International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

A Review of Federated Learning: Privacy-Preserving Machine Learning

Authors: Rathod Neha , Mojidra kirtika, khandhediya Isha
Co-author: Harkishan sir
Institute: Gyanmanjari Innovative University

Abstract- Federated Learning (FL), was created by McMahan et al (14), has become of interest because it offers a decentralized machine learning framework for developing large scale ML models. This allows many users (or clients) to collaborate on training a shared model while retaining control of their own data. FL is ultimately designed to provide a solution to the conflict between the data demands of machine learning systems and the desire of individuals/companies to keep their personal and commercial data private. This paper is a review of the privacy and confidentiality aspects of Federated Learning. A critical review of the fundamental algorithms used in FL, possible attacks against FL systems, and the four primary techniques for enhancing privacy in FL; Differential Privacy (DP), Secure Multi-Party Computation (SMPC), Homomorphic Encryption (HE), and hardware based Trusted Execution Environments (TEE), is provided. We will review aggregation protocols, determine the strength of FL systems against poisoning and inference attacks, and compare various FL systems implemented in three industries; healthcare, mobile communication and finance. A detailed review of FL reveals research issues related to; statistical heterogeneity, communication overhead, system heterogeneity and fairness. Finally, this review presents a prioritized set of research objectives for the next ten years, with an emphasis on situating FL within the larger context of privacy-preserving ML and potential regulatory developments.

Keywords: Federated Learning's many advantages include: Privacy-preserving machine learning, Differential privacy, Secure multi-party computation, Homomorphic encryption, Byzantine robustness, Non-IID data, Communication efficiency, GDPR compliance, Edge AI .

I. INTRODUCTION

Traditionally, the machine learning pipeline has been based on the premise that the training dataset can be placed in a centralized location for collection from multiple sources and use in developing a machine learning model. This premise has been increasingly challenged. There are multiple data protection laws in place around the world that restrict the collection and sharing of data between cross-border jurisdictions (e.g., the European Union's General Data Protection Regulation [GDPR; 2018], California's California Consumer Privacy Act [CCPA], and India's Digital Personal Data Protection Act [2023]. Further, proprietary or regulatory restrictions have made it impossible to share data within certain domains (e.g., health, finance, telecommunications) despite the fact that there may be the ability to share data due to technological capability. The continuing rapid growth of edge computing has led to the proliferation of large amounts of data being stored on personal electronic devices and many of these devices are bandwidth limited and thus cannot upload raw data to the cloud.

The dilemma of how to train machine learning models using sensitive data without compromising the privacy of individuals was addressed by Federated Learning (FL), as proposed by McMahan et al. (14). In FL, a global model is distributed by a central coordination server to several clients who each train on their own private data locally and then send model updates (gradients or weights) back to the server for aggregation to build an improved global model. This iterative process repeats until convergence is reached, at which point there is a fully trained global model. Most importantly, raw training data is kept at its source and does not leave its original location.

Federated Averaging (FedAvg), the original FL algorithm, demonstrated the ability to train competitive language and image models on heterogeneous mobile data; this was

largely responsible for the rapid acceleration of FL research; there are now more than 10,000 FL papers published since 2017.

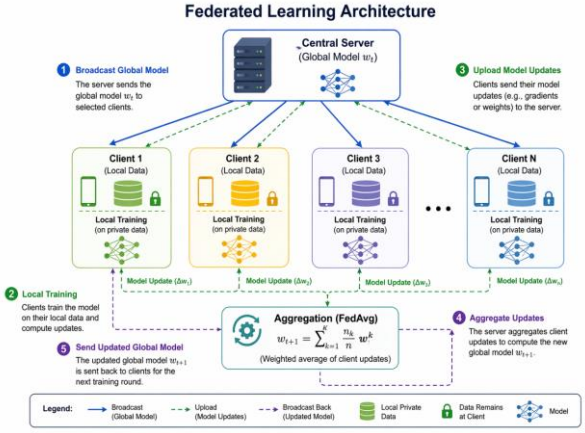
Simply sharing model updates as opposed to raw data does not provide adequate confidentiality. Various attack types, including gradient inversion [1] (Zhu et al. 2019) to access the gradients and obtain sensitive information, have shown that sharing gradient information can leak sensitive information about clients' data, including the ability to reconstruct very closely the training example. Other attacks include membership inference (Shokri et al. 2017), model inversion attacks. Thus, much work has been done at the intersection of federated learning (FL) and formal methods to develop privacy-preserving assurances (e.g., differential privacy (DP), secure multi-party computation (SMPC), homomorphic encryption (HE), and trusted execution environments (TEE)).

The present review serves as a technical synthesis of federated learning with privacy preservation. We organize our review into five dimensions — foundational algorithms and architectures, the privacy threat landscape, techniques for enhancing privacy, real-world applications and deployments, and open challenges. Our goal is to provide researchers, practitioners, and policymakers with a comprehensive overview of the current state of the art and an assessment of the outstanding problems associated with privacy and federated learning.

A. Contributions and Organisation

This review contributes to the field of FL by providing an overall picture for how local and remote FL architectures fit into an established taxonomy of FL architectures; by presenting a threat model that matches attack vectors to countermeasures for all privacy-preserving techniques; by providing a comparison of 7 deployed FL systems on 6 dimensions in a table; and by synthesizing 6 open research challenges with actionable research directions in a table.

The rest of this paper is structured as follows: section 2 discusses foundational principles of FL; section 3 discusses the landscape of privacy threats; section 4 discusses the types of privacy mechanisms; section 5 discusses types of aggregation protocols; section 6 gives examples of real-world deployments of FL; section 7 identifies cross-cutting challenges; section 8 discusses ethical and regulatory issues; section 9 presents a research agenda to address the above issues; and section 10 provides a conclusion for the review.



II. FOUNDATIONS OF FEDERATED LEARNING

Problem Formulation

Let K represent a fixed number of clients each having their own unique and disjoint local dataset, D_k. The objective of federated learning is to minimize a common global cost function, F(w) = sum_k [|D_k| / |D|] * F_k(w) , where F_k is the local cost on client k via client k's local dataset D_k, and |D| = sum_k |D_k| is the total volume of datasets on all clients in federated learning. In Traditional centralized forms of ML a single server/central machine receives the complete pool of all local datasets and minimizes F with the standard method of Stochastic Gradient Descent (SGD). In Federated Learning, however, the optimization process cannot occur on all clients without each client having access to their own dataset(s). McMahan et al demonstrate how Federated Averaging (FedAvg) in Federated Learning has clients perform multiple local SGD iterations prior to transmitting a shared weight update, which gives substantial benefits over naive forms of sharing gradients and requires considerably fewer round trips to the server to complete a global model update.

Taxonomy of FL Architectures

Three major configurations of Federated Learning have been identified (FL):
- Cross-device FL, which typically involves millions of resource-constrained clients (like smartphones, IoT devices, etc.) with small amounts of highly diverse local datasets, but suffers from a lack of available communication bandwidth and clients may not often be available. An example of a Cross-device FL deployment is Google's GBoard keyboard prediction system.

- Cross-silo FL, where there are tens to hundreds of institutional clients (for example, hospitals, banks, research consortia) with sizable, quality-curated datasets, but who are consistently online and can use significant computing power due to the tight data governance policy of their institution. Examples of Cross-silo FL applications include healthcare imaging, credit risk scoring, etc.

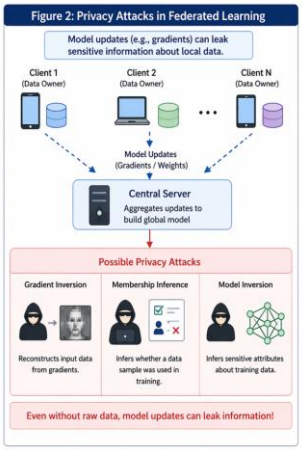
Vertical federated learning (split learning) allows several clients to have different features on the same set of individuals instead of having different samples from the same population. The clients create partial forward passes (through an appropriate model) and send only the intermediate representation instead of the gradient to each other. This type of arrangement between clients can often be found in finance, where a banking institution and a retail institution might want to train a common fraud detection model together, using the unique features of each client.

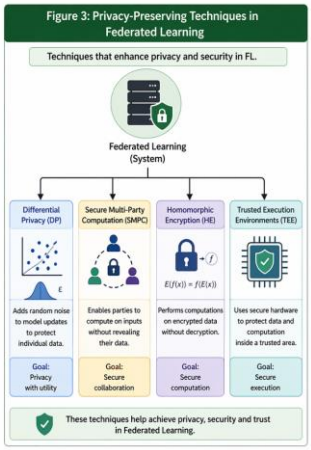
To illustrate the above point, we can look at a fourth dimension of the taxonomy — horizontal vs. vertical: horizontal federated learning assumes that the clients have different samples (or data records) but share the same feature space; vertical federated learning assumes that the clients have the same set of sample IDs but different features on those IDs. Federated transfer learning would apply to a situation where neither the clients share the same sample IDs nor do they share the same features on those sample IDs.

The FedAvg Algorithm and Its Variants

FedAvg (McMahan et al., 2017) is done in rounds. During each round, the server randomly selects a subset of C*K clients, sends out the current global model w_t, each selected client runs E epochs of local SGD and calculates w_t^k, and the server aggregates using w_{t+1} = sum_k (|D_k|/|D|) * w_t^k. The FedAvg algorithm will converge if the data is IID across clients, but will converge at a much slower rate or may diverge if the data is statistically heterogeneous (i.e., not IID).
Addressing these limitations are key variants. The proximal term added to the local objective of FedProx (Li et al., 2020) penalizes deviation from the global model, stabilizing convergence across heterogeneous data. Control variates are introduced for correction of client drift in SCAFFOLD (Karimireddy et al., 2020). MOON (Li et al., 2021) aligns local representations via contrastive learning. Local updates are normalized in FedNova to correct for different numbers of local steps taken by clients.

III. PRIVACY THREAT LANDSCAPE IN FEDERATED LEARNING





Attack Surface Overview

FL becomes less dependent on the distribution of unstructured data. However, the model updates shared between client devices and central server represent a unique threat vector. There are multiple types of passive and active threats associated with the risk that an attacker could either monitor or modify the model updates exchanged between client devices and the central server. As such, there will be multiple strategies for both passive and active attacks originating from a malicious server or client or through eavesdropping on the communication channel used to distribute model updates.

Inference Attacks

Gradient inversion attacks ([20] [9]) demonstrated that shared gradients from a single training step can be inverted to reconstruct training images with pixel-level accuracy for small batch sizes. The R-GAP attack (Zhu and Blaschko, 2021) extended this to deep networks by exploiting the recursive structure of back-propagated gradients. Membership inference attacks (Shokri et al., 2017) determine, with above-chance accuracy, whether a given data record was part of a client's training set by querying the global model's confidence outputs. Property inference attacks reconstruct global statistics of the training data — e.g., the fraction of a hospital's patients with a particular diagnosis — from shared model parameters.

Poisoning Attacks

Take for example attacks on the model through alterations made by malicious clients uploading specifically crafted updates to influence global behaviour of the model in a certain way: normally this will result in the model misclassifying inputs containing some kind of trigger(s) with respect to how it would normally behave. Backdoor attacks are especially stealthy in federated learning because the server cannot see the data used by clients to train their models; thus, if one malicious client is able to create an update which dominates the aggregate created at the server (due to no robust aggregation rule), then the impact on the overall model will also be significant. Data poisoning attacks corrupt a client's local dataset (as opposed to modifying directly), making them more difficult to detect.

IV. PRIVACY-PRESERVING MECHANISMS

Federated learning (FL) often utilises several mechanisms to preserve clients' privacy, while also providing clients with an output for use during the FL model training. The structures of the privacy-preserving mechanisms are compared according to their formal guarantees of privacy (where applicable), computationally related trade-offs or costs, and the most appropriate contexts for deployment. Summarised in Table 2 are the four main privacy-preserving mechanisms found in FL systems.

Table 2. Privacy-Preservation Mechanisms in Federated Learning.

Technique	Core Mechanism	Privacy Guarantee	Main Trade-off	Overhead	Best Fit
Differential Privacy	Gaussian / Laplace noise injection	Formal (epsilon, delta)-DP	Accuracy vs. privacy budget	Low	Cross-device FL
Secure Aggregation	Secret-sharing over gradients	Information-theoretic	Communication cost	Medium	Small silo FL
Homomorphic Encrypt.	Computation on encrypted ciphertext	Computational security	Very high compute cost	High	Low-freq. updates
Trusted Exec. Env.	Hardware enclave (SGX / TrustZone)	Platform-level isolation	Hardware dependency	Medium	Server-side aggr.
SMPC	Garbled circuits / secret shares	Information-theoretic (honest)	Communication rounds	High	Cross-silo FL

Differential Privacy

One of the most widely recognised definitions of privacy is provided by differential privacy (DP), which has been developed since its inception in the early 2000s. A randomised mechanism M satisfies (epsilon, delta)-DP if for any two adjacent datasets (D and D') differing in one

record and for any output set S, Pr[M(D) in S] < e^epsilon P[M(D') in S] + delta (Dwork et al., 2006). In FL systems, differential privacy can be implemented using the differential privacy via stochastic gradient descent (DP-SGD) algorithm (Abadi et al., 2016), where the local client gradients are clipped to a maximum L2 norm C, and calibrated Gaussian noise is added to the clipped gradients and scaled according to C/batch size before they are uploaded to the FL server. The privacy loss tracked by the privacy budget epsilon is the sum cumulative privacy loss across all rounds using either the moments accountant or Renyi difference privacy (DP) composition techniques.

The server uses aggregated (global) noise in Centralised DP (C-DP), which creates less secure per-client (client) guarantees but increases overall utility for all clients. In the case of Local DP (L-DP), the client sends their data with added local noise, which provides all clients with their own per-client guarantees; however, the amount of local noise each client has to add to their data is much larger than the global noise that is added to the aggregated record, and thus decreases the accuracy of the model. User-level DP protects the entire set of user information from appended, which also requires larger amounts of noise, and is the most relevant for mobile FL([15]) showed the application of DP through Google Gboard (GooG), where they applied C-DP and achieved an overall e-8 at user level with competitive accuracy for next-word prediction.

Secure Multi-Party Computation and Secure Aggregation

While Secure Aggregation (Moffie et al., 2020) uses cryptographic secret-sharing to create a communication-efficient method to hide client updates from servers, its performance degrades substantially when large models are involved since the cost of communication for SecAgg rises with the square of the number of clients per round.

Most existing secure multi-party computation (SMPC) protocols like garbled circuits (Yao, 1982) or SPDZ () provide the ability to compute on secret-shared inputs but are designed for cross-silo settings. where only a few participants are involved and remain connected. WeBank's FATE (Federated AI Technology Enabler), which implements SMPC-based vertical Federated Learning (FL), enables multiple banks to conduct joint credit scoring without either exposing its customer's input features to any other bank.

Homomorphic Encryption

HE allows calculations to be made while keeping data hidden (in encrypted form) so that when the calculations are complete they can be shared (in decrypted form) and be equal to the original data that was used to generate the calculations. PHE supports one function of either addition or multiplication, but the BGV and CKKS schemes support both functions (FHE) therefore both types of FHE can be used in Federated Learning. In Federated Learning, HE can provide clients with encrypted versions of their gradient updates that a server can aggregate without decrypting first so that only the clients can decrypt them using their decryption keys, or a trusted third-party holds their decryption keys. As a result of these properties, the clients will have a very high level of seclusion from their server because the server has a limited amount of access to the decrypted, raw training data. However, it is necessary to note that due to the fact that you will have to encrypt and aggregate the gradients of large, deep learning networks it is going to be orders of magnitude slower than using plaintext, therefore Federated Learning will be limited to only shallow networks or periodic updates at best.

Trusted Execution Environments

Hardware-enforced isolated execution enclaves protect code and data from being accessed by the underlying operating system or hypervisor via trusted execution environments (TEEs) such as Intel SGX and ARM TrustZone. TEEs can be used to protect aggregation logic from the server (that is, a compromised OS cannot read the content of the enclave) in FL or model execution on client devices. The privacy guarantee associated with TEE-enabled FL is at the platform level (rather than being based on mathematically formal proofs) and relies upon the trustworthiness of the hardware vendor and the absence of any known side-channel attacks. NVIDIA FLARE has implemented TEE-based FL for privacy-preserving medical image analysis across multiple hospital consortia.

V. AGGREGATION PROTOCOLS AND BYZANTINE ROBUSTNESS

Standard Aggregation

FedAvg uses weighted averages of the client model weight with the weight being dependent on the size of the client's local dataset to produce an efficient average that approximates a centralized model when the dataset is IID, but because of the arbitrary nature of the Byzantine client, this method is vulnerable to poisoning attacks whereby even one Byzantine client can move the aggregated model to any point in the parameter space, even if there are 100 other honest clients providing honest updates.

Robust Aggregation Rules

Robust aggregation rules replace the use of mean statistics with statistics that are more resistant to outliers. Krum uses the k nearest-neighbor similarity method to find the point that is most similar to k of the nearest client updates and rejects any of the outlier client updates as part of the aggregation. In addition, the coordinate-wise median and trimmed mean are both created by replacing the extremely high or low values with the closest dimension-by dimension average during aggregation. FLTrust assigns a trust score for each client update based on how similar the client's cosine similarity is to the reference update created by the server; this means that decreases in trust will correspond to less weight given to the client update when creating the average. Flame combines both clustering and noise injection in order to protect against backdoor attacks while still providing utility to the user. Each of the above aggregation rules creates a tradeoff between Byzantine fault tolerance and the accuracy of the average when working with heterogeneous non-IID datasets; as such, clients that are honest but have a very different distribution than the average client in the federation may be wrongly identified as malicious.

Table 2. Comparison of Major Federated Learning Frameworks and Deployments

Framework	FL Type	Aggregation	Privacy Tech	Application	Year
FedAvg	Cross-device	FedAvg SGD	None (baseline)	Mobile NLP	2017
Google Gboard	Cross-device	FedAvg + DP	DP-SGD	Keyboard predict	2019
PySyft / OpenFL	Cross-silo	SecAgg	SMPC	Healthcare	2020
FATE	Cross-silo	Vertical FL	HE + SMPC	Finance	2019
Flower (Flwr)	Cross-device	Flexible	Pluggable	Research / Edge	2020
TensorFlow Fed.	Cross-device	FedAvg / FedProx	DP-SGD	On-device ML	2019
NVIDIA FLARE	Cross-silo	FedProx / SCAFFOLD	DP + TEE	Medical Imaging	2021

VI. REAL-WORLD DEPLOYMENTS AND APPLICATION DOMAINS

Table 1 provides a structured comparison of seven major FL frameworks and deployments. We discuss three domains in depth.

Mobile and On-Device Learning

Google's provision of Federated Learning (FL) for next word prediction via Gboard ([10]) is the largest and most referenced real-world example. FL uses a process called federation and it performs training on millions of Android phones via Wi-Fi overnight as users charge their phones. FedAvg (Federated Average) using DP-SGD (differential

private Stochastic Gradient Descent) yields an approximate User level of ε 8 (epsilon) and FL with secure aggregation will significantly improve recall when working with an A/B test results of the Federated model compared to a model which has been trained centrally or out-of-band. Apple uses similar forms of FL for Predicting Emoji, Keyboard Personalizing and Siri Intent Classification using both on-device learning (i.e. no need for Internet connectivity) in addition to using DP prior to any gradient exiting the device.

Healthcare and Medical Imaging

FL (federated learning) is considered to be one of the most influential industries regarding the implementation of federated learning systems. For example, Rieke et al. (2020) demonstrated the results of performing FL on 10 hospitals to achieve the performance of a model trained using all of the hospital's data via a centralised approach for brain tumour segmentation, while keeping the patient scans on-site. Additionally, NVIDIA FLARE ([18]) has been used in the Intel-Penn Medicine partnership for COVID-19 lung lesion segmentation and in the NIH-funded consortium for federated genomic analyses. Finally, the FeTS Challenge (2021) successfully combined FL across 23 countries for brain tumour segmentation, further confirming that FL can be utilized by multiple institutions to derive clinical AI solutions

Financial Services.

Another example of FL application across different siloed data sources is within the financial services sector, as data-sharing regulations prevent institutional cross-sharing of data; however, cross-institutional collaboration would provide significant benefits in regards to fraud detection, credit scoring, and anti-money-laundering model development. WeBank's FATE platform provides vertical FL between banks and insurance companies and employs SMPC and HE to protect the sensitive customer feature data while jointly training machine learning models. Given that a joint credit scoring model trained via FATE on data from a bank and retailer generated an approximately 12 percent relative AUC improvement compared to models trained solely on each entity's data (13), the significant value proposition associated with privacy-preserving collaboration is clearly evident.

VII. CROSS-CUTTING CHALLENGES

W. Statistical Heterogeneity (Non-IID Data)

The assumption that client data can be considered as independent, identically distributed (IID) data is often violated in practice. For example, the keyboard FL data collected from a given client reflects the idiosyncrasy of that user's language. Similarly, imaging data for hospitals will depend upon the type of patient population that the hospital services, the types of scanners that the facility uses to collect imaging data, and the hospital's clinical protocols for collecting, storing, and managing the data. Typically, non-IID client data will cause the local model trained on the client's data to specialize to the distribution of the client's data; therefore, the average of all local FL models will not correspond to any good solution for any particular client. Zhao et al. (2018) also found that the accuracy of FL was reduced by as much as 55 percent in a highly skewed non-IID scenario as opposed to IID baselines. One approach to personalizing FL is to develop FL models for each of the clients, rather than a single FL model representing the average performance of all clients. This can be accomplished through using per-client adaptations learned through MAML-based meta learning, fine-tuning, or mixture-of-experts techniques.

Communication Efficiency

It is not practical to upload full model gradients every round in cross-device federated learning for large models and bandwidth-limited devices. A ResNet-50 model has over twenty-five million parameters so sending a copy of all 32-bit floats associated with those parameters will take 100 MB to send each round from each client. Gradient compression techniques – such as top-k sparsification (transmitting only the k largest-gradient magnitudes), quantising (reducing precision from 32 bits to 8 or 4), and using error feedback (accumulating compression errors across rounds) – can help by reducing communication costs by 100x to 1000x while maintaining reasonable accuracy. Other approaches, like one shot FL, as well as knowledge distillation-based protocols, aim to reduce the number of rounds of communication needed to just one or a small constant number of rounds.

System Heterogeneity and Stragglers

Devices that are clients in a cross-device federated learning system can differ significantly from each other with respect to their compute capability, memory, battery life and network bandwidth. Asynchronous federated learning protocols aggregate (average) the updates as they are received instead of waiting for all of the clients that were selected to provide their updates before proceeding to the next round of computation and this creates a straggler effect, where the round cannot be completed until the slowest client provides their updates. Asynchronous federated learning protocols also use some form of staleness penalty to discount the updates to the federated model from clients that are still training on outdated versions of the federated model. The proximal term in FedProx enables clients to still contribute an update to the federated model after providing less than E local epochs of training (although any contribution will likely be less accurate).

Fairness Across Clients

Standard FedAvg maximizes average performance as calculated from the total size of each client's dataset. For example, if a global FL model is trained using data from 100 different hospitals, it can reach an average segmentation accuracy that is outstanding; however, when the periphery to a smaller rural hospital (in the same region) does not have a patient population that is representative of the model, then an automatic result will be that this smaller hospital will consistently produce poor segmentation results. One of the advantages of using the Hybrid per-char and q-FedAvg method of federated learning ([12]) is that they integrate a second, fairness, objective into the computation of model weights, by increasing the penalty to high-loss clients; therefore, the model will change weights more quickly for high-loss clients than it will for low-loss clients. On the other hand, the Agnostic Method ([15]) does not seek to produce average performance above all else; rather, it looks at the worst-case performance across all clients and guarantees minimax fairness, albeit at the expense of average performance.

Federated learning is frequently described as a technological solution to address privacy concerns; however, the relationship between federated learning and regulatory requirements is more complex than many popular accounts describe. For example, the GDPR provides broad definitions of personal data, and model parameters and gradients do not receive a blanket exemption from this definition. As an example, a gradient update may potentially be reversed and used to reconstruct the original training image; in this case, the gradient update would meet the definition of personal data under the GDPR. Additionally, there has been no definitive guidance issued, from either the Article 29 Working Party or the European Data Protection Board, regarding the use of federated learning, leading to continued legal uncertainty for compliance officers.

The artificial intelligence (AI) act set forth by the European Union (EU) establishes new laws that require companies to adhere to additional requirements for high-risk (HR) AI products that may be used in different areas such as healthcare, human resources (or employment), and infrastructure. The EU AI act will therefore have a direct impact on the use of federated learning (FL) applications in these markets. For HR applications developed using FL, AI applications will require maintaining logs of data/transactions, establishing human oversight and an assessment of conformity to the applicable laws. In order for FL infrastructure to comply with the EU AI act, FL infrastructures must be clearly established as the controlling entity for the audit of a particular data set and the tracking of data lineage associated with a particular data set across many different data sources.

Apart from compliance with the law, FL also has ethical challenges concerning the power imbalance that exists between the central coordination of the FL ecosystem and the clients (the clients are the people that own the device). For example, the central provider of the FL model (the device manufacturer) will have ultimate control of the model goals, aggregation logic and privacy budget; therefore, clients of the device will have limited awareness of what is occurring with respect to their own data or how their data will be used with respect to the model that will be created. Therefore, the question of how to provide clients with informed consent for participating in an FL ecosystem is an open research question.

IX. RESEARCH AGENDA AND OPEN PROBLEMS

Table 3 maps six open challenges to their current state and recommended research directions. We expand on the three most pressing below.

Table 3. Open Research Challenges in Federated Learning and Recommended Directions.

Challenge	Current State	Recommended Research Direction
Non-IID Data	Global model accuracy degrades severely on heterogeneous clients	Personalised FL (pFL); clustered FL; meta-learning approaches
Communication Cost	Naive FL requires many rounds of large gradient uploads	Gradient compression; event-driven updates; one-shot FL
Byzantine Robustness	Poisoning attacks can corrupt global model with few malicious clients	Robust aggregation (Krum, Median, FLTrust); anomaly detection
Privacy-Utility Gap	DP noise and encryption reduce model accuracy substantially	Adaptive noise scheduling; composition theorems; HE acceleration
Fairness	Global model performs unequally across client subpopulations	Fairness-aware aggregation; agnostic FL; multi-objective optimisation
Regulatory Compliance	GDPR / AI Act obligations are ambiguous for FL pipelines	Legal FL frameworks; auditable aggregation; data lineage tracking

Unifying Privacy, Robustness, and Utility

Federated Learning has a real balancing act: keeping data priv33ate, staying tough against attacks, and still getting solid model performance. Take Differential Privacy, for example—it protects privacy by adding noise, but that usually means the model loses some accuracy. Robust aggregation algorithms, built to weed out malicious clients, sometimes end up punishing honest ones too, especially when everyone's data looks so different. What's needed are smarter, all-in-one solutions that juggle privacy, security, and performance without dropping any of those balls—and back it all up with strong theoretical guarantees.

Scalable Privacy-Preserving Technologies

On the security side, methods like Homomorphic Encryption and Secure Multi-Party Computation offer powerful protection. But let's be honest: they're hungry for resources, both in terms of computation and communication. That makes them tough to deploy when you're dealing with big networks or need real-time results. The way forward? Research should lean into hardware acceleration—think GPUs and FPGAs—as well as look at lighter encryption tricks and hybrid methods that blend multiple privacy tools. That's how we can push these systems to scale up and stay fast without sacrificing privacy or security.

Federated Learning for Foundation Models

There are many new challenges posed by using FL with Large Foundation Models (e.g., Large Language Models and Vision Transformers). The sheer volume of information contained in Large Foundation Models necessitates a huge amount of bandwidth to communicate between devices, rendering classic FL approaches very ineffective. To mitigate the communication cost, there are parameter-efficient techniques such as LoRA, Adapters and Prompt Tuning, among others. However, the significant challenges of privacy leakage, model inversion attacks, and secure aggregation have generally not been resolved. Further studying scalable and secure FL methods which are tailored for foundation models is essential.

X. CONCLUSION

Federated Learning (FL) has gone from being just an area of academic study to having high level, real-world use... the models used to train machine learning algorithms to determine how FL can be effectively and securely deployed across multi-user devices like smartphones, tablets, computer systems (lab & cloud) are: Differential Privacy, Secure Aggregation, Homomorphic Encryption, and Trusted Execution Environments. These are privacy-centric methodologies with the goal of improving privacy and security in FL.

A number of issues still exist with FL, including a disconnect between theoretical privacy protections as established in the literature and actual privacy protections or implementations that are used in real life; additional problems posed by gradient leakage attacks, dealing withNon-IID data, constraints on communications, and ambiguities with regard to liability and regulatory compliance.

Researchers should concentrate on the following target areas: establishing a balance between privacy, robustness and accuracy; increasing the scalability of privacy techniques; making FL suitable for training artificial intelligence systems using large foundation models; and ensuring that FL is safe for participants and complies with all applicable laws and regulations.

In conclusion, Federated Learning has tremendous potential to be used as a method of establishing a safer and more robust way to collaborate across multiple parties, leveraging their knowledge and experience in order to develop and deliver products and services, particularly within critical areas such as health care, finance, and public services.

REFERENCES

1. Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (pp. 308-318). ACM.
2. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). How to backdoor federated learning. Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (pp. 2938-2948). PMLR.
3. Bhagoji, A. N., Chakraborty, S., Mittal, P., & Calo, S. (2019). Analyzing federated learning through an adversarial lens. Proceedings of the 36th International Conference on Machine Learning (pp. 634-643). PMLR.
4. Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in Neural Information Processing Systems, 30. Curran Associates.
5. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., & Seth, K. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1175-1191). ACM.
6. Cao, X., Fang, M., Liu, J., & Gong, N. Z. (2020). FLTrust: Byzantine-robust federated learning via trust bootstrapping. Proceedings of the 28th Network and Distributed System Security Symposium. Internet Society.
7. Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. Proceedings of the 3rd Theory of Cryptography Conference (pp. 265 284). Springer.
8. European Parliament and Council. (2024). Regulation (EU) 2024/1689 on Artificial Intelligence (Artificial

Intelligence Act). Official Journal of the European Union. P a ge 14 | 15

9. Geiping, J., Bauermeister, H., Dröge, H., & Moeller, M. (2020). Inverting gradients: How easy is it to break privacy in federated learning? Advances in Neural Information Processing Systems, 33, 16937-16947. Curran Associates.
10. Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., & Ramage, D. (2018). Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604.
11. Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., & Suresh, A. T. (2020). SCAFFOLD: Stochastic controlled averaging for federated learning. Proceedings of the 37th International Conference on Machine Learning (pp. 5132-5143). PMLR.
12. Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems, 2, 429-450.
13. Liu, Y., Chen, T., & Yang, Q. (2019). Secure federated transfer learning. arXiv preprint arXiv:1812.03337.
14. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (pp. 1273-1282). PMLR.
15. McMahan, H. B., Ramage, D., Talwar, K., & Zhang, L. (2018). Learning differentially private recurrent language models. Proceedings of the International Conference on Learning Representations. OpenReview.
16. Nguyen, T. D., Rieger, P., De Viti, R., Chen, H., Brandenburg, B. B., Yalame, H., ... & Picek, S. (2022). FLAME: Taming backdoors in federated learning. Proceedings of the 31st USENIX Security Symposium (pp. 1415-1432). USENIX.
17. Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., ... & Cardoso, M. J. (2020). The future of digital health with federated learning. NPJ Digital Medicine, 3(1), 119.
18. Roth, H. R., Yang, D., Xu, Z., Wang, X., & Xu, D. (2021). Federated learning for breast density classification: A real-world implementation. Domain Adaptation and Representation Transfer (pp. 181-191). Springer.
19. Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against machine learning models. Proceedings of the 2017 IEEE Symposium on Security and Privacy (pp. 3-18). IEEE.
20. Zhu, L., Liu, Z., & Han, S. (2019). Deep leakage from gradients. Advances in Neural Information Processing Systems, 32. Curran Associates.