applied sciences

MDPI

Review

Synergizing Intelligence and Privacy: A Review of Integrating Internet of Things, Large Language Models, and Federated Learning in Advanced Networked Systems

Hongming Yang [1,†], Hao Liu [1,†], Xin Yuan [2], Kai Wu [1,3], Wei Ni [2,*], J. Andrew Zhang [1,3] and Ren Ping Liu [1,3]

[1] School of Electrical and Data Engineering (SEDE), University of Technology Sydney (UTS), Sydney, NSW 2007, Australia; hongming.yang@student.uts.edu.au (H.Y.); hao.liu-13@student.uts.edu.au (H.L.); kai.wu@uts.edu.au (K.W.); andrew.zhang@uts.edu.au (J.A.Z.); renping.liu@uts.edu.au (R.P.L.)
[2] Data61, Commonwealth Scientific and Industrial Research Organization, Marsfield, Sydney, NSW 2122, Australia; xin.yuan@ieee.org
[3] Global Big Data Technologies Centre (GBDTC), University of Technology Sydney (UTS), Sydney, NSW 2007, Australia
* Correspondence: wei.ni@ieee.org
† These authors contributed equally to this work.

Abstract: Bringing together the Internet of Things (IoT), LLMs, and Federated Learning (FL) offers exciting possibilities, creating a synergy to build smarter, privacy-preserving distributed systems. This review explores the merging of these technologies, particularly within edge computing environments. We examine current architectures and practical methods enabling this fusion, such as efficient low-rank adaptation (LoRA) for fine-tuning large models and memory-efficient Split Federated Learning (SFL) for collaborative edge training. However, this integration faces significant hurdles: the resource limitations of IoT devices, unreliable network communication, data heterogeneity, diverse security threats, fairness considerations, and regulatory demands. While other surveys cover pairwise combinations, this review distinctively analyzes the three-way synergy, highlighting how IoT, LLMs, and FL working in concert unlock capabilities unattainable otherwise. Our analysis compares various strategies proposed to tackle these issues (e.g., federated vs. centralized, SFL vs. standard FL, DP vs. cryptographic privacy), outlining their practical trade-offs. We showcase real-world progress and potential applications in domains like Industrial IoT and smart cities, considering both opportunities and limitations. Finally, this review identifies critical open questions and promising future research paths, including ultra-lightweight models, robust algorithms for heterogeneity, machine unlearning, standardized benchmarks, novel FL paradigms, and next-generation security. Addressing these areas is essential for responsibly harnessing this powerful technological blend.

Keywords: Internet of Things (IoT); Large Language Model (LLM); Federated Learning (FL); privacy-preserving techniques (PETs); edge computing; Parameter-Efficient Fine-Tuning (PEFT); Split Federated Learning (SFL); data heterogeneity; network security; distributed systems

Academic Editors: Rui Pinto, Pedro M. B. Torres and Volker Lohweg

Received: 23 April 2025
Revised: 9 June 2025
Accepted: 10 June 2025
Published: 11 June 2025

Citation: Yang, H.; Liu, H.; Yuan, X.; Wu, K.; Ni, W.; Zhang, J.A.; Liu, R.P. Synergizing Intelligence and Privacy: A Review of Integrating Internet of Things, Large Language Models, and Federated Learning in Advanced Networked Systems. *Appl. Sci.* **2025**, *15*, 6587. https://doi.org/10.3390/app15126587

1. Introduction

Internet of Things (IoT) and Artificial Intelligence (AI) are reshaping the way we live. IoT is penetrating every aspect of our modern society. It features the explosion of interconnected devices generating vast amounts of real-world data, driving significant and innovative insights to improve our lives. Simultaneously, emerging LLMs like the GPT series have shown a remarkable ability to understand and process complex information [1,2].

The power of Large Language Models (LLMs) arises from a vast amount of training data, while IoT systems are excellent means to provide such data. Combining the two fields is a natural move. This, however, incurs significant challenges. A core question is how we can leverage the intelligence of resource-hungry LLMs to make sense of the massive, diverse, and often sensitive data streams produced by countless IoT devices, especially when the data is mostly heterogeneous, multimodality, high-dimensional, sparse, and needs to be processed quickly and, in many cases, locally [3–5]. This is further elaborated on below.

Sending huge volumes of IoT data to a central cloud for AI analysis is often not practical [6]. It can be too slow for applications needing real-time responses (like industrial control or autonomous systems), consumes too much bandwidth, and raises significant privacy concerns [7]. Many critical IoT applications simply demand intelligence closer to the data source [3]. On the other hand, while LLMs possess the analytical power needed for complex IoT tasks, they face their own hurdles: they require massive datasets for training, and accessing the rich, real-world, but often private, data held on distributed IoT devices is difficult [8]. Moreover, deploying these powerful models effectively within the constraints of real-world distributed systems like IoT remains a significant challenge, considering limited hardware resources and power supply, data access, and privacy. This is precisely where Federated Learning (FL) enters the picture [9]. FL revolutionizes traditional approaches by enabling collaborative model training across decentralized data sources, eliminating the need for raw data centralization. This creates a compelling opportunity: using FL to train powerful LLMs on diverse, distributed IoT data while preserving user privacy and data locality [10,11]. This combination promises smarter, more responsive, and privacy-respecting systems, potentially leading to more efficient factories, safer autonomous vehicles, or more personalized healthcare, all leveraging local data securely. However, integrating these three sophisticated technologies (IoT, LLMs, FL) creates unique complexities and challenges related to efficiency, security, fairness, and scalability [12]. Given the significance of the integration and the increasing attention it has gained recently, this review aims to provide a timely overview of the state of the art in synergizing IoT, LLMs, and FL, particularly for edge environments, hoping to highlight current capabilities, identify key challenges, and inspire future research directions that enable intelligent, privacy-preserving, and resource-efficient edge intelligence systems. Specifically, we will explore the architectures, methods, inherent challenges, and promising solutions, highlighting why this three-way integration is crucial for building the next generation of intelligent, distributed systems.

The burgeoning interest in deploying advanced AI models like LLMs within distributed environments like IoT, often facilitated by techniques such as FL and edge computing, has spurred a number of valuable survey papers. While these reviews provide essential insights, they typically focus on specific sub-domains or pairwise interactions. Some representative survey works are reviewed below. Table 1 summarizes their primary focus and key differentiating aspects alongside our current work.

- Qu et al. [13] focus on how mobile edge intelligence (MEI) infrastructure can support the deployment (caching, delivery, training, inference) of LLMs, emphasizing resource efficiency in mobile networks. Their core contribution lies in detailing MEI mechanisms specifically tailored for LLMs, especially in caching and delivery, within 6G.
- Adam et al. [14] provide a comprehensive overview of FL applied to the broad domain of IoT, covering FL fundamentals, diverse IoT applications (healthcare, smart cities, autonomous driving), architectures (CFL, HFL, DFL), a detailed FL-IoT taxonomy, and challenges like heterogeneity and resource constraints. LLMs are treated as an emerging FL trend within the IoT ecosystem.

- Friha et al. [15] examine the integration of LLMs as a core component of edge intelligence (EI), detailing architectures, optimization strategies (e.g., compression, caching), applications (driving, software engineering, healthcare, etc.), and offering an extensive analysis of the security and trustworthiness aspects specific to deploying LLMs at the edge.
- Cheng et al. [10] specifically target the intersection of FL and LLMs, providing an exhaustive review of motivations, methodologies (pre-training, fine-tuning, Parameter-Efficient Fine-Tuning (PEFT), backpropagation-free), privacy (DP, HE, SMPC), and robustness (Byzantine, poisoning, prompt attacks) within the "Federated LLM" paradigm, largely independent of the specific application domain (like IoT) or deployment infrastructure (like MEI).

Table 1. Comparison with related surveys.

Survey	Primary Focus	Key Strengths	Distinction from Our Work
Qu et al. [13]	MEI supporting LLMs	Deep dive into edge resource optimization (compute, comms, storage); mobile network (6G); detailed edge caching/delivery for LLMs.	Focuses on infrastructure for LLMs; less depth on FL specifics, security/trust, or the unique synergy of IoT + LLM + FL. Less emphasis on IoT data characteristics.
Adam et al. [14]	FL for IoT applications	Comprehensive FL principles in IoT context; detailed IoT application case studies; broad FL taxonomy for IoT.	IoT application-driven; LLMs are only one emerging aspect; less depth on LLM specifics or the challenges arising from the three-way synergy.
Friha et al. [15]	LLMs integrated into EI	Deep analysis of security and trustworthiness for LLM-based EI; covers architectures, optimization, autonomy, applications broadly.	Focuses on LLM as EI component; less depth on FL methods specifically for training LLMs on distributed IoT data.
Cheng et al. [10]	Federated LLMs (FL + LLM)	Exhaustive review of FL methods for LLMs (PEFT, init, etc.); deep dive into privacy/robustness specific to federated LLMs.	Focuses narrowly on FL+LLM interaction; less emphasis on the specific IoT context (data types, device constraints) or the edge infrastructure aspects.
This Survey	Synergy of IoT + LLM + FL for privacy-preserving edge intelligence	Unique focus on the three-way interaction; explicit analysis of synergistic effects (Section 5); addresses challenges arising specifically from the integration; compares trade-offs in the specific IoT + LLM + FL@Edge context.	Provides a holistic view of the integration, bridging gaps between surveys focused on pairwise interactions or single components. Emphasizes the unique capabilities, privacy considerations, and challenges born from the specific combination of IoT data richness, LLM intelligence, and FL's distributed privacy paradigm within advanced edge networks.

While prior reviews cover areas like edge resources for LLMs [13], FL for IoT [14], edge LLM security [15], or federated LLM methods [10], they mainly look at pairs of these technologies. This survey distinctively examines the combined power and challenges of

integrating all three, including IoT, LLMs, and FL, particularly for privacy-focused intelligence at the network edge. This synergy is depicted in Figure 1. It illustrates a conceptual framework in which synergistic AI solutions emerge from the integration of IoT, LLMs, FL, and privacy-preserving techniques (PETs). Each component contributes uniquely, where IoT provides pervasive data sources, LLMs offer powerful reasoning and language capabilities, FL supports decentralized learning, and PETs ensure data confidentiality, together forming a foundation for scalable, intelligent, and privacy-aware edge AI systems.

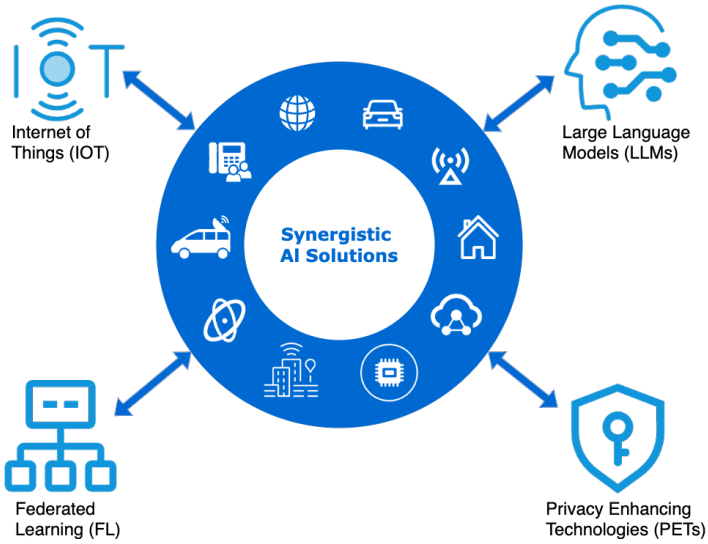

Figure 1. Conceptual overview of technological convergence for synergistic AI solutions. This diagram illustrates the roles of key components: IoT for data provision, LLMs for intelligence, FL for privacy-preserving distributed training, and PETs for security, together forming a foundation for advanced edge AI systems.

More specifically, this review provides a comprehensive analysis of the state of the art regarding architectures, methodologies, challenges, and potential solutions for integrating IoT, LLMs, and FL, with a specific emphasis on achieving privacy-preserving intelligence in edge computing environments. We explore architectural paradigms conducive to edge deployment based on [3], investigate key enabling techniques including PEFT methods like low-rank adaptation (LoRA) [16] and distributed training strategies such as Split Federated Learning (SFL) [17,18], and systematically analyze the inherent multifaceted challenges spanning resource constraints, communication efficiency, data/system heterogeneity, privacy/security threats, fairness, and scalability [3]. Mitigation strategies are discussed alongside critical comparisons highlighting advantages and disadvantages. We survey recent applications to illustrate practical relevance [19]. While existing surveys may cover subsets of this intersection, such as FL for IoT [20,21] or FL for LLMs [22], this review offers a unique contribution by focusing specifically on the three-way synergy (IoT + LLM + FL) and its implications for privacy-preserving edge intelligence [10]. We aim to provide a structured taxonomy of relevant techniques, critically compare their suitability for resource-constrained and distributed IoT settings, identify research gaps specifically arising from this unique technological confluence, and propose targeted future research directions essential for advancing the field of trustworthy, decentralized AI [23].

As summarized in Figure 2, the subsequent sections are structured as follows: Section 2 introduces foundational concepts related to IoT systems, LLMs, FL principles, and PETs. Section 3 discusses architectural considerations for deploying LLMs within IoT ecosystems. Section 4 examines FL methodologies specifically adapted for LLM training and fine-tuning in this context, including frameworks and data considerations. Section 5 analyzes the

unique synergistic effects arising from the integration of IoT, LLMs, and FL, highlighting emergent capabilities. Section 6 provides an expanded analysis of key challenges encountered in the integration, discusses mitigation strategies, and evaluates inherent trade-offs. Section 7 identifies critical research gaps and elaborates on future research directions stemming from the synergistic integration. Section 8 concludes the review, summarizing the key insights and forward-looking perspectives on privacy-preserving, intelligent distributed systems enabled by IoT, LLMs, and FL.

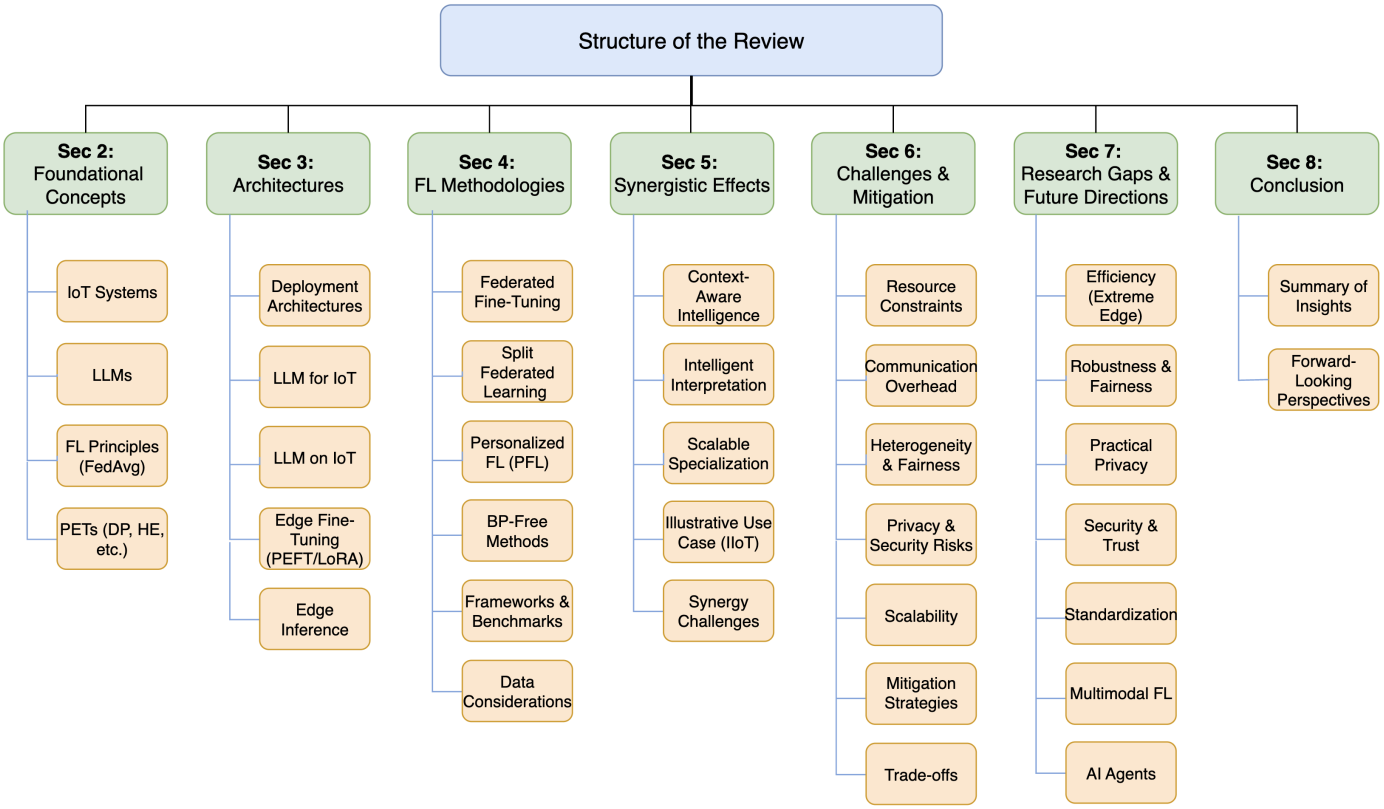

Figure 2. Overview of the review's organizational structure. The diagram outlines the paper's progression through its main sections: Foundational Concepts (Section 2), Architectures (Section 3), FL Methodologies (Section 4), Synergistic Effects (Section 5), Challenges and Mitigation (Section 6), Research Gaps and Future Directions (Section 7), and Conclusion (Section 8). Key topics within each section are indicated, offering a reader roadmap.

To ensure a comprehensive and systematic review, we adopted a structured literature search and selection methodology.

- Search Strategy and Databases: We conducted extensive searches in prominent academic databases, including Google Scholar, IEEE Xplore, ACM Digital Library, Scopus, and ArXiv (for pre-prints). The search was performed between February 2024 and May 2025 to capture the most recent advancements.
- Search Keywords: A combination of keywords was used, including, but not limited to "Internet of Things" OR "IoT" AND "Large Language Models" OR "LLMs" AND "Federated Learning" OR "FL"; "LLMs on edge devices"; "Federated LLMs for IoT"; "privacy-preserving LLMs in IoT"; "LoRA for Federated Learning"; "Split Federated Learning for LLMs"; "efficient LLM deployment on IoT"; "AIoT AND LLMs"; "Industrial IoT AND Federated Learning".
- Inclusion Criteria: Papers were included if they were peer-reviewed journal articles, conference proceedings, or highly cited pre-prints directly relevant to the integration of IoT, LLMs, and FL. We prioritized studies that discussed system architectures, method-

ologies, applications, challenges, or future directions related to this tripartite synergy, particularly those addressing resource constraints and privacy in IoT environments.

- Exclusion Criteria: Papers were excluded if they focused solely on one technology without significant discussion of its integration with the other two, were not written in English, or were not accessible in full text. Short abstracts, posters, and non-academic articles were also excluded.
- Literature Screening and Selection Statistics: Our initial search across the specified databases (Google Scholar, IEEE Xplore, ACM Digital Library, Scopus, and ArXiv using keywords such as ((“Internet of Things” OR “IoT”) AND (“Large Language Models” OR “LLMs”) AND (“Federated Learning” OR “FL”)) AND (“Edge Computing” OR “Privacy”)) yielded 223 unique articles. After screening titles and abstracts for relevance to the tripartite synergy of IoT, LLMs, and FL, particularly in edge environments, 160 articles were retained. These 160 articles underwent a full-text review against our predefined inclusion and exclusion criteria. From this detailed assessment, 78 articles were identified as directly pertinent to the core research questions of this review and were selected for in-depth data extraction and synthesis. The final manuscript cites a total of 135 references, which encompass these 78 core articles along with foundational papers and other supporting literature.
- Bias Assessment and Mitigation: To ensure a balanced review, potential sources of bias were considered. Publication bias, the tendency to publish positive or significant results, was mitigated by including pre-prints from ArXiv, allowing for the inclusion of recent and ongoing research that may not yet have undergone peer review. To counteract database bias, we utilized multiple prominent and diverse academic databases. Furthermore, keyword bias was addressed by developing a comprehensive list of search terms, including synonyms and variations, related to IoT, LLMs, FL, and their intersection with edge computing and privacy. The selection and data extraction were primarily conducted by two authors, with discrepancies resolved through discussion to minimize individual researcher bias.
- Data Extraction and Synthesis: Relevant information regarding methodologies, challenges, proposed solutions, applications, and future trends was extracted from the selected papers. This information was then synthesized to identify common themes, research gaps, and the overall state of the art, forming the basis of this review.

2. Foundational Concepts

This section lays the groundwork for our review by introducing the fundamental concepts underpinning the integration of IoT, LLMs, and FL. We will briefly define each core technology, including IoT systems and their characteristics in advanced networks, the capabilities and challenges of LLMs, the principles of FL such as FedAvg, and key PETs like differential privacy and Homomorphic Encryption. Understanding these foundational elements is crucial for appreciating the synergistic approach and addressing the complexities discussed in later sections.

2.1. IoT in Advanced Networks

The IoT encompasses vast networks of interconnected physical objects and devices, enabling them to collect, exchange, and act upon data, often without direct human intervention [24,25]. This ecosystem is characterized by its massive scale and significant heterogeneity in terms of hardware capabilities, power sources, connectivity, and the types of data generated in real time [6], as summarized in Table 2. To better understand the context for integrating advanced AI, it is useful to consider key IoT sub-domains:

- Massive IoT (MIoT): This segment focuses on connecting a very large number of low-cost, low-power devices (e.g., in smart metering or environmental monitoring) that typically transmit small amounts of data infrequently. Key challenges include ensuring scalability to billions of devices, long battery life, and managing connectivity for devices with severely limited local processing capabilities. The data, while individually small, can be voluminous in aggregate [26].
- Industrial IoT (IIoT): Applied in sectors like manufacturing and energy, IIoT prioritizes high reliability, low latency, and robust security for critical operations such as predictive maintenance and process automation. IIoT systems often generate large volumes of high-frequency, time-sensitive data from sophisticated sensors and machinery, frequently necessitating a strong trend towards edge computing for localized processing and real-time analytics [27].
- Artificial Intelligence of Things (AIoT): AIoT signifies the convergence of AI technologies with IoT infrastructure, aiming to embed AI capabilities, including machine learning and potentially Large Language Models (LLMs), into IoT devices, edge gateways, or associated platforms. This facilitates intelligent decision-making and autonomous operations across diverse applications (e.g., smart homes, intelligent transportation). A primary challenge in AIoT is managing the computational demands of AI models on typically resource-constrained IoT hardware [28].

Table 2. Comparison of IoT characteristics.

Characteristic	Massive IoT (MIoT)	Industrial IoT (IIoT)	Artificial Intelligence of Things (AIoT)	General Consumer IoT
Primary Goal	Wide-scale, low-cost data collection from numerous simple devices [29]	High-reliability, low-latency control and monitoring of critical industrial processes	Enhanced automation, intelligent decision-making, and adaptive behavior through AI/ML at the edge/device	Convenience, automation, and enhanced user experience in daily life
Typical Applications	Smart metering, environmental monitoring, asset tracking, smart agriculture (large-scale sensor networks)	Manufacturing automation (PLCs, SCADA), predictive maintenance, robotics, process control, smart grids	Smart surveillance (intelligent video analytics), autonomous vehicles/drones, advanced robotics, personalized healthcare monitors, smart retail	Smart home devices (lights, thermostats, speakers), wearables (fitness trackers), connected appliances
Data Types and Volume	Small data packets, often infrequent; high volume due to massive device numbers; primarily sensor readings (temperature, humidity, location, status)	Time-series sensor data (vibration, pressure, temperature), control signals, machine status, production data; moderate to high volume per device, often continuous	Multimodal data (video, audio, sensor fusion, text), complex features extracted by AI models; volume varies greatly depending on AI task [30]	User commands (voice, app), sensor data (activity, environment), media streams; volume varies, can be high for media

Table 2. *Cont.*

Characteristic	Massive IoT (MIoT)	Industrial IoT (IIoT)	Artificial Intelligence of Things (AIoT)	General Consumer IoT
Network Topology and Connectivity	LPWAN (LoRaWAN, NB-IoT, Sigfox), cellular IoT; star or mesh topologies; focus on long range, low power	Wired (Industrial Ethernet, Profibus), reliable wireless (e.g., private 5G, Wi-Fi HaLow); often deterministic networks; Focus on reliability, low latency	Diverse: Wi-Fi, 5G/6G, Bluetooth, Zigbee, wired; edge-centric or device-to-device communication; focus on bandwidth and latency for AI processing	Wi-Fi, Bluetooth, Zigbee, Z-Wave; typically star or mesh connected to a home hub/router; focus on ease of use and interoperability
Key Constraints and Challenges	Extreme low power, low cost per device, massive scalability, simple device management, intermittent connectivity	Ultra-high reliability, low and deterministic latency, security against cyber–physical attacks, harsh operating environments, interoperability of legacy systems [31]	Computational power for AI on device/edge, energy for AI processing, real-time AI inference, complexity of AI model deployment and management, data quality for AI	User privacy, security vulnerabilities, ease of setup and use, interoperability between vendor ecosystems, device cost
LLM/FL Relevance	FL for anomaly detection across massive datasets, simple status summarization by LLMs (if data aggregated); SFL for very basic feature extraction.	FL for predictive maintenance models, LLMs for analyzing maintenance logs and generating reports, LLMs for human–machine interfaces (NL queries about machine status).	FL for training sophisticated AI models (e.g., vision, speech) at the edge, LLMs for complex scene understanding, natural language interaction, and autonomous decision-making.	FL for personalized models (e.g., smart home routines), LLMs for voice assistants and intuitive control; SFL for privacy-preserving on-device learning [32]

Across these varied IoT deployments, and particularly as AIoT applications become more sophisticated, the inherent resource limitations (such as CPU, memory, battery, and power budgets) of many end devices and even edge nodes represent a primary bottleneck. Executing complex AI models, such as LLMs, directly at the extreme edge is thus particularly challenging [33], underscoring the critical need for resource-efficient AI techniques, including PEFT and FL, which are central to this review.

2.2. Large Language Models

LLMs are deep learning models, primarily Transformer-based [34], possessing billions of parameters and demonstrating powerful emergent capabilities derived from extensive pre-training [1,35]. They typically undergo fine-tuning for task adaptation [36]. Their significant size imposes high computational costs for training and inference, making deployment on standard IoT hardware challenging [4]. Ethical considerations regarding potential biases and responsible use are also critical [5,37].

2.3. Federated Learning

FL enables collaborative training on decentralized data [9]. The most widely known FL algorithm is Federated Averaging (FedAvg) [9,38,39]. In each communication round t, local clients receive the current global model weights w_t from the central server. K selected clients then train the model locally using its data D_k for E epochs, and update local weights

$w_{t+1}^k, k \in K$. The server aggregates these local weights to produce the updated global model w_{t+1}, as defined in (1):

$$w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k, \tag{1}$$

where $n_k = |D_k|$ is the number of data points on client k, and $n = \sum_{k=1}^{K} n_i$ is the total number of data points across the selected clients [40]. This weighted average aims to give more importance to updates from clients with more data. The adoption of FL, particularly in sensitive or distributed environments like IoT, is driven by several key advantages over traditional centralized approaches [7]:

- Enhanced Privacy: Data remains localized on user devices, reducing risks associated with central data aggregation.
- Communication Efficiency: Transmitting model updates instead of raw data significantly reduces network load.
- Utilizing Distributed Resources: Leverages the computational power available at the edge devices [41].

While FedAvg provides a foundational approach, practical FL implementations involve several key characteristics, architectural choices, and challenges:

- CFL vs. DFL: Centralized FL (CFL) uses a server for coordination and aggregation, offering simplicity but creating a potential bottleneck and single point of failure [42]. Decentralized FL (DFL) employs peer-to-peer communication, potentially increasing robustness and scalability for certain network topologies (like mesh networks common in IoT scenarios) but adding complexity in coordination and convergence analysis [43].
- Non-IID Data: A central challenge in FL stems from heterogeneous data distributions across clients, commonly referred to as Non-Independent and Identically Distributed (Non-IID) data [44]. This means the statistical properties of data significantly vary between clients; for instance, clients might hold data with different label distributions (label skew) or different feature characteristics for the same label (feature skew). Such heterogeneity can substantially degrade the performance of standard algorithms like FedAvg, as the single global model aggregated from diverse local models may not generalize well to each client's specific data distribution [7].

2.4. Privacy-Preserving Techniques

FL's privacy benefits can be further enhanced using PETs, with significant advantages and disadvantages, particularly relevant in the resource-constrained IoT context:

Differential Privacy (DP): DP provides a formal, mathematical definition of privacy guarantees [45,46]. A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-DP if, for any two adjacent datasets D_1 and D_2 (differing by at most one element), and for any possible subset of outputs S, the following inequality holds:

$$\mathbb{P}[\mathcal{M}(D_1) \in S] \leq e^{\epsilon} \mathbb{P}[\mathcal{M}(D_2) \in S] + \delta, \tag{2}$$

where ϵ is the privacy budget, and δ represents the probability that the strict ϵ-DP guarantee might be violated. The privacy budget (ϵ) is a fundamental concept in differential privacy that quantifies the maximum amount of information leakage or privacy loss permitted in a privacy-preserving mechanism. A smaller privacy budget (a smaller ϵ value) indicates stronger privacy by limiting the influence of any single data point on the output, thereby making it harder to infer individual information. However, this typically results in lower utility of the data or model, as more noise is often required to achieve stronger privacy. In the context of Federated Learning with LLMs, the privacy budget must be carefully managed across multiple rounds of training and participating clients to balance privacy

protection with model performance, especially in sensitive IoT applications like healthcare or smart homes. For δ, it is typically set to a very small value (e.g., less than the inverse of the dataset size $|D|$), representing a small probability that the pure ϵ-DP guarantee is broken. This definition ensures that the output distribution of the mechanism is statistically similar regardless of the presence or absence of any single individual's data [47]. DP guarantees are commonly achieved by adding carefully calibrated noise (e.g., following a Gaussian or Laplace distribution) to function outputs, gradients, or model updates, as implemented in algorithms like DP-SGD [48].

DP offers strong, mathematically rigorous privacy guarantees against inference attacks. Its computational overhead is generally lower compared to cryptographic methods like HE or SMPC. However, a key challenge of DP is the inherent trade-off between privacy and utility, where increasing noise (reducing ϵ) to enhance privacy typically degrades model accuracy [49], as conceptually illustrated in Figure 3. This figure compares the relative computational and communication overheads of various privacy-preserving techniques in FL. It highlights that while DP introduces additional costs, its overhead remains modest compared to more complex methods like SMPC and HE. Notably, homomorphic encryption incurs the highest total overhead, underscoring the practicality of DP in resource-constrained edge scenarios. Managing privacy budgets effectively across rounds and clients is complex [50–52], and DP noise can disproportionately affect fairness for underrepresented groups [3].

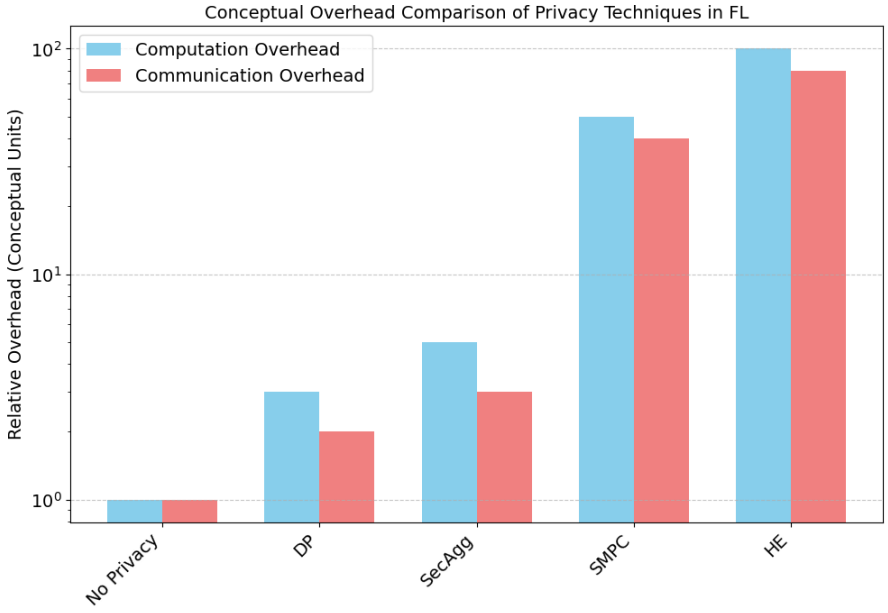

Figure 3. Conceptual illustration of the privacy–utility trade-off in DP. Stronger privacy guarantees (lower ϵ) often correlate with a decrease in model utility or accuracy. The exact curve depends heavily on the dataset, model, and specific DP mechanism.

Homomorphic Encryption (HE): HE allows specific computations (e.g., addition for averaging updates) on encrypted data [53]. The server aggregates ciphertexts without decrypting them. The advantage of HE lies in the fact that it provides strong confidentiality against the server (server learns nothing about individual updates), hence no impact on model accuracy (utility) compared to non-private aggregation. However, HE can have extremely high computational overhead for encryption/decryption and homomorphic operations, significantly expanding the communication data size (ciphertext size). Thus, HE is currently impractical for direct implementation on most resource-constrained IoT devices [7].

Secure Multi-Party Computation (SMPC): SMPC enables multiple parties to jointly compute a function, such as the sum of updates, using cryptographic protocols like secret sharing, without revealing their private inputs [54]. The primary advantage of SMPC lies in its strong privacy guarantees achieved by distributing trust among participants, including potentially the server and clients, with no impact on model accuracy [55]. However, SMPC protocols often require complex multi-round interactions, leading to significant communication overhead. Furthermore, assumptions of synchronous participation or the need for fault tolerance mechanisms add complexity, posing challenges for deployment in dynamic IoT environments [3].

Secure Aggregation: Secure Aggregation utilizes specialized protocols, often based on secret sharing or lightweight cryptography, optimized specifically for the FL aggregation task [56]. These protocols allow the server to securely compute only the sum or average of client updates [57]. Compared to general HE or SMPC, Secure Aggregation is significantly more efficient computationally and communication-wise for this specific task, leading to its widespread adoption in practical FL systems. Nevertheless, while it protects individual updates from the server during the aggregation phase, it does not shield the final aggregated result from potential inference attacks, nor does it secure the updates during transmission unless combined with additional encryption methods.

Table 3 provides a comparative summary of these key privacy-preserving techniques, highlighting their mechanisms, pros, and cons within the FL context. The practical choice often involves secure aggregation, potentially combined with DP for stronger client-level guarantees, or relies on trust in the server, depending heavily on the threat model, system capabilities, and regulatory environment (e.g., GDPR, HIPAA constraints on data processing and transfer) [3,58].

Table 3. Comparison of key privacy-preserving techniques in the FL context.

Technique	Mechanism	Pros	Cons
Differential Privacy	Adds calibrated noise to gradients, updates, or data for formal (ϵ, δ)-privacy guarantees.	Strong, mathematical privacy guarantees; relatively lower computational overhead than cryptographic methods.	Direct privacy–utility trade-off (noise vs. accuracy); complex privacy budget management; can impact fairness; overhead can still be significant for resource-poor IoT devices.
Homomorphic Encryption	Allows specific computations (e.g., addition) on encrypted data; server aggregates ciphertexts without decryption.	Strong confidentiality against the server; no impact on model accuracy (utility).	Extremely high computational overhead (encryption, decryption, operations); significant communication overhead (ciphertext size); largely impractical for direct use on most IoT devices.
Secure Multi-Party Computation	Enables joint computation (e.g., sum) via cryptographic protocols without parties revealing private inputs.	Strong privacy guarantees (distributed trust); no impact on model accuracy.	Requires complex multi-round interaction protocols; significant communication overhead; often assumes synchronicity or fault tolerance mechanisms, challenging in dynamic IoT.
Secure Aggregation	Specialized protocols (often secret sharing-based) optimized for securely computing the sum/average of client updates.	More efficient (computationally and communication-wise) than general HE/SMPC for the aggregation task; widely adopted.	Protects individual updates from the server during aggregation, but not the final aggregated model from inference, nor updates during transmission without extra encryption.

3. LLM-Empowered IoT Architecture for Distributed Systems

Having established the foundational concepts of IoT, LLMs, and FL in the preceding section, this section transitions to explore the architectural frameworks necessary for effectively deploying LLM-empowered IoT systems within distributed environments. We begin by outlining a general multi-tier (Cloud–Edge–Device) architecture that balances computational demands with data locality and latency requirements. Subsequently, we delve into two key operational perspectives: first, how LLMs can augment and enhance

the capabilities of IoT systems (termed "LLM for IoT") by enabling intelligent interfaces, advanced data analytics, and automated control; second, the crucial strategies and optimization techniques for efficiently running LLMs on or near resource-constrained IoT devices (termed "LLM on IoT"), covering essential aspects like edge fine-tuning and edge inference.

3.1. Architectural Overview

Deploying LLMs within IoT often favors multi-tier architectures (Cloud–Edge–Device) to balance computation, latency, and data locality [33]. This involves strategically placing LLM-related tasks: heavy pre-training in the cloud, fine-tuning and inference closer to the edge, and potentially highly optimized inference on capable end devices [25]. This architecture supports both leveraging LLMs for IoT enhancement ("LLM for IoT") and efficiently managing LLMs within IoT constraints ("LLM on IoT") [10].

3.2. LLM for IoT

LLMs can significantly enhance IoT system capabilities through the following:

- Intelligent Interfaces and Interaction: Enabling sophisticated natural language control (e.g., complex conditional commands for smart environments) and dialogue-based interaction with IoT systems for status reporting or troubleshooting [59].
- Advanced Data Analytics and Reasoning: Fusing data from multiple sensors (e.g., correlating camera feeds with environmental sensor data for scene understanding in smart cities), performing complex event detection, predicting future states (e.g., equipment failure prediction in IIoT based on subtle degradation patterns), and providing causal explanations for system behavior.
- Automated Optimization and Control: Learning complex control policies directly from high-dimensional sensor data for optimizing resource usage (e.g., dynamic energy management in buildings considering real-time occupancy, weather forecasts, and energy prices) or network performance (e.g., adaptive traffic routing in vehicular networks).

3.3. LLM on IoT: Deployment Strategies

Efficiently running LLMs on or near IoT devices requires optimization. In the training stage, model pruning is a typical strategy, while inference adaptation can also be performed for edge devices. These techniques are reviewed next.

3.3.1. Edge Fine-Tuning

Adapting pre-trained models locally using PEFT is key. To adapt large pre-trained models like LLMs without incurring the high computational and memory costs of full fine-tuning, PEFT methods can be employed. A prominent example is the popular LoRA [16]. Instead of updating the entire pre-trained weight matrix $\mathbf{W}_0 \in \mathbb{R}^{d \times k}$, LoRA introduces two smaller, low-rank matrices, $\mathbf{A} \in \mathbb{R}^{d \times r}$ and $\mathbf{B} \in \mathbb{R}^{r \times k}$, where the rank r is typically much smaller than d or k (i.e., $r \ll \min(d, k)$). The core idea is to represent the weight update $\Delta\mathbf{W}$ as the product of these low-rank matrices ($\Delta\mathbf{W} = \mathbf{BA}$). During fine-tuning, the original weights $\mathbf{W}_0$ remain frozen, and only the parameters in $\mathbf{A}$ and $\mathbf{B}$ are trained. This mechanism is illustrated in Figure 4. The effective weight matrix used in the forward pass is then computed as

$$\mathbf{W} = \mathbf{W}_0 + \Delta\mathbf{W} = \mathbf{W}_0 + \mathbf{BA}. \tag{3}$$

This approach drastically reduces the number of trainable parameters from $d \times k$ for full fine-tuning down to only $r \times (d + k)$ for LoRA [60]. This significant reduction in parameters, memory usage, and computation makes fine-tuning large models feasible even on resource-constrained edge devices and substantially decreases communication

overhead in Federated Learning scenarios where only the small **A** and **B** matrices need to be exchanged [61].

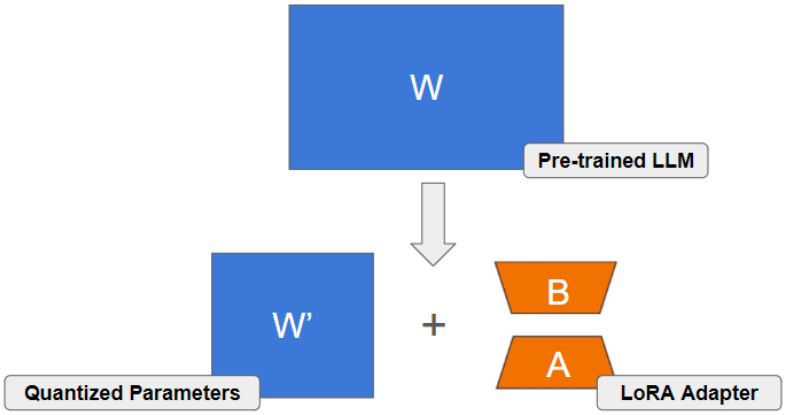

Figure 4. Illustration of the LoRA adapter mechanism, potentially used with quantized base model weights (as in QLoRA [62]). The large pre-trained weights (W) might be stored in a quantized format (W'), while the task-specific update is learned via the small, trainable low-rank adapter matrices (B and A).

Recent advancements in LoRA have produced variants that further address challenges pertinent to resource-constrained IoT and FL settings. For instance, DoRA (Weight-Decomposed Low-Rank Adaptation) [63] extends LoRA by decomposing the pre-trained weights into magnitude and direction components for fine-tuning. This allows LoRA updates to refine the model with a different learning mechanism compared to full fine-tuning of magnitudes and directions, reportedly enabling more efficient and effective training by potentially better capturing nuanced parameter adjustments while maintaining or even improving performance over standard LoRA. Such an approach can be particularly valuable in Federated Learning rounds over bandwidth-constrained IoT environments where efficient yet impactful updates are crucial.

Building upon such decomposed approaches, EDoRA (Efficient DoRA) [64] aims to further optimize training efficiency. The EDoRA methodology is designed to significantly reduce the computational burden and memory footprint during the fine-tuning process, potentially involving techniques like quantization or structured pruning tailored for the decomposed components. For example, EDoRA is reported to achieve substantial reductions in communication overhead compared to standard LoRA while maintaining comparable task performance by employing sparse updating mechanisms or more aggressive compression. These characteristics make variants like EDoRA highly suitable for IoT+FL scenarios where both computational and communication efficiency are critical for practical deployment on diverse and often limited edge devices.

However, PEFT methods, including LoRA and its variants, have benefits and drawbacks. The choice of parameters, such as the rank r in LoRA, directly impacts the balance between efficiency and the model's adaptation capacity; a very low rank might limit the model's ability to capture complex task-specific nuances [65]. Furthermore, the generalization capability of PEFT methods, especially when adapting models to tasks significantly different from the pre-training data, compared to full fine-tuning, remains an active area of investigation [66].

3.3.2. Edge Inference

Prediction/generation performance can be optimized through the following techniques.

- On-device Inference: Utilizes model compression (quantization, pruning, distillation) [62,67]. Compression inherently risks degrading model accuracy or robustness, and the extent depends heavily on the technique and compression ratio [68].
- Co-inference/Split Inference: Divides layers between device and edge server [18]. It introduces network latency and dependency on the edge server, although it keeps raw data local. This is distinct from SFL used for training.
- Edge Caching: Reduces latency for repeated queries [3].

4. Federated Learning for Privacy-Preserving LLM Training in IoT

Having established the foundational concepts and architectural considerations, this section delves into the specific methodologies required to effectively train and adapt LLMs within distributed IoT environments using FL. We examine various techniques designed to overcome the inherent challenges of resource constraints, communication overhead, data heterogeneity, and privacy concerns that arise when integrating these powerful models with FL paradigms at the edge [61]. Key topics include core federated fine-tuning strategies tailored for LLMs, methods for personalization, alternative training approaches, essential supporting frameworks and data handling techniques, the emerging role of LLMs in aiding the FL process itself, and crucial evaluation metrics specific to this context [69]. Understanding these methodologies is crucial for realizing the practical potential of the synergistic IoT, LLM, and FL integration.

4.1. Federated Fine-Tuning of LLMs

Applying FL to fine-tune Large Language Models enables collaborative adaptation on decentralized IoT data, crucial for personalization and domain specialization while preserving privacy [23]. The integration of FL with PEFT methods, particularly LoRA, significantly reduces communication overhead by transmitting only lightweight parameter updates (typically <1% of total model parameters) [70]. Beyond CFL approaches, research is exploring decentralized fine-tuning methods [71]. For instance, Dec-LoRA is an algorithm designed for decentralized fine-tuning of LLMs using LoRA without relying on a central parameter server [72]. Experimental results suggest that Dec-LoRA can achieve performance comparable to centralized LoRA, even when facing challenges like data heterogeneity and quantization constraints, offering a potential pathway for more robust and scalable federated fine-tuning in certain network topologies [60].

4.2. Split Federated Learning

SFL addresses the critical memory limitations on edge devices during the training phase of large models within an FL context [19]. By partitioning the model and offloading a significant portion of the computation (especially backward passes through deeper layers) to a server, SFL allows memory-constrained devices to participate [18]. Integrating LoRA further optimizes this [17]. However, SFL introduces latency due to the necessary exchange of activations and gradients between client and server per iteration, and its performance is sensitive to the network bandwidth and the choice of the model split point [18].

4.3. Personalized Federated LLMs

PFL methods are vital for addressing client heterogeneity in FL, aiming to provide models better suited to individual client data or capabilities than a single global model [73,74]. Table 4 provides a comparative overview.

Table 4. Comparative overview of personalized PFL approaches for LLMs.

Approach	Mechanism	Heterogeneity Handled	Efficiency	Trade-Offs
PEFT-based PFL (Prompts, Adapters, LoRA)	Clients train personalized PEFT components attached to a shared, frozen LLM backbone. Global aggregation might occur on these PEFT components or parts thereof [16,61,75–77].	Primarily statistical (data); some methods adapt to system heterogeneity by adjusting PEFT complexity (e.g., heterogeneous LoRA ranks) [78,79].	High communication efficiency (small updates); moderate computation (only PEFT tuning) [80].	Personalization depth limited by PEFT capacity; potential for negative interference if global components are poorly aggregated [81].
Model Decomposition/Partial Training	Global model is structurally divided; clients train only specific assigned layers or blocks [69,82,83].	Primarily system (computation/memory); can assign smaller parts to weaker clients.	Reduced client computation; communication depends on the size of the trained part.	Less flexible personalization compared to PEFT; requires careful model partitioning design; potential information loss between components.
Knowledge Distillation (KD)-based PFL	Uses outputs (logits) or intermediate features from a global "teacher" model (or ensemble of client models) to guide the training of personalized local "student" models [84–86].	Can handle model heterogeneity (different student architectures); adaptable to data heterogeneity.	Communication involves logits/features, potentially smaller than parameters; client computation depends on student model size.	Distillation process can be complex; potential privacy leakage from shared logits; the student's model might not perfectly capture the teacher's knowledge.
Meta-Learning-based PFL (e.g., Reptile, MAML adaptations)	Learns a global model initialization that can be rapidly adapted (fine-tuned) to each client's local data with few gradient steps [74].	Focuses on adapting to statistical (data) heterogeneity.	Communication similar to standard FL; potentially more local computation during adaptation phase.	Can be sensitive to task diversity across clients; training the meta-model can be computationally intensive.

4.4. Backpropagation-Free Methods

These methods (e.g., zeroth-order optimization) bypass standard backpropagation, reducing peak memory usage by eliminating the need to store activations [87–90]. Limitations: They often require significantly more function evaluations (slower convergence) and can be less stable or scalable for very high-dimensional parameter spaces compared to gradient-based methods [87,91]. Their practical application in large-scale federated LLM training remains an active research topic.

4.5. Frameworks and Benchmarks

The practical implementation and evaluation of federated LLMs rely on specialized software frameworks and benchmarks:

Frameworks: Libraries like FedML [92] with its FedLLM component [92], Flower [93,94], FATE-LLM [95], and FederatedScope-LLM [96] provide infrastructure for simulating or deploying FL. Features relevant to IoT/edge include support for heterogeneous devices, PEFT methods (e.g., LoRA), various aggregation algorithms, security mechanisms (DP, secure aggregation), and sometimes specific optimizations for edge deployment (e.g., efficient client runtimes, handling intermittent connectivity). Selecting a framework depends on the specific research or deployment needs regarding scale, flexibility, supported models, and available privacy/security features.

Benchmarks: Standardized datasets and evaluation protocols are crucial for comparing different algorithms. Efforts like FedIT [97] focus on benchmarking federated

instruction tuning. FedNLP [98] provided early benchmarks for standard NLP tasks in FL. OpenFedLLM aims to offer a comprehensive platform with multiple datasets and metrics [99]. However, benchmarks specifically capturing the complexities of real-world IoT data heterogeneity, network conditions, and device constraints for LLMs are still needed.

4.6. Initialization and Data Considerations

Effective federated LLM training depends significantly on model initialization and data handling:

Model Initialization: Starting FL from a well-pre-trained LLM, rather than random initialization, significantly improves convergence speed, final model performance, and robustness to Non-IID data [100]. It allows FL to focus on adaptation rather than learning foundational knowledge from scratch [101].

Data Processing: Handling massive, distributed datasets requires scalable tools. Libraries like Dataset Grouper aim to facilitate partitioning large datasets for FL simulation [102].

Synthetic Data Generation: When local data is scarce or highly skewed, generating synthetic data can augment training [10]. LLMs show promise for generating high-quality synthetic data that reflects complex real-world distributions, potentially overcoming limitations of earlier generative models used in FL [103]. Frameworks like GPT-FL explore using LLM-generated data to aid FL. Selecting relevant public data using distribution matching techniques can also enhance privacy-preserving training via knowledge distillation [104].

4.7. LLM-Assisted Federated Learning

Beyond using FL to train LLMs, the reciprocal relationship where LLMs assist FL is also emerging [23]:

Mitigating Data Heterogeneity: LLMs pre-trained on vast datasets can generate high-quality synthetic data reflecting diverse distributions. This synthetic data can be used centrally or shared (with privacy considerations) to augment clients' local datasets, helping to alleviate the negative impacts of Non-IID data on FL convergence [103].

Knowledge Distillation: A large, powerful LLM (potentially centrally available or trained via FL itself) can act as a "teacher" model. Its knowledge (e.g., predictions, representations) can be distilled into smaller "student" models trained by clients in the FL network, improving the efficiency and performance of client models, especially on resource-constrained devices [65].

Intelligent FL Orchestration: LLMs could potentially be used for more sophisticated FL management tasks, such as predicting client resource availability, assessing data quality for client selection, or even dynamically tuning FL hyperparameters based on observed training dynamics.

4.8. Evaluation Metrics

Evaluating federated LLM systems requires a multifaceted approach beyond standard accuracy measures, particularly in the IoT context (see Table 5). Developing standardized benchmarks that allow for consistent evaluation across these diverse metrics is a key challenge and future direction [97]. Table 5 summarizes the key categories of evaluation, including model utility, efficiency, privacy, fairness, and scalability, each with specific metrics tailored to the constraints and demands of federated IoT settings. For instance, communication and computation efficiency metrics reflect the limited bandwidth, energy, and processing power typical of edge devices. Privacy is evaluated through both theoretical guarantees (such as differential privacy parameters) and empirical attack resistance, while fairness and scalability ensure inclusiveness and robustness across heterogeneous clients. Together, these metrics offer a comprehensive framework for assessing the real-world

feasibility and trustworthiness of federated LLM systems deployed across diverse and distributed IoT environments.

Table 5. Key evaluation metrics for federated LLM systems in IoT contexts.

Category	Specific Metric Examples	Relevance/Notes
Model Utility	Accuracy, F1-score, BLEU, ROUGE, perplexity, calibration, robustness (to noise, adversarial inputs)	Task-specific performance and reliability of the learned model.
Efficiency:		Crucial for resource-constrained and bandwidth-limited IoT environments.
- Communication	Total bytes/bits transmitted, number of rounds, compression rates	Impacts network load, energy consumption on wireless devices.
- Computation	Client training time/round, server aggregation time, edge inference latency, total FLOPs, energy consumption	Determines feasibility on device, overall system speed, battery life.
- Memory	Peak RAM usage (client/server), model storage size	Critical for devices with limited memory capacity.
Privacy	Formal guarantees (e.g., (ϵ, δ)-DP values), empirical leakage (e.g., Membership Inference Attack success rate)	Quantifies the level of privacy protection provided against specific threats.
Fairness	Variance in accuracy across clients/groups, worst-group performance vs. average	Measures consistency of performance for diverse participants or data subpopulations.
Scalability	Performance/efficiency degradation as the number of clients increases	Assesses the system's ability to handle large-scale IoT deployments.

5. Synergistic Effects of Integrating IoT, LLMs, and Federated Learning

The previous sections have laid the groundwork by introducing the core concepts and individual capabilities of IoT, LLM and FL. While pairwise integrations—such as applying LLMs to IoT data analytics [105], using FL for privacy-preserving IoT applications [20,21], or employing FL to train LLMs [23]—offer significant advancements, they often encounter inherent limitations [10]. Centralized LLM processing of IoT data raises critical privacy and communication bottlenecks [13]; traditional FL models struggle with the complexity and scale of raw IoT data [14]; and federated LLMs without direct access to real-world IoT streams lack crucial grounding and context [7].

This section argues that the true transformative potential lies in the synergistic convergence of all three technologies, IoT, LLMs, and FL, explicitly enhanced by Privacy-Enhancing Technologies [49]. This three-way integration creates a powerful ecosystem where the strengths of each component compensate for the weaknesses of the others, enabling capabilities and solutions that are fundamentally unattainable or significantly less effective otherwise [45]. We posit that this synergy is not merely additive but multiplicative, paving the way for a new generation of advanced, privacy-preserving, context-aware distributed intelligence operating directly at the network edge [15]. We will explore this "1 + 1 + 1 > 3" effect through three core synergistic themes, building upon the motivations discussed in works like [56].

5.1. Theme 1: Privacy-Preserving, Context-Aware Intelligence from Distributed Real-World Data

The Challenge: LLMs thrive on vast, diverse, and timely data to develop nuanced understanding and maintain relevance [8]. IoT environments generate precisely this type of data—rich, real-time, multimodal streams reflecting the complexities of the physical world [7,49]. However, this data is inherently distributed across countless devices and

locations [14], and often contains highly sensitive personal, operational, or commercial information, making centralized collection legally problematic (e.g., GDPR, HIPAA compliance [21]), technically challenging (bandwidth costs, latency [13]), and ethically undesirable [5,22]. Relying solely on public datasets limits LLM grounding and domain specificity [10].

The Synergy (IoT + LLM + FL): Federated Learning acts as the crucial enabling mechanism [9] that allows LLMs to tap into the rich, distributed data streams generated by IoT devices without compromising data locality and privacy [15]. IoT provides the continuous flow of real-world, multimodal data (the "what" and "where") [14]. FL provides the privacy-preserving framework for collaborative learning across these distributed sources (the "how") [10]. The LLM provides the advanced cognitive capabilities to learn deep representations, understand context, and extract meaningful intelligence from this data (the "why" and "so what?") [69].

Emergent Capability: This synergy empowers LLMs to maintain robust general capabilities while dynamically adapting to specific real-world contexts. By leveraging fresh, diverse, and privacy-sensitive IoT data, these models achieve continuous grounding in evolving environments. This allows for the following:

- Hyper-Personalization: Training models tailored to individual users or specific environments (e.g., a smart home assistant learning user routines from sensor data via FL [14]).
- Real-time Domain Adaptation: Continuously fine-tuning LLMs (e.g., using PEFT like LoRA [61]) with the latest IoT data to adapt to changing conditions (e.g., adapting a traffic prediction LLM based on real-time sensor feeds from different city zones [106]).
- Enhanced Robustness: Learning from diverse, real-world IoT data sources via FL can make LLMs more robust to noise and domain shifts compared to training solely on cleaner, but potentially less representative, centralized datasets [44].

5.2. Theme 2: Intelligent Interpretation and Action within Complex IoT Environments

The Challenge: IoT environments produce data that is often complex, noisy, unstructured, and multimodal (e.g., raw sensor time series, machine logs, video feeds, acoustic signals) [14]. Traditional FL, while preserving privacy, often employs simpler models that struggle to extract deep semantic meaning or perform complex reasoning on such data [49]. Conversely, powerful LLMs, while capable of understanding complexity [15], lack the direct connection to the physical world for sensing and actuation and struggle with distributed private data access [107].

The Synergy (IoT + LLM + FL): LLMs bring sophisticated natural language understanding, reasoning, and generation capabilities to the table [1], allowing the system to interpret intricate patterns, correlate information across different IoT modalities, and even generate human-readable explanations or reports [105]. FL provides the means to train these powerful LLMs collaboratively using the relevant complex IoT data distributed across the network [61]. Crucially, IoT devices provide the physical grounding, acting as the sensors collecting the complex data and potentially as actuators executing decisions derived from LLM insights [3]. Furthermore, LLMs can enhance the FL process itself by intelligently guiding client selection based on interpreting the relevance or quality of their IoT data, or even assisting in designing personalized FL strategies [15].

Emergent Capability: The combination allows for systems that can deeply understand complex physical environments and interact intelligently within them. This goes beyond simple data aggregation or pattern matching:

Contextual Anomaly Detection: Identifying subtle anomalies in IIoT machine behavior by correlating multi-sensor data and unstructured logs, understood and explained by an LLM trained via FL [108]. Causal Reasoning in Smart Cities: Using FL-trained LLMs to

analyze diverse IoT data (traffic, pollution, events) to infer causal relationships and predict cascading effects [14,106]. Goal-Oriented Dialogue with Physical Systems: Enabling users to interact with complex IoT environments (e.g., a smart factory floor) using natural language, where an LLM interprets the request, queries relevant IoT data (potentially involving FL for aggregation), and generates responses or even commands for actuators [15].

5.3. Theme 3: Scalable and Adaptive Domain Specialization at the Edge

The Challenge: Deploying large, general-purpose LLMs directly onto resource-constrained IoT devices is often infeasible due to their size and computational requirements [62]. While smaller, specialized models can run on the edge, training them from scratch for every specific IoT application or location is inefficient and does not leverage the power of large pre-trained models [15]. Centralized fine-tuning of large models for specific domains requires access to potentially private or distributed IoT data [13].

The Synergy (IoT + LLM + FL): FL combined with PEFT techniques like LoRA [70] provides a highly scalable and resource-efficient way to specialize pre-trained LLMs for diverse IoT domains using distributed edge data [13,60]. IoT devices/edge servers provide the specific local data needed for adaptation [14]. PEFT ensures that only a small fraction of parameters need to be trained and communicated during the FL process, drastically reducing computation and communication overhead [61,82]. The base LLM provides the powerful foundational knowledge, while FL+PEFT enables distributed, privacy-preserving specialization [71].

Emergent Capability: This synergy enables the mass customization and deployment of powerful, specialized AI capabilities directly within diverse IoT environments. Key outcomes include the following:

- Locally Optimized Performance: Models fine-tuned via FL+PEFT on local IoT data will likely outperform generic models for specific edge tasks (e.g., a traffic sign recognition LLM adapted via FL to local signage variations [14]).
- Rapid Adaptation: New IoT devices or locations can quickly join the FL process and adapt the shared base LLM using PEFT without needing massive data transfers or full retraining [10].
- Resource-Aware Deployment: Allows for leveraging powerful base LLMs even when end devices can only handle the computation for small PEFT updates during FL training [79], or optimized inference models (potentially distilled using FL-trained knowledge [86]). Frameworks like Split Federated Learning can further distribute the load [17,18].

5.4. Illustrative Use Case: Predictive Maintenance in Federated Industrial IoT

Consider a scenario involving multiple manufacturing plants belonging to different subsidiaries of a large corporation, or even different collaborating companies [108]. Each plant operates similar types of critical machinery (e.g., CNC machines, robotic arms) equipped with various sensors (vibration, temperature, acoustic, power consumption—the IoT component). The goal is to predict potential machine failures proactively across the entire fleet to minimize downtime and optimize maintenance schedules, while ensuring that proprietary operational data and specific machine performance characteristics from one plant are not shared with others.

Below, we summarize the limitations without synergy.

- IoT only: Basic thresholding or simple local models on sensor data might miss complex failure patterns. No collaborative learning.
- IoT + Cloud LLM: Requires sending massive, potentially sensitive sensor streams and logs to the cloud, incurring high costs, latency, and privacy risks [13].

- IoT + FL (Simple Models): Can learn collaboratively but struggles to interpret unstructured maintenance logs or complex multi-sensor correlations indicative of subtle wear patterns [14].
- LLM + FL (No IoT): Lacks real-time grounding; trained on potentially outdated or generic data, not the specific, current state of the machines [10].

To address the issues highlighted above, a synergistic solution (IoT + LLM + FL) is illustrated next.

- Data Generation: Sensors on machines continuously generate multimodal time-series data and operational logs.
- Model Choice (LLM): A powerful foundation LLM (potentially pre-trained on general engineering texts and machine manuals) is chosen as the base model. It possesses the capability to understand technical language in logs and potentially process time-series data patterns [15].
- Collaborative Fine-Tuning (FL + PEFT): FL is used to fine-tune this LLM across the plants using their local IoT sensor data and maintenance logs [69]. To manage resources and communication, PEFT (e.g., LoRA [16]) is employed. Only the small LoRA adapter updates are shared with a central FL server (or aggregated decentrally [72])—preserving privacy regarding raw data and detailed operational parameters [61].
- Intelligence and Action (LLM + IoT): The fine-tuned LLM (potentially deployed at edge servers within each plant [13]) analyzes incoming IoT data streams and logs in near-real time. It identifies complex failure precursors missed by simpler models, correlates sensor data with log entries, predicts remaining useful life, and generates concise, human-readable alerts and maintenance recommendations for specific machines [108]. These alerts can be directly integrated into the plant's maintenance workflow system (potentially an IoT actuation).

This integrated system can achieve highly accurate, context-aware predictive maintenance across multiple entities by leveraging diverse operational data (IoT) through privacy-preserving collaborative learning (FL), powered by the deep analytical and interpretive capabilities of LLMs, all achieved efficiently using PEFT. This outcome would be significantly harder, if not impossible, to achieve with only two of the three components.

5.5. Challenges Arising from the Synergy

While powerful, the tight integration of IoT, LLMs, and FL introduces unique challenges beyond those of the individual components:

Cross-Domain Data Alignment and Fusion: Effectively aligning and fusing heterogeneous, multimodal IoT data streams within an FL framework before feeding them to an LLM requires sophisticated alignment and representation techniques [105].

Resource Allocation Complexity: How to jointly optimize computation (LLM inference/training, FL aggregation), communication (IoT data upload, FL updates), and privacy (PET overhead) across heterogeneous IoT devices, edge servers, and potentially the cloud specifically for this integrated task [13].

Model Synchronization vs. Real-time Needs: Balancing the need for FL model synchronization (potentially slow for large LLM updates [10]) with the real-time data processing and decision-making requirements of many IoT applications.

Emergent Security Vulnerabilities: New attack surfaces emerge at the interfaces, e.g., malicious IoT data poisoning FL training specifically to mislead the LLM's interpretation [109], or FL privacy attacks aiming to reconstruct sensitive IoT context interpreted by the LLM [110]. Verifying the integrity of both IoT data and FL updates becomes critical [15].

5.6. Concluding Remarks on Synergy

The convergence of IoT, Large Language Models, and Federated Learning represents a fundamental paradigm shift in designing intelligent distributed systems. As demonstrated, their synergy unlocks capabilities far exceeding the sum of their individual parts. By enabling powerful LLMs to learn from diverse, real-world, privacy-sensitive IoT data through the secure framework of FL, we can create adaptive, context-aware, and specialized AI solutions deployable at the network edge. This synergy directly addresses the limitations inherent in previous approaches, paving the way for truly intelligent, efficient, and trustworthy applications across critical domains like Industrial IoT, autonomous systems, and smart infrastructure. While unique challenges arise from this tight integration, they also define fertile ground for future research focused on realizing the full, transformative potential of this powerful technological triad.

6. Key Challenges and Mitigation Strategies

In this section, we identify the key challenges of the synergy of IoT, LLM, and FL, and suggest potential mitigation strategies based on relevant techniques found in the open literature. Table 6 summarizes the main challenges and mitigation methods, as elaborated on next.

Table 6. Major challenges in integrating IoT, LLMs, and FL, with mitigation strategies.

Challenge	Description	Mitigation Strategies	Trade-Offs/Notes
Resource Constraints (Compute, Memory, Energy)	Severe limitations on many IoT devices conflict with LLM computational demands [3,24].	Model compression [62]; split computing [3,18]; PEFT [16]; adaptive distribution.	Accuracy loss (compression); latency/sync needs (split); limited adaptivity; orchestration complexity (adaptive).
Communication Overhead	High cost of transmitting large model updates frequently over constrained IoT networks [10,111].	PEFT [16]; update compression [111]; reduced frequency [9]; asynchronous protocols [101,112].	Smaller updates limit model changes; info loss risk (compression); slower convergence (frequency); staleness issues (Async).
Data Heterogeneity (Non-IID) and Fairness	Non-IID data hinders convergence and fairness [44,113]; biases can be amplified [37]; decentralized bias mitigation is hard.	Robust Aggregation (e.g., FedProx) [44]; PFL [73,74]; fairness-aware algorithms; synthetic/public data augmentation [103,104].	Complexity (PFL); potential avg. accuracy reduction (fairness); privacy concerns with data augmentation.
Privacy and Security Risks	Balancing privacy vs. utility; protecting against leakage [110,114], poisoning [109,115], Byzantine [116], backdoor attacks [117–119]; regulatory compliance (GDPR, HIPAA).	PETs (DP [48], HE [53], SMPC [55], secure aggregation [56]); Robust Aggregation (e.g., Krum [116], PEAR [120]); attack detection [121,122]; TEEs [123]; ZKP-based methods (e.g., ByzSFL [124]).	Accuracy loss (DP) [49]; high overhead (HE/SMPC) [3]; limited protection (SecAgg); assumptions fail (Robust Agg.); verifiability (ZKP). See Table 3.
On-Demand Deployment and Scalability	Efficiently managing FL training and LLM inference across massive, dynamic IoT populations [3].	Edge infrastructure optimization (caching, serving) [3]; scalable FL orchestration (hierarchical, decentralized, asynchronous) [42,101,125]; resource-aware management [3,43].	Orchestration trade-offs; incentive complexity.

6.1. Resource Constraints

A primary obstacle when deploying LLMs within IoT ecosystems arises from the stark mismatch between the models' demands and the typically severe resource constraints of edge devices [3]. Edge units often provide limited processing power, small memory capacities (e.g., typically 1–4 GB of RAM), and must operate under strict power budgets (often ≤ 10 W) [24]. Yet, even moderately sized models, like a 7-billion-parameter LLM, can require approximately 4 GB of memory just for inference, making deployment challenging [10].

To bridge this gap and enable on-device LLM adaptation and execution, several mitigation strategies focusing on efficiency can be employed. Model compression techniques, notably quantization (e.g., to 4-bit precision), can significantly slash memory usage by roughly 75% while often preserving a high percentage (e.g., 92–97%) of the original model's accuracy on tasks like text classification [62]. Another approach is split computing, particularly SFL, which partitions the model layers between the device and a more capable edge server. This can cut on-device memory requirements substantially (e.g., by 40–60%), though it introduces trade-offs such as increased round-trip latency (e.g., 150–300 ms) during operations like federated training iterations [18]. Furthermore, PEFT methods have emerged as a highly promising strategy. Techniques like LoRA drastically reduce the number of trainable parameters by updating only a small fraction (e.g., about 1–2%) of the model's weights, achieving massive reductions (up to 98%) in parameters needing training and storage [16]. Impressively, this efficiency often comes with only a modest decrease in performance, retaining substantial percentages (e.g., around 89%) of full fine-tuning performance on standard benchmarks.

The trade-offs between parameter efficiency and task performance for various PEFT methods, including LoRA, adapter tuning, and prompt tuning compared to full fine-tuning, are clearly visualized in Figure 5. Full fine-tuning involves updating all model parameters, which leads to the highest performance but at a substantial computational and memory cost. In contrast, PEFT methods significantly reduce the number of trainable parameters—LoRA updates approximately 1% of parameters, adapters around 2%, and prompt tuning fewer than 0.1%—while still achieving competitive downstream task performance. As the figure illustrates, these methods strike different balances between efficiency and effectiveness, making them particularly attractive for resource-constrained IoT and Federated Learning settings where full fine-tuning is often impractical. This visual comparison underscores the growing importance of PEFT techniques in scaling LLM applications to diverse, decentralized edge environments.

Finally, complementing these model-level optimizations, adaptive distribution techniques employing dynamic workload schedulers can monitor real-time device telemetry (available RAM, CPU load, network bandwidth) to adjust model partitioning or batch sizes on the fly, maximizing the utilization of available resources. Together, these diverse approaches—compression, splitting, parameter-efficient adaptation, and dynamic scheduling—make it increasingly practical to deploy and adapt sophisticated LLMs effectively on resource-constrained IoT hardware.

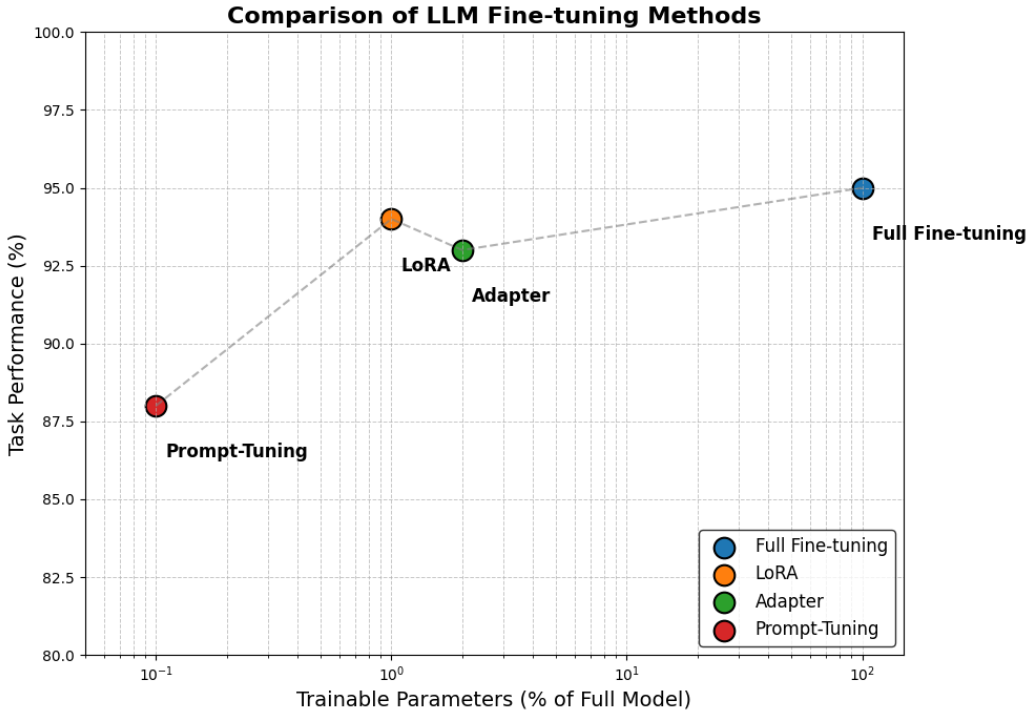

Figure 5. Trade-off between trainable parameter ratio and downstream task performance (e.g., typical accuracy observed on natural language understanding tasks from benchmarks like GLUE) for various LLM fine-tuning methods. Full fine-tuning updates 100% of parameters, whereas PEFT approaches such as LoRA (1%), adapter (2%), and prompt tuning (<0.1%) offer large savings in parameter updates at the cost of some performance.

6.2. Communication Overhead

The high communication overhead associated with FL poses another significant challenge, particularly in IoT networks characterized by potentially unreliable or low-bandwidth connections [10]. Transmitting large model updates frequently between numerous devices and a central server can saturate the network and consume considerable energy. Several approaches aim to mitigate this communication burden. As mentioned, PEFT methods are highly effective, as only the small adapter updates need to be transmitted [101]. Update compression techniques can further reduce the size of transmitted data, but carry a risk of information loss [111]. Reducing the frequency of communication rounds can save bandwidth, but typically slows down the convergence of the global model [9]. Additionally, asynchronous protocols allow devices to communicate more flexibly based on their availability, alleviating delays caused by stragglers, but they introduce challenges related to model staleness and potential inconsistencies [112].

6.3. Data Heterogeneity and Fairness

The performance and fairness of FL systems are significantly impacted by data heterogeneity, commonly referred to as Non-IID data, which is prevalent in IoT environments [113]. Data distributions often vary substantially across devices due to differing local environments, usage patterns, or sensor types (e.g., label or feature skew). This heterogeneity can hinder the convergence of standard FL algorithms like FedAvg and lead to a global model that performs poorly for specific clients. Furthermore, biases present in local data or even within the pre-trained base LLM can be amplified or unfairly distributed across participants through the FL process, and measuring or mitigating such biases in a decentralized manner remains difficult [37]. Strategies to address Non-IID data and promote fairness include using Robust Aggregation algorithms (like FedProx), designed to be less sensitive to diverging updates [44], and employing PFL techniques that tailor parts

of the model to local data, although this adds complexity [73,74]. Fairness-aware algorithms explicitly try to balance performance across different client groups, sometimes at the cost of overall average accuracy. Another approach involves augmenting local data with synthetic data (potentially generated by LLMs) or relevant public data, but this requires careful consideration of privacy implications [103,104].

6.4. Privacy and Security Risks

Ensuring robust privacy and security is perhaps the most critical challenge, given the sensitive nature of IoT data and the distributed nature of FL. Key concerns involve balancing model utility against privacy guarantees, protecting against various attacks such as data leakage from model updates [110,114], data or model poisoning by malicious clients [109,115], Byzantine failures [116], and backdoor attacks targeting the models [117–119], all while complying with regulatory mandates like GDPR or HIPAA.

A variety of techniques, often referred to as PETs and robust mechanisms, are used to mitigate these risks, each with distinct trade-offs in aspects like overhead (conceptually compared in Figure 6) and utility. DP offers strong, mathematical guarantees against inference attacks by adding calibrated noise. While generally having lower computational overhead than cryptographic methods, it introduces a direct privacy–utility trade-off, where increasing noise to enhance privacy typically degrades model accuracy [49], as illustrated conceptually in Figure 6. Cryptographic approaches like HE allow computations (like aggregation) on encrypted data, providing strong confidentiality against the server without accuracy loss, but their extremely high computational and communication overhead makes them largely impractical for direct use on most IoT clients [3,53]. Similarly, SMPC enables joint computations without revealing private inputs, offering strong security through distributed trust with no accuracy loss, but typically requires complex, multi-round interactions unsuitable for dynamic IoT environments [55]. Secure aggregation protocols are optimized specifically for the FL summation task, offering much better efficiency than general HE/SMPC and protecting individual updates from the server during aggregation, but they do not protect the final model from inference or updates during transmission without additional measures [56].

To defend against malicious clients sending faulty updates (poisoning or Byzantine attacks), Robust Aggregation methods like Krum [116], Bulyan [126], coordinate-wise median, or trimmed mean are employed to filter outlier updates. However, their effectiveness can decrease with sophisticated attacks or high Non-IID levels [127,128]. Recent advancements show promise, such as the PEAR mechanism using cosine similarity and trust scores for better robustness in Non-IID settings [120], or techniques like ByzSFL that integrate Byzantine robustness with secure computation using zero-knowledge proofs (ZKPs) for efficient verification without revealing private data [124]. Complementary strategies include explicit attack detection and verification mechanisms [121,122] and leveraging hardware security through Trusted Execution Environments (TEEs) to provide protected enclaves for computation [123].

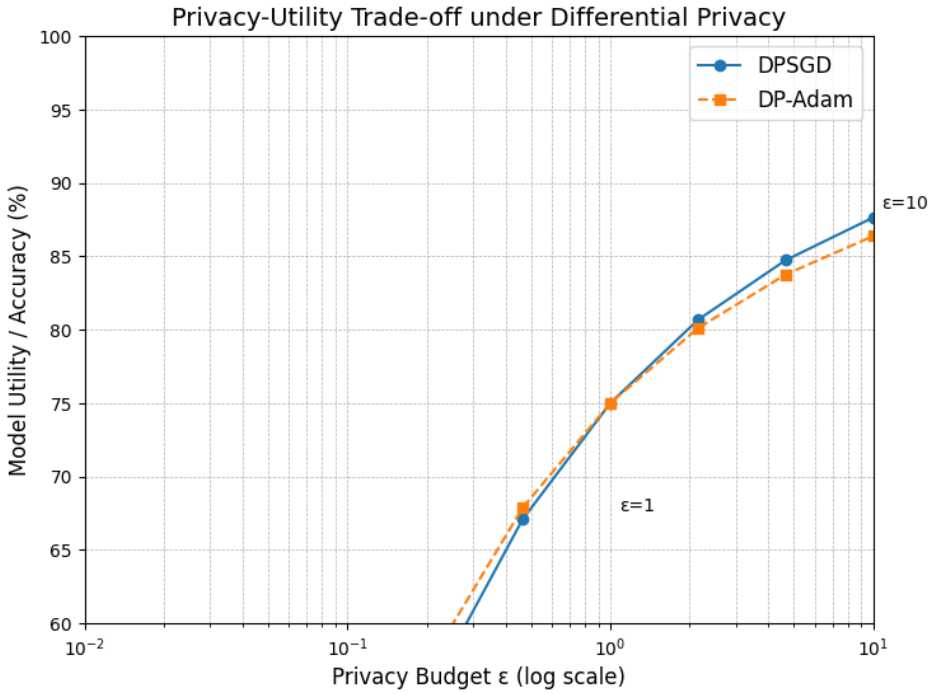

Figure 6. Illustration of the privacy–utility trade-off under differential privacy for Federated Learning models. The plot compares the model accuracy (%) of two different DP-based optimization methods, DPSGD and DP-Adam, across a range of privacy budgets (ϵ) on a logarithmic scale. As the privacy guarantee strengthens (smaller ϵ), model performance consistently degrades, highlighting the inherent trade-off between privacy protection and predictive utility. Key privacy levels (ϵ = 0.01, 0.1, 1, and 10) are annotated to demonstrate performance sensitivity.

6.5. Scalability and On-Demand Deployment

Finally, achieving efficient scalability and supporting on-demand deployment is crucial for applying FL-trained LLMs across massive and dynamic IoT populations [42]. Managing the training process and subsequent inference efficiently requires optimized edge infrastructure, including techniques like caching and optimized model serving [43]. Scalable FL orchestration is also essential, employing architectures like hierarchical, decentralized, or asynchronous FL, each presenting different trade-offs in coordination complexity, robustness to failures or stragglers, and communication latency [101]. Furthermore, effective resource-aware management, incorporating adaptive scheduling, intelligent client selection strategies, and potentially incentive mechanisms, is needed to handle the dynamic nature of device availability and network conditions [125].

7. Research Gaps and Future Directions

The IoT, LLMs, and FL have seen rapid progress, establishing a notable current state of development and research. However, despite these advancements, substantial challenges persist. This section aims to provide a structured overview by first briefly acknowledging key aspects of the current landscape within specific domains of this integration. Building on this, we then identify critical research gaps, supported by detailed evidence and insights from recent literature. Finally, based on these identified gaps, we delineate promising future directions for advancing the synergistic application of these technologies.

Efficiency for Extreme Edge: LLMs are notoriously resource-intensive, but edge IoT devices often operate on milliwatts of power with kilobytes of RAM. Techniques like QLoRA [62] reduce fine-tuning memory use by combining 4-bit quantization and low-rank adaptation, making LLMs tractable for edge execution. Similarly, SparseGPT achieves one-shot pruning with negligible accuracy drop on billion-parameter models [67]. SmoothQuant

enhances post-training quantization by aligning activations and weights to improve stability under int8 quantization [68]. Backpropagation-free training is emerging as a potential direction to eliminate memory-heavy gradient calculations; the survey in [87] reviews biologically inspired and forward–forward alternatives relevant to constrained hardware. These are particularly promising when combined with hardware-aware co-design, as advocated in [3], for FL in 6G IoT networks.

Robustness to Heterogeneity and Fairness: Extreme client heterogeneity in IoT-FL, both in data and hardware, poses serious convergence and fairness challenges. Pfeiffer et al. [24] analyze system-level disparities and advocate for client-specific adaptation layers. Carlini et al. [37] further highlight how adversarial alignment in neural networks can propagate biases, underscoring the need for fairness constraints in model design. Multi-prototype FL, as discussed in the Wevolver report [12], enables clients to specialize on subsets of prototypes that better represent their local distributions. Deng et al. [73] propose a hierarchical knowledge transfer scheme that separates global, cluster, and local models, reducing the negative transfer from outlier clients. Formal fairness-aware FL protocols, however, are still lacking.

Practical Privacy Guarantees: Applying PETs to LLM-based FL is non-trivial. While traditional DP mechanisms such as those in [45,48] remain foundational, Ahmadi et al. [49] show that when applied to LLMs in FL, DP introduces substantial performance degradation unless combined with hybrid masking and adaptive clipping strategies. Liu et al. [70] propose DP-LoRA, which selectively adds noise only to low-rank adaptation matrices, achieving a trade-off between utility and formal privacy. Yet, computational cost remains high. HE and SMPC offer stronger privacy but with significant communication and computational overheads unsuitable for IoT [53,55]. Efficient and scalable PET integration into low-power FL deployments remains an open issue.

Advanced Security and Trust: Foundation models open new attack surfaces in FL. Li et al. [118] demonstrate that compromised foundation models can inject imperceptible backdoors into global models during federated fine-tuning. Wu et al. [119] study adversarial adaptations where model updates mimic benign behavior, bypassing current anomaly detection. Existing aggregation defenses like Krum [116] and Bulyan [126] struggle when attackers use model-aligned poisoning. Fan et al. [124] propose using zero-knowledge proofs for secure update verification in FL, though integration into LLM systems is yet to be tested. Decentralized trust frameworks with verifiable integrity, such as those discussed in [42], could mitigate these threats in IoT federations.

Standardization and Benchmarking: Most existing FL benchmarks are designed for small NLP tasks (e.g., FedNLP [98]), lacking scale and modality diversity. Zhang et al. [97] introduce FederatedGPT to benchmark instruction tuning under FL settings, incorporating metrics like alignment score and robustness. FederatedScope-LLM [96] goes further, providing end-to-end support for parameter-efficient tuning (e.g., LoRA, prompt tuning) across diverse datasets. However, neither covers streaming sensor data, nor evaluates under network constraints typical in IoT. A comprehensive benchmark must include multimodal tasks, model size variability, privacy/utility/fairness trade-offs, and realistic simulation environments [129].

Multimodal Federated Learning: IoT deployments naturally involve multimodal data. ImageBind [130] demonstrates crossmodal LLMs trained on image, audio, depth, and IMU inputs in a single embedding space, but assumes centralized training. Cui et al. [105] highlight the challenges of decentralized multimodal alignment, including inter-client modality mismatch and unbalanced contributions. Communication-efficient multimodal fusion techniques and modality-specific adapters are needed. Sensor-based FL must incorporate asynchronous updates and crossmodal imputation to be practical in the real world.

Federated Learning for AI Agents: Li et al. [131] envision LLM-based AI agents capable of perception, planning, and actuation across decentralized IoT systems. Such agents require lifelong learning and task adaptation, which traditional FL lacks. PromptFL [80] proposes learning shared prompts instead of entire models, while FedPrompt [81] enhances this with privacy-preserving prompt updates. These methods significantly reduce communication and allow client-specific behavior, but lack the reasoning and memory modules required by generalist agents. Integration with reinforcement FL and safe exploration policies is a future direction.

Continual Learning and Adaptability: The temporal nature of IoT data leads to frequent concept drift. Shenaj et al. [132] propose online adaptation techniques but do not consider privacy. Wang et al. [107] review continual FL methods including regularization-based and rehearsal-based strategies. Xia et al. [108] propose FCLLM-DT, which maintains temporal awareness via digital twins. These approaches should be enhanced with memory-efficient adaptation and forgettable modules that meet legal obligations on data deletion.

Legal, Ethical, and Economic Considerations: Federated LLMs operating across jurisdictions must comply with evolving data governance policies. Cheng et al. [10] outline open legal questions in multi-party FL, such as liability for biased decisions and model misuse. Qu et al. [13] emphasize ethical concerns such as disproportionate access to computing resources and biased training data. Witt et al. [43] review incentive mechanisms like token-based payments or fairness-based credit allocation, critical for encouraging client participation. However, these are rarely tested in LLM-specific scenarios, and no consensus exists on equitable reward strategies.

Machine Unlearning and Data Erasure: Hu et al. [133] propose erasing LoRA-tuned knowledge via gradient projection and local retraining to remove specific client data contributions without damaging generalization. Patil et al. [134] leverage influence functions to reduce a sample's effect on final predictions, but require full access to model internals. Qiu et al. [135] address federated unlearning by designing reverse aggregation schemes, though practical validation on LLMs is absent. Verifiability and efficiency of unlearning remain open problems, especially in decentralized, heterogeneous FL contexts.

8. Conclusions

Before summarizing our findings, it is important to acknowledge certain limitations of this review. While we endeavored to conduct a comprehensive search across multiple prominent databases and included pre-prints to capture the latest advancements, the selection process may be subject to inherent biases. Our exclusion of non-English language articles and the specific keywords chosen might have inadvertently omitted some relevant studies. Furthermore, the field of integrating IoT, LLMs, and FL is exceptionally dynamic; consequently, new developments may have emerged subsequent to our literature search cutoff in May 2025 that are not encompassed in this work. The review's primary focus on the tripartite synergy also means that related pairwise integrations or broader technological aspects might have received less exhaustive coverage than in specialized surveys. These factors should be considered when interpreting the scope and conclusions of this review.

Bringing together the IoT, LLMs, and FL creates a powerful combination. This review has explored how this three-way synergy, backed by strong privacy techniques, paves the way for smarter, more responsive, and trustworthy distributed systems, achieving results that are not available when these technologies are used in pairs. We have mapped out the motivations, the edge-focused architectures, the key methods like PEFT and SFL that make it work, and importantly, the significant challenges involved. Making this powerful integration a reality means tackling some tough hurdles head-on. We need to find ways

to run demanding LLMs on resource-limited IoT devices using FL, manage data sharing across networks without overwhelming them, handle the inherent diversity in IoT data and systems, and ensure fairness for everyone involved. Above all, protecting user privacy and securing the entire system against attack, all while meeting legal requirements, is absolutely critical. Despite these difficulties, researchers are actively finding solutions. We are seeing progress with techniques like model compression, smarter communication strategies, personalized learning, advanced privacy methods, and robust ways to combine model updates, though finding the right balance is always key. Encouragingly, real-world applications are starting to emerge, showing the clear value of using FL to let LLMs learn from distributed IoT data privately and effectively.

However, there is still a gap between this potential and widespread, reliable use. To close this gap, the research community needs to focus on several key areas. We urgently need breakthroughs in on-device efficiency for tiny edge devices, more robust algorithms that can handle messy real-world data and potential attacks, reliable ways to guarantee privacy and fairness, standard benchmarks to measure progress fairly, and clear thinking on the legal, ethical, and economic implications. By taking on these challenges with focused, collaborative research, we can unlock the true promise of this technological convergence. Getting this right means building a future with distributed AI systems that are not only powerful and efficient but also fundamentally trustworthy and respectful of data rights, impacting critical areas from industry to healthcare and beyond.

Author Contributions: Conceptualization and methodology, H.Y., X.Y., K.W. and W.N.; software, H.Y.; validation, H.Y.; formal analysis, H.Y. and X.Y.; investigation, H.Y. and H.L.; resources, H.Y. and H.L.; writing—original draft preparation, H.Y.; writing—review and editing, H.Y., H.L., X.Y., K.W., W.N., J.A.Z. and R.P.L.; visualization, H.Y., X.Y., K.W. and W.N.; supervision, X.Y., K.W. and W.N.; project administration, J.A.Z. and R.P.L.; funding acquisition, J.A.Z. and R.P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
CFL	Centralized Federated Learning
DFL	Decentralized Federated Learning
DP	Differential Privacy
FL	Federated Learning
GDPR	General Data Protection Regulation
HE	Homomorphic Encryption
HIPAA	Health Insurance Portability and Accountability Act
IIoT	Industrial Internet of Things
IoT	Internet of Things
KD	Knowledge Distillation
LLM	Large Language Model
LoRA	Low-Rank Adaptation

Non-IID	Non-Independent and Identically Distributed
PEFT	Parameter-Efficient Fine-Tuning
PET	Privacy-Enhancing Technology
PFL	Personalized Federated Learning
SFL	Split Federated Learning
SMPC	Secure Multi-Party Computation
TEE	Trusted Execution Environment
ZKP	Zero-Knowledge Proof

References

1. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Proceedings of the Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; pp. 1877–1901.
2. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. LLaMA: Open and Efficient Foundation Language Models. *arXiv* **2023**, arXiv:2302.13971.
3. Chen, X.; Wu, W.; Li, Z.; Li, L.; Ji, F. LLM-Empowered IoT for 6G Networks: Architecture, Challenges, and Solutions. *arXiv* **2025**, arXiv:2503.13819.
4. Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T.B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; Amodei, D. Scaling Laws for Neural Language Models. *arXiv* **2020**, arXiv:2001.08361.
5. Weidinger, L.; Mellor, J.; Rauh, M.; Griffin, C.; Uesato, J.; Huang, P.S.; Cheng, M.; Glaese, M.; Balle, B.; Kasirzadeh, A.; et al. Ethical and social risks of harm from Language Models. *arXiv* **2021**, arXiv:2112.04359.
6. Mao, Y.; You, C.; Zhang, J.; Huang, K.; Letaief, K.B. A survey on mobile edge computing: The communication perspective. *IEEE Commun. Surv. Tutorials* **2017**, *19*, 2322–2358. [CrossRef]
7. Wang, J.; Liu, Z.; Yang, X.; Li, M.; Lyu, Z. The Internet of Things under Federated Learning: A Review of the Latest Advances and Applications. *Comput. Mater. Contin.* **2025**, *82*, 1–39.
8. Villalobos, P.; Sevilla, J.; Heim, L.; Besiroglu, T.; Hobbhahn, M.; Ho, A. Will We Run Out of Data? An Analysis of the Limits of Scaling Datasets in Machine Learning. *arXiv* **2022**, arXiv:2211.04325.
9. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; PMLR; pp. 1273–1282.
10. Cheng, Y.; Zhang, W.; Zhang, Z.; Zhang, C.; Wang, S.; Mao, S. Towards Federated Large Language Models: Motivations, Methods, and Future Directions. *IEEE Commun. Surv. Tutorials* **2024**, 1. [CrossRef]
11. Li, K.; Yuan, X.; Zheng, J.; Ni, W.; Dressler, F.; Jamalipour, A. Leverage Variational Graph Representation for Model Poisoning on Federated Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 116–128. [CrossRef]
12. Wevolver. Chapter 5: The Future of Edge AI. 2025. Available online: https://www.wevolver.com/article/2025-edge-ai-technology-report/the-future-of-edge-ai (accessed on 22 April 2025).
13. Qu, Y.; Ding, M.; Sun, N.; Thilakarathna, K.; Zhu, T.; Niyato, D. The frontier of data erasure: Machine unlearning for large language models. *arXiv* **2024**, arXiv:2403.15779. [CrossRef]
14. Adam, M.; Baroud, U. Federated Learning For IoT: Applications, Trends, Taxonomy, Challenges, Current Solutions, and Future Directions. *IEEE Open J. Commun. Soc.* **2024**, *5*, 7842–7877. [CrossRef]
15. Friha, O.; Ferrag, M.A.; Kantarci, B.; Cakmak, B.; Ozgun, A.; Ghoualmi-Zine, N. Llm-based edge intelligence: A comprehensive survey on architectures, applications, security and trustworthiness. *IEEE Open J. Commun. Soc.* **2024**, *5*, 5799–5856. [CrossRef]
16. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *arXiv* **2021**, arXiv:2106.09685.
17. Lin, Z.; Hu, X.; Zhang, Y.; Chen, Z.; Fang, Z.; Chen, X.; Li, A.; Vepakomma, P.; Gao, Y. Splitlora: A split parameter-efficient fine-tuning framework for large language models. *arXiv* **2024**, arXiv:2407.00952.
18. Wu, W.; Li, M.; Qu, K.; Zhou, C.; Shen, X.; Zhuang, W.; Li, X.; Shi, W. Split learning over wireless networks: Parallel design and resource management. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 1051–1066. [CrossRef]
19. Chen, H.Y.; Tu, C.H.; Li, Z.; Shen, H.W.; Chao, W.L. On the importance and applicability of pre-training for federated learning. *arXiv* **2022**, arXiv:2206.11488.
20. Khan, L.U.; Saad, W.; Han, Z.; Hossain, E.; Hong, C.S. Federated learning for internet of things: Recent advances, taxonomy, and open challenges. *IEEE Commun. Surv. Tutorials* **2021**, *23*, 1759–1799. [CrossRef]

21. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Poor, H.V. Federated learning for internet of things: A comprehensive survey. *IEEE Commun. Surv. Tutorials* **2021**, *23*, 1622–1658. [CrossRef]
22. Liu, M.; Ho, S.; Wang, M.; Gao, L.; Jin, Y.; Zhang, H. Federated learning meets natural language processing: A survey. *arXiv* **2021**, arXiv:2107.12603.
23. Zhuang, W.; Chen, C.; Lyu, L. When foundation model meets federated learning: Motivations, challenges, and future directions. *arXiv* **2023**, arXiv:2306.15546.
24. Pfeiffer, K.; Rapp, M.; Khalili, R.; Henkel, J. Federated learning for computationally constrained heterogeneous devices: A survey. *ACM Comput. Surv.* **2023**, *55*, 1–27. [CrossRef]
25. Xu, M.; Du, H.; Niyato, D.; Kang, J.; Xiong, Z.; Mao, S.; Han, Z.; Jamalipour, A.; Kim, D.I.; Shen, X.; et al. Unleashing the power of edge-cloud generative AI in mobile networks: A survey of AIGC services. *IEEE Commun. Surv. Tutorials* **2024**, *26*, 1127–1170. [CrossRef]
26. Guo, F.; Yu, F.R.; Zhang, H.; Li, X.; Ji, H.; Leung, V.C. Enabling massive IoT toward 6G: A comprehensive survey. *IEEE Internet Things J.* **2021**, *8*, 11891–11915. [CrossRef]
27. Boyes, H.; Hallaq, B.; Cunningham, J.; Watson, T. The industrial internet of things (IIoT): An analysis framework. *Comput. Ind.* **2018**, *101*, 1–12. [CrossRef]
28. Zhang, J.; Tao, D. Empowering things with intelligence: A survey of the progress, challenges, and opportunities in artificial intelligence of things. *IEEE Internet Things J.* **2020**, *8*, 7789–7817. [CrossRef]
29. Stusek, M.; Zeman, K.; Masek, P.; Sedova, J.; Hosek, J. IoT protocols for low-power massive IoT: A communication perspective. In Proceedings of the 2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Dublin, Ireland, 28–30 October 2019; pp. 1–7.
30. Ghosh, A.; Chakraborty, D.; Law, A. Artificial intelligence in Internet of things. *CAAI Trans. Intell. Technol.* **2018**, *3*, 208–218. [CrossRef]
31. Al-Turjman, F.; Alturjman, S. Context-sensitive access in industrial internet of things (IIoT) healthcare applications. *IEEE Trans. Ind. Inform.* **2018**, *14*, 2736–2744. [CrossRef]
32. Khan, W.Z.; Aalsalem, M.Y.; Khan, M.K. Communal acts of IoT consumers: A potential threat to security and privacy. *IEEE Trans. Consum. Electron.* **2018**, *65*, 64–72. [CrossRef]
33. Gong, X. Delay-optimal distributed edge computing in wireless edge networks. In Proceedings of the IEEE INFOCOM 2020—IEEE Conference on Computer Communications, Toronto, ON, Canada, 6–9 July 2020; pp. 2629–2638.
34. Ashish, V. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2027; pp. 6000–6010.
35. Lee, J.; Toutanova, K. Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
36. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training Language Models to Follow Instructions with Human Feedback. In *Proceedings of the Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2022; pp. 27730–27744.
37. Carlini, N.; Nasr, M.; Choquette-Choo, C.A.; Jagielski, M.; Gao, I.; Koh, P.W.W.; Ippolito, D.; Tramer, F.; Schmidt, L. Are aligned neural networks adversarially aligned? *arXiv* **2023**, arXiv:2306.15447.
38. Wu, N.; Yuan, X.; Wang, S.; Hu, H.; Xue, M. Cardinality Counting in "Alcatraz": A Privacy-aware Federated Learning Approach. In Proceedings of the ACM Web Conference 2024, Singapore, 13–17 May 2024; pp. 3076–3084.
39. Hu, S.; Yuan, X.; Ni, W.; Wang, X.; Hossain, E.; Vincent Poor, H. OFDMA-F^{2}L: Federated Learning With Flexible Aggregation Over an OFDMA Air Interface. *IEEE Trans. Wirel. Commun.* **2024**, *23*, 6793–6807. [CrossRef]
40. Bhavsar, M.; Bekele, Y.; Roy, K.; Kelly, J.; Limbrick, D. FL-IDS: Federated learning-based intrusion detection system using edge devices for transportation IoT. *IEEE Access* **2024**, *12*, 52215–52226. [CrossRef]
41. Tian, Y.; Wang, J.; Wang, Y.; Zhao, C.; Yao, F.; Wang, X. Federated vehicular transformers and their federations: Privacy-preserving computing and cooperation for autonomous driving. *IEEE Trans. Intell. Veh.* **2022**, *7*, 456–465. [CrossRef]
42. Beltrán, E.T.M.; Pérez, M.Q.; Sánchez, P.M.S.; Bernal, S.L.; Bovet, G.; Pérez, M.G.; Pérez, G.M.; Celdrán, A.H. Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges. *IEEE Commun. Surv. Tutorials* **2023**, *25*, 2983–3013. [CrossRef]
43. Witt, L.; Heyer, M.; Toyoda, K.; Samek, W.; Li, D. Decentral and incentivized federated learning frameworks: A systematic literature review. *IEEE Internet Things J.* **2022**, *10*, 3642–3663. [CrossRef]
44. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [CrossRef]
45. Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, 4–7 March 2006*; Proceedings 3; Springer: Berlin/Heidelberg, Germany, 2006; pp. 265–284.

46. Shan, B.; Yuan, X.; Ni, W.; Wang, X.; Liu, R.P.; Dutkiewicz, E. Preserving the privacy of latent information for graph-structured data. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 5041–5055. [CrossRef]
47. Ragab, M.; Ashary, E.B.; Alghamdi, B.M.; Aboalela, R.; Alsaadi, N.; Maghrabi, L.A.; Allehaibi, K.H. Advanced artificial intelligence with federated learning framework for privacy-preserving cyberthreat detection in IoT-assisted sustainable smart cities. *Sci. Rep.* **2025**, *15*, 4470. [CrossRef]
48. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 308–318.
49. Ahmadi, K.; Behnia, R.; Ebrahimi, R.; Kermani, M.M.; Birrell, J.; Pacheco, J.; Yavuz, A.A. An Interactive Framework for Implementing Privacy-Preserving Federated Learning: Experiments on Large Language Models. *arXiv* **2025**, arXiv:2502.08008.
50. Basu, P.; Roy, T.S.; Naidu, R.; Muftuoglu, Z.; Singh, S.; Mireshghallah, F. Benchmarking differential privacy and federated learning for bert models. *arXiv* **2021**, arXiv:2106.13973.
51. Hu, S.; Yuan, X.; Ni, W.; Wang, X.; Hossain, E.; Vincent Poor, H. Differentially Private Wireless Federated Learning With Integrated Sensing and Communication. *IEEE Trans. Wirel. Commun.* **2025**, 1. [CrossRef]
52. Yuan, X.; Ni, W.; Ding, M.; Wei, K.; Li, J.; Poor, H.V. Amplitude-Varying Perturbation for Balancing Privacy and Utility in Federated Learning. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 1884–1897. [CrossRef]
53. Paillier, P. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology—EUROCRYPT '99*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 223–238.
54. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
55. Yao, A.C. Protocols for secure computations. In Proceedings of the 23rd Annual Symposium on Foundations of Computer Science (SFCS 1982), Chicago, IL, USA, 3–5 November 1982; pp. 160–164.
56. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical secure aggregation for privacy-preserving machine learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 1175–1191.
57. Che, T.; Liu, J.; Zhou, Y.; Ren, J.; Zhou, J.; Sheng, V.S.; Dai, H.; Dou, D. Federated learning of large language models with parameter-efficient prompt tuning and adaptive optimization. *arXiv* **2023**, arXiv:2310.15080.
58. Lyu, L.; Yu, H.; Ma, X.; Chen, C.; Sun, L.; Zhao, J.; Yang, Q.; Philip, S.Y. Privacy and robustness in federated learning: Attacks and defenses. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 8726–8746. [CrossRef]
59. Xi, Z.; Chen, W.; Guo, X.; He, W.; Ding, Y.; Hong, B.; Zhang, M.; Wang, J.; Jin, S.; Zhou, E.; et al. The rise and potential of large language model based agents: A survey. *Sci. China Inf. Sci.* **2025**, *68*, 121101. [CrossRef]
60. Malaviya, S.; Shukla, M.; Lodha, S. Reducing communication overhead in federated learning for pre-trained language models using parameter-efficient finetuning. In Proceedings of the Conference on Lifelong Learning Agents, Montréal, QC, Canada, 22–25 August 2023; PMLR; pp. 456–469.
61. Jiang, J.; Jiang, H.; Ma, Y.; Liu, X.; Fan, C. Low-parameter federated learning with large language models. *arXiv* **2024**, arXiv:2307.13896.
62. Dettmers, T.; Pagnoni, A.; Holtzman, A.; Zettlemoyer, L. Qlora: Efficient finetuning of quantized llms. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 10088–10115.
63. Mao, Y.; Huang, K.; Guan, C.; Bao, G.; Mo, F.; Xu, J. Dora: Enhancing parameter-efficient fine-tuning with dynamic rank distribution. *arXiv* **2024**, arXiv:2405.17357.
64. Nasiri, H.; Garraghan, P. EDoRA: Efficient Weight-Decomposed Low-Rank Adaptation via Singular Value Decomposition. *arXiv* **2025**, arXiv:2501.12067.
65. Gu, Y.; Dong, L.; Wei, F.; Huang, M. MiniLLM: Knowledge distillation of large language models. *arXiv* **2023**, arXiv:2306.08543.
66. Naik, K. LLM Fine Tuning with LoRA. 2025. Available online: https://medium.com/@kednaik/llm-fine-tuning-with-lora-8e06f2227183 (accessed on 22 April 2025).
67. Frantar, E.; Alistarh, D. SparseGPT: Massive Language Models Can Be Accurately Pruned in One-Shot. In Proceedings of the 40th International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023; Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J., Eds.; Volume 202, Proceedings of Machine Learning Research; pp. 10280–10295.
68. Xiao, G.; Lin, J.; Seznec, M.; Wu, H.; Demouth, J.; Han, S. SmoothQuant: Accurate and Efficient Post-Training Quantization for Large Language Models. In Proceedings of the 40th International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023; Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J., Eds.; Volume 202, Proceedings of Machine Learning Research; pp. 38087–38099.
69. Tian, Y.; Wan, Y.; Lyu, L.; Yao, D.; Jin, H.; Sun, L. FedBERT: When federated learning meets pre-training. *ACM Trans. Intell. Syst. Technol. (TIST)* **2022**, *13*, 1–26. [CrossRef]
70. Liu, X.Y.; Zhu, R.; Zha, D.; Gao, J.; Zhong, S.; White, M.; Qiu, M. Differentially private low-rank adaptation of large language model using federated learning. *ACM Trans. Manag. Inf. Syst.* **2025**, *16*, 11. [CrossRef]

71. Zhang, Z.; Yang, Y.; Dai, Y.; Wang, Q.; Yu, Y.; Qu, L.; Xu, Z. FedPETuning: When federated learning meets the parameter-efficient tuning methods of pre-trained language models. In Proceedings of the Annual Meeting of the Association of Computational Linguistics, Toronto, Canada, 9–14 July 2023; Association for Computational Linguistics (ACL): Stroudsburg, PA, USA, 2023; pp. 9963–9977.
72. Ghiasvand, S.; Alizadeh, M.; Pedarsani, R. Decentralized Low-Rank Fine-Tuning of Large Language Models. *arXiv* **2025**, arXiv:2501.15361.
73. Deng, Y.; Ren, J.; Tang, C.; Lyu, F.; Liu, Y.; Zhang, Y. A hierarchical knowledge transfer framework for heterogeneous federated learning. In Proceedings of the IEEE INFOCOM 2023—IEEE Conference on Computer Communications, New York, NY, USA, 17–20 May 2023; pp. 1–10.
74. Fallah, A.; Mokhtari, A.; Ozdaglar, A. Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach. In *Proceedings of the Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; pp. 3557–3568.
75. Collins, L.; Wu, S.; Oh, S.; Sim, K.C. PROFIT: Benchmarking Personalization and Robustness Trade-off in Federated Prompt Tuning. *arXiv* **2023**, arXiv:2310.04627.
76. Yang, F.E.; Wang, C.Y.; Wang, Y.C.F. Efficient model personalization in federated learning via client-specific prompt generation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 19159–19168.
77. Yi, L.; Yu, H.; Wang, G.; Liu, X.; Li, X. pFedLoRA: Model-heterogeneous personalized federated learning with LoRA tuning. *arXiv* **2023**, arXiv:2310.13283.
78. Cho, Y.J.; Liu, L.; Xu, Z.; Fahrezi, A.; Joshi, G. Heterogeneous lora for federated fine-tuning of on-device foundation models. *arXiv* **2024**, arXiv:2401.06432.
79. Su, S.; Li, B.; Xue, X. Fedra: A random allocation strategy for federated tuning to unleash the power of heterogeneous clients. In *Computer Vision—ECCV 2024*; Springer: Cham, Switzerland, 2025; pp. 342–358.
80. Guo, T.; Guo, S.; Wang, J.; Tang, X.; Xu, W. Promptfl: Let federated participants cooperatively learn prompts instead of models–federated learning in age of foundation model. *IEEE Trans. Mob. Comput.* **2023**, *23*, 5179–5194. [CrossRef]
81. Zhao, H.; Du, W.; Li, F.; Li, P.; Liu, G. Fedprompt: Communication-efficient and privacy-preserving prompt tuning in federated learning. In Proceedings of the ICASSP 2023—2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
82. Chen, Y.; Chen, Z.; Wu, P.; Yu, H. FedOBD: Opportunistic block dropout for efficiently training large-scale neural networks through federated learning. *arXiv* **2022**, arXiv:2208.05174.
83. Sun, G.; Mendieta, M.; Luo, J.; Wu, S.; Chen, C. Fedperfix: Towards partial model personalization of vision transformers in federated learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 4988–4998.
84. Chen, D.; Yao, L.; Gao, D.; Ding, B.; Li, Y. Efficient personalized federated learning via sparse model-adaptation. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; PMLR; pp. 5234–5256.
85. Cho, Y.J.; Manoel, A.; Joshi, G.; Sim, R.; Dimitriadis, D. Heterogeneous ensemble knowledge transfer for training large models in federated learning. *arXiv* **2022**, arXiv:2204.12703.
86. Sui, D.; Chen, Y.; Zhao, J.; Jia, Y.; Xie, Y.; Sun, W. Feded: Federated learning via ensemble distillation for medical relation extraction. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 2118–2128.
87. Mei, H.; Cai, D.; Wu, Y.; Wang, S.; Xu, M. A Survey of Backpropagation-free Training For LLMS. *TechRxiv* **2024**. [CrossRef]
88. Xu, M.; Wu, Y.; Cai, D.; Li, X.; Wang, S. Federated Fine-tuning of Billion-sized Language Models Across Mobile Devices. *arXiv* **2023**, arXiv:2308.13894.
89. Qin, Z.; Chen, D.; Qian, B.; Ding, B.; Li, Y.; Deng, S. Federated full-parameter tuning of billion-sized language models with communication cost under 18 kilobytes. *arXiv* **2023**, arXiv:2312.06353.
90. Sun, J.; Xu, Z.; Yin, H.; Yang, D.; Xu, D.; Chen, Y.; Roth, H.R. Fedbpt: Efficient federated black-box prompt tuning for large language models. *arXiv* **2023**, arXiv:2310.01467.
91. Pau, D.P.; Aymone, F.M. Suitability of forward-forward and pepita learning to mlcommons-tiny benchmarks. In Proceedings of the 2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS), Berlin, Germany, 23–25 July 2023; pp. 1–6.
92. He, C.; Li, S.; So, J.; Zeng, X.; Zhang, M.; Wang, H.; Wang, X.; Vepakomma, P.; Singh, A.; Qiu, H.; et al. FedML: A Research Library and Benchmark for Federated Machine Learning. *arXiv* **2020**, arXiv:2007.13518.
93. Beutel, D.J.; Topal, T.; Mathur, A.; Qiu, X.; Fernandez-Marques, J.; Gao, Y.; Sani, L.; Li, K.H.; Parcollet, T.; de Gusmão, P.P.B.; et al. Flower: A Friendly Federated Learning Research Framework. *arXiv* **2020**, arXiv:2007.14390.
94. Arisdakessian, S.; Wahab, O.A.; Mourad, A.; Otrok, H.; Guizani, M. A survey on IoT intrusion detection: Federated learning, game theory, social psychology, and explainable AI as future directions. *IEEE Internet Things J.* **2023**, *10*, 4059–4092. [CrossRef]

95. Fan, T.; Kang, Y.; Ma, G.; Chen, W.; Wei, W.; Fan, L.; Yang, Q. FATE-LLM: A Industrial Grade Federated Learning Framework for Large Language Models. *arXiv* **2023**, arXiv:2310.10049.
96. Kuang, W.; Qian, B.; Li, Z.; Chen, D.; Gao, D.; Pan, X.; Xie, Y.; Li, Y.; Ding, B.; Zhou, J. Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 5260–5271.
97. Zhang, J.; Vahidian, S.; Kuo, M.; Li, C.; Zhang, R.; Yu, T.; Wang, G.; Chen, Y. Towards Building The Federatedgpt: Federated Instruction Tuning. In Proceedings of the ICASSP, Seoul, Republic of Korea, 14–19 April 2024.
98. Lin, B.Y.; He, C.; Zeng, Z.; Wang, H.; Huang, Y.; Dupuy, C.; Gupta, R.; Soltanolkotabi, M.; Ren, X.; Avestimehr, S. FedNLP: Benchmarking Federated Learning Methods for Natural Language Processing Tasks. *arXiv* **2021**, arXiv:2104.08815.
99. Ye, R.; Wang, W.; Chai, J.; Li, D.; Li, Z.; Xu, Y.; Du, Y.; Wang, Y.; Chen, S. Openfedllm: Training large language models on decentralized private data via federated learning. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 6137–6147.
100. Chakshu, N.K.; Nithiarasu, P. Orbital learning: A novel, actively orchestrated decentralised learning for healthcare. *Sci. Rep.* **2024**, *14*, 10459. [CrossRef]
101. Nguyen, J.; Malik, K.; Zhan, H.; Yousefpour, A.; Rabbat, M.; Malek, M.; Huba, D. Federated learning with buffered asynchronous aggregation. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Valencia, Spain, 28–30 March 2022; PMLR; pp. 3581–3607.
102. Charles, Z.; Mitchell, N.; Pillutla, K.; Reneer, M.; Garrett, Z. Towards federated foundation models: Scalable dataset pipelines for group-structured learning. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 32299–32327.
103. Zhang, T.; Feng, T.; Alam, S.; Dimitriadis, D.; Lee, S.; Zhang, M.; Narayanan, S.S.; Avestimehr, S. Gpt-fl: Generative pre-trained model-assisted federated learning. *arXiv* **2023**, arXiv:2306.02210.
104. Wang, B.; Zhang, Y.J.; Cao, Y.; Li, B.; McMahan, H.B.; Oh, S.; Xu, Z.; Zaheer, M. Can public large language models help private cross-device federated learning? *arXiv* **2023**, arXiv:2305.12132.
105. Cui, C.; Ma, Y.; Cao, X.; Ye, W.; Zhou, Y.; Liang, K.; Chen, J.; Lu, J.; Yang, Z.; Liao, K.D.; et al. A Survey on Multimodal Large Language Models for Autonomous Driving. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 1–6 January 2024; pp. 958–979.
106. Pandya, S.; Srivastava, G.; Jhaveri, R.; Babu, M.R.; Bhattacharya, S.; Maddikunta, P.K.R.; Mastorakis, S.; Piran, M.J.; Gadekallu, T.R. Federated learning for smart cities: A comprehensive survey. *Sustain. Energy Technol. Assess.* **2023**, *55*, 102987. [CrossRef]
107. Wang, L.; Zhang, X.; Su, H.; Zhu, J. A comprehensive survey of continual learning: Theory, method and application. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 5362–5383. [CrossRef]
108. Xia, Y.; Chen, Y.; Zhao, Y.; Kuang, L.; Liu, X.; Hu, J.; Liu, Z. FCLLM-DT: Enpowering Federated Continual Learning with Large Language Models for Digital Twin-based Industrial IoT. *IEEE Internet Things J.* **2025**, *12*, 6070–6081. [CrossRef]
109. Fang, M.; Cao, X.; Jia, J.; Gong, N.Z. Local Model Poisoning Attacks to Byzantine-Robust Federated Learning. In Proceedings of the 29th USENIX Security Symposium (USENIX Security 20), Virtual Event, 12–14 August 2020; pp. 1605–1622.
110. Melis, L.; Song, C.; De Cristofaro, E.; Shmatikov, V. Exploiting unintended feature leakage in collaborative learning. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 691–706.
111. Yang, Y.; Dang, S.; Zhang, Z. An adaptive compression and communication framework for wireless federated learning. *IEEE Trans. Mob. Comput.* **2024**, *23*, 10835–10854. [CrossRef]
112. Hu, C.H.; Chen, Z.; Larsson, E.G. Scheduling and aggregation design for asynchronous federated learning over wireless networks. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 874–886. [CrossRef]
113. Vahidian, S.; Morafah, M.; Chen, C.; Shah, M.; Lin, B. Rethinking data heterogeneity in federated learning: Introducing a new notion and standard benchmarks. *IEEE Trans. Artif. Intell.* **2023**, *5*, 1386–1397. [CrossRef]
114. Zhu, L.; Liu, Z.; Han, S. Deep Leakage from Gradients. In *Proceedings of the Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019, pp. 14774–14784.
115. Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; Shmatikov, V. How to Backdoor Federated Learning. In Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS), Palermo, Italy, 26–28 August 2020; Chiappa, S., Calandra, R., Eds.; Volume 108, Proceedings of Machine Learning Research; pp. 2938–2948.
116. Blanchard, P.; El Mhamdi, E.M.; Guerraoui, R.; Stainer, J. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. In *Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017)*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; pp. 119–129.
117. Li, C.; Pang, R.; Xi, Z.; Du, T.; Ji, S.; Yao, Y.; Wang, T. An embarrassingly simple backdoor attack on self-supervised learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 4367–4378.
118. Li, X.; Wang, S.; Wu, C.; Zhou, H.; Wang, J. Backdoor threats from compromised foundation models to federated learning. *arXiv* **2023**, arXiv:2311.00144.

119. Wu, C.; Li, X.; Wang, J. Vulnerabilities of foundation model integrated federated learning under adversarial threats. *arXiv* **2024**, arXiv:2401.10375.
120. Sun, H.; Zhang, Y.; Zhuang, H.; Li, J.; Xu, Z.; Wu, L. PEAR: Privacy-preserving and effective aggregation for byzantine-robust federated learning in real-world scenarios. *Comput. J.* **2025**, bxae086. [CrossRef]
121. Gu, Z.; Yang, Y. Detecting malicious model updates from federated learning on conditional variational autoencoder. In Proceedings of the 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS), Portland, OR, USA, 17–21 May 2021; pp. 671–680.
122. Zhang, Z.; Cao, X.; Jia, J.; Gong, N.Z. Fldetector: Defending federated learning against model poisoning attacks via detecting malicious clients. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2022; pp. 2545–2555.
123. Huang, W.; Wang, Y.; Cheng, A.; Zhou, A.; Yu, C.; Wang, L. A fast, performant, secure distributed training framework for LLM. In Proceedings of the ICASSP 2024—2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; pp. 4800–4804.
124. Fan, Y.; Zhu, R.; Wang, Z.; Wang, C.; Tang, H.; Dong, Y.; Cho, H.; Ohno-Machado, L. ByzSFL: Achieving Byzantine-Robust Secure Federated Learning with Zero-Knowledge Proofs. *arXiv* **2025**, arXiv:2501.06953.
125. Wang, Z.; Xu, H.; Liu, J.; Huang, H.; Qiao, C.; Zhao, Y. Resource-efficient federated learning with hierarchical aggregation in edge computing. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10.
126. El Mhamdi, E.M.; Guerraoui, R.; Rouault, S. The Hidden Vulnerability of Distributed Learning in Byzantium. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018; Dy, J., Krause, A., Eds.; Volume 80, Proceedings of Machine Learning Research, pp. 1863–1872.
127. Li, S.; Ngai, E.C.H.; Voigt, T. An experimental study of byzantine-robust aggregation schemes in federated learning. *IEEE Trans. Big Data* **2024**, *10*, 975–988. [CrossRef]
128. Wu, Z.; Ling, Q.; Chen, T.; Giannakis, G.B. Federated variance-reduced stochastic gradient descent with robustness to byzantine attacks. *IEEE Trans. Signal Process.* **2020**, *68*, 4583–4596. [CrossRef]
129. Zhou, T.; Yan, H.; Han, B.; Liu, L.; Zhang, J. Learning a robust foundation model against clean-label data poisoning attacks at downstream tasks. *Neural Netw.* **2024**, *169*, 756–763. [CrossRef]
130. Girdhar, R.; El-Nouby, A.; Liu, Z.; Singh, M.; Alwala, K.V.; Joulin, A.; Misra, I. Imagebind: One embedding space to bind them all. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 15180–15190.
131. Li, Y.; Wen, H.; Wang, W.; Li, X.; Yuan, Y.; Liu, G.; Liu, J.; Xu, W.; Wang, X.; Sun, Y.; et al. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv* **2024**, arXiv:2401.05459.
132. Shenaj, D.; Toldo, M.; Rigon, A.; Zanuttigh, P. Asynchronous federated continual learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 5055–5063.
133. Hu, Z.; Zhang, Y.; Xiao, M.; Wang, W.; Feng, F.; He, X. Exact and efficient unlearning for large language model-based recommendation. *arXiv* **2024**, arXiv:2404.10327.
134. Patil, V.; Hase, P.; Bansal, M. Can sensitive information be deleted from llms? objectives for defending against extraction attacks. *arXiv* **2023**, arXiv:2309.17410.
135. Qiu, X.; Shen, W.F.; Chen, Y.; Cancedda, N.; Stenetorp, P.; Lane, N.D. Pistol: Dataset compilation pipeline for structural unlearning of llms. *arXiv* **2024**, arXiv:2406.16810.