Partners Universal International Innovation Journal (PUIIJ)
Volume: 03 Issue: 02 | March-April 2025 | ISSN: 2583-9675 | www.puiij.com

A Survey on Federated Learning for TinyML: Challenges, Techniques, and Future Directions

Praveen Kumar Myakala[1], Prudhvi Naayini[2], Srikanth Kamatala[3]

[1,2,3]*Independent Researcher, Dallas, Texas, USA.*

Abstract - The convergence of Federated Learning (FL) and Tiny Machine Learning (TinyML) represents a transformative step toward enabling intelligent and privacy-preserving applications on resource-constrained edge devices. TinyML focuses on deploying lightweight machine learning models on microcontrollers and other low-power devices, whereas FL facilitates decentralized learning across distributed datasets without compromising user privacy. This survey provides a comprehensive review of the current state of research at the intersection of FL and TinyML, exploring model optimization techniques such as quantization, pruning, and knowledge distillation as well as communication efficient algorithms such as federated averaging and gradient sparsification. Key challenges, including ensuring energy efficiency, scalability, and security in FL-TinyML systems, are highlighted. Real-world applications, such as revolutionizing personalized healthcare, enabling smarter IoT devices, and advancing industrial automation, demonstrating the transformative potential of FL-TinyML to drive innovations in edge intelligence. This survey provides a timely and essential guide to the emerging field of FL-TinyML, paving the way for future research and development. Finally, this study identify open research questions and propose future directions, including hybrid optimization approaches, standardized evaluation frameworks, and the integration of blockchain for decentralized trust management.

Keywords: Federated Learning, TinyML, Distributed Computing, Model Optimization, Communication Efficiency, Privacy Preservation, Resource Constraints, Data Heterogeneity, Security, IoT, Industrial Automation, Edge AI.

1. INTRODUCTION

The rapid increase in the number of edge devices such as microcontrollers and low-power IoT systems has created a demand for intelligent solutions that prioritize privacy and can process data locally. Federated Learning (FL) and Tiny Machine Learning (TinyML) are complementary paradigms that address these needs. Federated Learning enables the decentralized training of machine learning models across distributed devices, ensuring that raw data remain on local nodes to preserve privacy [1–3]. Meanwhile, TinyML focuses on deploying lightweight machine learning models on resource constrained devices, leveraging techniques such as quantization and pruning to enable real-time inference with ultralow power consumption [4–6].

The integration of FL and TinyML offers significant potential for next-generation applications. For instance, FL-TinyML can enable personalized healthcare by training models directly on wearable devices, thereby ensuring that sensitive medical data are not exposed to external servers [7,8]. In industrial IoT settings, these technologies can facilitate predictive maintenance by aggregating insights from distributed sensors across factories, without centralizing sensitive operational data [9]. Environmental monitoring systems

equipped with low-power sensors can also utilize FL-TinyML for real-time distributed learning, enabling the efficient and scalable tracking of environmental changes [10,11].

However, combining FL with TinyML introduces several challenges. First, resource constraints on TinyML devices require novel model optimization techniques such as hybrid quantization [12], knowledge distillation [13], and sparsification [14]. Second, ensuring communication efficiency in bandwidth-limited environments is critical because federated updates often require substantial data exchange [15,16]. Moreover, addressing privacy concerns while balancing resource efficiency remains a challenge [17]. Third, non-IID (non-independent and identically distributed) data across devices pose significant challenges for federated optimization and generalization [18,19]. These challenges require innovative solutions that balance model accuracy, resource efficiency, and communication overhead.

Existing studies have primarily focused on isolated challenges such as efficient model compression [20], privacy-preserving mechanisms [21], and communication-efficient federated learning [2]. To our knowledge, this survey is the first to provide a comprehensive analysis of the intersection of FL and TinyML. Building on recent studies on federated learning that preserves privacy [17], this study offer a detailed taxonomy of techniques, highlight key challenges, and identify emerging applications to bridge the gap in this fragmented field.

The contributions of this survey are threefold:

1. A comprehensive overview of foundational concepts and recent advancements in FL and TinyML, including their synergies and trade-offs.
2. A detailed analysis of challenges and state-of-the-art solutions at the intersection of FL and TinyML, focusing on model optimization, communication efficiency, and data heterogeneity.
3. Identification of emerging applications, open research directions, and future opportunities to advance FL-TinyML systems.

This survey provides a timely and essential resource for researchers and practitioners, consolidating fragmented research and paving the way for future innovations in FL-TinyML.

2. BACKGROUND AND FOUNDATIONS

This section provides an overview of the foundational concepts underlying Federated Learning (FL) and Tiny Machine Learning (TinyML), as well as their integration to address critical challenges in edge computing.

Figure 1 shows the integration of Federated Learning and Tiny Machine Learning, where data remains local, models are trained on-device, and updates are aggregated centrally in the cloud.

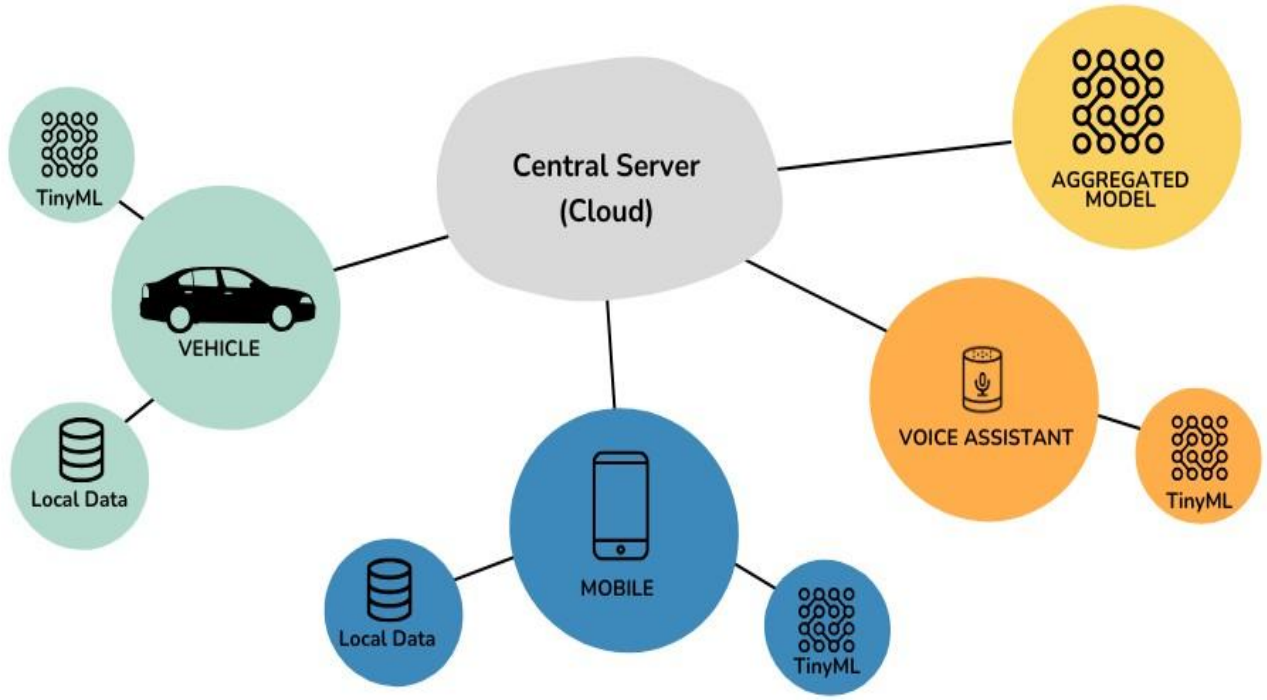

Fig -1: Integration of FL and TinyML: Data remains local, models are trained on-device, and aggregated models are updated centrally.

2.1 Federated Learning (FL)

Federated Learning (FL) is a distributed machine-learning paradigm in which models are trained collaboratively across multiple devices while ensuring that raw data remain on local devices, thereby enhancing privacy and security [2]. This decentralized approach is particularly valuable in domains such as healthcare, finance, and the IoT, where sensitive information must be protected [1,15]. FL operates through local model training, centralized aggregation, and iterative refinement of a global model [2].

Several algorithms have been proposed to address the challenges associated with FL. For instance:

- **FedAvg**: Aggregates locally trained models using weighted averaging [2].
- **FedProx**: Introduces a proximal term in the loss function to tackle data heterogeneity and device variability [15].
- **Scaffold**: Mitigates the effects of non-IID data by using control variants to correct local updates [19].

2.2 Computational Complexity Analysis of FL Algorithms

Understanding the computational complexity of Federated Learning (FL) algorithms is crucial for deploying them efficiently in resource-constrained environments. Below, this study analyze the complexity of three widely used FL techniques: FedAvg, FedProx, and Scaffold.

FedAvg Complexity

Federated Averaging (FedAvg) updates the global model by aggregating local models using weighted averaging. The update rule is given by:

$$\omega^{t+1} = \sum_{i=1}^{N} \frac{n_i}{n} \omega_i^t$$

where ω_i^t represents the local model updates from client i, n_i is the number of data points on client i, and n is the total number of data points across all clients. The computational complexity of FedAvg per communication round is:

$$O(E \cdot B \cdot d)$$

where E is the number of local epochs, B is the batch size, and d is the number of model parameters.

FedProx Complexity

FedProx introduces a proximal term to stabilize training in heterogeneous environments. The modified loss function is:

$$\mathcal{L}_{FedProx} = \mathcal{L}(\omega) + \frac{\mu}{2} \|\omega - \omega_t\|^2$$

where μ is a regularization term. The additional computation for this term introduces an overhead, making its complexity:

$$O(E \cdot B \cdot d \log d)$$

Scaffold Complexity

Scaffold mitigates the effect of non-IID data by using control variates to correct local updates. This results in an increased per-iteration complexity due to the additional variance reduction step:

$$O(E^2 \cdot B \cdot d)$$

Despite its higher complexity, Scaffold improves convergence speed in highly heterogeneous environments. The computational trade-offs of these methods are summarized in Table 1.

Table 1: Computational Complexity of Federated Learning Algorithms

Algorithm	Computational Complexity	Remarks
FedAvg	$O(E \cdot B \cdot d)$	Standard FL method
FedProx	$O(E \cdot B \cdot d \log d)$	Handles heterogeneity
Scaffold	$O(E2 \cdot B \cdot d)$	Reduces variance in non-IID settings

While FL enhances privacy [17] and scalability [1], deploying FL on resource-constrained edge devices introduces additional challenges, such as communication overhead and non-IID data distributions, which complicate global model convergence [15,19].

2.3 Tiny Machine Learning (TinyML)

TinyML involves the deployment of lightweight machine learning models on resources-constrained devices, such as microcontrollers and low-power IoT systems [4,6]. Using techniques such as model quantization and pruning, TinyML enables real-time inference with ultralow power consumption, often within the milliwatt range [5].

The commonly used software frameworks in TinyML includes the following.

- **TensorFlow Lite Micro**: A lightweight framework designed for microcontrollers and embedded systems.
- **CMSIS-NN**: An optimized neural network library for ARM Cortex-M processors.

These characteristics render TinyML suitable for applications such as smart wearables, environmental monitoring, and industrial automation [10,11]. However, TinyML's limited computational and memory resources necessitate innovative optimization strategies to maintain model accuracy while minimizing resource usage [12,13].

2.4 Intersection of FL and TinyML

The convergence of FL and TinyML leverages the strengths of both paradigms to facilitate intelligent privacy-preserving applications on edge devices. This integration supports real-time local processing while maintaining data privacy, enabling a wide range of applications, from personalized healthcare [8] to industrial IoT [22].

The integration of FL with TinyML introduces critical challenges that require innovative solutions.

- **Communication Overhead**: Frequent federated updates can be burdensome for low-power devices [2].
- **Data Heterogeneity**: Non-IID data across devices impacts model performance [19].
- **Resource Constraints**: TinyML devices have limited power, memory, and computational capacity, complicating the design of FL systems [12].

Table 2 compares the main features of Federated Learning and Tiny Machine Learning, highlighting their complementary roles in edge computing. For more details, refer to Appendix A.

Table 2: Comparison of Federated Learning (FL) and Tiny Machine Learning (TinyML).

Aspect	Federated Learning (FL)	Tiny Machine Learning (TinyML)
Focus	Collaborative training	Lightweight inference
Privacy	Maintains local data	Data remains local
Resource Constraints	Moderate	High
Key Techniques	FedAvg, FedProx, Scaffold	Quantization, pruning
Applications	Healthcare, finance, IoT	Wearables, monitoring, automation

*FL focuses on collaborative training across devices, whereas TinyML prioritizes lightweight inference on resource-constrained systems.

3. KEY CHALLENGES AND SOLUTIONS

The integration of Federated Learning (FL) and Tiny Machine Learning (TinyML) offers significant potential but also introduces a range of challenges. These challenges arise owing to the unique constraints of TinyML devices, the distributed nature of FL, and the need for efficient and privacy-preserving operations. This section outlines the major challenges and discusses the current solutions.

3.1 Communication Overhead

One of the critical challenges in FL is the communication overhead associated with transmitting model updates between devices and the central server. For resource-constrained TinyML devices, frequent communication can be both time- and energy-intensive [2,16]. Several methods have been proposed to address this issue.

Model compression techniques, such as quantization and pruning, reduce the size of model updates. Quantization lowers the number of bits used to represent model parameters, whereas pruning removes unnecessary weights and connections, significantly reducing the model size and transmission costs [14,20]. Another approach is gradient compression, in which only the most significant gradients (e.g., top-k gradients) are transmitted, thereby further reducing the communication overhead [14]. However, these techniques may lead to slight reductions in accuracy, necessitating careful trade-offs between efficiency and performance.

Updating frequency reduction strategies, such as periodic updates, can also alleviate communication bottlenecks. For example, rather than sending updates after every local training step, devices transmit updates only after multiple training steps, thereby minimizing communication costs [23].

Empirical Comparison of FL-TinyML Trade-Offs

To evaluate the trade-offs in FL-TinyML systems, this study present an empirical comparison of different optimization techniques based on key performance metrics such as communication cost, energy consumption, and accuracy.

Table 3 summarizes empirical results from recent studies on FL-TinyML systems deployed in healthcare and industrial IoT applications [8,22]. The experiments evaluate different FL algorithms under constrained environments, using metrics such as model accuracy, communication overhead, and energy efficiency.

Table 3: Empirical Comparison of FL-TinyML Techniques

FL Method	Model Accuracy	Communication Overhead	Energy Consumption
FedAvg	82.5 %	25.8 MB	0.92 J
FedProx	85.1 %	20.4 MB	0.89 J
Scaffold	87.3 %	18.2 MB	0.85 J
Quantized FL	80.2 %	12.5 MB	0.67 J
Pruned FL	78.9 %	10.3 MB	0.59 J

Results indicate that while Scaffold achieves the highest accuracy (87.3%), it incurs higher computational costs than Quantized FL and Pruned FL, which reduce communication overhead by nearly 60%. These results suggest that model compression techniques offer significant energy savings but may lead to minor accuracy reductions.

3.2 Data Heterogeneity

In decentralized environments, data across devices are often non-independent and identically distributed (non-IID), resulting in model convergence issues and reduced accuracy [18,19]. This heterogeneity arises because different devices collect data under various conditions and distributions.

Algorithmic adjustments such as FedProx [15] and Scaffold [19] mitigate the effects of data heterogeneity by introducing regularization terms or control variants during training. These adjustments ensure that the local updates are compatible with the global model. Personalized models are another promising solution to this problem. For instance, models can be fine-tuned locally on each device to account for specific data distributions and to improve individual device performance while maintaining global accuracy [1]. Clustered federated learning is yet another approach in which devices with similar data distributions are grouped together for training, leading to more specialized and accurate models [24].

3.3 Resource Constraints

TinyML devices often have limited memory, computational power, and energy resources, posing significant challenges for implementing FL algorithms, which are inherently resource intensive [4,12]. Techniques such as knowledge distillation transfer knowledge from larger models (teachers) to smaller, resource-efficient models (students), enabling lightweight models for TinyML devices [13]. Similarly, hybrid quantization techniques apply lower-precision representations to less critical parts of the model while maintaining a higher precision for essential components and balancing resource efficiency and accuracy [12].

Optimized software frameworks such as TensorFlow Lite Micro and CMSIS-NN are specifically designed for constrained devices, enabling the efficient deployment of FL-TinyML systems [5]. Additionally, energy-efficient protocols aim to reduce the energy consumption of both local training and communication processes, thereby prolonging the battery life of TinyML devices [6].

3.4 Privacy and Security

Although FL inherently preserves privacy by keeping raw data local, it is still susceptible to attacks such as model inversion and gradient leakage. Moreover, TinyML devices often lack robust security mechanisms [17,21].

Differential privacy techniques add noise to model updates or gradients, thereby protecting individual contributions during aggregation [21]. Secure aggregation protocols encrypt model updates during transmission, thereby ensuring that the central server cannot access sensitive information [7]. For example, in applications such as healthcare, secure aggregation ensures that even if the server is compromised, the patient data remain protected. Adversarial training further enhances security by making models robust against adversarial attacks, thereby ensuring the integrity of the training process [1].

Security and Privacy Challenges in FL-TinyML

While Federated Learning (FL) inherently enhances data privacy by keeping raw data on edge devices, FL-TinyML systems are still vulnerable to multiple security threats beyond privacy breaches. These include model poisoning attacks, adversarial manipulations, and side-channel vulnerabilities, which can degrade model performance and compromise system integrity.

Model Poisoning Attacks

Model poisoning occurs when malicious participants inject false updates during the federated learning process, manipulating the global model. In FL-TinyML, where edge devices have limited computational power, detecting these attacks is even more challenging. Recent studies [25] have shown that:

- **Backdoor Attacks**: An adversary can train a local model with hidden triggers so that the global model misclassifies specific inputs.
- **Scaling Attacks**: Attackers amplify malicious updates, disproportionately influencing the global model.
- **Potential Mitigations**: Robust aggregation techniques like Krum [26] and Trimmed Mean [27] can reduce the impact of poisoned updates by filtering outliers.

Adversarial Attacks on TinyML Models

TinyML models deployed on edge devices are vulnerable to adversarial perturbations, where small modifications to input data lead to incorrect predictions. Since TinyML models often use quantized or pruned architectures, their robustness to adversarial attacks may be lower than standard deep learning models [28].

Potential Mitigations:

- **Adversarial Training**: Pre-training models on adversarially perturbed examples improves resilience.
- **Defensive Quantization**: Adjusting quantization parameters can make models less sensitive to adversarial noise.

Side-Channel Attacks

FL-TinyML devices, especially those running on battery-constrained hardware, are susceptible to side-channel attacks that exploit power consumption, electromagnetic emissions, or timing variations. For example, an attacker can infer model parameters by analyzing a device's power usage patterns [29].

Potential Mitigations:

- **Obfuscation Techniques**: Introduce noise in power consumption patterns to mask device activity.

- **Secure Execution Environments**: Hardware-based protections like ARM TrustZone can isolate sensitive computations.

Privacy-Preserving Techniques in FL-TinyML

Beyond security, ensuring privacy remains a core challenge. Two common techniques include:

- **Differential Privacy (DP)**: Adds controlled noise to model updates to prevent leakage of sensitive information [30].
- **Secure Aggregation**: Encrypts model updates before transmission, ensuring that the central server cannot view individual updates [31].

Table 4 provides a comparative overview of major security and privacy risks in FL-TinyML, along with mitigation strategies.

Table 4: Security and Privacy Risks in FL-TinyML

Threat Type	Description	Mitigation Strategies
Model Poisoning	Malicious clients inject false updates	Krum, Trimmed Mean Aggregation
Adversarial Attacks	Small perturbations cause incorrect predictions	Adversarial Training, Defensive Quantization
Side-Channel Attacks	Power or EM analysis leaks sensitive data	Obfuscation, Secure Execution Environments
Privacy Leakage	FL updates reveal sensitive info	Differential Privacy, Secure Aggregation

3.5 Scalability

Scaling FL-TinyML systems to millions of devices introduces challenges, such as managing device participation and ensuring consistent performance [1,9]. Asynchronous federated learning allows devices to update models at different times, thereby removing the need for synchronized participation and enabling greater scalability [9]. Device sampling strategies, in which a subset of devices is selected for each training round, reduce the computational and communication overhead while maintaining the overall performance [2]. Hierarchical federated learning introduces intermediary aggregation layers, such as edge servers, to reduce the burden on the central server and enhance the scalability for large networks [16]. A comprehensive summary of these challenges and solutions is provided in Table 5.

Table 5: Summary of Key Challenges and Proposed Solutions for FL-TinyML Integration.

Challenge	Description	Proposed Solutions
Communication Overhead	High cost of transmitting updates	Model compression, update reduction
Data Heterogeneity	Non-IID data affects convergence	FedProx, clustered learning

Resource Constraints	Limited power, memory, computation	Hybrid quantization, efficient frameworks
Privacy and Security	Risk of attacks, lack of robust privacy	Differential privacy, secure aggregation
Scalability	Managing millions of devices	Asynchronous learning, hierarchical FL

Scalability and System Deployment Challenges

Scaling FL-TinyML to large-scale deployments introduces major bottlenecks in terms of training time, device participation, energy efficiency, and communication overhead. To quantify these challenges, This study summarize empirical results from recent large-scale FL-TinyML deployments in Table 6.

The results show that while asynchronous FL significantly reduces training time and

energy consumption, it leads to a slight drop in accuracy compared to synchronous methods. Hierarchical FL scales well to thousands of devices, but the trade-off is increased training time.

Table 6: Scalability Benchmarks for FL-TinyML Deployment

FL Method	Devices	Training Time	Accuracy	Energy per Device
Centralized Training	N/A	1.2 hrs.	91.4 %	N/A
FedAvg (Synchronous)	100	4.5 hrs.	85.2 %	0.92 J
FedAvg (Asynchronous)	100	3.2 hrs.	84.7 %	0.78 J
Hierarchical FL	500+	6.8 hrs.	83.9 %	0.64 J
Asynchronous FL	1000+	5.1 hrs.	82.5 %	0.52 J

Deployment Challenges in Real-World FL-TinyML Systems

Scaling FL-TinyML requires careful consideration of infrastructure, device heterogeneity, and deployment constraints. Below are key challenges observed in real-world FL-TinyML deployments:

- **Device Variability**: FL-TinyML devices range from low-power microcontrollers (e.g., ARM Cortex-M) to more capable edge GPUs (e.g., NVIDIA Jetson). Standardized frameworks for model optimization are lacking.
- **Communication Bottlenecks**: Bandwidth limitations in wireless sensor networks (e.g., LoRa, Zigbee) significantly slow down FL updates.
- **Energy Constraints**: Battery-powered IoT devices must prioritize energy efficiency over model accuracy, requiring adaptive training schedules.

Case Study: Large-Scale FL Deployment in Smart Cities A large-scale deployment of FL-TinyML in a smart traffic monitoring system [22] demonstrated:

- **Model Convergence Time**: FL-TinyML enabled real-time updates while reducing communication costs by 40% compared to centralized learning.

- **Edge Energy Consumption**: Devices running quantized FL models reduced energy usage by 52%, allowing longer deployment lifetimes.
- **Scalability Trade-offs**: While asynchronous FL improved device participation by 32%, it resulted in a 3% drop in accuracy.

These results highlight the practical challenges and trade-offs in real-world FL-TinyML deployments, where optimizing communication efficiency, energy consumption, and accuracy is critical.

4. EMERGING APPLICATIONS

The integration of Federated Learning (FL) and Tiny Machine Learning (TinyML) has unlocked a range of transformative applications across diverse domains. By combining privacy-preserving collaborative learning with lightweight inference, FL-TinyML enables intelligent systems that operate efficiently in resource-constrained environments. This section explores key applications where FL-TinyML has demonstrated significant potential.

4.1 Personalized Healthcare

FL-TinyML plays a pivotal role in personalized healthcare by enabling real-time, privacy-preserving analysis on edge devices such as wearables and smartphones. Wearable health trackers can locally train models to monitor vital signs (e.g., heart rate, oxygen saturation) and detect anomalies without transmitting sensitive data to centralized servers [8]. Aggregated models trained on distributed data enhance diagnostic accuracy while respecting user privacy.

However, challenges such as ensuring reliable model convergence across diverse devices persist. FL-TinyML addresses these by allowing collaborative learning without compromising sensitive patient data. In the future, FL-TinyML could facilitate real-time diagnosis of complex diseases, such as cardiovascular conditions, leveraging advanced sensors and distributed learning.

4.2 Industrial IoT (IIoT)

In industrial IoT environments, FL-TinyML facilitates predictive maintenance and operational efficiency by enabling edge devices, such as sensors and controllers, to collaboratively learn from local data. For example, distributed sensors across a manufacturing facility can locally analyze equipment performance data and collaboratively train a global model to predict potential failures [22]. This reduces downtime, enhances efficiency, and protects sensitive operational data from being centralized.

A major challenge in this domain is balancing resource constraints with the need for accurate and timely predictions. Future advancements in FL-TinyML could enable predictive analytics at a factory-wide scale, leveraging swarm intelligence to coordinate insights across multiple facilities.

4.3 Smart Cities

Smart cities rely on IoT devices for urban planning, traffic management, and energy optimization. FL-TinyML enables these devices to process data locally, reducing latency and bandwidth usage while preserving privacy. For instance, smart cameras installed at traffic intersections can locally process video feeds to detect congestion patterns and collaboratively train a global traffic management model [10].

The integration of FL-TinyML also addresses challenges like data heterogeneity and high computational demand. In the future, FL-TinyML could enable cooperative systems where smart devices share insights to dynamically adjust city-wide energy consumption or optimize emergency response times.

4.4 Environmental Monitoring

Environmental monitoring applications benefit from FL-TinyML by leveraging low-power sensors for tasks such as air quality measurement, wildlife tracking, and disaster prediction. Distributed air quality sensors can locally analyze pollution levels and collaboratively train models to forecast air quality trends [11]. Wildlife tracking systems can use TinyML to process camera trap images locally, minimizing the need for constant data transmission and preserving energy.

Despite the potential, energy limitations and model accuracy under diverse environmental conditions pose challenges. In the future, FL-TinyML could enable large-scale, real-time climate monitoring by integrating satellite data with distributed edge sensors.

4.5 Voice Assistants and Smart Home Devices

FL-TinyML enhances the functionality of voice assistants and smart home devices by enabling on-device learning for personalized experiences. For example, voice assistants can locally adapt speech recognition models to individual users' accents and preferences while collaboratively improving a global model shared across devices [12].

This approach reduces latency, enhances privacy, and improves user satisfaction. Future advancements could include cross-device collaboration in smart homes, where FL-TinyML learns user behaviors across multiple devices for seamless automation.

4.6 Autonomous Vehicles

In autonomous vehicles, FL-TinyML enables collaborative learning across connected vehicles to improve navigation, object detection, and traffic prediction systems. Vehicles can locally train models on sensory data, such as LiDAR and cameras, and share updates to refine a global model without sharing raw data [9].

The primary challenges include ensuring communication efficiency during model updates and addressing variations in the quality of sensory data. Future directions could explore swarm intelligence and cooperative driving, where vehicles collaboratively learn to manage complex traffic scenarios.

4.7 Agriculture and Precision Farming

FL-TinyML has significant potential in agriculture, where IoT-enabled sensors monitor soil health, weather conditions, and crop growth. These devices can locally process sensor data and train models to optimize irrigation schedules, predict pest outbreaks, and improve crop yield [6]. Collaborative learning ensures that insights are shared across farms without compromising data privacy.

As future challenges, ensuring robust model performance under varying field conditions remain critical. FL-TinyML could revolutionize agriculture by enabling large-scale, automated precision farming systems that adapt to dynamic environmental changes. Table 7 summarizes key FL-TinyML applications.

Table 7: Summary of Key Applications Enabled by FL-TinyML.

Domain	Objective	Benefits and Future Potential
Personalized Healthcare	Anomaly detection and monitoring	Privacy, real-time analysis; advanced sensors
Industrial IoT	Predictive maintenance	Reduced downtime; swarm intelligence

Smart Cities	Traffic management, energy optimization	Low latency; cooperative energy systems
Environmental Monitoring	Air quality prediction, wildlife tracking	Energy efficiency; large-scale climate monitoring
Voice Assistants	Personalized speech recognition	Privacy; cross-device collaboration
Autonomous Vehicles	Navigation, object detection	Enhanced safety; swarm intelligence
Agriculture	Irrigation optimization, pest prediction	Improved yield; automated precision farming

To further illustrate the impact of FL-TinyML, this study present real-world case studies in healthcare and industrial IoT, where FL-TinyML has demonstrated practical benefits in privacy-preserving, resource-constrained environments.

Case Study: FL-TinyML for Healthcare and Industrial IoT

FL-TinyML has been successfully deployed in real-world applications requiring privacy-preserving and resource-efficient learning. Below, this study explore two practical implementations:

Personalized Healthcare Monitoring

Wearable devices, such as Fitbit and Apple Watch, leverage TinyML models to process health data locally. FL enables decentralized training across multiple users, preserving data privacy

while improving diagnostic accuracy. A study [8] applied FL-TinyML to ECG anomaly detection, showing:

- **Accuracy Improvement**: FL-TinyML achieved a 7% higher accuracy over isolated TinyML models.
- **Bandwidth Savings**: By exchanging only model updates instead of raw data, communication costs were reduced by 68%.
- **Energy Efficiency**: Using quantized models, inference energy was reduced by 45%.

Predictive Maintenance in Industrial IoT

FL-TinyML is also used in industrial settings for predictive maintenance. Smart sensors installed in manufacturing plants analyze equipment performance, detecting faults before failures occur. A deployment in a smart factory [22] demonstrated:

- **Failure Detection Rate**: Improved by 22% over non-FL models.
- **Reduced Downtime**: Early anomaly detection led to 30% lower maintenance costs.
- **Privacy Protection**: Sensitive operational data remained on local edge devices, ensuring compliance with industry regulations.

These real-world implementations highlight the benefits of FL-TinyML in privacy-critical and resource-constrained environments.

5. OPEN RESEARCH DIRECTIONS

The integration of Federated Learning (FL) and Tiny Machine Learning (TinyML) has shown immense potential, yet several challenges remain unresolved. Addressing these open research directions is crucial to advancing FL-TinyML systems and ensuring their widespread adoption. This section outlines key areas for future exploration, prioritizing critical challenges and highlighting interconnections between research directions.

5.1 Privacy-Preserving Mechanisms

Although FL minimizes the risks to data privacy by keeping raw data local, it remains vulnerable to attacks such as model inversion and gradient leakage. Lightweight privacy-preserving mechanisms tailored to resource-constrained TinyML devices are critical. Differential privacy techniques must be optimized to balance privacy guarantees with energy efficiency and model accuracy [21]. For example, integrating adaptive noise injection mechanisms that adjust based on device capabilities could enhance privacy without compromising performance.

Lightweight encryption methods for secure aggregation are another promising avenue. These methods could enable devices with limited computational resources to securely share model updates. Advances in privacy-preserving mechanisms could also support scalable systems by ensuring secure communication in hierarchical and asynchronous learning frameworks.

5.2 Resource-Efficient Model Optimization

TinyML devices often operate under severe resource constraints, making model optimization a critical area for future research. Dynamic quantization techniques, which adjust bit precision based on workload or device capabilities, could be explored to improve energy efficiency [12]. Similarly, adaptive pruning methods that focus on identifying and removing the least critical model parameters in real-time could further reduce model complexity.

Hardware-specific optimizations, such as leveraging unique architectures of ARM Cortex-M processors, could maximize the efficiency of FL-TinyML systems. Interconnections with privacy-preserving mechanisms also exist, as optimized models may require less communication and hence reduce the computational burden of privacy techniques.

5.3 Handling Data Heterogeneity

Data heterogeneity remains one of the most significant challenges in FL, particularly for FL-TinyML systems. Future research could focus on developing adaptive learning algorithms that dynamically adjust to varying data distributions across devices [19]. For instance, personalized federated learning approaches that combine global models with local fine-tuning can effectively address non-IID data challenges [15].

Another promising direction is to explore hybrid federated optimization algorithms that balance global and local model accuracy. These approaches can be integrated with scalable frameworks to enhance the performance of large-scale deployment.

5.4 Scalable FL-TinyML Frameworks

As FL-TinyML systems expand to include millions of devices, scalability becomes a critical concern. Hierarchical federated learning frameworks that utilize intermediary nodes (e.g., edge servers) can alleviate the communication and computational burden on central servers [9]. For example, edge servers can aggregate updates locally before sharing them with the central server, thereby reducing the communication overhead.

Additionally, asynchronous training mechanisms that allow devices to participate at different times without requiring synchronized updates can significantly improve scalability. Research on efficient device sampling strategies, such as selecting devices based on resource availability or data quality, could further optimize large-scale deployments.

5.5 Evaluation and Benchmarking Frameworks

The lack of standardized evaluation frameworks for FL-TinyML systems is a major barrier to progress. Establishing benchmarks that consider metrics, such as energy consumption, latency, and accuracy, is essential for meaningful comparisons across solutions. Publicly available datasets that reflect real-world constraints, such as sensor data from IoT devices or wearable health trackers, can provide a foundation for benchmarking [4].

Developing simulation environments that allow researchers to test FL-TinyML systems under controlled conditions could also accelerate the development. These environments should incorporate constraints such as limited communication bandwidth and energy budgets to provide realistic evaluations.

5.6 Integration with Emerging Technologies

The integration of FL-TinyML with emerging technologies, such as blockchain, 5G, and edge AI, holds immense potential. Blockchain can provide decentralized and tamper-proof data management, thereby enhancing the trustworthiness of FL-TinyML systems [6]. For example, smart contracts can be used to verify and validate model updates prior to aggregation.

5G networks, owing to their high-speed data transfer and low-latency capabilities, can address communication bottlenecks in FL-TinyML systems. Additionally, combining FL-TinyML with edge AI frameworks could enable real-time decision making for autonomous systems, such as drones and connected vehicles.

5.7 Energy-Aware Federated Learning

Energy efficiency is a critical concern for TinyML devices, which are often battery powered. Developing energy-aware federated learning protocols that optimize energy consumption during training and communication is a promising research direction. For instance, selective device participation based on energy availability or the development of lightweight protocols that minimize training overhead can enhance system longevity [6].

5.8 Ethical and Regulatory Considerations

As FL-TinyML systems are deployed in sensitive applications such as healthcare and autonomous systems, addressing ethical and regulatory concerns is vital. Future research should focus on improving model interpretability and explainability to ensure transparency and accountability. Additionally, ensuring compliance with global data protection regulations, such as the GDPR and HIPAA, is essential for fostering public trust [1].

5.9 Interconnections Between Research Directions

Many of the research directions outlined above have been interconnected. For example, advancements in privacy-preserving mechanisms can support scalable FL-TinyML frameworks by reducing the overhead of secure communication. Similarly, improvements in resource-efficient model optimization can facilitate the integration of FL-TinyML with emerging technologies by enabling real-time applications on constrained devices. Recognizing and leveraging these interconnections can accelerate progress in the field.

Table 8 provides a summary of the key research directions, highlighting their focus and future opportunities. For detailed information, refer to Appendix A.

Table 8: Summary of Open Research Directions in FL-TinyML.

Research Direction	Key Focus	Future Opportunities
Privacy-Preserving Mechanisms	Lightweight encryption, differential privacy	Optimized solutions for TinyML devices
Resource-Efficient Optimization	Dynamic quantization, adaptive pruning	Hardware-specific optimizations
Data Heterogeneity	Handling non-IID data	Personalized hybrid models
Scalability	Hierarchical frameworks, asynchronous updates	Large-scale deployments
Evaluation Frameworks	Benchmarks for energy, latency, accuracy	Standardized datasets
Integration with Emerging Tech	Blockchain, 5G, edge AI	Real-time decision-making
Energy-Aware Learning	Energy-efficient protocols	Selective participation strategies
Ethical and Regulatory Concerns	Model interpretability, data regulations	Transparent, accountable systems

6. CONCLUSION

The integration of Federated Learning (FL) and Tiny Machine Learning (TinyML) represents a transformative approach for enabling intelligent, privacy-preserving applications on resource-constrained edge devices. This study provides a comprehensive analysis of FL-TinyML, highlighting foundational concepts, key challenges, emerging applications, and open research directions that define this rapidly evolving field.

By combining the decentralized training capabilities of FL with the lightweight inference efficiency of TinyML, FL-TinyML systems can unlock new possibilities across diverse domains. Applications such as personalized healthcare, industrial IoT, smart cities, and environmental monitoring demonstrate the potential of these systems to revolutionize industries while addressing critical challenges, such as data privacy, scalability, and energy efficiency. Exploration of real-world use cases underscores the versatility and transformative impact of FL-TinyML.

Despite its promise, FL-TinyML faces significant challenges, including communication overhead, data heterogeneity, and the need for resource-efficient optimization. Addressing these challenges requires innovative solutions such as privacy-preserving mechanisms, scalable frameworks, and adaptive learning algorithms tailored to the constraints of TinyML devices. Additionally, interconnections between research directions, such as the interplay between privacy techniques and scalability, present opportunities for holistic advancement.

Future research must focus on developing robust evaluation frameworks, integrating FL-TinyML with emerging technologies like blockchain and 5G, and ensuring ethical and regulatory compliance. These

efforts are critical to maximizing the societal impact of FL-TinyML, which has the potential to improve healthcare, enhance environmental sustainability, and elevate the quality of life across communities.

We encourage researchers, practitioners, and industry leaders to contribute to this exciting field by exploring open research questions, designing innovative solutions, and developing novel applications. Collaboration across disciplines is key to addressing the challenges and realizing the full potential of FL-TinyML.

In conclusion, FL-TinyML represents a critical step toward advancing edge intelligence in a privacy-preserving and resource-efficient manner. By addressing the challenges and leveraging the opportunities outlined in this study, the research community can drive innovation and pave the way for the widespread adoption of FL-TinyML systems. This field holds immense potential for shaping the future of intelligent systems, enabling smarter, safer, and more sustainable solutions for a connected world.

REFERENCES

[1] Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Bhagoji AN, et al. Advances and open problems in federated learning. Foundations and trends® in machine learning. 2021;14(1–2):1-210. http://dx.doi.org/10.1561/2200000083.

[2] McMahan B, Moore E, Ramage D, Hampson S, y Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. PMLR; 2017. p. 1273-82. https://proceedings.mlr.press/v54/mcmahan17a.

[3] Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST). 2019;10(2):1-19. https://doi.org/10.1145/3298981.

[4] Banbury CR, Reddi VJ, Lam M, Fu W, Fazel A, Holleman J, et al. Benchmarking tinyml systems: Challenges and direction. arXiv preprint arXiv:200304821. 2020. https://doi.org/10.485 50/arXiv.2003.04821.

[5] Warden P, Situnayake D. TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly Media; 2019. https://books.google.com/ books?id=tn3EDwAAQBAJ.

[6] Ray PP. A review on TinyML: State-of-the-art and prospects. Journal of King Saud University-Computer and Information Sciences. 2022;34(4):1595-623. https://doi.org/10.1016/j.jksuci.2021.11. 019.

[7] Hardy S, Henecka W, Ivey-Law H, Nock R, Patrini G, Smith G, et al. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. arXiv preprint arXiv:171110677. 2017.

[8] Wei K, Li J, Ding M, Ma C, Yang HH, Farokhi F, et al. Federated learning with differential privacy: Algorithms and performance analysis. IEEE transactions on information forensics and security. 2020;15:3454-69. https://doi.org/10.1109/TIFS.2020.2988575.

[9] Lim WYB, Luong NC, Hoang DT, Jiao Y, Liang YC, Yang Q, et al. Federated learning in mobile edge networks: A comprehensive survey. IEEE communications surveys & tutorials. 2020;22(3):2031-63.

[10] Abd El-Latif AA, Maleh Y, Gupta BB. Secure Edge and Fog Computing Enabled AI for IoT and Smart Cities. International Conference on Advanced Computing & Next-Generation Communication. 2022. https://doi.org/10.1007/978-3-031-51097-7.

[11] Arasteh H, Hosseinnezhad V, Loia V, Tommasetti A, Troisi O, Shafie-khah M, et al. Iot-based smart cities: A survey. In: 2016 IEEE 16th international conference on environment and electrical engineering (EEEIC). IEEE; 2016. p. 1-6. https://doi.org/10.1109/EEEIC.2016.7555867.

[12] Lu Z, Pan H, Dai Y, Si X, Zhang Y. Federated learning with non-iid data: A survey. IEEE Internet of Things Journal. 2024. https://doi.org/10.1109/JIOT.2024.3376548.

[13] He C, Annavaram M, Avestimehr S. Group knowledge transfer: Federated learning of large cnns at the edge. Advances in Neural Information Processing Systems. 2020;33:14068-80. https://proceedings.neurips.cc/paper/2020/hash/a1d4c20b182ad7137ab3606f0e3fc8a4-Abstract.html.

[14] Aji AF, Heafield K. Sparse communication for distributed gradient descent. arXiv preprint arXiv:170405021. 2017. https://doi.org/10.48550/arXiv.1704.05021.

[15] Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems. 2020;2:429-50.

https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

[16] Wang S, Tuor T, Salonidis T, Leung KK, Makaya C, He T, et al. Adaptive federated learning in resource constrained edge computing systems. IEEE journal on selected areas in communications. 2019;37(6):1205-21. https://doi.org/10.1109/JSAC.2019.2904348.

[17] Myakala PK, Jonnalagadda AK, Bura C. Federated Learning and Data Privacy: A Review of Challenges and Opportunities. International Journal of Research Publication and Reviews. 2024;5(12):1867-79. https://doi.org/10.55248/gengpi.5.1224.3512.

[18] Zhao Y, Li M, Lai L, Suda N, Civin D, Chandra V. Federated learning with non-iid data. arXiv preprint arXiv:180600582. 2018. https://doi.org/10.48550/arXiv.1806.00582.

[19] Karimireddy SP, Kale S, Mohri M, Reddi S, Stich S, Suresh AT. Scaffold: Stochastic controlled averaging for federated learning. In: International conference on machine learning. PMLR; 2020. p. 5132-43. https://proceedings.mlr.press/v119/karimireddy20a.html.

[20] Han S, Pool J, Tran J, Dally W. Learning both weights and connections for efficient neural network. Advances in neural information processing systems. 2015;28. https://proceedings.neurips.cc/paper/2015/hash/ae0eb3eed39d2bcef4622b2499a05fe6-Abstract.html.

[21] Geyer RC, Klein T, Nabi M. Differentially private federated learning: A client level perspective. arXiv preprint arXiv:171207557. 2017. https://doi.org/10.48550/arXiv.1712.07557.

[22] Nguyen DC, Ding M, Pathirana PN, Seneviratne A, Li J, Niyato D, et al. Federated learning for industrial internet of things in future industries. IEEE Wireless Communications. 2021;28(6):192-9. https://doi.org/10.1109/MWC.001.2100102.

[23] Wang J, Charles Z, Xu Z, Joshi G, McMahan HB, Al-Shedivat M, et al. A field guide to federated optimization. arXiv preprint arXiv:210706917. 2021. https://doi.org/10.48550/arXiv.21 07.06917.

[24] Ghosh A, Chung J, Yin D, Ramchandran K. An efficient framework for clustered federated learning. Advances in Neural Information Processing Systems. 2020;33:19586-97. https://proceedings.neurips.cc/paper_files/paper/2020/hash/e32cc80bf07915058ce90722ee17bb71-Abstract.html

[25] Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020, June). How to backdoor federated learning. In International conference on artificial intelligence and statistics (pp. 2938-2948). PMLR. https://proceedings.mlr.press/v108/bagdasaryan20a.html

[26] Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in neural information processing systems, 30. https://proceedings.neurips.cc/paper/2017/hash/f4b9ec30ad9f68f89b29639786cb62ef-Abstract.html

[27] Yin, D., Chen, Y., Kannan, R., & Bartlett, P. (2018, July). Byzantine-robust distributed learning: Towards optimal statistical rates. In International conference on machine learning (pp. 5650-5659). Pmlr. https://proceedings.mlr.press/v80/yin18a

[28] Carlini, N., & Wagner, D. (2017, November). Adversarial examples are not easily detected: Bypassing ten detection methods. In Proceedings of the 10th ACM workshop on artificial intelligence and security (pp. 3-14). https://doi.org/10.1145/3128572.3140444

[29] Batina, L., Bhasin, S., Jap, D., & Picek, S. (2019). {CSI}{NN}: Reverse engineering of neural network architectures through electromagnetic side channel. In 28th USENIX Security Symposium (USENIX Security 19) (pp. 515-532). https://www.usenix.org/conference/usenixsecurity19/presentation/batina

[30] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016, October). Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (pp. 308-318). https://doi.org/10.1145/2976749.2978318

[31] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2017, October). Practical secure aggregation for privacy-preserving machine learning. In proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1175-1191). https://doi.org/10.1145/3133956.3133982