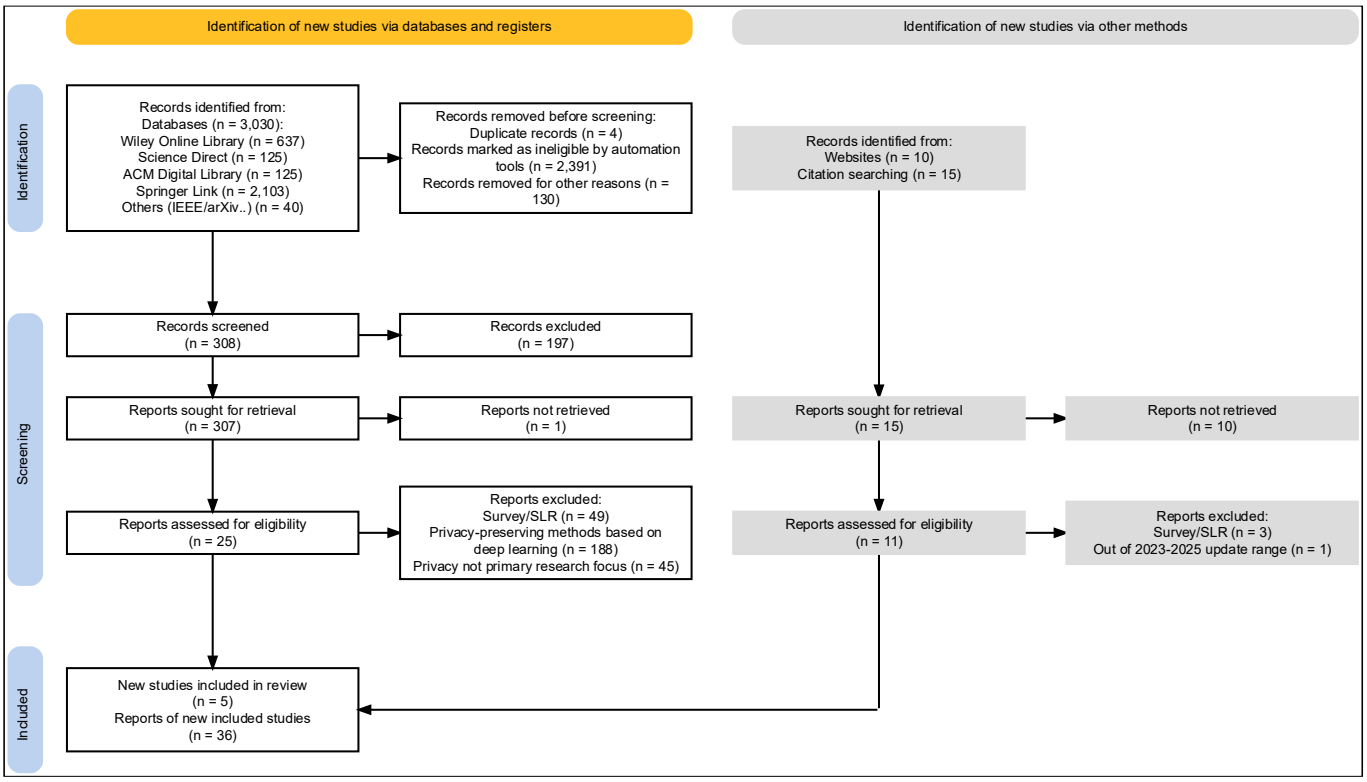

FIGURE 2. PRISMA 2020 FLOW DIAGRAM FOR THE STUDY SELECTION

(MAIN SEARCH 2013–2022; UPDATE 2023–2025).

5. **Flow diagram.** Figure 2 presents the PRISMA 2020 flow summarizing identification (databases/registers and other methods), screening, eligibility, and inclusion.

C- Corpus Construction

The bibliographic dataset was managed with Zotero. After filtering, the corpus comprises *36* papers: *15* journal articles, *16* conference papers, and *5* preprints. This distribution reflects both field maturity (journal publications) and ongoing dynamism (conference and preprint dissemination). Summary lists appear in **Table 10** (Appendix~A).

D- Bibliometric Keyword Co-occurrence Analysis

In addition to manual review, we conducted a bibliometric keyword co-occurrence analysis to identify dominant concepts and thematic clusters within the selected corpus. To strengthen the methodological framework, we conducted a bibliometric analysis using VOSviewer. A dataset of 36 articles was exported from Zotero in CSV format, including title and abstract fields. Terms were extracted with full counting, setting a minimum occurrence threshold of 10. Out of 1179 candidate terms, 22 met the threshold and were retained for visualization. Each term was assigned a relevance score, and 100% of the qualified terms were included.

Figure 3 illustrates the resulting keyword co-occurrence network. Node size reflects the frequency of a term, while edge thickness indicates co-occurrence strength. Three thematic clusters clearly emerge: (i) **cryptographic approaches** (red cluster: *homomorphic encryption*, *FHE*, *protocol*, *server*); (ii) **deep learning and differential privacy** (blue cluster: *differential privacy*, *deep neural network*, *model*, *framework*); and (iii) **system-level performance and applications** (green cluster: *accuracy*, *training*, *work*, *user*, *security*, *party*). At the center of the map, the terms **data** and **privacy** act as transversal concepts, strongly connected to all clusters, emphasizing their pivotal role in privacy-preserving deep learning research.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

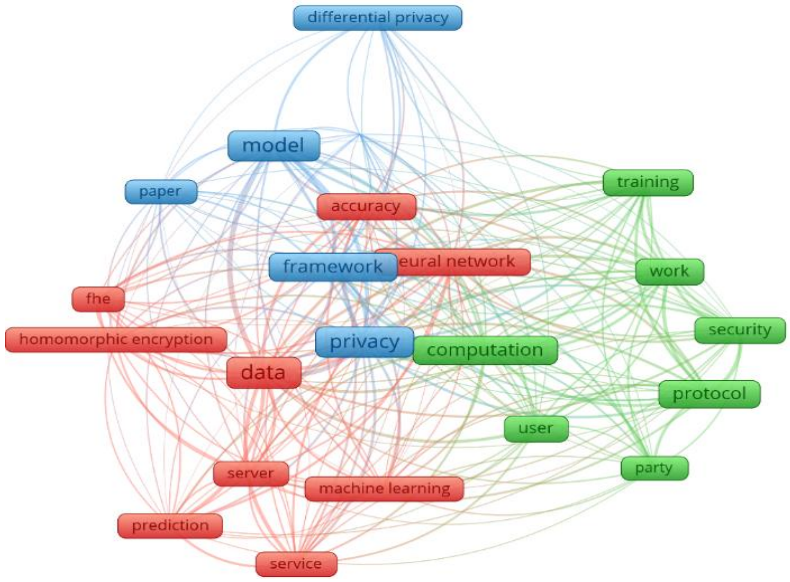

FIGURE 3. KEYWORD CO-OCCURRENCE NETWORK GENERATED WITH VOSVIEWER (36 BIBLIOGRAPHIC RECORDS, THRESHOLD = 10 OCCURRENCES, 22 TERMS RETAINED)

E- Attributes for Analysis

For each article, we extracted the following attributes:

Bibliographic attributes: reference, authors, title, year, type (journal, conference, preprint), and publication venue.

Technical attributes: dataset used, neural network architecture, training or inference phase (and perturbation phase for DP methods), type of threats addressed (e.g., reconstruction, model inversion, membership inference), system access model (white-box or black-box), techniques employed (HE, DP, SE, MPC), and reported limitations.

F- Metrics for Evaluation

The selected works were assessed across three dimensions: efficacy, privacy, and efficiency.

4. ***Efficacy:*** This dimension evaluates model performance, typically through:
    a. *Accuracy:* classification accuracy of the proposed methods.
    b. *Latency:* delay introduced by computation and communication overhead.
5. ***Privacy***
    a. *Data:* raw data remain inaccessible to servers or third parties.
    b. *Model:* except for prediction results, no party learns details of the trained model.
    c. *Result:* neither server nor client can infer information from prediction outputs.
6. ***Efficiency:*** is assessed through computational complexity, training and inference time, and communication cost.

This methodological framework establishes the foundation for the subsequent analysis. In the next section, we apply these attributes and metrics to construct a taxonomy of privacy-preserving deep learning techniques, evaluate their efficacy, and compare their strengths and limitations.

RESULTS

We classify PPDL approaches into four main technique families—HE, SMPC/hybrid, DP, and SE/TEEs—and evaluate how FL composes these techniques at scale under practical constraints.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A- PPDL based on Homomorphic Encryption

CryptoNets [20] is one of the first works on HE inference, researchers present a method to convert a trained neural network into an encrypted network, called CryptoNet by combining cryptography and deep learning.

This allows clients to send their data in an encrypted format using HE and receive the encrypted result as well, without the data being decrypted during the transfer between the client and the cloud server. Afterward, the clients can use their private key (which the cloud server has generated for each client) to decrypt the prediction result. This guarantees the confidentiality of the client and the confidentiality of the result. The performance evaluations on MNIST datasets reach an accuracy of 98.95% and can make more than 51,000 predictions per hour on a single PC. However, the weakness of Cryptonets is the performance limitation due to their complexity. The multilevel HE does not work well on deeper NNs that have a large number of non-linear layers. In this case, the accuracy decreases, and the error rate increases.

To improve the performance of [20], which is only good when the number of layers is restricted, the [21] researchers combined it with a polynomial approximation for the activation function and the batch normalization layer proposed by [22]. As a result, the structure of the regular NN will change. For the learning phase; the addition of a Batch normalization layer between the Pooling layer and the activation layer in order to avoid a strong degradation of the accuracy. The Max-pooling layer is not a linear function, it will be replaced by an Average-pooling layer which is more favorable to the FHE and has a low impact on the accuracy. Before classification; the ReLu function is replaced by a low-degree polynomial approximation as it gives a small error, which is very suitable to be used in this model. A batch normalization layer is added before each ReLu layer, it helps to restrict the input of each activation layer, which helps to obtain a stable distribution to avoid the strong degradation of accuracy. Performance evaluations on MNIST datasets, reach an accuracy of 99.30%, which significantly improves CryptoNets. However, they do not show any results on the latency of their method.

In order to overcome the biggest weakness of FHE which requires considerable time to evaluate deep learning models on encrypted data. The researchers [23] propose **TAPAS**; which uses binarised neural networks to speed up their HE inference method. The TAPAS architecture is composed of a fully connected layer, a convolution layer, and a batch normalization layer with sparse encrypted computation to reduce computation time. The key idea is the co-contribution of a new algorithm to accelerate binary computations in binary neural networks. Thus, support for parallel computing. They claim that, unlike [20] and [21] which only protect the data, their proposed scheme can also protect the confidentiality of the model. (Indeed, clients should generate parameters for the encryption based on the structure of f, so we are able to make inferences about the model). Nevertheless, the only limitation is that it only supports binary neural networks.

Faster CryptoNets [24], speeds up homomorphic evaluation in [20] by pruning the network parameters so that many multiplication operations can be omitted. The main weakness of Faster Cryptonets is that it is vulnerable to membership inference attacks, and model stealing [seen in Section III].

[25] propose **CryptoNN**; which is a privacy-preserving method that uses functional encryption for arithmetic computation on encrypted data. The FE scheme protects data in the form of a feature vector inside matrices. In this way, the matrix computation for the formation of the NN can be performed in encrypted form.

Orion [26] introduces a fully automated FHE framework for PyTorch models, enabling efficient private inference on complex networks with accuracy comparable to plaintext (e.g., 93.4% on ResNet-20 with CIFAR-10) and reduced latency (618s for ResNet-20).

HE-LRM [27] applies FHE to large recommendation models, achieving 85% accuracy on the UCI Heart Disease dataset and inference latencies ranging from 24 to 488 seconds on UCI and Criteo datasets.

Active ME! [28], Advances in activation functions for FHE, such as optimized polynomial approximations of Square and ReLU, have enabled up to 99.4% accuracy on LeNet-5 with MNIST and 89.8% on ResNet-20 with CIFAR-10, with latencies between 95 and 1697 seconds. **Table 3** summarizes these methods alongside earlier HE-based approaches.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

TABLE 3. COMPARISON OF DIFFERENT PPDL HE-BASED WORKS

Work	PP(HE)	Network	Dataset	Phase
CryptoNets [20]	HE	CNN	MNIST	Inference
Chabanne [21]	FHE	CNN	MNIST	Training, Inference
Face Match [29]	FHE	CNN	LFW, IJB-A/B, CASIA	Inference
Tapas [23]	FHE	CNN	Cancer, Diabtes, Faces, MNIST	Inference
Faster CryptoNets [24]	HE	CNN	MNIST	Inference
CryptoNNS [25]	FE	CNN	MNIST	Training, Inference
Orion [26]	FHE	CNN ResNet	MNIST, CIFAR-10	Inference
HE-LRM [27]	FHE	DLRM	UCI Heart, Criteo	Training, Inference
Activate ME! [28]	FHE	CNN	MNIST, CIFAR-10	Inference

B- PPDL based on Homomorphic Encryption

1. Secure-MPC Frameworks

The **Chameleon** framework, proposed by [30], introduces a novel PPDL approach that integrates Secure-MPC with CNNs. Chameleon operates in two distinct phases: an online phase employing protocols such as Additive Secret Sharing (ASS) and Garbled Yao Circuits (GC) to enable joint computations between two parties without revealing their inputs, and an offline phase utilizing a semi-honest third party (STP) to precompute Oblivious Transfers (OTs) and multiplication triples. Performance evaluations on the MNIST dataset demonstrate that Chameleon processes handwritten digit images 133 times faster than Microsoft's [20], highlighting its efficiency in secure inference. However, its reliance on a semi-honest third party introduces a trust assumption that could be a vulnerability if the STP is compromised, and the framework's efficiency is limited to inference, with no significant support for training phases.

Similarly, **SecureNN** [31] advances Secure-MPC by developing a system integrated with CNNs, with a notable contribution being a new protocol for Boolean operations (e.g., ReLU, Maxpool, and their derivatives). This protocol reduces communication overhead compared to the Yao GC used in Chameleon. Evaluations on the MNIST dataset reveal that SecureNN achieves a prediction accuracy exceeding 99% during training, with execution times 2-4 times faster than other MPC-based PPDL methods, surpassing frameworks like [30] and [32]. Despite these gains, SecureNN's three-party computation model assumes semi-honest behavior, and its performance may degrade with malicious adversaries, while the lack of GPU support limits scalability for larger datasets.

Recent research has focused on enhancing Secure-MPC efficiency. **Efficient Shamir-MPC** [33] utilizes Shamir secret sharing and fixed-point arithmetic to accelerate convolution and Softmax computations, achieving over 50% performance gains in simulations with three or more parties. However, its simulation-based evaluation lacks real-world dataset validation, and the fixed-point arithmetic may introduce precision errors in complex models. **Low-Latency MPC** [34] reduces communication rounds and optimizes non-linear functions, yielding 10-20% latency improvements on MNIST and CIFAR datasets. Yet, its optimization is tailored to specific non-linear functions, potentially limiting adaptability to diverse network architectures, and it assumes a semi-honest threat model that may not hold against malicious actors.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. Hybrid PPDL Approaches

Hybrid PPDL methods combine multiple privacy-preserving techniques to enhance performance and security.

Gazelle [32] exemplifies this by merging HE with traditional two-party computation (GC) techniques for secure inference. To optimize HE speed, Gazelle employs Single Instruction Multiple Data (SIMD) for ciphertext addition and multiplication, alongside Approximate Homomorphic Encryption (AHE) and ASS to accelerate convolution and matrix multiplication. The framework features a CNN architecture with two convolutional layers, two ReLU activation layers, a pooling layer, and a fully connected layer. Performance on MNIST and CIFAR-10 datasets outperforms [20] in execution time. However, Gazelle's limitation to two-party schemes and restrictions on client classification requests to mitigate link attacks pose challenges. Gazelle has spent a lot of time optimizing each of its components. However, there is still an overhead compared to native execution, among the reasons; the libraries used do not support GPU acceleration so it is much slower than linear layer evaluation techniques.

Building on Gazelle, **Delphi** [35] addresses communication and latency issues critical for latency-sensitive applications. The Delphi planner inputs all ReLU activations and a precision threshold *t*, outputting a hybrid CNN that integrates ReLU and quadratic activation functions while maintaining precision above *t*. Evaluations on CIFAR-10 and CIFAR-100 datasets show Delphi outperforming Gazelle by 9 to 22 times in efficiency, depending on parameters such as transferred data and execution time, through a synthesis of system, cryptographic, and machine learning techniques. Nevertheless, Delphi's reliance on a two-party HE/MPC hybrid assumes semi-honest participants, and its performance may suffer with increased network latency or when handling deeper networks due to computational complexity.

CrypTFlow [36] adopts a hybrid approach by combining Secure Enclaves with secret sharing for DNNs and CNNs, targeting the MNIST, CIFAR, and ImageNet datasets. Despite its innovative design, CrypTFlow's inability to support GPU processing results in significant computational overhead during secure learning, and its dependence on SE introduces risks if the trusted execution environment is compromised, limiting its robustness against advanced attacks.

3. Spectrum of Party-Computation Settings

As outlined in **Table 4**, Secure-MPC and hybrid PPDL frameworks span a rich spectrum of party-computation (PC) settings, threat models, and protocol guarantees.

Two-party computation (2PC). Protocols such as *Delphi* and *Gazelle* rely on hybrid HE/MPC constructions (HE + GC + ASS) for private inference. They operate under a "two-party client–server model" without a semi-honest third party (STP), and assume semi-honest adversaries (MAL = X). These designs are lightweight but limited to inference and vulnerable if one party deviates from the protocol. **Three-party computation (3PC)**. Frameworks like *Chameleon*, *SecureNN*, and *CryptFlow* incorporate a semi-honest third party (STP = √) that helps generate correlated randomness or assist with offline precomputation. This setting balances efficiency and security: operations such as Beaver triples, ASS, and Boolean gates are accelerated, while data, model, and intermediate results remain hidden. However, these protocols generally assume "semi-honest adversaries" (MAL = X). **3+ party Shamir-based protocols**. The work of *Efficient Shamir-MPC* extends beyond 3PC by leveraging Shamir secret sharing (Shamir-SS). STP is required (√), and the scheme remains secure under the semi-honest assumption. This setup demonstrates scalability and robustness in simulations but has not yet been validated under malicious adversaries. **Four-party computation (4PC)**. Protocols such as *FLASH* and *SWIFT* distribute the computation across four parties, incorporating both Additive and Robust Secret Sharing (ASS/RSS). These are the only surveyed frameworks explicitly securing against malicious adversaries, through *Guaranteed Output Delivery (GOD)*. GOD ensures that even if one or more parties deviate, all honest parties still receive the correct output. This robustness comes at the cost of higher online communication but significantly strengthens security guarantees.

This progression from 2PC to 4PC illustrates a clear trade-off: increasing the number of parties and adversarial protections improves security and robustness (especially under malicious models) but imposes higher computational and communication costs.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

TABLE 4. COMPARISON OF DIFFERENT PPDL SECURE MPC-BASED AND HYBRID WORKS

Work	PC	Threat model		Tech	Network	Dataset	Phase
		STP	MAL				
Gazelle [32]	2	X	X	HE GC ASS	CNN	MNIST CIFAR-10	Inference
DELPHI [35]		X	X	HE GC ASS	CNN	CIFAR-10,100	Inference
Low-Latency MPC [34]		X	X	ASS OT Beaver triples	DNN	MNIST CIFAR	Training Inference
Chameleon [30]	3	√	X	GC ASS	CNN SVM	MNIST Credit Approval	Inference
SecureNN [31]		√	X	ASS	CNN	MNIST	Training Inference
CryptFlow [36]		√	X	SE ASS	DNN CNN	MNIST CIFAR Image-Net	Inference
Shamir-MPC [33]	3+	√	X	Shamir-SS	CNN	Simulation	Inference
FLASH [37]	4	X	GOD	ASS RSS	DNN BNN	MNIST	Training Inference
SWIFT [38]	3+	X	GOD	ASS RSS	CNN	MNIST CIFAR-10	Training Inference

C- PPDL based on Secure Enclaves

Chiron, [39] provides a black-box system for PPDL. It uses SGX enclaves and the Ryoan sandbox. Chiron uses a secure enclave environment where model parameters are exchanged via the server. It runs the untrusted code to update the model and implements protection using sandboxes so that the code does not leak data outside the enclave.

TABLE 5. COMPARISON OF DIFFERENT PPDL SE-BASED WORKS

Work	Tech	Network	Dataset	Phase
CHIRON [39]	SGX SandBox Ryoan	DNN	CIFAR ImageNet	Training
SLALOM [40]	SGX TEE	DNN	ImageNet	Inference

SLALOM, [40] uses Trusted Execution Environments (TEE), which isolate the computational process from untrusted software. The DNN computation is divided into trusted and untrusted parts. SLALOM runs DNN in the Intel SGX enclave which delegates the computational process to an untrusted GPU. The weakness of this approach is that it limits the operation of the CPU since the TEE does not allow access to the GPU. A vulnerability through a side channel attack can occur, as shown by [41].

D- PPDL based on Differential Privacy

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

This section presents recent research based on DP to protect user privacy in DL. We classify this section according to the level at which DP can be applied, as shown in **Figure 4** below.

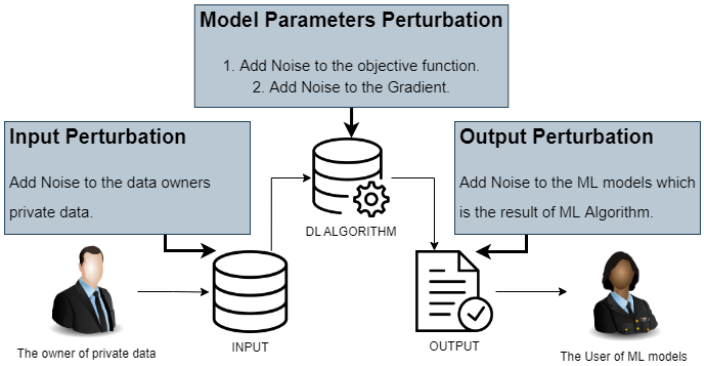

FIGURE 4. THE DESIGN PRINCIPLES OF DIFFERENTIAL PRIVATE MACHINE LEARNING

1. Input Perturbation

These approaches add noise to the original dataset to produce a new privacy-preserving (PP) dataset.

Researchers [42] proposed **ACIES**, a differential privacy-preserving classification system for edge computing, utilizing models such as Support Vector Machines (SVM), k-Nearest Neighbors (kNN), and Sparse Representation Classification (SRC). Unlike native approaches that add noise to raw data, ACIES injects Laplace noise during feature extraction to indirectly control information leakage from training data, with these feature vectors used for training and evaluation. Performance evaluation on diverse datasets including YaleB, CSI, MNIST, and HAR demonstrates resilience against reconstruction attacks, with a maximum accuracy impact of 5%.

2. Perturbation of Model Parameters

This subsection presents recent research addressing confidentiality in DL by applying DP during training to produce privacy-preserving models.

1. Perturbation of the objective:

The perturbation of the objective function is explored for machine learning tasks with convex objective functions. To identify the intensity of the added noise; calculating the sensitivity of the objective function is crucial due to the non-convexity of typical DL objective functions. A solution proposed by [43] involves replacing the non-convex function with an approximate convex polynomial function, followed by objective function perturbation. However, this approximation limits the power and applicability of traditional DNNs.

The Adaptive Laplace Mechanism (**AdLM**) [43] combines DP with relevance-guided noise placement. Layer-wise Relevance Propagation (LRP) is used *only* to estimate feature relevance (utility signal); it is not a privacy mechanism. First, the average relevance scores are *privatized* by adding Laplace noise to obtain DP relevance estimates. Then, AdLM injects adaptive Laplace noise into each layer's affine transformation, assigning larger noise to less-relevant features. A normalization layer is inserted before non-linearities to control sensitivity, and the output layer is privatized via a polynomial loss approximation with noisy coefficients. Under comparable privacy budgets, AdLM reports better utility than [43] on MNIST and CIFAR-10. The privacy guarantee in AdLM comes exclusively from the calibrated Laplace mechanisms; LRP serves to steer the noise, not to provide privacy by itself.

[45] adopted similar techniques to [43], enhancing accuracy by integrating DP with LRP. They perturb the target value at each batch via the loss function, ensuring each data access point is protected to yield a reliable privacy-preserving model. The loss function is approximated using the Maclaurin series instead of the Taylor series used by [43]. Performance evaluations on the WDBC dataset, with dense noise addition, achieved accuracy close to the unprotected version.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. **Gradient Perturbation:** Gradient perturbation is a widely adopted approach for private learning in deep learning.

[44] proposed a Differential Private Stochastic Gradient Descent (**DP-SGD**). During training, at each SGD step, gradients of a random subset of examples are computed, with the L2 norm of each gradient clipped. These gradients are averaged, and Gaussian noise is added. The model is then output, and the overall privacy loss is tracked using a Moments Accountant (MA), which accumulates costs for each training data access and determines the optimal **δ** leakage parameter for a given privacy budget. Performance evaluation on MNIST and CIFAR-10 for image classification shows accuracies of 97% on MNIST (1.3% below the non-private baseline) and a larger 7% gap on CIFAR-10.

In [46], researchers introduced f-DP, a recent privacy definition, for training private DL models using SGD or ADAM. They proposed Differential Gaussian Privacy (**GDP**), analyzing privacy budget depletion in DNN training with the Adam optimizer without relying on the Moments Accountant in the (ϵ, δ)-DP framework. Performance results across MNIST (image classification), IMDb (text classification), and MovieLens (recommender systems) indicate that trained networks are private under f-DP (e.g., 1.13-GDP) but not under (ϵ, δ)-DP due to conservative privacy bounds.

To address the high noise costs of traditional DP-SGD, [47] introduce **Spectral-DP**, a novel framework that performs gradient perturbation in the spectral (frequency) domain. Instead of directly adding Gaussian noise to raw gradients, Spectral-DP applies a spectral transformation and filtering process that suppresses high-variance components before perturbation, thereby reducing sensitivity and minimizing the amount of noise required to achieve rigorous (ϵ, δ)-DP guarantees. This design naturally aligns with convolutional operations in CNNs and further leverages block-circulant compression to enable efficient spectral processing in fully connected layers. Empirical evaluations on benchmark datasets such as CIFAR-10 and ImageNet demonstrate that Spectral-DP consistently achieves superior privacy–utility trade-offs compared to standard DP-SGD, yielding accuracy improvements of 3–5% in both training from scratch and transfer learning scenarios.

[48] propose a **layer-level adaptive gradient perturbation** mechanism to enhance the privacy-utility balance in differentially private deep learning. The method dynamically allocates privacy budgets across layers during training: starting with equal budgets and progressively adjusting them based on iteration progress, assigning less noise (increased budget) to input-proximal layers and more noise to output-proximal layers to counter membership inference attacks. By perturbing only selected hidden layers and leveraging DP's post-processing immunity, it maintains strong privacy guarantees. Experiments on five well-known datasets reveal higher accuracy and greater resilience against attacks compared to uniform noise baselines at equivalent privacy levels.

3. Output Perturbation

Output perturbation involves running a non-private learning algorithm and adding noise to the result. This approach can degrade the model's utility, particularly in deep learning, where high-dimensional outputs are sensitive to noise. To mitigate this, techniques like noisy aggregation of predictions are employed. One prominent example is the Private Aggregation of Teacher Ensembles **PATE** framework, introduced by [49], which leverages a teacher-student paradigm to preserve privacy. In PATE, multiple teacher models are trained on private data and aggregate their predictions into a single output, with Laplace noise added to the vote counts to ensure (ϵ, δ)-DP. The student model, trained on publicly labeled data annotated by these noisy teacher outputs using a differential privacy method (e.g., noisy voting in a GAN), cannot access the original data or teacher parameters, thus preventing adversaries from extracting confidential information. Performance evaluations on MNIST and SVHN datasets indicate reduced accuracy for complex or diverse data, attributed to the added noise, though it remains effective for simpler tasks.

Building on PATE, [50] proposes a solution for information retrieval (IR) applications, such as document re-classification, achieving acceptable performance with low privacy risk. This teacher-student approach also addresses the challenge of limited large datasets in IR by leveraging unlabelled public data, highlighting mimic learning's broader utility in privacy-preserving deep learning. **Table 6** summarizes these methods alongside earlier DP-based approaches.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

TABLE 6. COMPARISON OF DIFFERENT PPDL DP-BASED WORKS

Work	Tech	Network	Dataset	Phase
Acies [42]	ϵ-DP, SVD Laplace mechanism kNN, SVM, SRC	/	YaleB HAR MNIST CSI	Inference, Feature extraction perturbation
AdLM [43]	ϵ-DP, LRP Laplace mechanism Taylor expansion Polynomial app.	DNN	MNIST CIFAR-10	Model parameters perturbation
Adesu. [45]	ϵ-DP, LRP Laplace mechanism Maclaurin series Polynomial app.	DNN	WDBC	Label perturbation
DPSGD [44]	(ϵ, δ)-DP SGD Gaussian noise Moment account.	DNN	MNIST CIFAR-10	Training, Model parameters perturbation
GDP [46]	f-DP SGD/Adam Gaussian noise	DNN LSTM RNN	MNIST IMDb Movie-Lens	Training, Model parameters perturbation
S-DP [47]	(ϵ, δ)-DP Spectral perturbation Gaussian noise	CNN DNN	Benchmark	Training, Model parameters perturbation
Layer. [48]	(ϵ, δ)-DP Adaptive gradient Gaussian noise	DNN	MNIST CIFAR-100	Training, Model parameters perturbation
PATE [49]	(ϵ, δ)-DP Laplace mechanism SSL Moment Account.	DNN GAN	MNIST SVHN	Training, Label/Output perturbation
Dahgh. [50]	DP Laplace mechanism	DNN	IR	Training, Label/Output perturbation

E- PPDL via Federated Learning

Federated Learning (FL) enables decentralized training across multiple data holders (clients) without centralizing raw data. In its canonical cross-device variant, a coordinator broadcasts the global model parameters $\boldsymbol{\theta_t}$ at round $\boldsymbol{t}$; each client $\boldsymbol{k}$ performs local updates on its private dataset $\mathbf{D_k}$ starting from $\boldsymbol{\theta_t}$, obtaining an updated local model $\boldsymbol{\theta_{t+1}}^{(k)}$. The server then aggregates these updates typically via FedAvg rule:

$$\theta_{t+1} \leftarrow \sum_{k=1}^{k} \frac{n_k}{\Sigma_j n_j} \theta_{t+1}^{(k)} \ , \ n^k = |\mathbf{D_k}|. \qquad (1)$$

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

FL reduces raw-data exposure but does *not* guarantee privacy by itself: model updates and gradients may leak information about local records and even enable reconstruction or membership inference (see **Section II**).

1. Threat Model in FL

We consider three adversarial settings: (i) an "*honest-but-curious server*" attempting to infer client data from per-round updates, (ii) "*malicious clients*" performing poisoning/backdoor attacks (which can also amplify privacy leakage of benign clients), and (iii) "*system-level adversaries*" exploiting side-channels, traffic-analysis, or TEE leakages when hardware protection is used.

Compared to centralized training, FL exposes additional surfaces: update/gradient inversion, property inference, round-wise membership inference, client deanonymization, and colluding-client attacks.

2. Core Privacy Techniques for FL

a) **Secure Aggregation (SA).** Cryptographic SA lets the server learn only the *sum* of client updates (hiding each $\theta_t^{(k)}$, typically through pairwise masks or additive secret sharing [51]. SA is orthogonal to DP and reduces the server's visibility, but does not by itself bound leakage from the aggregate.

b) **Differential Privacy for FL.** Two deployment styles are prevalent.
Central DP-FL clips each client update to $\boldsymbol{\ell_2 - norm}$ C and adds Gaussian noise at the server after aggregation:

$$\tilde{g}_t = \frac{1}{K}\sum_{k=1}^{K} \text{clip}\left(g_t^{(k)}, C\right) + N\left(0, \sigma^2 C^2 I\right). \qquad (2)$$

with privacy accounting over rounds (e.g., RDP/GDP accountants) [46, 52, 53].
Local DP-FL adds noise client-side before SA, strengthening per-client protection against a curious server at the cost of larger utility degradation [54]. "DP-FedLoRA" Recent work adapts [55].

c) **Hybrid FL with HE/MPC/TEE.** Hybrid FL frameworks increasingly combine advanced cryptographic techniques to enhance privacy and efficiency. Multi-key homomorphic encryption (HE) allows each participant to encrypt updates with their own key, enabling secure aggregation without exposing individual data, as demonstrated in recent efficient federated learning schemes [56-58]. Complementarily, [59] introduce a pure MPC-based framework using secret sharing techniques (e.g., Sharemind) to achieve secure aggregation, enhancing robustness against inference attacks in distributed settings. Trusted Execution Environments (TEEs) are also used to offload sensitive operations, improving computational efficiency while maintaining strong security guarantees [56]. Additionally, Jin et al. [60] present an efficient HE-based FL system (FedML-HE) that reduces computational overhead for deep networks, achieving up to a 10-fold reduction in latency for ResNet-50 training by optimizing HE schemes such as CKKS, making it scalable for cross-silo FL deployments with minimal accuracy loss (e.g., 94.1% on ImageNet). Similarly, Kalapaaking et al. [61] propose a blockchain-enhanced TEE framework for secure aggregation in IoT contexts, leveraging Intel SGX to execute tamper-proof updates and Hyperledger Fabric for decentralized auditability, which mitigates model poisoning attacks with a reported 98% success rate in detecting malicious updates across distributed IoT nodes.

TABLE 7. COMPARISON OF DIFFERENT PPDL VIA FL-BASED WORKS

Work	PP(FL)	Network	Dataset	Phase
DP-FedAvg [52]	CDP	LSTM	Mobile keyboard data	Training Aggregation
DP-Fed LoRA [55]	LDP	Transformer	Alpaca-GPT-4	Training

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

		LLM/LoRA		Fine-tuning
SecureAgg [51]	SA/MPC	DNN	Large scale	Aggregation
FedML-HE [60]	HE	ResNet-50 BERT	Wikitext CIFAR-100	Training Aggregation
MPC-FL [59]	MPC	CNN LSTM	MNIST CIFAR-10 CASA	Training
HE-Key [58]	HE	CNN	UP-FALL	Training Aggregation
TEE-FL [61]	TEE	CNN	MNIST HAR(IoT)	Training Aggregation

3. FL-Specific Attacks and Defenses

Gradient/Update Inversion. Recent studies show that exposing model updates in FL still leaks sensitive information. [62] analyze the feasibility of gradient inversion attacks under different modes, showing that models using fixed batch-normalization statistics in inference mode are significantly more vulnerable than during training. [63] further demonstrate that in FL with text data, discrete optimization over embedding and fully-connected layer gradients (via their FET method) can recover private text sequences. Defenses include stronger clipping, central DP noise, avoiding inference-mode normalization, and limiting the number of local steps [64].

Membership/Property Inference. remains a persistent threat in FL. [65] propose "*FedMIA*", which leverages the "all-for-one" principle by combining updates from non-target clients across multiple rounds, substantially improving attack performance even under defenses. A recent survey by Bai et al. [66] categorizes MIAs and defenses, highlighting that larger client datasets, higher model complexity, careful privacy accounting, and differential privacy mechanisms help reduce attack success.

Poisoning/Backdoors. Malicious clients may still steer the global model toward targeted misbehavior. Defenses include Byzantine-robust aggregation rules such as Krum, Trimmed Mean, and Median [67], anomaly detection on updates, and combining differential privacy with gradient clipping to bound adversarial influence.

Deanonymization/Traffic Analysis. Side channels such as update timing, size, and client participation patterns can deanonymize users or link updates to clients. Mitigations include fixed update sizes, randomized batching and padding, and secure aggregation protocols that hide individual client contributions.

4. Positioning FL in the PPDL Taxonomy

FL is an orchestration paradigm that leverages DP, HE/MPC, TEEs, and robust aggregation rather than replacing them. In our taxonomy, FL forms a top-level branch with sub-techniques (DP-FL, SA, HE-FL, MPC-FL, TEE-FL), and cross-references to cryptographic and DP **Sections II, III**. Practically, combining SA + central DP achieves a favorable privacy–utility–efficiency trade-off for many cross-device deployments, while HE/MPC/TEE variants address stronger adversaries or regulatory constraints.

F- Attack-Defense Mapping

The following mapping **Table 8** summarizes representative attack classes against ML/DL systems and the principal defenses evaluated in the literature. For each attack, we list a characteristic attack vector (example papers) and the defensive families that have been proposed and empirically tested. This compact reference helps practitioners select targeted countermeasures and highlights gaps where defenses remain immature or impose high system costs.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cryptographic approaches and TEEs provide strong guarantees against reconstruction and gradient-inversion attacks, but impose significant computational and communication overhead. DP remains the most versatile defense against membership and inversion threats, though at the expense of model utility—particularly in large-scale architectures where tighter accounting and per-layer perturbation are required. FL–specific vulnerabilities (e.g., poisoning, Sybil, free-rider) are most effectively countered through robust aggregation, anomaly detection, and access control, ideally combined with SA/DP for a balanced trade-off between privacy, robustness, and efficiency.

TABLE 8. MAPPING OF PRIVACY ATTACKS IN ML/DL TO REPRESENTATIVE DEFENSE TECHNIQUES

Attack Type	Attack Vector / Example	Representative Defenses
Reconstruction Attacks	Recover raw inputs from feature vectors or gradients [14, 16, 17]	HE [20, 21], Secure-MPC protocols (GC, ASS) [30, 31], Hybrid HE+MPC [32, 35], TEE [36, 40]
Model Inversion Attacks	Infer representative inputs from model outputs or confidence scores [15, 62, 63]	DP (DP-SGD, AdLM) [43, 44], output perturbation [50], restricting access to logits/confidences, HE/MPC inference
Membership Inference Attacks	Decide whether a record was in the training set [18, 19, 65]	DP mechanisms (DP-SGD, GDP, PATE) [44, 46, 49], federated DP (DP-FedAvg) [52], limiting model outputs (label-only), robust aggregation in FL [67]
Gradient Leakage in FL	Reconstruct client data from shared updates/gradients [62-64]	SA [51], clipping + central DP noise [52, 53], local DP-FL [54], HE/MPC-FL [57, 59]
Poisoning / Backdoor Attacks in FL	Malicious clients inject corrupted updates or triggers [68]	Byzantine-robust aggregation (Krum, Trimmed-Mean, Median) [67], DP clipping, anomaly detection, TEE-based secure aggregation [61]
Side-channel / Hardware Attacks	Leakage from TEEs (e.g., SGX, TDX) or GPU memory [41]	TEE hardening [40, 41], constant-time protocols, hybrid TEE+MPC for enclave robustness [36]
Unintended Memorization (LLMs)	LLMs regurgitate rare sensitive sequences verbatim [3]	DP-SGD/GDP for transformers [3, 44, 46], dataset deduplication/redaction [3], secret filtering
Extraction / Prompt Attacks (LLMs)	Adversarial prompting to extract training snippets or inject instructions	DP training [3], retrieval/content sanitization, tool-use constraints, policy-tuned decoding

G- Metrics Evaluation

In this subsection, we consolidate all surveyed works and score them against the three global criteria introduced in **Section 4**: *efficacy*, *privacy*, and *efficiency*. Rather than re-describing each method, we present a unified comparison **Table 9** that surfaces the dominant trade-offs across approaches. Broadly, HE methods (e.g., CryptoNets, Orion) preserve accuracy but incur high computational latency; SMPC and hybrid schemes (e.g., Gazelle, SecureNN, MPCFL) strengthen privacy at the cost of communication overhead; DP techniques (e.g., DP-SGD, Spectral-DP) offer tunable privacy–utility trade-offs with noticeable accuracy drops on complex datasets; and federated-learning variants integrate these primitives at scale to balance accuracy and deployment constraints. No single paradigm dominates all three axes, which motivates hybrid designs tailored to task, threat model, and system constraints.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Legend. "√" =used/reported; "X" =not used/not reported. Privacy columns (Data/Model/ Result) indicate whether *confidentiality is guaranteed by design* (not merely assumed). Efficiency subcolumns report whether compute/communication cost and time metrics were *empirically measured*.

TABLE 9. METRIC EVALUATION

Work	PPDL METHOD					EFFICACY		PRIVACY			EFFICIENCY		
	HE	SMPC	DP	SE	FL	Accuracy (%)	Latency (ms)	Data	Model	Result	Computation/ Communication Cost	Inference Time (s)	Training Time (s)
CryptoNets [20]	√	X	X	X	X	√	√	√	X	√	X	√	X
Chabanne [21]	√	X	X	X	X	√	X	√	X	√	√	X	X
Face Match [29]	√	X	X	X	X	√	X	√	√	√	√	√	X
Tapas [23]	√	X	X	X	X	√	X	√	√	√	√	√	X
F-CryptoNets [24]	√	X	X	X	X	√	X	√	X	√	√	√	X
CryptoNNs [25]	√	X	X	X	X	√	X	√	√	X	√	X	√
Orion [26]	√	X	X	X	X	√	√	√	√	X	√	√	X
HE-LRM [27]	√	X	X	X	X	√	√	√	√	√	√	X	√
Activate ME! [28]	√	X	X	X	X	√	√	√	√	X	√	√	X
Chameleon [30]	X	√	X	X	X	√	√	√	X	√	√	√	X
SecureNN [31]	X	√	X	X	X	√	√	√	√	√	√	√	√
Flash [37]	X	√	X	X	X	√	√	√	√	X	√	√	√
Swift [38]	X	√	X	X	X	√	√	√	√	X	√	√	√
Efficient Shamir [33]	X	√	X	X	X	√	√	√	√	X	√	√	X
Low-Latency [34]	X	√	X	X	X	√	√	√	√	√	√	√	√
Gazelle [32]	√	√	X	X	X	X	√	√	√	√	√	√	X
Delphi [35]	√	√	X	X	X	√	√	√	√	√	√	√	X
CryptFlow [36]	X	√	X	√	X	√	√	√	√	√	√	√	X
ACIES [42]	X	X	√	X	X	√	√	X	√	√	√	√	X
AdLM [43]	X	X	√	X	X	√	X	X	√	√	√	X	X
Adesuyi [45]	X	X	√	X	X	√	X	X	√	√	√	X	X
DPSGD [44]	X	X	√	X	X	√	X	X	√	√	√	X	X
GDP [46]	X	X	√	X	X	√	X	√	√	X	√	X	X

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Spectral-DP [47]	X	X	√	X	X	√	X	X	√	√	√	X	√
Layer-LA [48]	X	X	√	X	X	√	X	X	√	√	√	X	√
Pate [49]	X	X	√	X	X	√	X	√	√	√	X	X	X
Dahghani [50]	X	X	√	X	X	√	X	√	√	√	X	√	X
DP-FedAvg [52]	X	X	√	X	√	√	X	√	X	X	√	X	X
DP-Fed LoRA [55]	X	X	√	X	√	√	X	√	√	X	√	X	X
SecureAgg [51]	X	√	X	X	√	√	√	√	X	X	√	X	√
FedML-HE [60]	√	X	X	X	√	√	√	√	√	X	√	X	√
MPC-FL [59]	X	√	X	X	√	√	√	√	X	X	√	X	X
HE-Key [58]	√	X	X	X	√	√	√	√	√	X	√	X	X
Chiron [39]	X	X	X	√	X	√	X	√	√	√	√	√	√
Slalom [40]	X	X	X	√	X	√	X	√	√	√	√	√	X
TEE-FL [61]	X	X	X	√	√	√	√	√	√	√	√	X	√

DISCUSSION

To consolidate the findings of this survey, we revisit the research questions posed in **Section 4** and provide evidence-based answers.

RQ.1: What attacks can compromise the privacy of private data in machine/deep learning?

Our review shows that privacy in ML/DL can be compromised through a wide spectrum of attacks. “Reconstruction” and “gradient inversion” attacks allow adversaries to recover raw input features from gradients or intermediate representations. “Model inversion” exploits output confidence scores to infer representative inputs, while “membership inference” identifies whether a specific record was used in training. In federated learning, “gradient leakage” and “poisoning/backdoor” attacks represent particularly severe risks, as client updates can be exploited or manipulated. Additionally, “side-channel exploits” in trusted hardware and “unintended memorization” in large generative models expand the attack surface beyond traditional learning paradigms. These findings, summarized in **Table 8**, confirm that vulnerabilities exist across all phases of the learning pipeline.

RQ.2: Can we quantify and control the rate of data leakage?

Differential privacy provides the most rigorous framework for quantifying privacy leakage through formal (ϵ, δ)-bounds. Advanced variants such as Rényi DP and Gaussian DP improve accounting under iterative optimization, making them suitable for deep learning. However, empirical evidence indicates that utility degrades as stricter privacy guarantees are enforced, particularly in large-scale or high-dimensional tasks. Recent refinements, such as spectral perturbation (Spectral-DP) and adaptive noise allocation at the layer level, show that leakage can be “controlled” more efficiently without incurring prohibitive accuracy loss. While complete elimination of leakage remains infeasible, these approaches demonstrate that privacy budgets can be tuned to balance protection with performance.

RQ.3: What is the most promising method to make DL models and data less vulnerable to attack?

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

While some works such as Delphi and Low-Latency MPC achieve excellent scores across efficacy, privacy, and efficiency, their strengths remain bounded by specific assumptions (e.g., inference-only setting, semi-honest adversaries). Similarly, systems like HE-LRM, CrypTFlow, and TEE-FL approach near-complete coverage but still leave certain attack vectors unaddressed. This confirms that no single method universally dominates across all contexts. Instead, the trajectory of PPDL research points toward hybrid approaches—integrating DP, cryptographic protocols, and system-level safeguards—as the most promising way to achieve both robustness and practicality across diverse deployment scenarios.

RQ.4: How have PPDL techniques evolved for generative models and distributed networks?

The evolution of PPDL reflects a shift from protecting classical supervised learning to addressing vulnerabilities in "generative models" (GANs, transformers, LLMs) and "distributed training frameworks" (federated learning). For LLMs, privacy risks include unintended memorization and adversarial extraction. Defenses now integrate DP-SGD for transformers, dataset deduplication, and secret filtering at inference. Federated learning has matured from DP-FedAvg to advanced designs such as local DP-FL, DP-FedLoRA for LLM fine-tuning, and hybrid aggregation frameworks leveraging HE, MPC, and TEEs. These developments highlight a trend toward "application-aware PPDL", where techniques are tailored to the practical threat surfaces of emerging architectures.

General Synthesis and Outlook

The survey reveals three overarching insights. First, privacy–utility–efficiency trade-offs remain central: stronger privacy mechanisms often entail computational or accuracy costs that must be carefully managed. Second, effective protection requires "layered defenses" combining DP, cryptography, and system-level measures (e.g., robust aggregation, anomaly detection). Third, future research must address the scalability of defenses to large models and decentralized infrastructures, particularly where regulatory or resource constraints limit the applicability of heavy cryptographic schemes. Hybrid PPDL designs, adaptive DP accounting, and privacy-aware system optimizations offer promising avenues for bridging the gap between theoretical guarantees and practical deployment.

CONCLUSION

This survey provided a comprehensive review of privacy-preserving deep learning (PPDL) methods, spanning homomorphic encryption (HE), secure multi-party computation (SMPC), differential privacy (DP), secure enclaves (SE/TEEs), and federated learning (FL) as an orchestration paradigm. We systematically analyzed how these approaches address reconstruction, inversion, membership inference, poisoning, and hardware-level attacks, while evaluating them under unified criteria of efficacy, privacy, and efficiency.

Our findings demonstrate that no single paradigm offers a complete solution: HE and MPC deliver strong confidentiality but at high computational and communication cost; DP achieves formal guarantees but reduces accuracy; SE/TEEs provide hardware-backed isolation but are vulnerable to side-channel leakage; and FL reduces raw-data exposure yet introduces novel vulnerabilities. Hybrid methods—integrating secure aggregation, differential privacy, and lightweight cryptographic or hardware-assisted protocols—emerge as the most promising path toward practical deployment.

Looking ahead, three challenges remain critical. First, scaling defenses to large generative models and decentralized infrastructures without prohibitive overhead. Second, tailoring privacy mechanisms to diverse application domains such as healthcare, finance, and IoT. Third, bridging the gap between theoretical guarantees and real-world robustness against adaptive adversaries.

Overall, the evolution of PPDL reflects a shift from isolated techniques toward holistic, layered strategies. By uniting formal privacy guarantees, efficient cryptographic protocols, TEE-backed isolation, and robust system designs, future research can enable deep learning systems that are both trustworthy and practical for sensitive, large-scale applications.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

REFRENCES

[1] Dwork, C., A. Smith, T. Steinke, and J. Ullman. (2017). "Exposed! A Survey of Attacks on Private Data." Annual Review of Statistics and Its Application 4 (1): 61–84. doi:10.1146/annurev-statistics-060116-054123
[2] Zhan, S., L. Huang, Guoliang L., Shijie Z., Zhen G., and H. Chao. (2025). "A Review on Federated Learning Architectures for Privacy-Preserving AI: Lightweight and Secure Cloud–Edge–End Collaboration." Electronics 14 (13): 2512. doi:10.3390/electronics14132512.
[3] Li, H., Y. Chen, J. Luo, J. Wang, H. Peng, Y. Kang, Y. Song, et al. 2023. "Privacy in Large Language Models : Attacks, Defenses and Future Directions."
[4] Rivest, R. L., L. Adleman, and M. L. Dertouzos. (1978). "On Data Banks and Privacy Homomorphisms." In Foundations of Secure Computation, 4:169–80. 11. Cambridge, MA, USA: Academia Press.
[5] Sahai, A., and B. Waters. (2005). "Fuzzy Identity-Based Encryption." In Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn., 457–73. Berlin, Germany: Springer.
[6] Boneh, D., A. Sahai, and B. Waters. (2011). "Functional Encryption: Definitions and Challenges." In Proc. Theory Cryptogr. Conf., 253–73. Berlin, Germany: Springer.
[7] Yao, Andrew C.-C. (1986). "How to Generate and Exchange Secrets." In Proc. 27th Annual Symposium on Foundations of Computer Science (FOCS), 162–67.
[8] Wood, A., Micah A., A. Bembenek, M. Bun, M. Gaboardi, J. Honaker, K. Nissim, David R. O'Brien, T. Steinke, and S. Vadhan. (2018). "Differential Privacy: A Primer for a Non-Technical Audience." Vanderbilt Journal of Entertainment & Technology Law 21: 209–75.
[9] Dwork, C., F. McSherry, K. Nissim, and A. Smith. (2006). "Calibrating Noise to Sensitivity in Private Data Analysis." In Proc. Theory Cryptogr. Conf., 265–84. Berlin: Springer.
[10] Pan, Ke, Y. Ong, M. Gong, H. Li, A. K. Qin, and Y. Gao. (2024). "Differential Privacy in Deep Learning: A Literature Survey." Neurocomputing 589: 127663. doi: 10.1016/j.neucom.2024.127663
[11] Intel. (2014). "Software Guard Extensions Programming Reference." Santa Clara, CA, USA: Intel Corporation.
[12] Haykin, Simon S. (2009). Neural Networks and Learning Machines. Pearson Prentice Hall.
[13] Benyamina, O. N. A., and Slama Z. (2025). "Privacy-Aware Income Prediction Using Deep Neural Networks on the UCI Adult Dataset." Journal of Information Systems Engineering and Management (JISEM) 10 (1): 1–11. doi:10.55267/iadt.12273
[14] Feng, J., and Anil K. Jain. (2011). "Fingerprint Reconstruction: From Minutiae to Phase." IEEE Transactions on Pattern Analysis & Machine Intelligence 33 (02): 209–23.
[15] Fredrikson, M., Somesh J., and T. Ristenpart. (2015). "Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures." In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS), 1322–33.
[16] Al-Rubaie, M., and J. M. Chang. (2016). "Reconstruction Attacks Against Mobile-Based Continuous Authentication Systems in the Cloud." IEEE Transactions on Information Forensics and Security 11 (12): 2648–63. doi:10.1109/TIFS.2016.2594132
[17] Zhang, R., Seira H., and F. Koushanfar. (2022). "Text Revealer: Private Text Reconstruction via Model Inversion Attacks Against Transformers."
[18] Shokri, R., M. Stronati, C. Song, and V. Shmatikov. (2017). "Membership Inference Attacks Against Machine Learning Models." In IEEE Symposium on Security and Privacy, 3–18.
[19] Song, C., and V. Shmatikov. (2019). "Auditing Data Provenance in Text-Generation Models." In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 196–206.
[20] Dowlin, N., Ran G., K. Laine, K. Lauter, M. Naehrig, and J. Wernsing. (2016). "CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy." MSR-TR-2016-3. Microsoft Research.
[21] Chabanne, H., Amaury W., J. Milgram, C. Morel, and E. Prouff. (2017). "Privacy-Preserving Classification on Deep Neural Network." IACR Cryptology ePrint Archive, 35.
[22] Ioffe, S., and C. Szegedy. (2015). "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift."
[23] Sanyal, A., M. J. Kusner, A. Gascón, and V. Kanade. (2018). "TAPAS: Tricks to Accelerate (Encrypted) Prediction as a Service."

[24] Chou, E., J. Beal, D. Levy, S. Yeung, A. Haque, and Li Fei-Fei. (2018). "Faster CryptoNets: Leveraging Sparsity for Real-World Encrypted Inference."
[25] Xu, R., James B. D. Joshi, and C. Li. (2019). "CryptoNN: Training Neural Networks over Encrypted Data." In 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), 1199–209.
[26] Ebel A., G. Karthik, and R. Brandon. (2025). "Orion: A Fully Homomorphic Encryption Framework for Deep Learning." In Proceedings of the 30th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, 734–49. ASPLOS '25. New York, NY, USA: Association for Computing Machinery. doi:10.1145/3676641.3716008.
[27] Karthik, G., A. Ebel, G. Micheli, and B. Reagen. (2025). "HE-LRM: Encrypted Deep Learning Recommendation Models Using Fully Homomorphic Encryption."
[28] Njungle, N. B., and M. A. Kinsy. (2025). "Activate Me! : Designing Efficient Activation Functions for Privacy-Preserving Machine Learning with Fully Homomorphic Encryption." In Progress in Cryptology - AFRICACRYPT 2025, 51–73. Cham: Springer Nature Switzerland. doi:10.1007/978-3-031-97260-7_3.
[29] Boddeti, V., N. (2018). "Secure Face Matching Using Fully Homomorphic Encryption."
[30] Riazi, M., S., C., Weinert, O., Tkachenko, E. M. Songhori, T. Schneider, and Farinaz Koushanfar. (2018). "Chameleon: A Hybrid Secure Computation Framework for Machine Learning Applications." In ASIACCS '18: Proceedings of the 2018 ACM Asia Conference on Computer and Communications Security, 707–21.
[31] Wagh, S., D. Gupta, and N. Chandran. (2019). "SecureNN: 3-Party Secure Computation for Neural Network Training." Proceedings on Privacy Enhancing Technologies 2019(3):26–49.
[32] Juvekar, C., Vinod V., and A. Chandrakasan. (2018). "GAZELLE: A Low Latency Framework for Secure Neural Network Inference." In Proceedings of the 27th USENIX Security Symposium, 1651–69.
[33] Shancheng Z., Z. Zhang, G. Qu, and L. Yang. (2025). "Efficient and Secure Multi-Party Computation Protocol Supporting Deep Learning." Cybersecurity 8: Article 46. doi:10.1186/s42400-024-00343-4
[34] Lin, Ke, Y. Glani, and P. Luo. (2024). "Low-Latency Privacy-Preserving Deep Learning Design via Secure MPC." ArXiv abs/2407.18982. doi:10.48550/arXiv.2407.18982
[35] Mishra, P., R. T. Lehmkuhl, A. Srinivasan, W. Zheng, and R. Ada Popa. (2020). "DELPHI: A Cryptographic Inference Service for Neural Networks." In 29th USENIX Security Symposium (USENIX Security 20), 2505–22.
[36] Kumar, N., M. Rathee, N. Chandran, D. Gupta, A. Rastogi, and R. Sharma. (2020). "CrypTFlow: Secure TensorFlow Inference." In 2020 IEEE Symposium on Security and Privacy, 336–53.
[37] Byali, M., Harsh C., A. Patra, and A. Suresh. (2020). "FLASH: Fast and Robust Framework for Privacy-Preserving Machine Learning." In Privacy Enhancing Technologies Symposium.
[38] Koti, N., M. Pancholi, A. Patra, and A. Suresh. (2021). "SWIFT: Super-Fast and Robust Privacy-Preserving Machine Learning." In 30th USENIX Security Symposium.
[39] Hunt, T., C. Song, R. Shokri, V. Shmatikov, and Emmett Witchel. 2018. "Chiron: Privacy-Preserving Machine Learning as a Service."
[40] Tramèr, F., and D. Boneh. (2019). "Slalom: Fast, Verifiable and Private Execution of Neural Networks in Trusted Hardware." In International Conference on Learning Representations.
[41] Bulck, J., M. Minkin, O. Weisse, D. Genkin, B. Kasikci, F. Piessens, M. Silberstein, T. F. Wenisch, Y. Yarom, and R. Strackx. (2018). "Foreshadow: Extracting the Keys to the Intel SGX Kingdom with Transient Out-of-Order Execution." In Proceedings of the 27th USENIX Security Symposium (USENIX Security), 991–1008.
[42] Luo, C. et al. (2022). "A Differential Privacy-Based Classification System for Edge Computing in IoT".Computer Communications 182:117–28.10.1016/j.comcom.2021.10.038
[43] Hu, H., N. H. Phan, X. Wu, and D. Dou. (2018). "Adaptive Laplace Mechanism: Differential Privacy Preservation in Deep Learning."
[44] Abadi, M., A. Chu, Ian G., H. B. McMahan, I. Mironov, K. Talwar, and Li Zhang. (2016). "Deep Learning with Differential Privacy." In Proceedings of the 2016 ACM SIGSAC (CCS).
[45] Adesuyi, T. A., and B. M. Kim. (2019). "A Layer-Wise Perturbation Based Privacy-Preserving Deep Neural Networks." In International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 389–94. doi:10.1109/ICAIIC.2019.8669014
[46] Bu, Z., Juo D., Qi L., and Su. (2020). "Deep Learning with Gaussian Differential Privacy."

[47] Feng Ce, W. Wujie, Xu N., V. Parv, and D. Caiwen. (2023). "Spectral-DP: Differentially Private Deep Learning Through Spectral Perturbation and Filtering." In 2023 IEEE Symposium on Security and Privacy (SP), 1944–60. doi:10.1109/SP46215.2023.10179457
[48] Xiangfei, Z., Z. Qingchen, and J. Liming. (2025). "Layer-level Adaptive Gradient Perturbation Protecting Deep Learning Based on Differential Privacy." CAAI Transactions on Intelligence Technology 10 (April): 929–44. doi:10.1049/cit2.70008
[49] Papernot, N., M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar. (2017). "Semi-Supervised Knowledge Transfer for Deep Learning from Private Training Data."
[50] Dehghani, M., H. Azarbonyad, J. Kamps, and M. Rijke. (2017). "Share Your Model Instead of Your Data: Privacy Preserving Mimic Learning for Ranking."
[51] Bonawitz, K., V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, Sarvar P., D. Ramage, A. Segal, and K. Seth. (2017). "Practical Secure Aggregation for Privacy-Preserving Machine Learning." In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS), 1175–91. doi:10.1145/3133956.3133982
[52] McMahan, H. B., D. Ramage, K. Talwar, and Li Z. (2018). "Learning Differentially Private Recurrent Language Models." CoRR, arXiv Preprint arXiv:1710.06963.
[53] Mironov, Ilya, K. Talwar, and Li Zhang. (2019). "Renyi Differential Privacy of the Sampled Gaussian Mechanism." CoRR abs/1908.10530.
[54] Sun, L., J. Qian, X. Chen, and P. S. Yu. (2021). "LDP-FL: Practical Private Aggregation in Federated Learning with Local Differential Privacy." CoRR abs/2007.15789.
[55] Xu, H., S. Shrestha, W. Chen, Z. Li, and Z. Cai. (2025). "DP-FedLoRA: Privacy-Enhanced Federated Fine-Tuning for on-Device Large Language Models." CoRR, arXiv Preprint arXiv:2509.09097.
[56] Cai, Y., W. D., Y. Xiao, Z. Yan, X. Liu, and Z. Wan. (2024). "SecFed: A Secure and Efficient Federated Learning Based on Multi-Key Homomorphic Encryption." IEEE Transactions on Dependable and Secure Computing 21: 3817–33 doi:10.1109/TDSC.2023.3336977
[57] Park, J., and H. Lim. (2022). "Privacy-Preserving Federated Learning Using Homomorphic Encryption." Applied Sciences. doi;10.3390/app12020734.
[58] Ma, J., S. Naas, S. Sigg, and X. Lyu. (2022). "Privacy-Preserving Federated Learning Based on Multi-Key Homomorphic Encryption." International Journal of Intelligent Systems 37 (9): 5880–5901. doi:10.1002/int.22818.
[59] Awaysheh, Feda A. et al. (2023). "MPCFL: Towards Multi-Party Computation for Secure Federated Learning Aggregation." In Proceedings of the 16th IEEE/ACM International Conference on Utility and Cloud Computing (UCC), 321–28. doi:10.1145/3603166.3603204.
[60] Jin, W., Y. Yao, S. Han, J. Gu, C. Joe-Wong, S. Ravi, S. Avestimehr, and C. He. (2023). "FedML-HE: An Efficient Homomorphic-Encryption-Based Privacy-Preserving Federated Learning System." arXiv Preprint arXiv:2303.10837.
[61] Kalapaaking, A. P. et al. (2023). "Blockchain-Based Federated Learning with Secure Aggregation in Trusted Execution Environment for Internet-of-Things" IEEE Transactions on Industrial Informatics 19 (2): 1703–14.
[62] Valadi, V., M. Åkesson, J. Östman, S. Toor, and A. Hellander. (2025). "From Research to Reality: Feasibility of Gradient Inversion Attacks in Federated Learning."
[63] Gao, Y., Y. Xie, et al. (2025). "Gradient Inversion Attack in Federated Learning: Exposing Text Data Through Discrete Optimization." In Proceedings of COLING 2025.
[64] Zhu, L., Z. Liu, and S. Han. (2019). "Deep Leakage from Gradients" CoRR abs/1906.08935.
[65] Zhu, G., D. Li, H. Gu, et al. (2024). "FedMIA: An Effective Membership Inference Attack Exploiting 'All for One' Principle in Federated Learning."
[66] Bai, Li et al. (2024). "Membership Inference Attacks and Defenses in Federated Learning: A Survey."
[67] Yin, D., Y. Chen, K. Ramchandran, and P. L. Bartlett. (2018). "Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates." CoRR abs/1803.01498.
[68] Bagdasaryan, E., A. Veit, Y. Hua, D. Estrin, and V. Shmatikov. (2020). "How to Backdoor Federated Learning." In Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics, 108:2938–48. Proceedings of Machine Learning Research.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

[69] Haddaway, N. R., et al. (2020). "PRISMA 2020: An R package and Shiny app for producing PRISMA 2020-compliant flow diagrams, with interactivity for optimised digital transparency and Open Synthesis". Campbell Systematic Reviews, 18-2. doi:10.1002/cl2.1230.

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

APPENDIX A

TABLE 10. LIST OF SELECTED PAPERS

Paper	Author	Title	Publication Venue
		HOMOMORPHIC ENCRYPTION	
CryptoNet [21] 2016	Ran G.B. et al.	CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy	Proceedings of the 33rd International Conference on Machine Learning, PMLR 48:201-210
Chabanne [22] 2017	Hervé C. et al.	Privacy-preserving classification on deep neural network	IACR Cryptology ePrint Archive, Volume 2017:35
FaceMatch [30] 2018	Vishnu N. B.	Secure Face Matching Using Fully Homomorphic Encryption	CoRR, Computer Science, arXiv:1805.00577
Tapas [24] 2018	Adrià G. et al.	Tapas: Tricks to Accelerate (Encrypted) Prediction as a Service	CoRR, Computer Science, arXiv:1806.03461
Faster Crypto. [25] 2018	Edward C. et al.	Faster Cryptonets: Leveraging Sparsity for Real-world Encrypted Inference	Computer Science Cryptography and Security, arXiv:1811.09953
CryptoNN [26] 2019	J. Joshi et al.	CryptoNN: Training Neural Networks over Encrypted Data	39th International Conference on Distributed Computing Systems
He-Key [59] 2022	J. Ma et al.	Privacy-preserving federated learning based on multi-key homomorphic	International Journal of Intelligent Systems, doi :10.1002/int.22818
FedML-HE [61] 2023	W. Jin et al.	FedML-HE: An efficient homomorphic-encryption-based privacy-preserving federated learning system	CoRR, Computer Science, arXiv:2303.10837
Orion [27] 2025	Austin E. et al.	Orion: A Fully Homomorphic Encryption Framework for Deep Learning	Proceedings of the 30th ACM International Conference on ASPLOS
HE-LRM [28] 2025	Karthik G. et al.	HE-LRM: Encrypted Deep Learning Recommendation Models using Fully Homomorphic Encryption	Computer Science Cryptography and Security, arXiv:2506.18150
Activate Me! [29] 2025	Nges B. et al.	Activate Me!: Designing Efficient Activation Functions for Privacy-Preserving Machine Learning with Fully	Conference Paper on Progress in Cryptology - AFRICACRYPT 2025
		Secure-MPC	
SecureAgg [52] 2017	K. Bonawitz et al.	Practical Secure Aggregation for Privacy-Preserving Machine Learning	Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security
Chameleon [31] 2018	O. Tkachenko et al.	Chameleon: A Hybrid Secure Computation Framework for Machine Learning Applications	Proceedings in ACM Asia Conference on Information, Computer and Communications Security
SecureNN [32] 2019	Sameer W. et al.	SecureNN: 3-Party Secure Computation for Neural Network Training	Proceedings on Privacy Enhancing Technologies
Flash [38] 2020	Arpita P. et al.	FLASH: Fast and Robust Framework for Privacy-preserving Machine Learning	Privacy Enhancing Technologies Symposium (PETS)
Swift [39] 2021	Megha B. et al.	SWIFT: Super-fast and Robust PPML-Preserving Machine Learning	30th USENIX Security Symposium (USENIX Security 21)

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

MPC-FL [60] 2023	F. Awaysheh et al.	MPCFL: Towards Multi-party Computation for Secure Federated Learning Aggregation	Proceedings of the 16th IEEE/ ACM International Conference on Utility and Cloud Computing
L-Latency MPC [35] 2024	Kun Lin et al.	Low-Latency Privacy-Preserving Deep Learning Design via Secure MPC	CoRR, Computer Science, Cryptography and Security, 10.48550/arXiv.2407.18982
E.Shamir-MPC [34] 2025	Shanchen g Z. et al.	Efficient and secure multi-party computation protocol supporting deep learning	Cybersecurity Journal doi.org /10.1186/s42400-024-00343-4
		HYBRID	
Gazelle [33] 2018	Chiraag J. et al.	GAZELLE: A Low Latency Framework for Secure Neural Network Inference	Proceedings of the 27th USENIX Security Symposium
Delphi [36] 2020	P. Mishra et al.	DELPHI: A Cryptographic Inference Service for Neural Networks	Conference 29th USENIX Security Symposium (USENIX Security 20)
CryptFlow [37] 2020	Kumar N. et al.	Cryptflow: Secure Tensorflow Inference	IEEE Symposium on Security and Privacy
		SECURE ENCLAVES	
Chiron [40] 2018	T. Hunt et al.	Chiron: Privacy-preserving Machine Learning as a Service	CoRR, Computer Science Cryptography & Security, arXiv:1803.05961
Slalom [41] 2019	Tramer F. et al.	Slalom: Fast, Verifiable and Private Execution of Neural Networks in Trusted Hardware	International Conference on Learning Representations (ICLR)
TEE-FL [62] 2023	K. Pribadi et al.	Blockchain-based federated learning with secure aggregation in trusted execution environment for internet-of-things	IEEE Transactions on Industrial Informatics Journal
		DIFFERENTIAL PRIVACY	
DPSGD [45] 2016	M. Abadi et al.	Deep Learning with Differential Privacy	ACM SIGSAC Conference on Computer & Communications Security
Pate [50] 2017	Nicolas P. et al.	Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data	International Conference on Learning Representations
Dahghani [51] 2017	M. Dehghani et al.	Share your Model instead of your Data: Privacy Preserving Mimic Learning for Ranking	CoRR, Computer Science, Information Retrieval, arXiv:1707.07605
AdLM [44] 2018	Nhat H. Phan et al	Adaptive Laplace mechanism: Differential privacy preservation in deep learning	Computer Science IEEE International Conference on Data Mining
DP-Fed Avg [53] 2018	H. McMahan et al.	Learning Differentially Private Recurrent Language Models	CoRR, Computer Science, Machine Learning, arXiv:1710.06963
Adesuyi [46] 2019	A. Adesuyi	A layer-wise Perturbation based Privacy Preserving Deep Neural Networks	International Conference on AI in Information and Communication
GDP [47] 2020	Zhiqi Bu et al.	Deep Learning with Gaussian Differential Privacy	Harvard Data Science Review, 2(3), DOI: 10.1162/99608f92.cfc5dd25
ACIES [43] 2022	C. Luo et al.	A Differential Privacy-based Classification System for Edge Computing in IoT	Computer Communications, arXiv:182:117–128
Spectral-DP [48] 2023	F. Ce et al.	Spectral-DP: Differentially Private Deep Learning through Spectral Perturbation and Filtering	Proceeding 2023 IEEE Symposium on Security and Privacy (SP)

Journal of Information Systems Engineering and Management
2025, 10(60s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Layer-LA [49] 2025	Q. Zhang et al.	Layer-Level Adaptive Gradient Perturbation Protecting Deep Learning Based on Differential Privacy	CAAI Transactions on Intelligence Technology, 10.1049/cit2.70008
DP-Fed LoRA [56] 2025	Xu Honghui et al.	DP-FedLoRA: Privacy-Enhanced Federated Fine-Tuning for On-Device Large Language Models	CoRR, Computer Science, Cryptography and Security, arXiv:2509.09097