International Journal of
Science, Architecture, Technology and Environment

Federated Learning for IoT and Edge Computing: A Comprehensive Review of Models, Optimization Strategies, Security, and Real-World Applications

Funminiyi Olagunju[1*]

[1]*Department of Electrical Engineering, North Carolina A & T State University*

[*]*Corresponding author*

DOI: https://doi.org/10.63680/ijsate0524112.06

Abstract

The rapid growth of Internet of Things (IoT) and edge computing devices has catalyzed the need for efficient, privacy-preserving machine learning methods tailored to resource-constrained and distributed environments. Federated Learning (FL) has emerged as a promising decentralized learning paradigm that enables collaborative model training without sharing raw data, thereby addressing critical privacy concerns. This review paper presents a comprehensive analysis of FL models designed specifically for IoT and edge scenarios, encompassing various architectural paradigms and personalization techniques. We explore resource-constrained optimization strategies that enhance communication, computation, and energy efficiency vital for practical deployments. Security, privacy, and trust challenges inherent in FL are examined alongside state-of-the-art defense mechanisms. Additionally, we survey real-world implementations of FL across diverse IoT domains including smart homes, industrial IoT, healthcare, and autonomous transportation. The paper concludes by discussing future research directions, highlighting the integration of FL with blockchain, 5G/6G networks, green AI, and reinforcement learning. This review aims to provide researchers and practitioners with a holistic understanding of the current landscape, challenges, and opportunities in FL for IoT and edge computing.

Keywords: Federated Learning, Internet of Things (IoT), Edge Computing, Privacy Preservation, Resource-Constrained Optimization, Security, 5G/6G Networks

1.0 Introduction

The rapid proliferation of Internet of Things (IoT) devices has ushered in an era characterized by ubiquitous connectivity and massive data generation. From smart home appliances to industrial sensors, these devices collectively produce vast volumes of data requiring efficient processing and analysis. Traditional cloud computing paradigms, which transmit data to centralized servers for processing, face significant challenges such as high latency, bandwidth constraints, and increased vulnerability to data breaches especially when handling sensitive information (Yang et al., 2019). To mitigate these issues, edge computing has emerged as a

promising solution by enabling data processing closer to the data source, either on or near IoT devices. This proximity reduces latency, alleviates bandwidth demands, and enhances data privacy (Medich, 2021). However, deploying machine learning (ML) models at the edge introduces new complexities due to limited computational resources on edge devices, necessitating lightweight ML models and efficient training approaches for feasible real-world implementation.

Federated Learning (FL) represents a decentralized ML paradigm that addresses many of these challenges by enabling multiple devices, or clients, to collaboratively train a shared global model while keeping their private data localized. Unlike traditional ML approaches that require data centralization, FL conducts training locally on the devices and only communicates model updates to a central server for aggregation (Kairouz et al., 2021). This approach is particularly well-suited to IoT and edge computing scenarios, where data privacy is paramount and computational capabilities are heterogeneous. By maintaining data on-device, FL not only helps preserve privacy but also aligns with regulatory frameworks such as the General Data Protection Regulation (GDPR), which restrict the movement and use of sensitive data (Briggs et al., 2020).

The adoption of FL in IoT and edge computing offers several distinct advantages. Primarily, FL enhances privacy preservation by ensuring sensitive data never leaves the local device, thereby mitigating risks of data breaches and unauthorized access (Briggs et al., 2020). Additionally, by transmitting only model updates instead of raw data, FL reduces communication overhead, which is critical in bandwidth-constrained and latency-sensitive IoT environments (Yang et al., 2019). FL also enables personalization at scale by allowing models to adapt to local data distributions and user behaviors, improving application relevance and performance (Nguyen et al., 2021). Furthermore, FL supports regulatory compliance by ensuring data residency and limiting cross-border data flow, thus addressing stringent data protection laws such as the GDPR (Briggs et al., 2020).

Despite its promise, FL for IoT and edge computing faces significant challenges. Resource constraints remain a major hurdle since edge devices typically have limited processing power, memory, and battery life, which restricts the complexity and scale of models that can be trained locally (Yang et al., 2019). The heterogeneity of devices, data distributions, and network conditions introduces additional complexity, causing variability in model performance and convergence (Nguyen et al., 2021). Communication bottlenecks present further challenges due to frequent synchronization requirements between devices and central servers, which may lead to network congestion and delay (Medich, 2021). Security vulnerabilities, including model poisoning and inference attacks, also threaten the integrity of FL systems in decentralized environments (Briggs et al., 2020).

This paper aims to provide a comprehensive review of Federated Learning specifically in the context of IoT and edge computing. It systematically examines FL models and architectures designed for these environments, highlighting their respective strengths and limitations. Additionally, the paper explores resource-constrained optimization strategies that improve the efficiency and scalability of FL deployments on edge devices. Finally, it reviews real-world implementations and applications across various IoT domains, such as smart healthcare, autonomous vehicles, and industrial automation. By synthesizing the current state-of-the-art research and practical deployments, this review identifies existing challenges and proposes promising directions for future research and development in FL for IoT and edge computing.

2.0 Literature Review

Many researchers have extensively investigated Federated Learning (FL) models and their applicability to IoT and edge computing environments, emphasizing how the unique constraints and characteristics of these domains shape FL design and implementation. According to Kairouz et al. (2021), FL can be categorized into several paradigms including horizontal, vertical, and federated transfer learning, each addressing different scenarios of data distribution and feature overlap among clients. Horizontal FL, the most widely applied paradigm in IoT, assumes clients share the same feature space but have different data samples, making it suitable for scenarios such as sensor networks or mobile devices collecting similar types of data (Li et al., 2020). Vertical FL, on the other hand, handles clients with different feature spaces but overlapping user sets and has been explored in contexts like cross-organizational data sharing where privacy is critical (Yang et al., 2019). Transfer FL further enhances FL's applicability by enabling knowledge transfer across domains with limited data overlap, a feature useful in heterogeneous IoT systems (Wang et al., 2020). Beyond the paradigms, research conducted by Bonawitz et al. (2019) and McMahan et al. (2017) has introduced several architectural variants of FL systems tailored for edge computing. Centralized FL, where a single server coordinates the aggregation of updates, remains the most common structure due to its relative simplicity and efficiency. However, centralized FL suffers from single points of failure and scalability bottlenecks in large IoT deployments. To mitigate these issues, hierarchical FL architectures have been proposed, where edge servers act as intermediate aggregators before forwarding updates to the cloud, reducing communication overhead and improving scalability (Liu et al., 2020). Decentralized FL, as explored by Lalitha et al. (2019), eliminates the central server entirely by enabling peer-to-peer communication between devices, offering robustness against server failures but at the cost of increased complexity in synchronization.

Personalization techniques in FL have gained significant attention, addressing the non-IID (non-independent and identically distributed) nature of IoT data, which often varies significantly across devices. Research conducted by Smith et al. (2017) introduced multi-task learning approaches to FL, enabling local models to adapt to device-specific data characteristics while sharing knowledge globally. More recently, meta-learning frameworks have been applied to FL to further enhance personalization by training models that can rapidly adapt to new devices with minimal data (Fallah et al., 2020). Clustered FL techniques, as proposed by Sattler et al. (2020), group clients with similar data distributions to improve overall model performance and reduce divergence caused by heterogeneous data.

Resource-constrained optimization strategies are critical in realizing FL on IoT and edge devices, which are often limited in computational power, memory, and energy. Communication efficiency has been a major focus of recent research. As per Lin et al. (2020), model compression techniques such as quantization and pruning significantly reduce the communication payload without substantial loss in model accuracy. Gradient sparsification and periodic update aggregation, investigated by Konečnỳ et al. (2016), have also demonstrated promise in reducing communication costs. From the computation perspective, strategies such as partial model updates and selective layer training have been proposed by Li et al. (2020), enabling devices to train only critical portions of the model to conserve resources. Additionally, scheduling algorithms that prioritize device participation based on resource availability and data relevance have been explored to optimize the training process (Wang et al., 2019).

Security and privacy remain paramount concerns in FL for IoT, due to the decentralized training process and sensitive nature of data involved. Research by Bagdasaryan et al. (2020) highlights the vulnerability of FL to model poisoning attacks, where malicious clients deliberately corrupt model updates to degrade overall

performance. In response, differential privacy mechanisms have been incorporated into FL frameworks to provide formal privacy guarantees by adding noise to local updates (Geyer et al., 2017). Secure Multi-party Computation (SMC) and homomorphic encryption techniques, as reviewed by Zhao et al. (2020), further strengthen FL systems by enabling encrypted model aggregation without exposing raw updates. Trust management systems leveraging blockchain technology have also been proposed to incentivize honest participation and prevent malicious behavior (Kim et al., 2021). Regarding real-world implementations, numerous studies have demonstrated the feasibility and benefits of FL in various IoT domains. For instance, in smart healthcare, Xu et al. (2021) showcased FL-based collaborative models across multiple hospitals, enabling disease prediction without compromising patient data privacy. In industrial IoT (IIoT), FL has been applied to predictive maintenance, allowing distributed sensors to collectively learn failure patterns (Wang et al., 2020). Autonomous vehicle networks leverage FL for cooperative learning of driving models without sharing raw sensor data, as demonstrated by Chen et al. (2020). Additionally, several open-source frameworks such as TensorFlow Federated (TFF), Flower, and NVIDIA FLARE provide essential tools that facilitate FL deployment on edge and IoT devices, contributing to accelerated research and development (Kairouz et al., 2021).

3.0 Discussion

3.1 FL Models Tailored for IoT and Edge

Federated Learning (FL) models must be adapted to meet the unique demands of IoT and edge environments, where data distribution, device capabilities, and privacy requirements vary considerably. Understanding the different FL paradigms is critical for designing efficient and effective federated systems. The three main FL paradigms Horizontal FL, Vertical FL, and Federated Transfer Learning each address distinct data partitioning scenarios and use cases in IoT and edge computing (Kairouz et al., 2021).

3.1.1 Federated Learning Paradigms

Horizontal Federated Learning (HFL)

Horizontal Federated Learning, also known as sample-partitioned FL, assumes that datasets across participating devices share the same feature space but have different samples. This means that each device or client holds data with the same attributes or features but for different users or time periods. For example, multiple IoT sensors collecting temperature readings from different locations all have the same type of data (e.g., temperature, humidity) but for distinct spatial points. HFL is the most common FL paradigm in IoT, as devices often monitor similar phenomena but collect data independently.

Research conducted by Li et al. (2020) shows that HFL is well suited to applications such as smart homes, where multiple sensors gather comparable data across households, enabling collaborative model training without centralizing sensitive data. The classical Federated Averaging (FedAvg) algorithm introduced by McMahan et al. (2017) is a prominent example of HFL, where local models trained on clients are averaged on a central server to update the global model iteratively.

Vertical Federated Learning (VFL)

Vertical Federated Learning, or feature-partitioned FL, applies to scenarios where multiple parties hold different sets of features for the same set of users. For instance, a smart manufacturing company might have

machine sensor data, while a maintenance service provider holds operational logs for the same equipment. Neither party alone has complete data to build a robust predictive model, but together, they can collaboratively train without exposing raw data to one another.

According to Yang et al. (2019), VFL enables joint model training by securely exchanging intermediate computations rather than raw data, preserving privacy while leveraging complementary feature spaces. VFL is particularly useful in industrial IoT and cross-organizational collaborations where feature spaces differ but entities share overlapping user bases or equipment identifiers.

Federated Transfer Learning (FTL)

Federated Transfer Learning is designed for cases where both the feature space and sample space are different but there exists some overlap in knowledge that can be transferred. This paradigm combines the benefits of FL and transfer learning to enable knowledge sharing between organizations or devices with heterogeneous datasets.

Wang et al. (2020) demonstrate the utility of FTL in healthcare IoT, where different hospitals may collect varying medical records and imaging data, making direct data or model sharing ineffective. FTL helps transfer learned representations across domains, improving performance on sparse or underrepresented data. This paradigm is especially relevant in IoT scenarios with diverse sensors or applications requiring cross-domain knowledge transfer.

Table 1: Illustrative Diagram Concept for Federated Learning Paradigms

FL Paradigm	Data Distribution	Example IoT Use Case	Key Idea
Horizontal FL	Same features, different samples	Smart home temperature sensors	Clients train on local data with same features but different users; models aggregated centrally.
Vertical FL	Different features, same users	Manufacturing + maintenance data	Different organizations hold complementary features of the same users/devices; joint training without sharing raw data.
Federated Transfer Learning	Different features, different users	Healthcare data across hospitals	Transfer knowledge between domains with minimal data overlap; leverages transfer learning.

3.1.2 Architecture Variants

Federated Learning architecture plays a crucial role in determining system efficiency, scalability, and robustness, especially in IoT and edge environments characterized by resource constraints and distributed device locations. The three primary architecture variants are centralized FL, hierarchical FL, and decentralized (peer-to-peer) FL.

Centralized Federated Learning

Centralized FL is the classical and most widely adopted architecture where a central server orchestrates the entire training process. In this setup, individual clients (IoT devices or edge nodes) perform local training on their private data and send model updates (e.g., gradients or weights) to the central server. The server aggregates these updates commonly through weighted averaging as in FedAvg and sends the updated global model back to the clients (McMahan et al., 2017). Centralized FL's simplicity and ease of implementation have made it the default approach in many IoT scenarios such as smart home systems and wearable health devices (Kairouz et al., 2021).

However, centralized FL faces challenges including a single point of failure, bottlenecks at the central server, and scalability issues as the number of clients grows, which may not suit large-scale IoT networks (Bonawitz et al., 2019). Furthermore, the communication overhead between clients and the central server can be substantial, particularly in bandwidth-constrained edge environments.

Hierarchical FL (Edge–Cloud Split)

Hierarchical FL introduces intermediate aggregation layers between the edge devices and the cloud to improve scalability and reduce communication latency. In this architecture, local devices first send their model updates to an edge server, which aggregates these local models before forwarding the aggregated updates to the cloud server (Liu et al., 2020). This edge–cloud split architecture takes advantage of proximity computing by offloading some computational tasks to nearby edge nodes, thus mitigating the communication load on the central server.

Hierarchical FL is particularly relevant for IoT deployments in smart cities and industrial automation, where clusters of devices are geographically close to edge servers. Research by Liu et al. (2020) demonstrates that hierarchical FL can reduce communication costs and accelerate convergence, making it suitable for latency-sensitive applications such as real-time monitoring and control.

Decentralized or Peer-to-Peer FL

Decentralized FL eliminates the central coordinating server, relying instead on direct peer-to-peer communication among clients. Devices exchange their locally trained model updates with neighboring nodes, collaboratively averaging these updates in a distributed manner (Lalitha et al., 2019). This architecture improves robustness and fault tolerance by removing single points of failure, which is advantageous in dynamic IoT networks with intermittent connectivity.

Nonetheless, decentralized FL introduces complexity in synchronization and convergence due to the lack of a central authority, and it may incur higher communication overhead among clients (Kairouz et al., 2021). Applications in vehicular networks and ad-hoc IoT systems have explored decentralized FL to leverage distributed intelligence and resilience (Lalitha et al., 2019).

3.1.3 Personalization Techniques

IoT and edge environments often exhibit high heterogeneity in data distributions and device capabilities. As a result, global FL models may not perform optimally for all clients. Personalization techniques tailor models to individual clients or clusters of clients, improving accuracy and user experience.

Clustered Federated Learning

Clustered FL groups clients based on similarities in their data distributions or model updates, training specialized models for each cluster (Sattler et al., 2020). By isolating clients with similar data characteristics, clustered FL reduces the negative effects of data heterogeneity and non-IID distributions common in IoT applications. This approach has been applied in smart healthcare, where patient data varies widely across different hospitals, allowing more accurate and personalized diagnostic models (Sattler et al., 2020).

Meta-learning-based Personalization

Meta-learning, or "learning to learn," enables models to quickly adapt to new clients using minimal local data. In the FL context, meta-learning trains a global model to serve as a good initialization point, which clients fine-tune locally to their specific data (Fallah et al., 2020). This approach is highly effective for IoT devices with limited data, allowing rapid personalization without extensive retraining. Fallah et al. (2020) demonstrate the effectiveness of meta-learning for personalized FL in resource-constrained IoT environments, showing improved adaptation speeds and model accuracy.

Transfer Learning and Fine-Tuning

Transfer learning in FL involves leveraging a pre-trained global model and fine-tuning it on local client data to capture individual characteristics (Wang et al., 2020). This technique benefits IoT scenarios where some devices have sparse or non-representative data. Fine-tuning allows the model to retain generalizable features from the global model while adapting to local nuances, which is critical in applications like autonomous vehicles and personalized health monitoring (Wang et al., 2020).

Table 2: Architecture and Personalization Variants

Variant	Description	Advantages	Challenges	Example IoT Use Case
Centralized FL	Central server aggregates client updates	Simple, efficient aggregation	Single point of failure, scalability bottleneck	Wearable health devices
Hierarchical FL	Edge servers aggregate before cloud	Reduced latency, scalable	Requires edge infrastructure	Smart cities, industrial IoT
Decentralized FL	Peer-to-peer communication without server	Fault tolerant, robust	Synchronization complexity	Vehicular networks
Clustered FL	Clients grouped by data similarity	Improved personalization	Cluster determination overhead	Healthcare data clusters
Meta-learning-based Personalization	Model learns initialization for fast adaptation	Fast client-specific adaptation	Computationally intensive	Low-data IoT devices

Transfer Learning and Fine-Tuning	Fine-tune global model on local data	Better local accuracy	Risk of overfitting on local data	Autonomous vehicles, personalized apps

3.2 Resource-Constrained Optimization Strategies

The deployment of Federated Learning (FL) in IoT and edge computing faces significant resource constraints such as limited bandwidth, computational capacity, and energy availability. To enable efficient FL under these constraints, researchers have proposed a variety of optimization strategies focusing on communication efficiency, computation and energy savings, and adaptive or asynchronous operation modes. These strategies help reduce overhead, improve convergence speed, and prolong device lifespan, ensuring FL's practical viability in real-world scenarios (Li et al., 2020; Kairouz et al., 2021).

3.2.1 Communication Efficiency

Communication overhead is one of the most critical bottlenecks in FL, especially for IoT devices connected over wireless or bandwidth-limited networks. To address this, several techniques have been developed to reduce the volume of data exchanged between clients and servers.

Model Compression (Quantization and Pruning)

Quantization reduces the precision of model parameters from 32-bit floats to lower-bit representations (e.g., 8-bit or even binary), significantly decreasing communication payloads without major accuracy loss (Alistarh et al., 2017). Similarly, pruning removes redundant or less significant connections in neural networks, resulting in sparse models with fewer parameters to transmit (Wang et al., 2018). According to research by Sattler et al. (2019), these methods can cut communication costs by up to 90% while maintaining comparable model performance.

Sparsification and Gradient Dropping

Rather than transmitting the full gradient updates, only the most significant gradients are communicated, and smaller ones are dropped or delayed. This sparsification approach is effective in reducing the size of updates and limiting network congestion (Stich et al., 2018). Jiang et al. (2018) show that sparsification with error compensation can maintain convergence guarantees despite the reduced communication.

Periodic Local Updates (FedAvg, FedPAQ)

Techniques like Federated Averaging (FedAvg) allow clients to perform multiple local training steps before sending aggregated model updates, thereby reducing communication rounds (McMahan et al., 2017). Extensions such as FedPAQ introduce periodic averaging with quantization, combining the benefits of compression and reduced communication frequency (Konečný et al., 2016). These methods effectively trade off local computation for fewer communication events, critical for edge networks.

3.2.2 Computation and Energy Efficiency

IoT and edge devices are often limited by low computational power and battery capacity. Resource-efficient FL models and training protocols are essential to accommodate these constraints.

Partial Training (Layer Freezing, Selective Updates)

Instead of training all layers of deep neural networks, partial training involves freezing some layers and only updating a subset of parameters locally (Wang et al., 2020). This reduces the computational workload and energy consumption on devices. For example, early layers pretrained on generic data may remain fixed, while final layers adapt to local client specifics.

Lightweight Models (TinyML, MobileNets)

Designing compact neural networks tailored for resource-limited devices, such as MobileNets (Howard et al., 2017) or TinyML frameworks (Warden & Situnayake, 2019), enables feasible on-device training and inference. Lightweight architectures balance accuracy and resource usage, making them ideal for FL on IoT edge devices.

Device Scheduling and Participation Control

Not all devices need to participate in every training round. Client selection strategies that consider device availability, battery level, and network conditions can optimize resource use (Nishio & Yonetani, 2019). This reduces unnecessary energy consumption and communication costs, improving overall FL efficiency.

3.2.3 Adaptive and Asynchronous FL

Adaptive strategies enhance FL's flexibility and robustness by adjusting training protocols based on device capabilities and network dynamics.

Client Selection Strategies

Intelligent client selection prioritizes devices with better resources or data quality to improve convergence speed and model accuracy (Nishio & Yonetani, 2019). Strategies may include fairness constraints to ensure diverse representation across heterogeneous IoT populations (Li et al., 2020).

Asynchronous FL

Traditional FL often operates synchronously, waiting for updates from all clients before aggregating. Asynchronous FL allows clients to send updates independently, enabling faster iteration cycles and resilience to stragglers or dropped connections (Xie et al., 2019). This is particularly beneficial in heterogeneous and intermittent IoT networks.

Event-triggered Updates

Instead of periodic communication, clients may trigger updates based on local conditions, such as significant model changes or energy thresholds (Ren et al., 2020). This strategy further reduces unnecessary communication and conserves device energy.

Table 3: Table of Resource-Constrained Optimization Strategies

Strategy	Key Techniques	Benefits	Challenges	Example IoT Application
Communication Efficiency	Quantization, pruning, sparsification, periodic updates	Reduced bandwidth usage, faster convergence	Possible accuracy trade-offs	Smart sensors over wireless networks
Computation and Energy Efficiency	Partial training, lightweight models, device scheduling	Lower energy consumption, feasible on-device training	Model capacity limits	Wearables, remote sensors
Adaptive and Asynchronous FL	Client selection, asynchronous updates, event-triggered updates	Improved robustness, reduced latency	Complexity in coordination	Dynamic IoT networks, vehicular FL

3.3 Security, Privacy, and Trust in Federated Learning

Federated Learning (FL) offers promising solutions for privacy-preserving ML by keeping raw data localized. However, this decentralized approach also introduces new security vulnerabilities, privacy risks, and challenges in establishing trust among heterogeneous participants (Li et al., 2020). To ensure reliable and secure FL systems, extensive research has focused on identifying threats, designing privacy-preserving mechanisms, and developing trust management frameworks.

3.3.1 Security Threats in FL

Several attack vectors threaten FL's integrity and reliability, particularly due to its distributed nature:

Model Poisoning: Adversaries can manipulate model updates sent to the central aggregator by submitting malicious gradients or parameters that degrade the global model's accuracy or bias it towards incorrect predictions (Bagdasaryan et al., 2020). These attacks are stealthy and difficult to detect since the aggregation process often assumes honest participants.

Data Poisoning: In this attack, compromised clients train local models on manipulated or corrupted datasets, causing the global model to learn incorrect or harmful patterns. This threat is particularly critical in IoT settings where devices might be physically accessible or less secure (Sun et al., 2020).

Backdoor Attacks: Attackers inject hidden malicious behaviors into the global model during training, which activate under specific inputs. Backdoors can cause the model to misclassify or malfunction under targeted conditions, posing risks in sensitive applications such as autonomous vehicles or healthcare (Gu et al., 2019).

3.3.2 Privacy-Preserving Techniques

To mitigate privacy risks and comply with regulations, various cryptographic and statistical methods are incorporated into FL frameworks:

Differential Privacy (DP): DP introduces carefully calibrated noise into model updates or gradients before transmission to ensure that the contribution of any single data point remains indistinguishable, thus protecting user privacy (Abadi et al., 2016). DP has been integrated into FL to prevent information leakage through model updates (Geyer et al., 2017).

Secure Multi-party Computation (SMC): SMC protocols enable multiple parties to jointly compute a function over their inputs while keeping those inputs private. Applied to FL, SMC can protect model aggregation processes from revealing individual client updates (Bonawitz et al., 2017).

Homomorphic Encryption (HE): HE allows computation directly on encrypted data without decryption. FL systems can aggregate encrypted model updates, ensuring that the server never accesses raw model parameters (Acar et al., 2018). Despite its computational overhead, HE offers strong privacy guarantees for FL.

3.3.3 Trust and Reputation Management

Trust plays a vital role in FL's collaborative framework, particularly when participants are untrusted or potentially malicious:

Trust Scoring Mechanisms: Reputation-based systems evaluate client behavior over time, scoring their reliability based on update quality, consistency, and history of participation. Trust scores help in filtering out suspicious clients and weighting their contributions (Zhao et al., 2020).

Incentive Schemes and Blockchain Integration: To encourage honest participation and mitigate freeloading, incentive mechanisms using tokens or rewards have been proposed (Khan et al., 2020). Blockchain technology has been explored to provide transparent, immutable records of client contributions, facilitating decentralized trust without central authorities (Lu et al., 2020).

3.4 Real-World Implementations and Applications

The versatility of FL makes it suitable for a wide range of IoT and edge use cases, where data privacy, latency, and bandwidth constraints are critical:

3.4.1 Smart Homes and Smart Cities

Smart meters, lighting systems, and surveillance cameras generate massive decentralized data. FL enables local training on devices to optimize energy consumption and enhance security without exposing sensitive personal data (Wang et al., 2019). For example, FL can improve anomaly detection in smart city sensors while respecting citizen privacy.

3.4.2 Industrial IoT (IIoT)

FL is applied in predictive maintenance to analyze machinery sensor data distributed across factories without centralizing sensitive industrial information (Xu et al., 2020). Robotics control and supply chain optimization also benefit from FL's ability to learn collaboratively across heterogeneous and resource-constrained devices.

3.4.3 Healthcare and Wearables

Federated hospital systems collaborate to train models for disease prediction without sharing patient records, addressing privacy and regulatory challenges (Sheller et al., 2020). Similarly, FL on wearables such as smartwatches and fitness trackers enables personalized health monitoring models that adapt locally while benefiting from collective intelligence (Rieke et al., 2020).

3.4.4 Autonomous Vehicles and Transportation

Vehicle-to-everything (V2X) communications generate large-scale data. FL enables cooperative learning among vehicles to improve safety, navigation, and traffic management without sharing raw sensor data (Chen et al., 2020). This decentralized approach enhances scalability and privacy in dynamic vehicular networks.

3.4.5 Frameworks and Tools

Several open-source frameworks and platforms support FL development and deployment:

Flower: A flexible FL framework supporting multiple ML libraries and scalable edge deployments (Beutel et al., 2020).

TensorFlow Federated: Developed by Google, this framework provides tools for simulation and deployment of FL workflows (Briggs et al., 2020).

PySyft: An open-source library focusing on privacy-preserving FL and secure computation (Ryffel et al., 2018).

NVIDIA FLARE: Designed for healthcare and edge use cases, supporting large-scale federated training on GPUs (NVIDIA, 2021).

Edge AI Hardware: Devices such as NVIDIA Jetson and Google Coral facilitate efficient on-device training and inference, enabling real-world FL applications (Lane et al., 2019).

3.5 Comparative overview of Federated Learning Model, Optimization strategies, and Real World Deployment

To better understand the landscape of Federated Learning (FL) tailored for IoT and edge computing, Table 4 below provides a comparative overview of key FL models, optimization strategies, and real-world deployments. This summary highlights their characteristics, typical use cases, advantages, and limitations, facilitating clearer insight into their practical applications and research directions.

Table 4: Comparative Summary of FL Models, Optimization Strategies, and Real-World Deployments

Category	Type/Technique	Use Case/Domain	Pros	Cons
FL Models	Horizontal FL	IoT devices with similar features	Simple data partitioning, easier aggregation	Assumes homogeneous feature space
	Vertical FL	Cross-organizational with different features	Enables collaboration on complementary data	Requires complex alignment of features
	Federated Transfer Learning	Heterogeneous data and domains	Handles non-overlapping feature and sample spaces	Computationally expensive; complex model adaptation
Optimization Strategies	Model Compression (Quantization, Pruning)	Bandwidth-constrained edge devices	Reduces communication overhead	Potential accuracy loss if overly compressed
	Sparsification and Gradient Dropping	High-frequency communication setups	Decreases communication load with minimal updates	Requires careful tuning; risk of stale gradients
	Periodic Local Updates (FedAvg, FedPAQ)	Resource-limited devices	Balances local computation and communication	May increase convergence time
	Partial Training (Layer Freezing)	Energy-constrained IoT devices	Saves computation and energy	May limit model capacity
	Lightweight Models (TinyML, MobileNets)	Mobile and embedded devices	Efficient inference and training	May reduce model expressiveness
	Client Scheduling and Participation Control	Heterogeneous device environments	Improves fairness and resource allocation	Scheduling complexity; possible bias in client selection
	Asynchronous FL	Unreliable network conditions	Reduces waiting time, improves efficiency	May cause model divergence
	Event-Triggered Updates	Dynamic edge scenarios	Reduces unnecessary communication	Event detection complexity
Real-World Deployments	Smart Homes & Cities	Energy management, security	Enhances privacy, reduces latency	Device heterogeneity, network reliability
	Industrial IoT (IIoT)	Predictive maintenance,	Data privacy, improved operational	Security threats,

		robotics	efficiency	resource constraints
	Healthcare & Wearables	Patient data, personalized health	Compliance with regulations, personalization	Data heterogeneity, stringent privacy requirements
	Autonomous Vehicles & Transportation	Cooperative driving, V2X learning	Improves safety and scalability	Real-time constraints, high security demands

Discussion

According to research conducted by Li et al. (2020), horizontal FL is most suitable when data across devices shares the same feature space, making it ideal for homogeneous IoT scenarios such as smart homes. Conversely, vertical FL (Hardy et al., 2017) facilitates collaboration between organizations holding different feature sets, common in healthcare or financial sectors. Federated transfer learning extends FL's applicability to heterogeneous environments but at increased computational costs (Zhao et al., 2020).

Optimization strategies like model compression and sparsification (Konečný et al., 2016; Aji & Heafield, 2017) effectively reduce communication overhead but can trade off model accuracy if not carefully designed. Partial training and lightweight models (Howard et al., 2017; Lane et al., 2019) cater to resource-constrained edge devices, ensuring feasible deployments while maintaining acceptable performance.

Real-world deployments demonstrate FL's versatility across domains. Smart city applications benefit from privacy-aware local training (Wang et al., 2019), while IIoT leverages FL for operational efficiency without central data pooling (Xu et al., 2020). Healthcare systems utilize FL to overcome regulatory barriers while harnessing rich medical data for predictive analytics (Sheller et al., 2020). Autonomous vehicles rely on FL for cooperative learning and safety-critical decision-making, balancing real-time constraints with data privacy (Chen et al., 2020). This comparative summary underscores that no one-size-fits-all solution exists; instead, the choice of FL model, optimization technique, and deployment strategy must align with application requirements, device capabilities, and privacy considerations.

4.0 Challenges

Despite the promising capabilities of Federated Learning (FL) for IoT and edge computing, several significant challenges remain to be addressed to enable scalable, secure, and efficient deployment in real-world scenarios.

4.1 Deployment at Scale

Deploying FL at scale across potentially millions of heterogeneous IoT and edge devices presents considerable technical and operational challenges. According to Li et al. (2020), large-scale FL systems must contend with diverse device capabilities, intermittent connectivity, and dynamic network conditions. The heterogeneity in hardware, network bandwidth, and energy availability complicates synchronization and

model aggregation. Moreover, coordinating millions of devices for synchronized training can result in high communication overhead and latency, especially when devices frequently join or leave the network (Kairouz et al., 2021).

Effective client selection and scheduling algorithms are critical to handle this scale. As noted by Bonawitz et al. (2019), scalable FL requires adaptive participation control mechanisms that can dynamically select a representative subset of devices to optimize communication efficiency without sacrificing model performance. However, the need to balance fairness, accuracy, and efficiency remains an open problem, especially under constrained resources.

4.2 Data Imbalance

Data imbalance, both in terms of quantity and quality, is another pervasive challenge in FL for IoT. Many IoT devices generate highly skewed and non-independent and identically distributed (non-IID) data due to variations in user behavior, environment, and sensor quality (Zhao et al., 2018). This heterogeneity in data distributions across clients can severely degrade the performance and convergence of federated models (Li et al., 2020).

Research by Wang et al. (2020) highlights that classical FL algorithms like FedAvg assume balanced and IID data, which rarely holds true in practical IoT deployments. Addressing this issue requires novel aggregation techniques and personalization strategies that can adapt to the skewed and unbalanced data, ensuring robustness and fairness across clients. Solutions such as clustered FL (Sattler et al., 2020) and meta-learning-based personalization (Fallah et al., 2020) are promising but still under active investigation.

4.3 Security at Inference Time

While FL primarily focuses on preserving privacy during training, securing the inference phase where the trained model is deployed and used is equally critical but less studied. Inference time vulnerabilities, such as model inversion, membership inference, and adversarial attacks, threaten the confidentiality and integrity of FL models in edge environments (Shokri et al., 2017).

A study by Hitaj et al. (2017) demonstrated that adversaries can reconstruct private training data from model queries, raising concerns about exposing sensitive information even after training. Moreover, edge devices used for inference are often resource-limited and physically exposed, making them susceptible to tampering and backdoor attacks (Bagdasaryan et al., 2020).

To mitigate these risks, advanced defense mechanisms such as runtime anomaly detection, secure enclaves, and robust model hardening are essential but remain challenging to implement efficiently on constrained IoT devices (Sun et al., 2020).

4.4 Benchmarking & Evaluation Standards

The lack of standardized benchmarking and evaluation metrics for FL in IoT and edge environments hinders the ability to fairly compare algorithms and implementations. According to Yang et al. (2019), existing FL evaluations often rely on idealized datasets and assumptions that do not reflect the complexity and heterogeneity of real-world IoT scenarios.

There is a growing need for comprehensive benchmark suites that incorporate diverse datasets, device heterogeneity, network conditions, and realistic threat models. Recent efforts like the LEAF benchmark (Caldas et al., 2018) have begun addressing this gap but do not yet fully capture all practical considerations such as energy consumption, latency, and scalability.

Additionally, evaluation metrics must extend beyond accuracy to include communication overhead, computational efficiency, privacy guarantees, and fairness across client devices (Kairouz et al., 2021). Establishing widely accepted standards will accelerate the development and adoption of FL solutions tailored for IoT and edge computing.

5.0 Future Directions

Federated Learning is a rapidly evolving field, and its intersection with emerging technologies opens up promising avenues for advancing IoT and edge computing. The following directions highlight key trends and potential research areas that can address existing challenges and enhance FL's capabilities.

5.1 FL and Blockchain Integration

Integrating blockchain technology with FL has garnered significant attention for enhancing security, trust, and decentralization in collaborative learning environments. According to Lu et al. (2020), blockchain can provide immutable audit trails and decentralized trust management, mitigating risks such as model poisoning and malicious client behavior. The synergy allows secure, transparent aggregation of model updates without relying on a centralized server, which is particularly valuable in IoT networks with distributed devices. However, challenges remain regarding blockchain scalability and energy consumption, especially for resource-constrained edge devices

5.2 FL with 5G and 6G Networks

The emergence of 5G and upcoming 6G networks presents transformative opportunities for FL by offering ultra-low latency, massive device connectivity, and high data throughput (Zhang et al., 2020). These advancements can alleviate communication bottlenecks and enable real-time federated training and inference for latency-sensitive applications like autonomous vehicles and smart cities. Moreover, 6G promises to integrate AI natively into communication infrastructures, creating a seamless edge-cloud continuum supporting FL at scale (Saad et al., 2020). Research must focus on optimizing FL protocols to fully leverage these networks' unique capabilities and address potential security vulnerabilities.

5.3 Green Federated Learning

Energy efficiency is a critical concern in IoT and edge environments where devices often operate on limited battery power. Green FL emphasizes developing energy-aware algorithms that minimize computational and communication costs without sacrificing model accuracy (Wang et al., 2021). Techniques such as adaptive client selection, dynamic resource allocation, and lightweight model architectures are pivotal to reducing the carbon footprint of FL deployments. Further work is needed to standardize energy consumption metrics and design FL frameworks that balance environmental sustainability with performance.

5.4 Federated Learning and Reinforcement Learning

The integration of Federated Learning with Reinforcement Learning (RL) offers exciting possibilities for distributed decision-making in complex, dynamic IoT systems. RL agents at edge devices can learn optimal policies collaboratively while preserving data privacy through FL frameworks (Zhu et al., 2021). Applications include intelligent traffic management, autonomous drones, and adaptive smart grids. However, combining FL and RL introduces new challenges such as non-stationary environments, exploration-exploitation trade-offs, and increased computational complexity, requiring innovative algorithmic designs and efficient communication protocols.

6. Conclusion

This review has examined the state-of-the-art in Federated Learning tailored for IoT and edge computing, covering FL models, resource-constrained optimization strategies, security and privacy considerations, and real-world implementations. FL emerges as a transformative paradigm that addresses critical challenges of data privacy, communication efficiency, and personalization in decentralized environments.

Despite its potential, deploying FL at scale remains challenging due to device heterogeneity, data imbalance, and security threats. Nonetheless, ongoing research integrating FL with blockchain, next-generation networks, energy-efficient approaches, and reinforcement learning promises to overcome these limitations and expand FL's applicability.

As IoT and edge computing continue to permeate diverse domains, Federated Learning will be indispensable for enabling intelligent, privacy-preserving, and scalable machine learning solutions. Continued interdisciplinary collaboration and innovation will be vital in realizing FL's full potential for the connected world.

Declaration of Conflicting Interests

The authors declare no potential conflicts of interest with respect to the research, authorship and publication of this article.

Funding

The author received no financial support for the research, authorship and publication of this article.

References

[1] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308-318.

[2] Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys (CSUR), 51(4),

[3] Alistarh, D., Grubic, D., Li, J., Tomioka, R., & Vojnović, M. (2017). QSGD: Communication-efficient SGD via gradient quantization and encoding. Advances in Neural Information Processing Systems, 30, 1709–1720. https://arxiv.org/abs/1610.02132

[4] Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). How to backdoor federated learning. Proceedings of the Twenty-Third International Conference on Artificial Intelligence and Statistics (AISTATS), 2938-2948. https://arxiv.org/abs/1807.00459

[5] Beutel, D., Huba, D., Josifovski, V., & McMahan, H. B. (2020). Flower: A friendly federated learning research framework. arXiv preprint. https://arxiv.org/abs/2007.14390

[6] Bonawitz, K., et al. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 1175-1191. https://doi.org/10.1145/3133956.3133982

[7] Briggs, J., Zhang, C., & Gupta, S. (2020). TensorFlow Federated: Machine learning on decentralized data. arXiv preprint. https://arxiv.org/abs/1910.10497

[8] Caldas, S., Wu, P., Li, T., Konečný, J., McMahan, H. B., Smith, V., & Talwalkar, A. (2018). LEAF: A Benchmark for Federated Settings. arXiv preprint arXiv:1812.01097.

[9] Chen, X., Liu, C., Li, W., & Li, Y. (2020). Federated learning for autonomous vehicles: Opportunities and challenges. IEEE Wireless Communications, 27(6), 72-78. https://doi.org/10.1109/MWC.001.1900409

[10] Fallah, A., Mokhtari, A., & Ozdaglar, A. (2020). Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. Advances in Neural Information Processing Systems, 33, 3557-3568. https://arxiv.org/abs/2002.07948

[11] Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective. arXiv preprint. https://arxiv.org/abs/1712.07557

[12] Gu, T., Dolan-Gavitt, B., & Garg, S. (2019). Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv preprint. https://arxiv.org/abs/1708.06733

[13] Hitaj, B., Ateniese, G., & Perez-Cruz, F. (2017). Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 603–618.

[14] Jiang, Z., Chen, T., & Wang, Z. (2018). Model compression and acceleration for deep neural networks: The principles, progress, and challenges. IEEE Signal Processing Magazine, 35(1), 126-136. https://doi.org/10.1109/MSP.2017.2743240

[15] Kairouz, P., et al. (2021). Advances and open problems in federated learning. Foundations and Trends® in Machine Learning, 14(1–2), 1–210. https://doi.org/10.1561/2200000073

[16] Khan, S., Xie, C., Imran, M., & Li, K. (2020). Incentive mechanisms for federated learning: A survey. IEEE Communications Surveys & Tutorials, 23(2), 1224-1244. https://doi.org/10.1109/COMST.2020.2980021

[17] Kim, J., Kang, S., & Kim, J. (2021). Blockchain-based federated learning for secure data sharing in industrial IoT. Sensors, 21(15), 5076. https://doi.org/10.3390/s21155076

[18] Konečnỳ, J., McMahan, H. B., Ramage, D., & Richtárik, P. (2016). Federated optimization: Distributed optimization beyond the datacenter. arXiv preprint. https://arxiv.org/abs/1511.03575

[19] Lalitha, A., Jiang, J., & Poor, H. V. (2019). Peer-to-peer federated learning on graphs. arXiv preprint. https://arxiv.org/abs/1901.11173

[20] Lane, N. D., Bhattacharya, S., Mathur, A., Forlivesi, C., Kawsar, F., & Sun, Y. (2019). DeepX: A software accelerator for low-power deep learning inference on mobile devices. Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services, 1-14. https://doi.org/10.1145/3307334.3326060

[21] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated Learning: Challenges, Methods, and Future Directions. IEEE Signal Processing Magazine, 37(3), 50–60.

[22] Liu, Y., Chen, M., Wang, Z., & Mao, S. (2020). Hierarchical federated learning for mobile edge computing: Algorithms and performance analysis. IEEE Transactions on Wireless Communications, 19(7), 4564-4577. https://doi.org/10.1109/TWC.2020.2975275

[23] Lu, Y., et al. (2020). Blockchain and federated learning for privacy-preserving smart healthcare. IEEE Network, 34(2), 1-6. https://doi.org/10.1109/MNET.011.1900366

[24] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of

deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 54, 1273-1282. https://arxiv.org/abs/1602.05629

[25] Medich, J. (2021). Edge computing is about to solve the IoT's biggest problems. Wired. https://www.wired.com/story/edge-computing-iot

[26] Nguyen, D. C., Ding, M., Pathirana, P. N., Seneviratne, A., Li, J., & Poor, H. V. (2021). Federated learning for Internet of Things: A comprehensive survey. arXiv. https://arxiv.org/abs/2104.07914

[27] Nishio, T., & Yonetani, R. (2019). Client selection for federated learning with heterogeneous resources in mobile edge. IEEE International Conference on Communications (ICC), 1-7. https://doi.org/10.1109/ICC.2019.8761466

[28] NVIDIA. (2021). NVIDIA FLARE: Federated learning application runtime environment. https://developer.nvidia.com/nvidia-flare

[29] Ren, S., et al. (2020). Efficient event-triggered communication in federated learning for IoT. IEEE Internet of Things Journal, 7(8), 7356-7365. https://doi.org/10.1109/JIOT.2020.2984894

[30] Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., ... & Kaissis, G. (2020). The future of digital health with federated learning. NPJ Digital Medicine, 3(1), 119. https://doi.org/10.1038/s41746-020-00323-1

[31] Ryffel, T., et al. (2018). A generic framework for privacy-preserving deep learning. arXiv preprint. https://arxiv.org/abs/1811.04017

[32] Saad, W., Bennis, M., & Chen, M. (2020). A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. IEEE Network, 34(3), 134-142.

[33] Sattler, F., Wiedemann, S., Müller, K.-R., & Samek, W. (2020). Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. IEEE Transactions on Neural Networks and Learning Systems, 32(8), 3710-3722. https://doi.org/10.1109/TNNLS.2020.2982976

[34] Sheller, M. J., Reina, G. A., Edwards, B., Martin, J., & Bakas, S. (2020). Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, 92-104. https://doi.org/10.1007/978-3-030-46640-4_10

[35] Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership Inference Attacks Against Machine Learning Models. 2017 IEEE Symposium on Security and Privacy (SP), 3–18.

[36] Stich, S. U., Cordonnier, J.-B., & Jaggi, M. (2018). Sparsified SGD with memory. Advances in Neural Information Processing Systems, 31, 4447–4458. https://arxiv.org/abs/1809.07599

[37] Sun, L., Yu, H., & He, X. (2020). Federated learning with data poisoning attacks. IEEE Transactions on Neural Networks and Learning Systems. Advance online publication. https://doi.org/10.1109/TNNLS.2020.3026907

[38] Wang, J., Liu, Q., Liang, H., Joshi, G., & Poor, H. V. (2019). Tackling the objective inconsistency problem in heterogeneous federated optimization. Advances in Neural Information Processing Systems, 33, 7611-7623. https://arxiv.org/abs/2007.07481

[39] Wang, S., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., & Chan, K. (2020). Adaptive federated learning in resource constrained edge computing systems. IEEE Journal on Selected Areas in Communications, 37(6), 1205-1221. https://doi.org/10.1109/JSAC.2019.2946097

[40] Wang, X., Han, Y., Wang, C., & Liu, Z. (2019). Smart city and smart home applications enabled by federated learning. IEEE Internet of Things Journal, 6(4), 6791-6800. https://doi.org/10.1109/JIOT.2019.2924442

[41] Wang, Z., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., & Chan, K. (2021). Energy-Efficient Federated Learning for Edge Computing Systems. IEEE Journal on Selected Areas in Communications, 39(1), 1-16.

[42] Warden, P., & Situnayake, D. (2019). TinyML: Machine learning with TensorFlow Lite on Arduino and ultra-low-power microcontrollers. O'Reilly Media, Inc.

[43] Xie, C., Koyejo, O., & Gupta, I. (2019). Asynchronous federated optimization. arXiv preprint. https://arxiv.org/abs/1903.03934

[44] Xu, J., Glicksberg, B. S., Su, C., Walker, P., Bian, J., & Wang, F. (2021). Federated learning for healthcare informatics. Journal of Healthcare Informatics Research, 5(1), 1-19. https://doi.org/10.1007/s41666-020-00090-1

[45] Xu, X., Li, H., & Chen, S. (2020). Federated learning for industrial IoT: Applications, challenges, and opportunities. IEEE Communications Magazine, 58(12), 24-29. https://doi.org/10.1109/MCOM.001.2000226

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

[47] Yang, W., Lim, B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y.-C., Yang, Q., Niyato, D., & Miao, C. (2019). Federated

learning in mobile edge networks: A comprehensive survey. arXiv. https://arxiv.org/abs/1909.11875

[48] Zhang, C., Zhang, K., Yang, Z., & Wang, S. (2021). A Survey on Federated Learning. Knowledge-Based Systems, 216, 106775.

[49] Zhang, Z., Zheng, K., & Yang, K. (2020). 5G Millimeter Wave Communications for IoT and Edge AI: Opportunities and Challenges. IEEE Wireless Communications, 27(3), 41-47.

[50] Zhao, J., Liu, S., Han, Y., Zhao, R., & Yang, Q. (2020). Federated learning with privacy: Challenges and solutions. IEEE Network, 34(6), 36-43. https://doi.org/10.1109/MNET.001.1900545

[51] Zhao, Y., Chen, H., & Xu, J. (2020). Trustworthy federated learning with reputation management in edge computing. IEEE Access, 8, 188245-188256. https://doi.org/10.1109/ACCESS.2020.3028641

[52] Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). Federated Learning with Non-IID Data. arXiv preprint arXiv:1806.00582.

[53] Zhu, X., Wang, J., Wang, Y., & Zhang, J. (2021). Federated Reinforcement Learning for Cooperative Edge Caching. IEEE Transactions on Wireless Communications, 20(7), 4317-4330.