future internet

MDPI

Review

Intelligent Edge Computing and Machine Learning: A Survey of Optimization and Applications

Sebastián A. Cajas Ordóñez, Jaydeep Samanta, Andrés L. Suárez-Cetrulo * and Ricardo Simón Carbajo

Ireland's Centre for Artificial Intelligence (CeADAR), University College Dublin, Belfield, D04 V2N9 Dublin, Ireland; sebastian.cajasordonez@ucd.ie (S.A.C.O.); jaydeep.samanta@ucd.ie (J.S.); ricardo.simoncarbajo@ucd.ie (R.S.C.)
* Correspondence: andres.suarez-cetrulo@ucd.ie

Abstract

Intelligent edge machine learning has emerged as a paradigm for deploying smart applications across resource-constrained devices in next-generation network infrastructures. This survey addresses the critical challenges of implementing machine learning models on edge devices within distributed network environments, including computational limitations, memory constraints, and energy-efficiency requirements for real-time intelligent inference. We provide comprehensive analysis of soft computing optimization strategies essential for intelligent edge deployment, systematically examining model compression techniques including pruning, quantization methods, knowledge distillation, and low-rank decomposition approaches. The survey explores intelligent MLOps frameworks tailored for network edge environments, addressing continuous model adaptation, monitoring under data drift, and federated learning for distributed intelligence while preserving privacy in next-generation networks. Our work covers practical applications across intelligent smart agriculture, energy management, healthcare, and industrial monitoring within network infrastructures, highlighting domain-specific challenges and emerging solutions. We analyze specialized hardware architectures, cloud offloading strategies, and distributed learning approaches that enable intelligent edge computing in heterogeneous network environments. The survey identifies critical research gaps in multimodal model deployment, streaming learning under concept drift, and integration of soft computing techniques with intelligent edge orchestration frameworks for network applications. These gaps directly manifest as open challenges in balancing computational efficiency with model robustness due to limited multimodal optimization techniques, developing sustainable intelligent edge AI systems arising from inadequate streaming learning adaptation, and creating adaptive network applications for dynamic environments resulting from insufficient soft computing integration. This comprehensive roadmap synthesizes current intelligent edge machine learning solutions with emerging soft computing approaches, providing researchers and practitioners with insights for developing next-generation intelligent edge computing systems that leverage machine learning capabilities in distributed network infrastructures.

Keywords: edge machine learning; edge AI; IoT; model optimization; MLOps; quantization; knowledge distillation; low-rank adaptation; federated learning; data drift; multimodal fusion; resource-constrained devices; system resilience; ethical AI; large language models

Academic Editors: Giovanni Pau and Fabio Arena

Received: 19 August 2025
Revised: 4 September 2025
Accepted: 9 September 2025
Published: 11 September 2025

Citation: Cajas Ordóñez, S.A.; Samanta, J.; Suárez-Cetrulo, A.L.; Carbajo, R.S. Intelligent Edge Computing and Machine Learning: A Survey of Optimization and Applications. *Future Internet* **2025**, *17*, 417. https://doi.org/10.3390/fi17090417

1. Introduction

Intelligent edge computing has emerged as a crucial research frontier that bridges artificial intelligence with resource-constrained network environments, driven by the exponential growth of IoT devices and the increasing demand for real-time data processing [1,2]. As edge computing becomes pervasive across next-generation networks, edge machine learning (Edge ML) aims to deliver low-latency, privacy-preserving, and energy-efficient inference through intelligent orchestration of heterogeneous devices spanning the IoT–Edge–Cloud continuum [3,4]. This paradigm shift from centralized cloud computing to distributed edge intelligence addresses critical limitations in bandwidth, latency, and privacy while enabling sophisticated AI applications in resource-constrained environments [5,6]. This survey focuses on emerging optimization strategies, soft computing techniques, and system-level integrations designed to enable intelligent network applications.

The deployment of sophisticated ML models on intelligent edge networks faces challenges due to the growing complexity of modern architectures and the inherent limitations of distributed network environments. While deep learning has advanced rapidly, particularly through transformer architectures [7], the associated computational demands remain prohibitive for next-generation network applications, especially for Large Language Models (LLMs) and Large Multimodal Models (LMMs). Despite the wide adoption of deep neural networks [7–9], very few leverage soft computing principles for adaptive edge environments. This challenge extends to multimodal data processing [10] and the deployment of Visual Language Models (VLMs) on mobile network hardware [11–13].

The proliferation of intelligent edge nodes deployed across diverse network infrastructures further compounds these challenges [6,14–16]. In parallel, federated learning has emerged as a promising paradigm for distributing intelligent model training while preserving data locality in network environments [17,18], though challenges persist in managing energy, model updates, and system robustness across network topologies [1,3,5,19,20]. Moreover, connectivity limitations in edge networks continue to pose challenges for intelligent update synchronization and streaming stability [2,21].

To address these network resource constraints, researchers have developed various model compression and soft computing optimization techniques. Model compression methods such as pruning, quantization, low-rank approximation, and distillation have gained significant traction for intelligent network applications [22,23]. Lightweight fine-tuning methods like LoRA [24], QA-LoRA [25], QLoRA [26], and DoRA [27] enable large models to run efficiently on embedded network systems by reducing latency and memory usage [13]. These approaches are often combined with hybrid and extreme quantization methods (e.g., BitNet, BitNet b1.58) [13,28–30] and knowledge distillation techniques for intelligent edge computing [31].

In practical next-generation network deployments, intelligent task offloading becomes essential, where computational tasks are distributed from central systems to edge devices [2,32] to support intelligent IoT scenarios [33,34], while maintaining data quality [15] and ethical safeguards [35]. Given the increasing adoption of intelligent Edge ML in critical network sectors like transportation, energy, and urban systems, it is vital to investigate how model adaptation, orchestration, and system resilience interact in practice through soft computing approaches.

Despite these advances, significant gaps remain in current intelligent edge computing research. While previous surveys have explored various aspects of fog and edge computing [4,36–39], they often lack coverage of soft computing techniques and intelligent network application trends. For example, Singh et al. [40] highlight edge task scheduling and caching but omit advances in lightweight ML techniques like quantization and soft computing approaches. Similarly, while Verbraeken et al. [41] survey distributed ML, they

overlook intelligent edge-specific systems with orchestration and modular deployment. Yu et al. [42] provide foundational architectural perspectives, but do not address ethics, multimodal alignment, and intelligent deployment pipelines.

Recent surveys have investigated specific aspects such as green design principles [43], containerized orchestration systems [44–46], distributed streaming pipelines [47], multimodal fusion [48–51], transfer learning [52], hardware-aware ML [23,53,54], and privacy-preserving ML via federated learning [55–57]. However, these optimization techniques remain underexplored in critical areas such as stream learning [58–61], anomaly detection [62], and distributed MLOps pipelines like dataClay [63]. Furthermore, quantization techniques lack full integration with orchestration frameworks [63,64], and most studies fail to incorporate lightweight tuning methods (e.g., BitNet) [28–30], stream learning under concept drift, and edge-ready anomaly detection or MLOps systems.

This survey addresses these gaps by providing a comprehensive review of modern intelligent Edge ML challenges, emphasizing real-world deployment in next-generation networks, modular system composition, and resilience under continuous data drift. We focus on dynamic, high-throughput settings such as intelligent smart grids [65–67], industrial monitoring [68,69], and autonomous mobility networks [70–72]. We also explore the intricacies of deploying intelligent ML systems in production environments with constrained network resources, advocating for effective monitoring and continual adaptation through soft computing principles [73,74].

To structure our survey, we begin with foundational optimization strategies and intelligent Edge ML techniques. We then review application areas and emerging toolchains for next-generation network applications. Finally, we highlight critical open questions to guide future research focused on balancing efficiency, trustworthiness, and sustainability in intelligent edge computing environments.

2. Machine Learning Background

The critical aspects for real-world implementation of intelligent edge machine learning (ML) models in next-generation networks include addressing model efficiency, robustness, footprint, AI degradation, and trust challenges within distributed computing environments. To tackle footprint concerns, deep compression techniques through pruning unnecessary connections and intelligent weight distribution have gained significant attention [22,23]. Collectively, these studies highlight how crucial it is to address these issues when deploying intelligent Edge ML models in network-aware practical applications. In this section, we cover essential concepts for intelligent model optimization and MLOps at the edge, model degradation, and data drift, as well as security and privacy for intelligent edge machine learning systems.

2.1. Intelligent Model Optimization at the Edge

This section introduces the main soft computing techniques to reduce GPU consumption, RAM utilization, computational needs by decreasing FLOPs count, and inference delay for intelligent network applications. This involves introducing sparsely connected deep neural networks for efficient deployment on resource-limited edge devices within next-generation network infrastructures, enhancing accessibility to training these networks with high-quality IoT datasets [75,76]. This approach is intended to reduce parameters and optimize networks for intelligent edge limitations in distributed environments.

Multiple soft computing techniques can be used for faster intelligent inference at the edge; these include parallelization across distributed network devices, memory offloading to speed up inference time by intelligently managing temporary data [40], and adaptive model optimization techniques such as (i) pruning, (ii) distillation, and (iii) quantiza-

tion [77,78], as well as (iv) low-rank decomposition methods for intelligent language model optimization, which substantially reduce trainable parameters. These methods adjust neural network architectures to edge network restrictions such as latency and memory, allowing complex models to operate seamlessly on computationally limited network devices while achieving energy efficiency [22].

Table 1 provides a structured comparison of pruning, quantization, distillation, and low-rank factorization, emphasizing their impact on memory usage, accuracy impact, latency improvement, and best-suited use cases. This allows for informed decisions when selecting optimization techniques for resource-constrained devices.

Table 1. Comparison of neural network optimization techniques: ranges reflect variability across model architectures, datasets, and hardware platforms. Performance improvements are hardware-dependent and may not be additive when techniques are combined. Based on studies from [26,64,79–86].

Technique	Memory Reduction	Accuracy Impact	Latency Improvement	Typical Use Case
Structured Pruning	2×–10× smaller	0.1–5% loss	1.2×–3× faster	Hardware-friendly edge deployment
Unstructured Pruning	5×–50× smaller	1–8% loss	Limited improvement	Memory-constrained scenarios
INT8 Quantization	4× smaller	0.5–3% loss	1.5×–3× faster (edge devices)	Mobile inference optimization
INT4/Binary Quantization	8×–16× smaller	2–15% loss	2×–4× faster (specialized HW)	Ultra-low resource deployment
Knowledge Distillation	2×–5× smaller	0.5–3% loss	Proportional to compression	Model compression with accuracy retention
Low-Rank Factorization	1.5×–4× smaller	0.1–2% loss	1.2×–2.5× faster	Fine-tuning large models

2.1.1. Intelligent Pruning Techniques

Pruning improves model performance by removing unnecessary connections via sparse connectivity for intelligent edge applications [79,80]. It maintains performance on edge nodes to allow high efficiency in network environments, reducing overparameterization while retaining accuracy. Excess components are removed using techniques such as weight, unit, and structural pruning, which are optimized for next-generation network applications. These methods optimize neural networks for resource-efficient, high-performance deployment in distributed intelligent systems. Pruning approaches can lower the parameters of trained networks by up to 90% [87].

Advanced techniques include local pruning for layer-wise optimization, global pruning for network-wide efficiency, and custom pruning with intelligent masking for specific network requirements [77]. Complementing these optimization approaches, frameworks like TensorFlow Lite have been specifically designed for scalable intelligent neural network deployment on resource-constrained network devices [88]. These frameworks provide optimized ML models for next-generation network applications, offering local data processing capabilities that eliminate cloud dependency, reduce latency through intelligent real-time operation, have offline functionality, and enhance data security for distributed edge environments.

2.1.2. Intelligent Quantization for Network Applications

Quantization is a fundamental soft computing optimization technique for converting tensors to lower precision, such as integers rather than floating-point values, enabling faster and more efficient computation in next-generation network environments [64,81]. An 8-bit integer-quantized version of a 32-bit floating-point model is approximately 4× smaller in size and 1.5–4.0× faster in computation. As a result, quantized models have a smaller memory footprint, making them ideal for deployment on resource-constrained edge devices in intelligent network infrastructures.

Deep learning has demonstrated significant performance improvements but at high computational cost, necessitating quantization tools for efficient and accurate intelligent inference schemes that address accuracy and latency in network applications [81]. TensorFlow Lite and PyTorch [85,89] enable efficient quantization with different methods, supporting customization for intelligent edge deployment in distributed networks.

There are two primary quantization approaches for creating low-bit models in intelligent network applications:

- Post-training quantization: This approach reduces the precision of weights and activations after model training, supporting various quantization levels including 8-bit variants [90], 4-bit variants [91,92], 2-bit quantization [93], and 1-bit quantization (BitNet variants) [13,28] that replace matrix multiplication with integer addition for intelligent edge applications. While simple to implement for network deployment, this method might result in accuracy loss. It includes
  1. Quantizing only weights;
  2. Quantizing both weights and activations [94].
- Quantization-aware training: This employs quantization during model training, achieving better accuracy for intelligent network applications. This technique incorporates simulated quantization operations using automated tools from the TensorFlow and PyTorch libraries [89].

2.1.3. Knowledge Distillation for Intelligent Edge Networks

Knowledge distillation enables smaller models to emulate larger ones through soft computing approaches, optimizing model size and accuracy for application-specific and resource-constrained network environments.

These approaches follow an intelligent teacher–student learning strategy, utilizing larger teacher networks to train compact student networks with minimal accuracy loss for edge deployment. Combined with transfer learning, network distillation significantly reduces model size without compromising performance in intelligent network applications [85]. Transferring knowledge from ensemble models or large regularized models into smaller, distilled models achieves comparable performance for next-generation network applications.

Knowledge distillation reduces training times considerably, facilitating easier deployment in intelligent network environments [84]. As shown in Figure 1, the teacher model transfers knowledge to the student model through intelligent distillation processes.

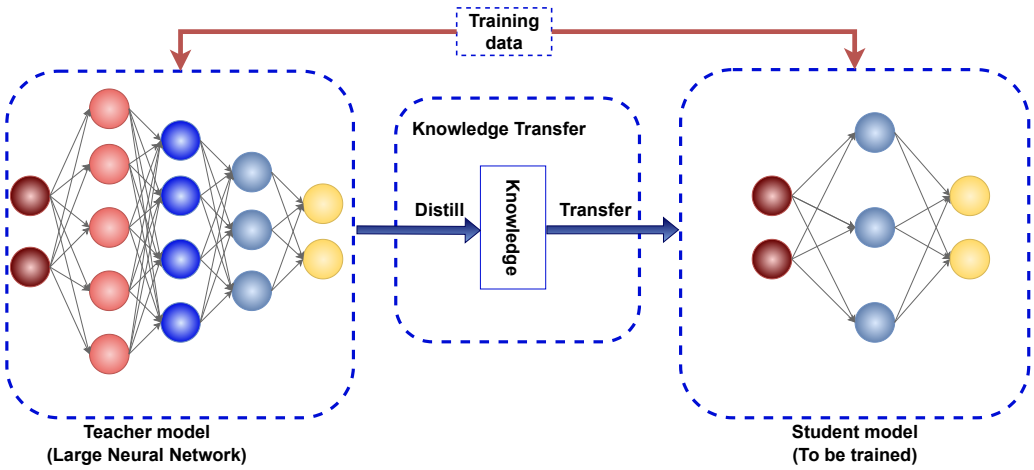

Figure 1. The intelligent teacher–student framework for knowledge distillation in edge networks [31].

Distillation techniques for intelligent edge applications, illustrated in Figure 2, include three categories:

- Response-based knowledge: The student model learns from teacher predictions, with distillation loss reducing logit differences for intelligent network optimization [95].
- Feature-based knowledge: Intermediate layers reduce feature discrepancies between models, enabling students to emulate teacher neuron activations in distributed environments [96].
- Relational knowledge: This evaluates feature maps and similarity matrices, understanding feature correlations across multiple representations for intelligent edge applications [97].

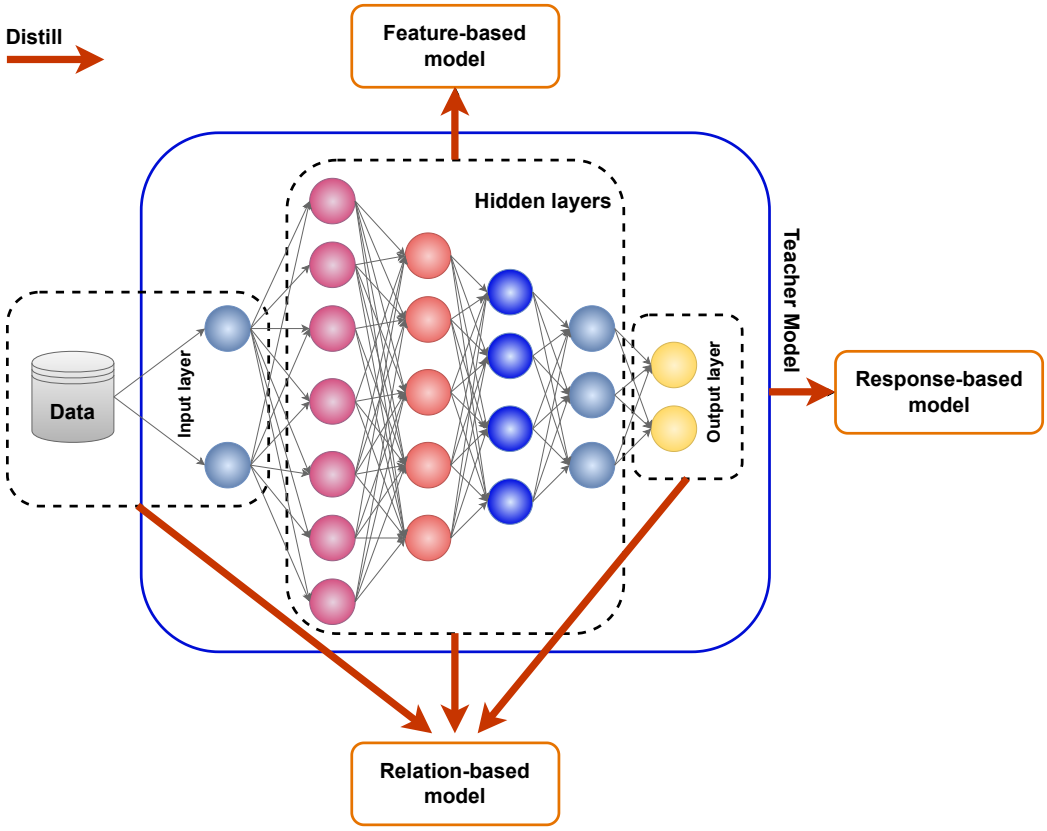

Figure 2. The three primary knowledge forms in intelligent distillation models for network applications [31].

Advanced distillation algorithms for intelligent network applications include (i) adversarial distillation, which uses generative principles for improved data representation [98]; (ii) multi-teacher distillation, which collects knowledge from multiple models; (iii) cross-modal distillation, which is used for multimodal network applications; and (iv) DistilGPT2, which demonstrates effective language model compression for intelligent edge deployment [99,100].

Additional techniques including attention-based, data-free, quantized, and lifelong distillation have emerged as valuable strategies for intelligent edge computing [101–103]. These methods enable more efficient and adaptable deep learning models for intelligent network inference while maintaining an optimal accuracy–latency balance.

2.1.4. Low-Rank Decomposition Methods for Intelligent Networks

Neural network models in next-generation network applications necessitate vast amounts of data and typically consist of millions to billions of parameters, creating significant computa-

tional challenges for intelligent edge deployment [7]. This reliance on expensive computational resources highlights the importance of devising specialized soft computing optimization techniques to effectively minimize costs in distributed network environments.

Research has demonstrated that over-parameterized models reside on low intrinsic dimensions [104,105], leading to the development of Low-Rank Adaptation (LoRA) [24]. LoRA focuses on training only parameter subsets from dense layers, optimizing resources and enhancing energy efficiency for intelligent network applications within transformer architectures. This method freezes pre-trained model weights while incorporating trainable rank decomposition matrices into each layer, significantly reducing trainable parameters for efficient intelligent edge deployment.

LoRA has enabled rapid development of advanced adaptations for next-generation network applications:

- *QLoRA* [26] optimizes weight parameters by reducing the 32-bit format to 4-bit quantization space, significantly reducing memory usage for intelligent edge networks while maintaining training effectiveness through dynamic precision switching.
- *QA-LoRA* [25] combines quantization and fine-tuning of LoRA parameters, balancing adapter and quantization parameters through group-wise operators for distributed network optimization.
- *DoRA* [27] enhances LoRA by decomposing pre-trained weights into magnitude and direction components, focusing on directional adaptation to improve scalability and learning capacity while reducing training overhead for intelligent network applications.

Low-rank matrix factorization plays a pivotal role in enhancing adaptability and efficiency for intelligent edge computing in network environments. Parameter-Efficient Fine-Tuning frameworks provide accessible deployment avenues for LoRA variations, enabling model optimization for next-generation network applications [106].

While optimization techniques reduce resource consumption, managing these models at scale introduces new challenges, as outlined in the following section on MLOps, which discusses the context where MLOps operates, particularly in real-world scenarios, highlighting its main significance in edge machine learning deployment:

2.2. Intelligent MLOps at the Edge

Recent developments in machine learning have focused on building intelligent models for diverse domains including health, finance, defense, entertainment, and commerce within next-generation network environments. These models, developed by domain experts and data scientists, constantly evolve as new data becomes available to enhance their capabilities for resource-constrained network devices. AI frameworks provide powerful toolkits for building complex predictive systems in distributed environments. However, real-world deployment involves maintenance costs that, if overlooked, may lead to technical debt in intelligent ML systems [107]. Unlike standard software design, ML solutions require data relationships that challenge long-term maintenance in network environments [108]. This necessitates complex pipelines for framework enhancement, making it challenging to retrain and deploy intelligent machine learning models into production [73]. This complex pipeline constitutes Machine Learning Operations or MLOps for intelligent edge computing [74].

Running intelligent MLOps at the edge becomes crucial with increasing IoT demand and the benefits provided by edge devices supporting ML frameworks in next-generation network infrastructures compared to traditional cloud-based approaches [109]. With the proliferation of IoT devices and edge data availability, leveraging intelligent edge computing for inference has become standard for efficient information processing and insight gathering from distributed network devices [3].

Recent frameworks have addressed intelligent MLOps deployment. Raj et al. [110] proposed synchronizing the development, deployment, and monitoring of ML models for real-time automation across cloud and edge operations in network environments. John et al. [111] developed frameworks helping organizations integrate intelligent MLOps into existing software development practices for next-generation network applications.

Since intelligent ML operations span all AI application stages, AI software sustainability presents significant risks. Recent research addresses the challenges and trends for sustainable intelligent ML operations in distributed network environments [112].

2.2.1. Intelligent MLOps Pillars and Goals

Successful intelligent MLOps implementation depends on four critical pillars for next-generation network applications, which are illustrated in Figure 3. This framework emphasizes intelligent MLOps technologies in accelerating the complete machine learning lifecycle for edge networks.

- Intelligent Model Deployment and Experimentation: Simplifies model creation and deployment by optimizing data procedures and verifying that intelligent models function as intended in real-world network environments.
- Intelligent Model Monitoring: Monitors model performance across various network situations, recognizing data drift and limiting risks associated with incorrect predictions in distributed environments.
- Intelligent Production Deployment: Automates critical operations including model upgrades, troubleshooting, approval, updates, and scalability for seamless integration into operational network settings.
- Preparation for Intelligent Production Release: Includes version control, automated documentation, update tracking, and risk assessment, ensuring seamless model releases in network environments.

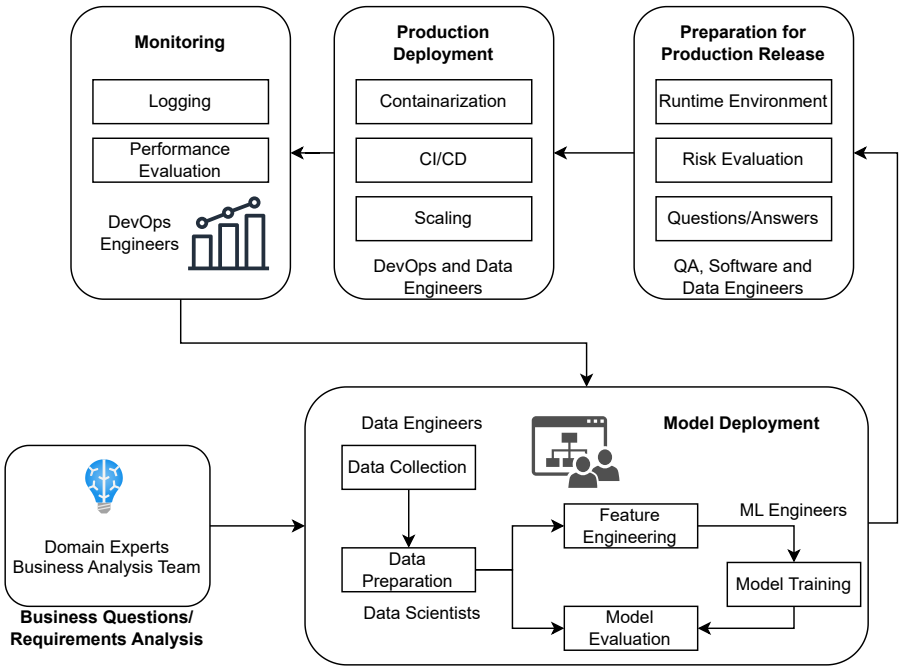

Figure 3. Intelligent MLOps pillars and associated roles for next-generation network applications [113].

2.2.2. Intelligent MLOps Tools for Network Applications

Building, deploying, and managing intelligent machine learning models in production requires efficient MLOps tools that deliver high-quality models and enhance collaboration between development and operations teams in next-generation network environments.

Numerous open-source platforms support intelligent ML project management and deployment. MLFlow [114] provides tracking, project management, and model registry functions for distributed networks. Kubeflow [115] enables scalable intelligent ML model deployments using Kubernetes for network infrastructures. MLReef [113] utilizes git-based repositories for collaborative intelligent ML development. ZenML [116] simplifies the setup and maintenance of intelligent ML pipelines for edge networks. MLRun [117] provides standardized interfaces for different ML libraries and data storage systems in distributed environments. Seldon Core [118] offers standardized interfaces for updating and deploying intelligent ML models in practical network settings. Tables 2 and 3 compare these frameworks for next-generation network applications looking into features such as data versioning, hyperparameter tuning, experiment and pipeline versioning, continuous integration and delivery, and model deployment and performance monitoring (Table 2), as well as their ease of use, scalability, edge compatibility, and use cases (Table 3).

Table 2. Comparison of key features across leading MLOps platforms, highlighting capabilities such as data versioning, experiment tracking, pipeline orchestration, and model deployment. Adapted from [113].

Platform	DV	HT	MEV	PV	CI/CD	MD	PM
AWS SageMaker	✓	✓	✓	✓	✓	✓	✓
MLFlow	✓	✓	✓	✓	✓	✓	
Kubeflow		✓	✓		✓	✓	✓
DataRobot		✓	✓			✓	✓
Iterative Enterprise	✓		✓		✓	✓	✓
ClearML	✓		✓	✓	✓	✓	✓
MLReef	✓	✓	✓	✓	✓	✓	✓
Streamlit	✓		✓			✓	✓

Abbreviations—DV: Data Versioning; HT: Hyperparameter Tuning; MEV: Experiment Versioning; PV: Pipeline Versioning; CI/CD: Continuous Integration/Delivery; MD: Model Deployment; PM: Performance Monitoring.

Table 3. Comparison of existing MLOps platforms adapted from [74,119].

Tool	Ease of Use	Scale	Edge Compat	Best Use Case
MLflow	Moderate	Good	Variable	Strong tracking/registry. Edge: model format.
W&B	High	Excellent	Variable	Excellent viz/tracking. Edge: model format.
Comet ML	High	Excellent	Variable	Robust tracking. Edge: model format.
Kubeflow	Complex	Excellent	Moderate	K8s-native, powerful but complex.
BentoML	Moderate	Good	Good	Optimized serving; edge-suitable.
SageMaker	Mod-High	Excellent	Good	Comprehensive suite, edge manager.
Databricks	Mod-High	Excellent	Variable	Big data scaling. Edge: model format.
Streamlit	High	Moderate	Variable	Quick dashboards, interactive viz.
MLReef	Moderate	Good	Good/Var.	Full-stack: deploy and monitor models.
DVC	Moderate	Excellent	Limited	Git-like versioning, reproducible ML.
DataRobot	High	Excellent	Good/Var.	End-to-end AutoML, explainability.

2.3. Intelligent AI Degradation and Data Drifts in Network Environments

Various factors contributing to data distribution changes or environmental variations in next-generation network infrastructures can cause intelligent model deterioration, significantly impacting predictive accuracy in distributed edge computing environments [59,120].

In intelligent network applications, concept drift represents "a change in data distribution and evolution of relationships between attributes and target features over time, or transitions between generative processes in network data streams. These transitions occur with different speeds, severity, and distribution patterns" [59].

Shifts and drifts may occur when data streams from edge devices in network infrastructures evolve, particularly when computational jobs change or environmental conditions surrounding sensors shift in intelligent IoT deployments [121]. Anomalies and data shifts can be monitored using statistical functions over sliding windows of data statistics and model performance metrics in distributed network environments [58,122].

Furthermore, intelligent models can be trained to adapt to data drifts through soft computing approaches utilizing forgetting mechanisms [123], offering better balance between performance and inference time compared to traditional algorithms in next-generation network applications [124]. Model explainability can be integrated with these mechanisms to increase transparency and comprehend intelligent model behavior in network environments, identifying improvement areas for enhanced performance in distributed edge computing systems [118].

2.4. Intelligent Federated Learning at the Edge

For real-world implementation in next-generation network applications, addressing intelligent Edge ML concerns of security, privacy, robustness, and trust is crucial. Federated learning (FL) is essential for enhancing privacy and security in distributed network environments. FL protects data privacy while producing accurate results by enabling edge devices to jointly train intelligent models without sharing raw data across network infrastructures. This approach creates the trust necessary for intelligent Edge ML to reach its potential in diverse network applications. Additionally, soft computing methods like bias reduction and anomaly identification contribute to the fairness and robustness of intelligent edge machine learning models in distributed environments. Open-source federated learning frameworks enable joint training of intelligent edge models, fostering transparency and trust while preserving privacy in network deployments.

Federated learning is an intelligent machine learning technique that enables model training on decentralized data sources across distributed network devices while optimizing privacy in next-generation network environments [1,17]. This strategy maximizes data utilization for training while maintaining data privacy in network infrastructures. Advantages of this method include data privacy preservation, scalability via distributed datasets, efficiency through minimized data transfer, and increased model resilience from varied data sources, improving robustness and generalizability for collaborative learning in intelligent network applications.

Figure 4 illustrates intelligent federated learning where data owners train local data within clusters A, B, and C, ultimately sharing updates through aggregation methods to update the global model in distributed network architectures.

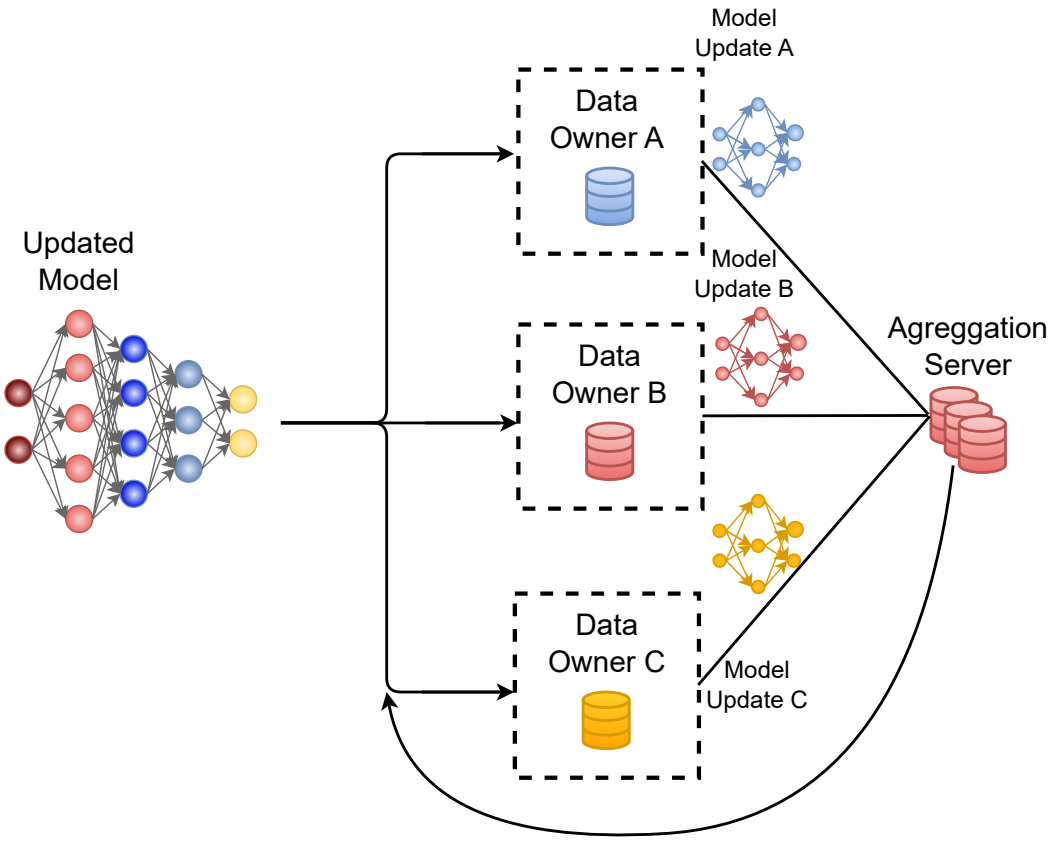

Figure 4. Intelligent client–server federated learning architecture for next-generation networks [125].

Several foundational approaches have been developed for intelligent federated learning to address various challenges in network environments. FedAvg [17] enables clients to update local models and aggregate at servers. FedProx [126] uses regularization terms to reduce weight discrepancies between local and global models in heterogeneous networks. FedPer [127] trains personalized models with shared base layers and individualized local layers for network clients.

Advanced approaches include FedOpt [128], which uses adaptive optimization methods for intelligent device selection, SplitFL [129], which employs dual models for local updates, and FedTL [130], which leverages pre-trained global models for device-specific refinement. FedEL [131] utilizes evolutionary algorithms for intelligent federated training. These strategies advance FL by addressing specific challenges in distributed network environments.

Alternative privacy protection approaches include hashing encryption techniques based on data fusion [132], exploring spatial-temporal privacy exposure risks in distributed intelligent edge environments.

Three categories of open-source intelligent federated learning frameworks support next-generation network applications [133]:

- All-in-one frameworks: Comprehensive solutions like FATE [134] and FedML [135] provide secure collaborative FL on decentralized network data.
- Horizontal-only frameworks: User-friendly APIs like Flower and FLUTE [136] emphasize simplicity for intelligent network applications.
- Specialized frameworks: Goal-specific solutions can also be used, like CrypTen [137] for secure multi-party computing and FedTree [138] for federated decision tree training in network environments.

GPU-accelerated tools, including Nvidia FLARE SDK [139] and PyTorch-based solutions like PySyft [140] enable scalable intelligent FL deployment. These frameworks offer

diverse functionalities for various network applications, advancing intelligent federated learning adoption in next-generation network infrastructures.

2.5. Performance Evaluation Metrics for Intelligent Edge AI

Evaluating the performance of intelligent edge AI systems often requires a number of factors listed below, consisting of a multi-dimensional assessment across computational efficiency, resource utilization, model quality, and system-level characteristics, among others, to ensure reliable deployment in next-generation network environments [141,142].

- **Computational metrics** form the foundation for evaluating edge AI performance, with latency measured as the time from input to output completion, mathematically expressed as $L = t_{end} - t_{start}$, where processing delays are critical for real-time applications. Throughput quantifies system capacity as the number of inference operations completed per unit time: $T = \frac{N_{operations}}{t_{elapsed}}$. Inference time specifically measures the duration required for model prediction on input data [143,144].
- **Resource utilization metrics** assess system efficiency through energy consumption measurement, typically expressed as energy per inference operation $E_{inference} = \frac{P_{avg} \times t_{inference}}{N_{inferences}}$, where P_{avg} represents average power consumption [145,146]. Memory utilization is quantified as $M_{util} = \frac{M_{used}}{M_{total}} \times 100\%$, while CPU and GPU utilization percentages indicate processing resource efficiency [147,148].
- **Model quality metrics** ensure intelligent edge systems maintain acceptable accuracy levels. These metrics are highly dependent on the task performed by the AI models. An example for classification tasks would be classification-accuracy-related metrics. For instance, $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$, $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, and $f_1score = 2 \times \frac{Precision \times Recall}{Precision+Recall}$, where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively [3,6].
- **System-level metrics** evaluate operational characteristics including availability measured as $Availability = \frac{MTBF}{MTBF+MTTR} \times 100\%$, where MTBF is mean time between failures and MTTR is mean time to repair. Scalability metrics assess system performance under varying loads, while reliability quantifies system stability over extended operation periods [37].
- **Standardized evaluation frameworks** provide consistent benchmarking approaches, with, for instance, MLPerf serving as the industry standard for measuring AI inference performance across diverse hardware platforms, supporting edge-specific benchmarks including MLPerf Inference Edge and MLPerf Mobile for comprehensive system evaluation [141,149]. These frameworks enable fair comparison across different edge AI implementations while supporting reproducible research and development efforts in intelligent edge computing environments.

3. Intelligent Edge ML Use Cases and Application Domains for Next-Generation Networks

In this section, we aggregate real-world use cases for intelligent edge machine learning in next-generation network applications that address the constraints of distributed edge deployment [109,150]. Table 4 provides a comprehensive comparison of edge AI characteristics, requirements, and challenges across major application domains, including agriculture, energy, healthcare, manufacturing, transportation, retail, smart cities, and finance.

We begin by introducing intelligent cloud offloading and highlighting real-time vision-assisted applications for network environments. We emphasize the need to reduce latency in next-generation network applications, particularly in scenarios like intelligent video offloading, where data streaming is essential for optimizing network consumption. For in-

stance, smart home scenarios in network infrastructures aim to reduce power usage and CO_2 emissions through intelligent edge computing [66,151]. In these and other use cases, collaborative intelligent edge computing [1,152] enables data communication across distributed network nodes and geographically dispersed intelligent systems.

This section examines applications including (i) intelligent smart cities and smart agriculture using edge computing for enhanced decision-making while minimizing environmental impact in network environments [67,153]; (ii) remote patient vital sign monitoring in intelligent healthcare networks [154,155]; (iii) intelligent internet of vehicles enabling autonomous driving and optimized routing in next-generation network infrastructures [156,157]; and (iv) intelligent smart industry approaches addressing failure detection and predictive maintenance in distributed network environments [158]. The domain-specific requirements and challenges outlined in Table 4 highlight the complexity of deploying edge AI solutions across these diverse applications. Finally, we discuss the optimization of intelligent operating systems and smart environments for next-generation network applications.

Table 4. Characteristics, requirements, and challenges of edge AI across different application domains. Sources [66,151,153–158].

Domain	Key Characteristics	Requirements	Main Challenges
Agriculture	Precision farming, crop tracking, weather prediction	Low power/wide coverage, weather resistance, real-time data	Rural connectivity, harsh conditions, cost
Energy	Smart grid, predictive maintenance, load balancing	High reliability, real-time decisions, system integration	Safety, regulatory compliance, scalability
Healthcare	Patient monitoring, diagnostics, wearables, emergency response	Ultra-low latency, high accuracy, privacy	Data privacy, life-critical accuracy, device size
Manufacturing	Quality control, predictive maintenance, robotics, supply chain	Real-time processing, high precision, system integration	Harsh environments, legacy systems, minimal downtime
Transportation	Autonomous vehicles, traffic management, fleet optimization	Ultra-low latency, high reliability/safety, real-time coordination	Safety, regulatory approval, infrastructure integration
Retail	Inventory, analytics, recommendations, checkout	Customer privacy, real-time analytics, scalability, cost	Privacy concerns, behavior patterns, POS integration
Smart Cities	Traffic/environmental monitoring, public safety	Wide area deployment, interoperability, scalability	Infrastructure complexity, data integration, public acceptance
Finance	Fraud detection, trading, risk assessment, automation	Ultra-low latency, high security, real-time processing	Regulatory demands, security threats, high-frequency decisions

3.1. Intelligent Energy Management for Network Applications

Intelligent AI systems can automate energy usage and storage in residential environments within next-generation network infrastructures, serving as crucial elements for decreasing costs, minimizing CO_2 emissions, and enhancing Quality of Service (QoS) [151,159]. The availability of micro-generation systems, electric vehicles, heat pumps, home energy storage, and smart meters provides essential data granularity for intelligent AI models that enable prediction, maintenance, and prevention of energy waste while reducing CO_2 emissions in distributed network environments [160].

Energy prediction in intelligent network applications depends on factors like building type, weather, structural features, and subsystem operations through soft computing approaches [65]. Occupancy behavior enhances statistical and ML models for consumption forecasting in smart network environments. Research demonstrates that simple intelligent models like logistic regression effectively support policy decisions for sustainable energy use in next-generation networks [67]. Real-time electricity usage data provision reduces residential consumption with robust results in intelligent network deployments [66]. Additional efforts include optimizing energy-efficient communication for video streaming in network applications [159].

3.2. Intelligent Smart Agriculture in Network Environments

Intelligent smart agriculture has gained attention for automating traditional farming processes, including precision agriculture monitoring, wildlife tracking, environmental detection, forest fire monitoring, pollution monitoring, and flood monitoring through next-generation network applications [161]. Nanotechnology advancement has enabled compact, inexpensive sensors for diverse applications from beekeeping to precision farming in distributed network infrastructures [162].

Intelligent smart agriculture utilizes modern technology including sensors mounted on farm machines and in fields to collect real-time data about planting, spraying, produce, soil types, and weather in network environments [153]. This data is analyzed using IoT and intelligent edge computing to inform decision-making and increase productivity while reducing environmental damage in next-generation network applications. Recent studies have proposed various systems using intelligent edge computing, fog computing, and cloud infrastructure to collect and analyze data, reduce response time, and improve overall efficiency [163–165].

Research focuses on decreasing energy consumption costs within heterogeneous networks for processing sensor information while enhancing energy efficiency through novel architectures, enabling real-time operations across multiple network layers [153].

3.3. Intelligent Smart Cities for Next-Generation Networks

Intelligent smart city technologies utilize data from devices and sensors to provide near-real-time information for addressing urban challenges in network environments. Data is collected, processed, and analyzed at intelligent edge, fog, or cloud computing layers to provide time-sensitive solutions using wireless sensor networks [154,161]. Large training datasets are essential for intelligent machine learning methods, particularly deep learning architectures in next-generation network applications [156].

Two pivotal aspects for intelligent smart cities include

- Quantity and quality of available data for intelligent city networks;
- Low-latency requirements for city edge nodes, varying by the specific functions required for each network application [37].

Intelligent applications encompass smart homes enabling remote control of energy and water consumption while functioning as security systems [166], smart lighting optimizing energy consumption based on city conditions [167], smart roads contributing to driver safety and traffic management [168,169], and intelligent wireless parking sensors facilitating parking space location while reducing congestion [170].

Location awareness enables faster processing times by allowing intelligent edge networks to collect and process data independently of physical location in distributed environments [109]. Research demonstrates that intelligent fog computing decreases processing time, hop traversal, and bandwidth usage compared to cloud approaches [171]. Mobile edge models for intelligent smart cities leverage network devices for computing tasks, re-

ducing latency for real-time applications like autonomous vehicles and emergency response systems [172].

Advanced approaches include 5G-based smartphone gateways for intelligent healthcare in smart cities, improving energy efficiency while reducing service response time in next-generation network infrastructures [173]. Given resource constraints and the dynamic nature of edge devices in intelligent smart cities, efficient resource distribution is crucial to avoid delays and QoS degradation, particularly as latency increases with device proliferation [6,174].

To solve these problems, one approach is to bring computing resources closer to the edge devices for enabling optimized data processing (see Figure 5). Hong et al. [6] collected multiple architectures and algorithms for resource management at the edge, considering the fog–edge computing model (see Figure 6).

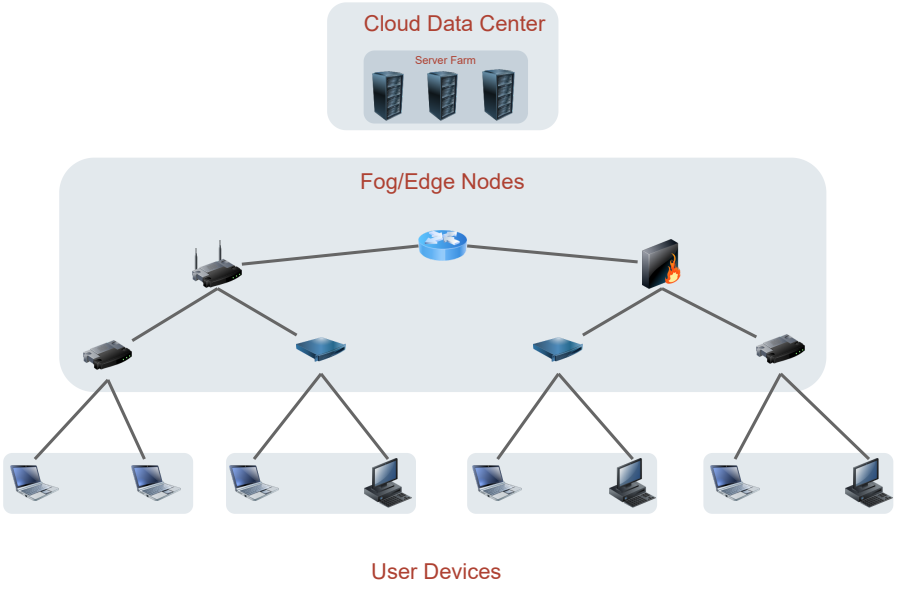

Figure 5. A fog/edge computing model encompassing cloud resources at the edge of the network [6].

3.4. Intelligent Healthcare Networks

In intelligent healthcare applications, edge and fog computing enable real-time patient data collection and local analysis, facilitating faster, more accurate diagnoses and treatment opportunities in next-generation network environments [154,155]. Intelligent sensors monitor patient vital signs, including blood pressure and heart rate. Connecting edge devices with cloud infrastructure enables cost-effective telemedicine with improved response times, reduced latency, and streamlined workflows in distributed healthcare networks [175].

Intelligent edge and fog computing enable remote monitoring and telemedicine, allowing patients to track their health from home while participating in real-time video conferences or teleoperations in network environments [176]. Research has established metrics for evaluating medical indicators including transmission, retrieval, encryption, and authentication in intelligent network applications [177]. IoT solutions using Raspberry Pi sensors enable measurement of vital body parameters with high accuracy and cost-effectiveness [178]. Specialized systems target diabetes monitoring using continuous insulin pumps and glucose monitors connected to intelligent fog layers for cloud data processing [176]. Edge ML-enabled IoT healthcare monitoring systems emphasize efficient data collection, processing, and distribution while minimizing latency in next-generation network infrastructures [154].

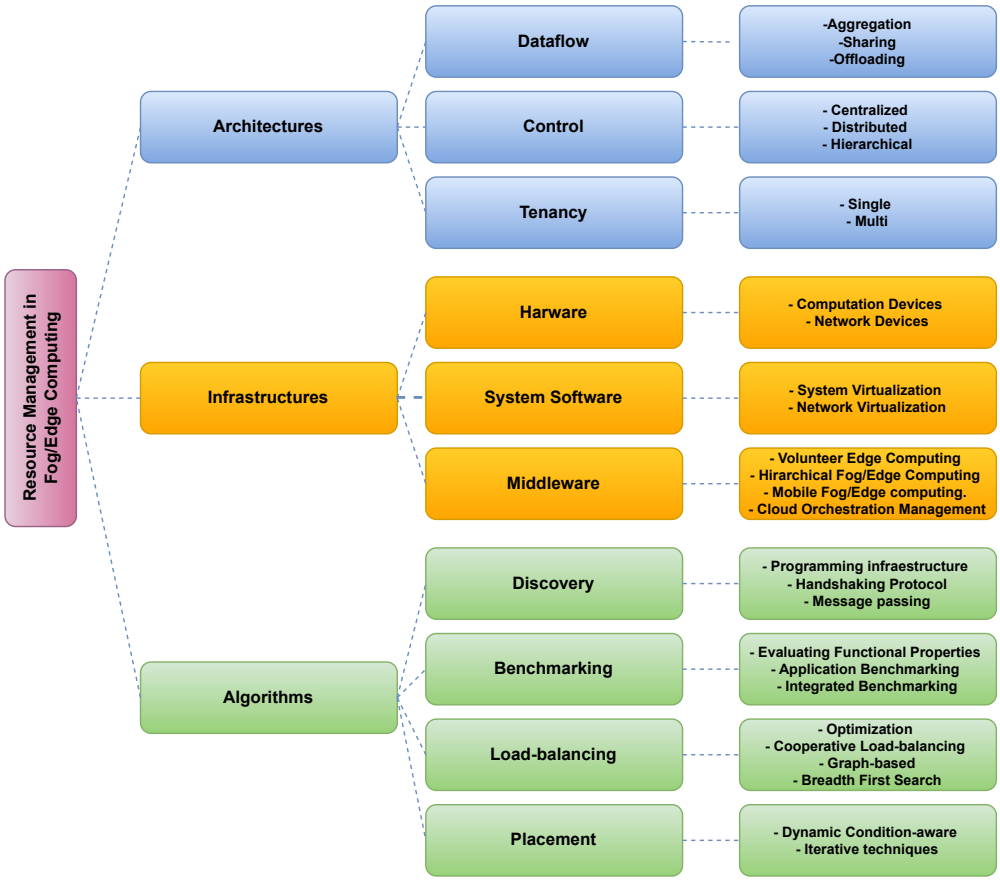

Figure 6. Architectures and algorithms for resource management on a fog/edge computing model [6].

3.5. Intelligent Smart Industry

Industry 4.0 benefits from intelligent AI solutions across multiple domains including predictive maintenance, waste reduction, quality control, sustainability, process optimization, production planning, safety, and human–robot collaboration, enhancing machine downtime, costs, and production quality in next-generation network environments [158].

Predictive maintenance (PM) plays a crucial role in manufacturing for monitoring and detecting equipment failure during operation in intelligent network applications. Manufacturers utilize predictive maintenance to improve production line stability [68]. PM detects equipment malfunctions before complete failure, enabling systems to take the necessary preventive action. With advancement in intelligent edge IoT and inference capabilities, real-time data collection and monitoring for detecting potential faults and abnormalities within distributed systems is now possible [69]. PM objectives include detecting abnormalities before occurrence, reducing maintenance costs, failure rates, and downtime while improving system reliability in intelligent network infrastructures.

Recent approaches utilize intelligent edge for industrial IoT applications [179,180]. Real-time frameworks using intelligent deep learning for manufacturing inspection detect defects, improve efficiency, and provide defect information in fog computing environments for next-generation network applications [181].

3.6. Intelligent Internet of Vehicles

The intelligent Internet of Vehicles (IoV), enabled by edge computing devices, facilitates vehicle monitoring to enhance road safety, efficient communication, congestion avoidance, traffic maintenance, and parking optimization in next-generation network infrastructures [157]. Intelligent IoV edge devices provide efficient road congestion estimation while ensuring user location privacy [70]. Advanced Driver Assistance Systems

built on intelligent edge platforms enable efficient cloud network communication, providing low-latency real-time driving assistance, including weather prediction and smart navigation [71].

Research demonstrates that offloading autonomous driving services via intelligent edge computing improves autonomous driving QoS, as edge devices process large amounts of sensor data for safe and reliable decision-making in distributed network environments [72,182].

3.7. Intelligent Smart Environment

Environmental safety challenges require intelligent smart environment technologies to provide safe, healthy environments and propose innovative solutions for efficient energy conservation in next-generation network applications [183]. Real-time air quality monitoring systems using intelligent sensor data improve indoor and outdoor air quality through distributed computing frameworks interacting across cloud and edge networks [184]. Raspberry Pi-based intelligent IoT edge solutions enable air quality monitoring and prediction [185].

Water quality monitoring utilizes IoT-based real-time frameworks for intelligent water quality management, monitoring, and alert generation based on contamination and toxic parameter levels in network environments [186]. Intelligent monitoring and prediction using drones and UAVs addresses geological hazards including landslides, earthquakes, and forest fires [187,188]. Efficient garbage collection management employs intelligent IoT edge devices with robotic systems for automated cleaning in next-generation network infrastructures [189].

3.8. Intelligent Operating Systems for Network Applications

Accommodating the resource limitations of low-end devices, including limited memory, computational resources, and power supply, requires selecting operating systems that optimize available resources effectively for next-generation network applications. This includes traditional systems like Linux [32], with Linux-Docker-based frameworks enhancing IoT security in network environments [190].

Operating systems for intelligent edge devices must be compact and efficient to accommodate limited processing and storage capabilities in distributed networks. Many OS systems aid intelligent edge application creation, addressing real-time performance, connectivity protocols, power management, and security. Key systems include TinyOS for sensor networks [191], EOS for telecommunications edge infrastructure [192], and ThingSpire OS for coherent processing across IoT devices and the cloud in network environments [193].

Specialized systems include Zephyr for minimal-footprint operations [194], RIOT and NuttX for real-time performance and low power consumption [195], Mbed OS, which emphasizes connectivity [196], and real-time systems like FreeRTOS and Azure RTOS ThreadX that have a reduced memory footprint [197,198].

The IoT2Cloud Operating System (ICOS) prioritizes device diversity, infrastructure virtualization, network diversity, scalability, privacy, security, and data sharing for intelligent edge market scenarios in next-generation network continuum paradigms.

4. State-of-the-Art Intelligent Edge ML Solutions for Next-Generation Network Applications

The future of intelligent machine learning at the edge is shaped by cloud offloading techniques, distributed learning approaches, context-awareness, soft computing model compression techniques, specialized hardware, and security and privacy via design principles [199–202]. These trends enable truly ubiquitous intelligent edge computing that can be popularized across AI applications in next-generation network infrastructures [1].

This section examines the existing technologies and methods currently used to address practical use cases across different application domains in intelligent network environments. These represent state-of-the-art solutions deployed today to meet specific needs in edge environments rather than addressing future research challenges. For each case, we evaluate technical maturity by incorporating concise assessments in the concluding paragraphs that categorize solutions based on their deployment readiness and commercial availability. These assessments distinguish between production-ready technologies with established commercial platforms, pilot-phase solutions requiring specialized expertise, and research-phase approaches still under development [203,204]. These technologies for next-generation network applications are depicted in Figure 7.

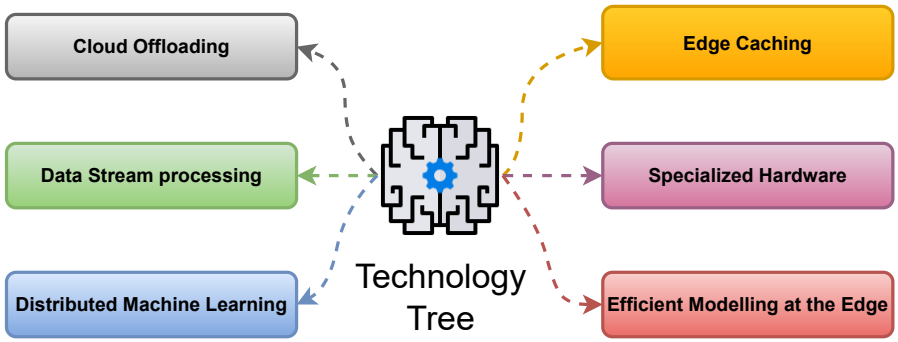

Figure 7. Solutions and technologies for edge machine learning.

4.1. Intelligent Cloud Offloading for Network Applications

Intelligent cloud offloading serves as a pivotal mechanism within next-generation edge cloud frameworks, enabling mechanisms that minimize latency, enhance energy efficiency, and optimize cost-effectiveness in distributed network environments. This approach streamlines and fine-tunes the coordination of computational resources, contributing to efficient operation of intelligent edge cloud systems by making them more responsive, sustainable, and economically viable for network applications.

Many intelligent cloud offloading technologies explore energy–performance trade-offs in IoT environments, focusing on real-time applications like vision-aided games, augmented reality, connected health, and vehicular multi-access edge computing networks [4,150,205]. Intelligent edge cloud offloading addresses centralized cloud computing limitations by relocating computation and storage resources closer to network devices, supporting resource-intensive applications in next-generation network infrastructures [4].

While intelligent edge computing reduces latency and improves energy efficiency through local data storage, systems like CloneCloud transform mobile applications to leverage cloud resources for specific tasks within edge devices [206]. Intelligent fog nodes drastically reduce latency between applications and the centralized cloud, improving QoS in network environments [150]. The primary goal is optimizing offloading for low latency and energy efficiency in computational offloading for intelligent network applications.

Advanced approaches utilize deep learning models for IoT applications with intelligent edge computing [207], while optimization techniques employ DQN for task offloading and wireless resource allocation to maximize network data acquisition and analysis capabilities [208,209]. The Internet of Vehicles exemplifies the necessity of real-time data collection and resource allocation optimization, highlighting the crucial synergy between AI and intelligent edge computing in next-generation network applications [210]. These cloud offloading approaches have reached commercial maturity, with widespread deployment across major cloud platforms including AWS, Azure, and Google Cloud, making them readily accessible for production implementations in next-generation network infrastructures.

4.2. Intelligent Edge Caching for Network Environments

Intelligent edge caching technology enables scenarios like object detection in edge device video analysis within smart city contexts, supporting AIoT platforms that collect video data from personal edge devices and transform it into valuable information for IoT and intelligent network applications [211]. However, video surveillance data analytics presents privacy concerns and unauthorized data exposure risks when uploading videos without user consent. Implementing intelligent video analytics with offloading approaches accelerates processing while reducing latency and resource consumption in network environments [109].

Internet of Vehicles applications demonstrate intelligent edge caching where multiple moving agents share data across networks, utilizing reinforcement learning for efficient networking in next-generation infrastructures. Research shows how intelligent edge caching with reinforcement learning facilitates computational workload offloading in 6G-enabled IoV environments [212].

Collaborative caching approaches among agents for sharing multimedia content minimize content access latencies and improve caching resource utilization in intelligent network applications [213]. Deep learning-based cloud video recommendation systems enhance accuracy through user profiles and video descriptions, incorporating federated learning for collaborative training across distributed cloud servers [214]. Federated video offloading via intelligent edge networks utilizes federated learning algorithms to optimize accuracy while reducing offloading latency [215].

Intelligent edge caching enables real-time data analysis and decision-making closer to data resources, offering faster processing, reduced latency, and resource efficiency for next-generation network applications. This provides viable solutions for federated learning involving heterogeneous edge devices through cache-driven learning approaches [216,217]. Self-trained and auto-tuned intelligent edge caching systems utilize deep reinforcement learning techniques for autonomous decision-making in network environments [217]. While basic edge caching has achieved production maturity through CDN providers like Cloudflare [218], intelligent caching approaches incorporating federated learning and deep reinforcement learning remain in pilot-testing phases, requiring specialized expertise for deployment in network infrastructures.

4.3. Intelligent Data Stream Processing for Network Applications

Intelligent data stream processing has gained attention across various fields, particularly for real-time inference use cases including streaming data display on mobile devices and the transfer of data mining results over limited-bandwidth wireless networks in next-generation infrastructures [219]. Selecting appropriate data streaming frameworks while maximizing the real-time processing of high-volume heterogeneous data streams remains challenging in intelligent network environments [47].

Intelligent edge analytics systems process data dynamically at edges and in the cloud in real-time, working in distributed contexts where analytical latency depends on the users, applications, and data from multiple regions [220]. Traffic video analytics systems produce high-quality analytical results while maintaining minimal resource consumption, including public cloud options and private edge nodes with various hardware for media processing and algorithm execution in intelligent network applications [221].

The rapid processing of continuous data streams within constrained timeframes emphasizes utilizing resource elasticity features from cloud computing, enabling dynamic system scaling in response to demand conditions in next-generation network environments [222].

Data streams in continuous ML scenarios leverage the intelligent stream learning literature, where algorithms continuously adapt to incoming data changes for sustained

performance [59]. Common solutions for AI degradation include model retraining, early stopping mechanisms, and constant ground-truth data access [121]. Alternative solutions utilize online or adaptive machine learning techniques, with forgetting mechanisms enabling faster real-time inference in intelligent network applications [59]. These data stream processing approaches exhibit varying technical maturity levels. Traditional streaming frameworks like Apache Kafka [223] have achieved production readiness, while adaptive machine learning techniques with forgetting mechanisms are still primarily in the research and pilot phases and require further development for robust network deployment.

4.4. Intelligent Distributed Machine Learning for Networks

Intelligent distributed machine learning at the edge enables knowledge interchange without centralized data storage, which is essential for resource-intensive training and inference of complex models in next-generation network environments [1,34]. Parallel computing methods demonstrate the superiority of GPUs over CPUs in latency reduction for intelligent network applications [224].

Intelligent distributed learning enhances user privacy protection across communication networks through training and inference on local data, reducing exposure to attacks in network environments [225]. Methods including model partitioning, federated learning baselines, and spatiotemporal data fusion techniques reduce private data exposure, particularly when applied to heterogeneous edge devices [19,132].

Privacy issues are addressed through access policies detecting anomalies or harmful threats, demonstrating that communication protocols and advanced techniques like data anonymization provide potential solutions for IoT security challenges in intelligent network infrastructures [150].

Recent approaches include distributed edge/cloud paradigms for lightweight virtualization using Docker [226]. These paradigms combine deep learning applications with edge optimizations across inference, computation, and training [39] and explore comprehensive Edge ML aspects including caching, training, inference, and offloading [227]. AI-to-edge architectures investigate intelligent Edge ML across domains, considering sensors, analytics, and ML across edge and fog layers for next-generation network applications [1].

Collaborative intelligent edge technologies facilitate ad hoc networking among stakeholders to coordinate collaborative edge devices and servers for processing geographically distributed data [1,228]. Advanced approaches include Variational Recurrent Neural Networks for distributed cooperative task offloading among multiple agents [229] and collaborative edge computing linking social relationships to physical domains through auction, coalition games, and federated learning solutions addressing incentive compatibility and security challenges in intelligent network environments [152]. Distributed machine learning technologies demonstrate mixed maturity levels, with federated learning frameworks like Pysift [140] reaching pilot deployment stages at major technology companies, and advanced approaches involving variational networks and game-theoretic solutions remaining primarily in the research phase, i.e., requiring substantial development before network-scale implementation.

4.5. Intelligent Efficient Modeling for Network Edge Applications

Intelligent machine learning pipelines must operate with stringent energy efficiency constraints in next-generation network edge environments, where computational resources are limited and power consumption directly impacts device battery life and operational costs. This challenge is particularly acute when training occurs locally or when minimizing data transmission overhead is critical for bandwidth optimization in intelligent network applications [34]. Large data volumes can significantly degrade Quality of Service (QoS)

and increase latency, creating cascading effects on pipeline management and scalability in distributed network environments.

Researchers have developed complementary strategies focusing on hardware optimization, algorithmic efficiency, and architectural design to address energy efficiency constraints in intelligent network applications. Key approaches include deploying energy-efficient hardware accelerators, implementing communication-efficient algorithms for distributed AI model training on edge nodes [2], and developing lightweight models specifically optimized for low-power edge inference in next-generation networks.

The intelligent lightweight model ecosystem has produced notable architectures designed for energy-constrained deployments, including ShuffleNet [230], ShuffleNet V2 [231], MobileNet [232], and SqueezeNet [233]. These models prioritize computational efficiency while maintaining acceptable accuracy levels for intelligent network applications. Supporting frameworks facilitate deployment in distributed network environments.

Recent advances have highlighted significant progress in intelligent energy management and optimization for next-generation networks. Self-learning energy management frameworks using Soft Actor–Critic algorithms achieve substantial energy reduction and battery life improvement through context-aware power optimization [234]. Enhanced IoT routing approaches utilize social attributes and energy awareness for intelligent network applications [235], while Rainbow DQN-based task offloading frameworks improve energy efficiency, reduce latency, and increase utility, demonstrating reinforcement learning effectiveness for energy-aware intelligent edge computing [236].

Beyond energy optimization, fault tolerance remains essential for maintaining reliable intelligent edge operations in network environments. Containerized architectures for fault-tolerant IoT applications [237], intelligent agent-based fault tolerance models [238], and clustering-based approaches [239] complement energy efficiency objectives in next-generation network applications. Efficient modeling techniques exhibit high technical maturity. Model compression methods like quantization and pruning are widely adopted in production frameworks including PyTorch, while fault tolerance approaches through containerization have reached commercial deployment via Docker and Kubernetes. However, intelligent agent-based fault tolerance models remain in the early stages of development.

4.6. Intelligent Specialized Hardware for Network Applications

Technologies in intelligent edge computing depend upon the specialized hardware used for computation in next-generation network environments. Devices with inbuilt CPUs and GPUs provide suitable alternatives for fast processing, model inference, and training in distributed networks. In addition to traditional CPU and GPU architectures, Neural Processing Units (NPUs) have emerged as specialized hardware accelerators specifically designed for AI and machine learning workloads, offering optimized computational architectures for neural network operations with superior energy efficiency compared to general-purpose processors [240–243]. The potential of intelligent edge machine learning is achieved through edge devices equipped with specialized hardware accelerators, including ARM-inbuilt GPUs, TPUs, NPUs, and FPGAs for network applications [199]. NPUs excel at the parallel processing of neural network computations through specialized architectures for matrix operations and support low-precision arithmetic operations such as INT4, INT8, and FP16 to maximize computational efficiency while minimizing power consumption. Modern NPUs integrate high-bandwidth memory architectures and work in heterogeneous computing paradigms alongside CPUs and GPUs, dynamically allocating computational tasks for optimal performance [241,242].

These intelligent hardware accelerators enable effective model inference while drastically reducing latency and power consumption, providing exceptional speed for next-

generation network applications [244–246]. This approach reduces dependencies on cloud servers and helps reduce operational costs since intelligent edge devices become powerful enough to handle heavy workloads independently in network environments. Moreover, these devices are significantly more power efficient than large servers, which require sophisticated cooling systems.

Intelligent systems provide enhanced security levels as data is processed and stored locally within edge devices for model training and serving in network applications. The local processing capabilities of these specialized processors support privacy-preserving applications in the healthcare, automotive, and industrial domains, where data confidentiality is paramount. This enables edge devices to build more the complex ML-based algorithms required for intrusion detection systems, cryptography, and quantum computing applications [247,248].

Specialized hardware in intelligent edge computing creates a greater need for novel methods to train high-quality models within devices, addressing on-device instruction challenges in network environments [249]. The spectrum of intelligent IoT applications continues to expand with GPU-enabled microcontroller capabilities, opening new real-time use cases and facilitating deployment of complex models in next-generation network infrastructures [245]. Specialized hardware technologies show varied maturity levels. Traditional CPUs and GPUs have achieved full production readiness across multiple vendors, while NPUs have reached commercial deployment in mobile devices and some edge systems through companies like Qualcomm and Apple. However, broader enterprise adoption and the standardization of NPU architectures need further development.

5. Research Challenges and Future Directions in Edge Machine Learning

One of the primary goals of this survey is to identify the key challenges and future directions of machine learning (ML) at the edge to expand applications across resource-constrained devices. While the AI space has experienced rapid progress in Large Language Models (LLMs) that extend to other branches of AI, it has become increasingly difficult to adapt such models within constrained settings and sparse data environments.

Figure 8 presents the major research challenges that span three critical dimensions: data-level challenges, including heterogeneity and label scarcity, optimized multimodal AI, and multimodal data fusion; systems-level challenges encompassing efficient orchestration and energy efficiency; and societal-level challenges addressing ethics in the Artificial Intelligence of Things and AI trustworthiness. These interconnected research areas demonstrate how advances in one domain directly influence progress in others, emphasizing the need for holistic approaches to Edge ML development.

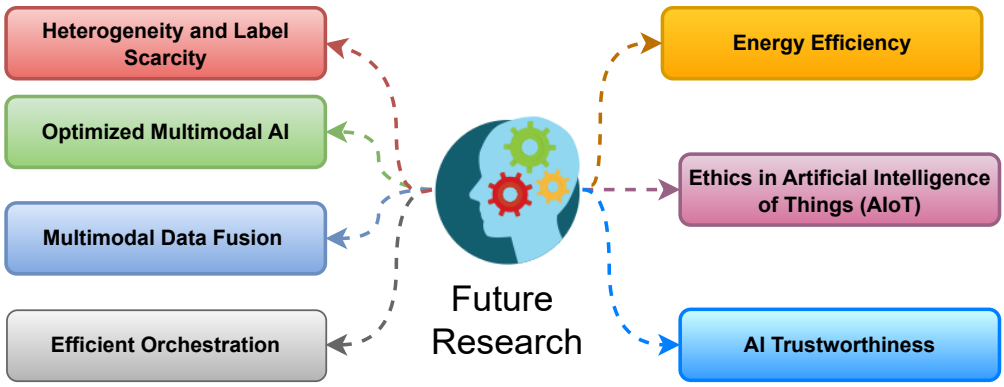

Figure 8. Future research areas for machine learning at the edge. The challenges span data (heterogeneity, label scarcity, multimodal fusion), systems (energy efficiency, orchestration), and societal dimensions (trustworthiness, ethics in AIoT).

The following subsections cover these challenges in more detail.

5.1. Intelligent Heterogeneity and Label Scarcity in Network Environments

A critical challenge in intelligent edge computing for next-generation networks is the limited availability of labeled data, especially within sparse data environments in distributed network infrastructures [43]. Manual data annotation is resource-intensive and typically infeasible in distributed, resource-constrained edge scenarios across network applications.

Several promising strategies have emerged to address this challenge in intelligent network environments. Transfer learning reduces the need for extensive labeled data by leveraging model weights trained on larger datasets for next-generation network applications [52]. Transfer learning combined with monitoring shows promising results in network deployments [250], while data distillation techniques train student models using smaller label amounts by inheriting knowledge from larger teacher models [251]. Semi-supervised learning techniques, combining limited labeled data with abundant unlabeled samples, have emerged as effective strategies for intelligent edge computing. Active learning methods optimize annotation efforts by selectively labeling only the most informative data samples, significantly reducing costs in network environments [252].

Beyond label scarcity, hardware and communication variations in intelligent edge computing profoundly influence data quality, impacting AI model accuracy in next-generation networks. Factors including data noise, incompleteness, heterogeneous hardware capabilities, and remote deployments exacerbate annotation-related challenges. Edge device diversity often necessitates developing and managing multiple tailored machine learning models, potentially resulting in underutilized architectures across network stakeholders [15,73].

Heterogeneity concerns persist regarding fragmented hardware and software ecosystems, complicating compatibility with intelligent MLOps workflows in network environments. Proposed solutions include containerized applications to standardize edge device environments [73] and strategically distributing training workloads between cloud and edge infrastructures according to the available network resources [253].

Integrating these approaches into intelligent Edge ML frameworks can significantly improve model performance while minimizing dependency on extensive, annotated datasets in next-generation network applications. Future research should emphasize developing lightweight, resource-conscious semi-supervised and active learning methods specifically adapted for heterogeneous intelligent edge environments.

5.2. Intelligent Optimized Multimodal AI for Network Applications

The rapid advancement of Large Multimodal Models shows promise across various fields, particularly in resource-constrained settings dominated by mobile devices in next-generation network infrastructures. Vision Language Models and Vision Language Pretraining hold significant potential for applications in real scenarios including mobile devices, self-driving cars, and embedded systems within intelligent network environments [12]. However, they present challenges for the IoT and constrained devices due to larger datasets and computational demands in network applications [254].

Deploying LMMs effectively on the intelligent IoT and edge devices while maintaining generalization and efficiency is essential for practical network applications. While VLMs exhibit exceptional reasoning abilities and effectively integrate multimodal data [50], they remain expensive, demand significant computational resources, and rely on vast datasets. Notable models include OPT [255], Flan-T5 [256], LlaVA [257], BLIP-2 [258], CogVLM [254], and Llama Series [259] for intelligent network applications.

Model optimization techniques for resource-constrained network devices involve complexity reduction via parameter optimization and quantization approaches [13,28,90].

Models like 1-bit BitNet exemplify this concept, enabling lighter implementations on CPUs for intelligent edge computing. Additionally, techniques including pruning [79], distillation [95], low-rank decomposition methods [24], and optimized architectures such as Sparse Mixture of Experts and MobileVLM [12] pave better paths towards optimized AI models for resource-constrained network environments.

Models like MoE LlaVA [260] follow similar approaches by incorporating a mixture of experts, significantly reducing model complexity for intelligent network applications. MobileVLM architectures offer efficient solutions for cross-modality methods and embedding alignment in next-generation networks [12]. However, integrating these models poses challenges, requiring the of exploration of better ways to combine them with low-rank decomposition methods for enhanced results in intelligent edge computing environments.

5.3. Intelligent Multimodal Data Alignment and Fusion for Network Applications

The primary goal of intelligent multimodal fusion is improving learning performance by leveraging complementary information from multiple data modalities, including text, images, and audio in next-generation network environments. A key objective is aligning and integrating heterogeneous data sources for intelligent network applications. Two major technical challenges emerge: alignment, ensuring data from different modalities is temporally or semantically synchronized, and fusion, combining the data into unified representations for network environments [261].

Proper intelligent multimodal alignment is crucial for effective fusion, particularly in spatio-temporal datasets where precise temporal coordination between modalities significantly impacts performance in intelligent network applications [262–264]. This alignment becomes even more critical over distributed network infrastructures [48]. Multimodal fusion creates unified representations of diverse modalities, allowing events to be interpreted from multiple perspectives and data sources in next-generation networks [265].

Intelligent multimodal fusion techniques can be categorized according to integration timing and fusion strategy for network applications. Data fusion typically occurs at the signal, feature, or decision levels [266]. Signal-level fusion combines raw signals directly, feature-level fusion integrates modalities after relevant feature extraction, and decision-level fusion independently processes each modality and then combines the outcomes.

The fusion strategies for intelligent network environments include

1. Early Fusion for Network Applications: This integrates low-level features from multiple modalities by concatenating or merging them into unified representations, enabling models to exploit cross-modal correlations for intelligent edge computing [267]. It has been successfully applied in semantic video analysis, audio–visual fusion, and healthcare applications in network environments.
2. Late Fusion for Network Environments: This integrates classification outcomes from independently trained modality-specific models, providing flexibility for heterogeneous data in intelligent network applications [51,268]. Common applications include multimedia data analysis, health monitoring, and stress detection systems in next-generation networks.
3. Hybrid Fusion for Intelligent Networks: This combines early and late fusion by integrating intermediate representations and final outputs, leveraging the strengths of both approaches for network applications [269]. Applications include emotion recognition, vehicle re-identification, and healthcare IoT-based multimodal fusion in intelligent network environments.

Recent intelligent transformer-based approaches have emerged as alternative fusion strategies for next-generation network applications. Transformer-based unified representations show superior performance compared to traditional fusion methods, particularly in

healthcare and multimedia applications [270,271]. Multi-agent transformer frameworks leverage modality-specific information effectively, demonstrating improved applicability in intelligent network environments [272].

5.4. Intelligent Efficient Orchestration for Network Applications

Intelligent efficient orchestration in edge-to-cloud computing addresses critical challenges for next-generation networks, including optimal resource distribution, load balancing, mobility management, task partitioning, and execution granularity. Effective orchestration ensures seamless coordination and optimal utilization of resources, improving overall system performance and reliability in distributed network environments. This approach emphasizes streamlined management and resource-aware decision-making processes for intelligent network applications.

Managing logical resources including containers or virtual machines is crucial for effective service chain performance in next-generation networks. This involves addressing optimization challenges considering various constraints, including hotspot-related performance issues [38], while proper traffic routing between VMs prevents congestion and ensures uninterrupted service chain operation in network environments [192]. Intelligent edge and fog computing play pivotal roles in scenarios where centralized cloud solutions are unsuitable for network applications [150].

Unlike centralized systems, intelligent edge computing demands high computational power, low response latency, and ample bandwidth, especially for AI-powered streaming or multimedia services in next-generation networks. Specialized intelligent edge OS solutions maintain minimal memory footprint while retaining standard functionality [32]. Recent interest focuses on modular, application-specific operating systems efficiently utilizing resources for intelligent network applications. This shift toward lightweight, tailored OS architectures reduces space and power consumption while optimizing performance across diverse sectors, including smart agriculture, industry, and healthcare [191,273].

Future research in intelligent orchestration for next-generation networks will focus on offloading intensive computation tasks and model compression with efficient computation resource allocation for network applications [274].

5.5. Intelligent Energy Efficiency and Infrastructure Optimization for Networks

The growing MEMS device manufacturing industry utilizes resource-constrained sensors in many intelligent edge applications for next-generation networks [1]. However, measuring energy efficiency in intelligent edge computing systems remains challenging, requiring optimization techniques for network environments. Heterogeneous intelligent edge computing systems encounter two main problems: reducing both size and cost and improving energy efficiency in distributed network infrastructures [16].

Latency, resource optimization, and Quality of Service can constitute problems when deployed in real-world network contexts, necessitating flexible systems designed to match end-user needs in intelligent network applications. Intelligent edge computing frameworks based on reinforcement learning improve energy efficiency by scheduling tasks to appropriate edge devices and the cloud in heterogeneous environments for AIoT task processing in next-generation networks [275]. Comprehensive approaches address energy-efficient communications and computation mechanisms in industrial IoT systems for network applications [276].

Real-time execution becomes complex as intelligent edge computing systems become more sophisticated in network environments. Resource constraints can hinder meeting timing requirements for real-time performance when multiple jobs compete for the same resources [197]. Hardware implementations of Real-Time Operating System functions

improve real-time performance, reducing software-based RTOS overheads and achieving faster task scheduling for intelligent network applications.

Mitigating latency and enabling real-time inference in intelligent edge computing allows exploration of creative solutions matching low-latency processing demands with edge device limitations in next-generation network environments [1].

5.6. Intelligent Ethics in AIoT for Network Applications

Ethical challenges in the intelligent AIoT remain major concerns for next-generation networks, particularly around fairness, accountability, and transparency. Addressing data bias requires embedding principles like justice, honesty, and sustainability directly into intelligent system design for network applications [277]. However, many systems lack mechanisms for human-centered decision-making and overlook ethical priorities in pursuit of operational goals in network environments.

Trustworthy intelligent AI must incorporate human values, including self-awareness and uncertainty estimation for next-generation network applications. Self-correcting models trained via multi-turn reinforcement learning achieve strong results in intelligent network environments [278]. Future intelligent models should learn to identify uncertainty, reduce hallucinations, and ask clarification questions in ambiguous scenarios for network applications.

Ethical deployment requires robust protection of intelligent models in production, especially on edge devices processing sensitive data in network environments. Model stealing, plagiarism, and adversarial access remain active threats requiring secure MLOps pipelines to preserve privacy and integrity in intelligent network applications [73].

AI ethics research must focus on integrating fairness, robustness, and accountability across the intelligent AI lifecycle for next-generation networks. This includes addressing data privacy, algorithmic bias, intellectual property protection, and embedding ethical reflection into development practices for network applications [279].

5.7. Intelligent AI Trustworthiness for Network Applications

Ensuring the trustworthiness of intelligent machine learning systems in production network environments requires robust practices for monitoring, observability, and explainability from both ethical and operational perspectives [74]. Continuous monitoring and timely alerting for anomalies or data distribution shifts enable proactive interventions, preventing performance degradation in intelligent network applications [118].

Challenges related to intelligent model degradation encompass model and data dependencies, including model decay, overfitting, data shift, data quality issues, and concept drift, necessitating robust strategies for next-generation networks [14]. Data-model dependencies can be location-specific or time-specific, with techniques like federated learning and transfer learning offering solutions for intelligent network environments [121].

Model retraining becomes crucial when input data streams evolve in intelligent network applications, addressing catastrophic forgetting and efficient retraining through federated learning for enhanced privacy and security. Continuous learning integrated with federated learning using multi-objective approaches balances accuracy and explainability while demonstrating privacy-preserving ML for intelligent edge computing [280].

Data dependencies involve the data provided during training and inference in network environments. The key challenges for this include handling data shifts affecting distribution and impacting model performance, data quality issues with an impact on preprocessing, and sensor reading issues generating noise in data streams. Concept drift, referring to environmental changes in data evolution, is analyzed from temporal data drift perspectives in next-generation networks [59].

Integration of these intelligent models for retraining enhances the adaptability of complex machine learning systems, introducing incremental learning approaches that enable informed self-adaptations while balancing precision and training time for network applications [124,281,282].

5.8. Intelligent Edge ML Challenges to Solutions Mapping for Networks

Current intelligent Edge ML challenges are addressed through emerging solutions for next-generation network applications, though many remain under active development. Figure 9 provides a comprehensive mapping framework that systematically connects seven major research challenges with their corresponding solution trends and detailed descriptions, organized in a three-column structure that facilitates understanding of the challenge–solution relationships in Edge ML.

The mapping reveals how heterogeneity and label scarcity challenges are addressed through transfer learning, data augmentation, knowledge distillation, and semi-/unsupervised learning approaches that mitigate annotation costs and improve generalization across diverse devices and datasets. Optimized multimodal AI challenges leverage lightweight LMMs, quantization, MobileVLM, and efficient backbones to enable multimodal learning in resource-constrained environments using specialized architectures.

Heterogeneity and label scarcity are tackled via Intelligent Distributed Machine Learning through federated learning and transfer learning approaches for network environments. Optimized multimodal AI and multimodal data alignment leverage intelligent efficient modeling using compressed architectures and lightweight fusion strategies for network applications.

Intelligent efficient orchestration utilizes cloud offloading and edge caching for intelligent task distribution across edge–cloud network infrastructure. Energy efficiency and infrastructure costs are addressed through intelligent specialized hardware, including neuromorphic chips and AI accelerators for next-generation networks.

Ethics in AIoT and AI trustworthiness employ intelligent data stream processing frameworks, incorporating privacy-preserving techniques and explainable AI methods for network applications. Finally, the framework addresses ethics in AIoT through fairness auditing, explainable AI, ethical design guidelines, and human-centered system design, which shape human–AI systems to advance well-being and ensure just outcomes, while AI trustworthiness is ensured through monitoring, drift detection, federated learning, and model retraining approaches that guarantee robustness, privacy, and transparency through observability and adaptive ML workflows.

These rapidly evolving intelligent solutions continue to emerge as the field advances toward mature Edge ML systems for next-generation network infrastructures.

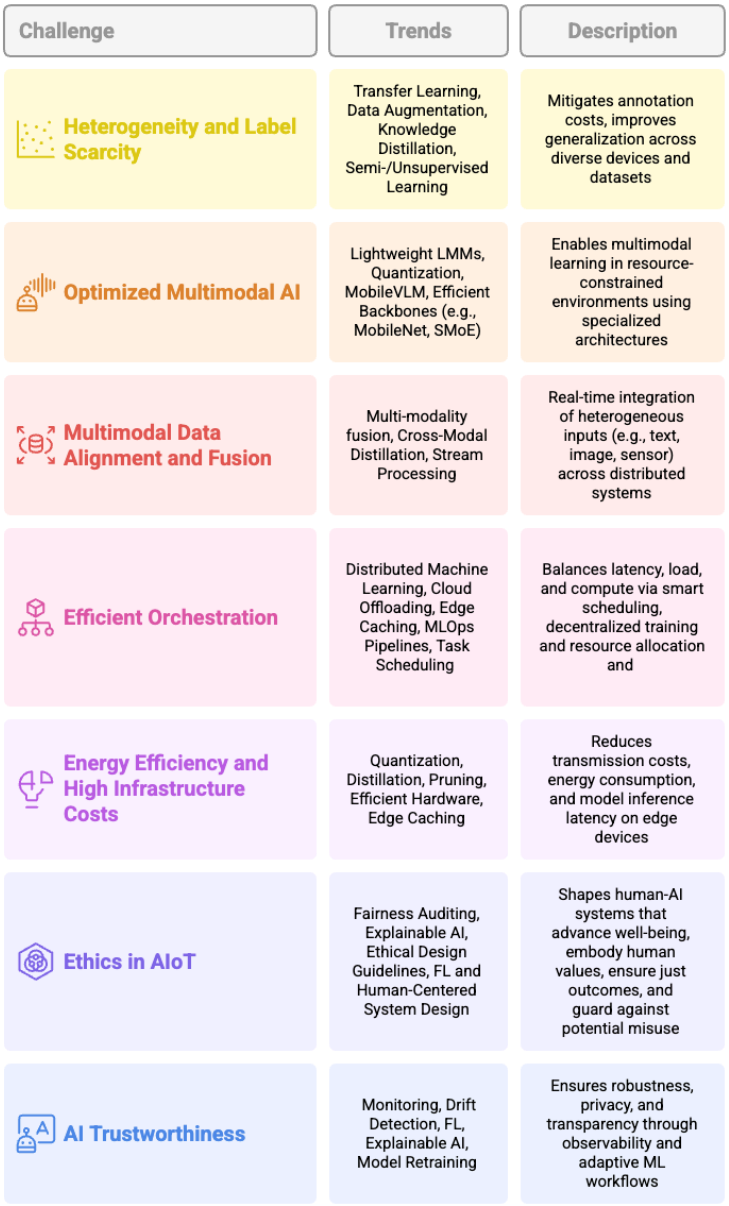

Figure 9. Mapping of future research challenges and research trends in Edge ML.

6. Conclusions

Intelligent machine learning at the edge has garnered significant attention from academia and industry through its plethora of real-world applications and optimizations for next-generation network environments. This survey addresses concerns about efficiency, latency in computing-constrained network infrastructures, accuracy maintenance challenges, and prevalent system heterogeneity in distributed intelligent systems. Additionally, we emphasize the influence of AI degradation, real-time data drifts, and ethical considerations regarding user privacy in intelligent network applications.

We explore the fundamental principles crucial for advancing intelligent edge machine learning, including the evolving domain of neural network optimization, federated learning, and resource optimization algorithms within intelligent MLOps frameworks for next-generation networks. We outline common scenarios for effective intelligent edge computing implementation and explore the approaches, solutions, and challenges they entail in distributed network environments.

Although intelligent machine learning at the edge remains in the early stages for network applications, we hope this survey illuminates promising directions for future research in next-generation infrastructures. We envision a future where the development of highly efficient large models capable of running on resource-limited devices in real-time becomes essential for intelligent network applications. This includes optimizing multimodal AI models for intelligent edge deployment, managing distributed heterogeneous data in network environments, and designing ethical human-centered AI systems that align with human values including humility, fairness, and transparency, while also being capable of self-calibration and self-correction in dynamic network environments.

These capabilities will be especially important in safety-critical domains and scenarios where internet access is unavailable, such as intelligent agriculture, military operations, and remote exploration, where reliable decision-making and ethical considerations are paramount for next-generation network applications. This survey draws attention to existing difficulties and emphasizes the wide range of applications and industries with significant development potential in intelligent network infrastructures.

Intelligent machine learning at the edge may provide better accuracy, lower latency, and more responsive user experiences by creating customized hardware that seamlessly integrates with GPUs and applying state-of-the-art neural network optimization algorithms, thereby improving the quality of service in next-generation network applications. We genuinely hope this study encourages researchers and stakeholders to investigate the plethora of opportunities and efforts that relate to this promising topic for intelligent edge computing in distributed network environments.

Author Contributions: Conceptualization, S.A.C.O. and A.L.S.-C.; methodology, S.A.C.O. and A.L.S.-C.; investigation, S.A.C.O., J.S. and A.L.S.-C.; writing—original draft preparation, S.A.C.O.; writing—review and editing, A.L.S.-C., J.S. and S.A.C.O.; visualization, S.A.C.O.; supervision, A.L.S.-C.; project administration, R.S.C.; funding acquisition, A.L.S.-C. and R.S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the HORIZON research and innovation program of the European Union under Grant No. 101070177 for the project "Towards a functional continuum operating system (ICOS)". Andrés L. Suárez-Cetrulo and Ricardo Simón Carbajo were additionally supported by the European Commission under Grant No. 101189589 through the HORIZON EU program for the project "Open CloudEdgeIoT Platform Uptake in Large Scale Cross-Domain Pilots (O-CEI)". The APC was funded by the European Union HORIZON research and innovation program.

Acknowledgments: We are grateful to our colleagues at ICOS and O-CEI Horizon projects and Ireland's Centre for Applied AI for helping to start and shape this research effort. The continual development of the intelligence layer and AIOps elements by CeADAR serves as the basis for this study.

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

References

1. Zhou, Z.; Chen, X.; Li, E.; Zeng, L.; Luo, K.; Zhang, J. Edge Intelligence: Paving the Last Mile of Artificial Intelligence with Edge Computing. *Proc. IEEE* **2019**, *107*, 1738–1762. [CrossRef]
2. Shi, Y.; Yang, K.; Jiang, T.; Zhang, J.; Letaief, K.B. Communication-Efficient Edge AI: Algorithms and Systems. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2167–2191. [CrossRef]
3. Deng, S.; Zhao, H.; Fang, W.; Yin, J.; Dustdar, S.; Zomaya, A.Y. Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence. *IEEE Internet Things J.* **2020**, *7*, 7457–7469. [CrossRef]
4. Wang, J.; Pan, J.; Esposito, F.; Calyam, P.; Yang, Z.; Mohapatra, P. Edge cloud offloading algorithms: Issues, methods, and perspectives. *ACM Comput. Surv.* **2019**, *52*, 23. [CrossRef]
5. Xu, D.; Li, T.; Li, Y.; Su, X.; Tarkoma, S.; Jiang, T.; Crowcroft, J.; Hui, P. Edge Intelligence: Architectures, Challenges, and Applications. *arXiv* **2020**, arXiv:2003.12172. [CrossRef]

6. Hong, C.H.; Varghese, B. Resource management in fog/edge computing: A survey on architectures, infrastructure, and algorithms. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 97. [CrossRef]
7. Ashish, V. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.
8. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
9. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
10. Gan, Z.; Li, L.; Li, C.; Wang, L.; Liu, Z.; Gao, J. Vision-language pre-training: Basics, recent advances, and future trends. *Found. Trends® Comput. Graph. Vis.* **2022**, *14*, 163–352. [CrossRef]
11. Chu, X.; Qiao, L.; Lin, X.; Xu, S.; Yang, Y.; Hu, Y.; Wei, F.; Zhang, X.; Zhang, B.; Wei, X.; et al. Mobilevlm: A fast, reproducible and strong vision language assistant for mobile devices. *arXiv* **2023**, arXiv:2312.16886.
12. Chu, X.; Qiao, L.; Zhang, X.; Xu, S.; Wei, F.; Yang, Y.; Sun, X.; Hu, Y.; Lin, X.; Zhang, B.; et al. MobileVLM V2: Faster and Stronger Baseline for Vision Language Model. *arXiv* **2024**, arXiv:2402.03766. [CrossRef]
13. Ma, S.; Wang, H.; Ma, L.; Wang, L.; Wang, W.; Huang, S.; Dong, L.; Wang, R.; Xue, J.; Wei, F. The Era of 1-bit LLMs: All Large Language Models are in 1.58 Bits. *arXiv* **2024**, arXiv:2402.17764. [CrossRef]
14. Bernabé-Sánchez, I.; Fernández, A.; Billhardt, H.; Ossowski, S. Problem Detection in the Edge of IoT Applications. *IJIMAI* **2023**, *8*. [CrossRef]
15. Makhija, D.; Han, X.; Ho, N.; Ghosh, J. Architecture Agnostic Federated Learning for Neural Networks; *arXiv* **2022**, arXiv:2202.07757. [CrossRef]
16. Jiang, C.; Fan, T.; Gao, H.; Shi, W.; Liu, L.; Cérin, C.; Wan, J. Energy aware edge computing: A survey. *Comput. Commun.* **2020**, *151*, 556–580. [CrossRef]
17. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.Y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the Artificial Intelligence and Statistics, PMLR, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
18. Bonawitz, K.; Eichner, H.; Grieskamp, W.; Huba, D.; Ingerman, A.; Ivanov, V.; Kiddon, C.; Konečný, J.; Mazzocchi, S.; McMahan, H.B.; et al. Towards Federated Learning at Scale: System Design *arXiv* **2019**, arXiv:1902.01046. [CrossRef]
19. Filho, C.P.; Marques, E.; Chang, V.; dos Santos, L.; Bernardini, F.; Pires, P.F.; Ochi, L.; Delicato, F.C. A Systematic Literature Review on Distributed Machine Learning in Edge Computing. *Sensors* **2022**, *22*, 2665. [CrossRef]
20. Plastiras, G.; Terzi, M.; Kyrkou, C.; Theocharidcs, T. Edge Intelligence: Challenges and Opportunities of Near-Sensor Machine Learning Applications. In Proceedings of the International Conference on Application-Specific Systems, Architectures and Processors, Milano, Italy, 10–12 July 2018. [CrossRef]
21. Wang, X.; Li, J.; Ning, Z.; Song, Q.; Guo, L.; Guo, S.; Obaidat, M.S. Wireless Powered Mobile Edge Computing Networks: A Survey. *ACM Comput. Surv.* **2022**, *55*, 263. [CrossRef]
22. Han, S.; Liu, X.; Mao, H.; Pu, J.; Pedram, A.; Horowitz, M.A.; Dally, W.J. EIE: Efficient Inference Engine on Compressed Deep Neural Network. In Proceedings of the—2016 43rd International Symposium on Computer Architecture, Seoul, Republic of Korea, 18–22 June 2016; pp. 243–254. [CrossRef]
23. Sze, V.; Chen, Y.H.; Yang, T.J.; Emer, J.S. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. *Proc. IEEE* **2017**, *105*, 2295–2329. [CrossRef]
24. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *arXiv* **2021**, arXiv:2106.09685.
25. Xu, Y.; Xie, L.; Gu, X.; Chen, X.; Chang, H.; Zhang, H.; Chen, Z.; Zhang, X.; Tian, Q. Qa-lora: Quantization-aware low-rank adaptation of large language models. *arXiv* **2023**, arXiv:2309.14717.
26. Dettmers, T.; Pagnoni, A.; Holtzman, A.; Zettlemoyer, L. Qlora: Efficient finetuning of quantized llms. *arXiv* **2023**, arXiv:2305.14314. [CrossRef]
27. Liu, S.Y.; Wang, C.Y.; Yin, H.; Molchanov, P.; Wang, Y.C.F.; Cheng, K.T.; Chen, M.H. DoRA: Weight-Decomposed Low-Rank Adaptation. *arXiv* **2024**, arXiv:2402.09353.
28. Wang, H.; Ma, S.; Dong, L.; Huang, S.; Wang, H.; Ma, L.; Yang, F.; Wang, R.; Wu, Y.; Wei, F. Bitnet: Scaling 1-bit transformers for large language models. *arXiv* **2023**, arXiv:2310.11453. [CrossRef]
29. Alemdar, H.; Leroy, V.; Prost-Boucle, A.; Pétrot, F. Ternary neural networks for resource-efficient AI applications. In Proceedings of the IEEE 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 2547–2554.
30. Courbariaux, M.; Hubara, I.; Soudry, D.; El-Yaniv, R.; Bengio, Y. Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or −1. *arXiv* **2016**, arXiv:1602.02830. [CrossRef]
31. Gou, J.; Yu, B.; Maybank, S.J.; Tao, D. Knowledge Distillation: A Survey. *Int. J. Comput. Vis.* **2020**, *129*, 1789–1819. [CrossRef]

32. Hahm, O.; Baccelli, E.; Petersen, H.; Tsiftes, N. Operating Systems for Low-End Devices in the Internet of Things: A Survey. *IEEE Internet Things J.* **2016**, *3*, 720–734. [CrossRef]
33. Pope, R.; Douglas, S.; Chowdhery, A.; Devlin, J.; Bradbury, J.; Levskaya, A.; Heek, J.; Xiao, K.; Agrawal, S.; Dean, J. Efficiently Scaling Transformer Inference. *arXiv* **2022**, arXiv:2211.05102. [CrossRef]
34. Wang, J.; Zhu, X.; Zhang, J.; Cao, B.; Bao, W.; Yu, P.S. Not Just Privacy: Improving Performance of Private Deep Learning in Mobile Cloud. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London, UK, 19–23 August 2018; pp. 2407–2416. [CrossRef]
35. Allhoff, F.; Henschke, A. The Internet of Things: Foundational ethical issues. *Internet Things* **2018**, *1–2*, 55–66. [CrossRef]
36. Iftikhar, S.; Gill, S.S.; Song, C.; Xu, M.; Aslanpour, M.S.; Toosi, A.N.; Du, J.; Wu, H.; Ghosh, S.; Chowdhury, D.; et al. AI-based fog and edge computing: A systematic review, taxonomy and future directions. *arXiv* **2023**, arXiv:022.100674. [CrossRef]
37. Hamdan, S.; Ayyash, M.; Almajali, S. Edge-Computing Architectures for Internet of Things Applications: A Survey. *Sensors* **2020**, *20*, 6441. [CrossRef]
38. Rodrigues, T.K.; Suto, K.; Nishiyama, H.; Liu, J.; Kato, N. Machine Learning Meets Computation and Communication Control in Evolving Edge and Cloud: Challenges and Future Perspective. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 38–67. [CrossRef]
39. Wang, X.; Han, Y.; Leung, V.C.; Niyato, D.; Yan, X.; Chen, X. Convergence of Edge Computing and Deep Learning: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 869–904. [CrossRef]
40. Singh, R.; Sukapuram, R.; Chakraborty, S. A survey of mobility-aware Multi-access Edge Computing: Challenges, use cases and future directions. *Ad Hoc Netw.* **2023**, *140*, 103044. [CrossRef]
41. Verbraeken, J.; Wolting, M.; Katzy, J.; Kloppenburg, J.; Verbelen, T.; Rellermeyer, J.S. A Survey on Distributed Machine Learning. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 30. [CrossRef]
42. Yu, W.; Liang, F.; He, X.; Hatcher, W.G.; Lu, C.; Lin, J.; Yang, X. A Survey on the Edge Computing for the Internet of Things. *IEEE Access* **2017**, *6*, 6900–6919. [CrossRef]
43. Mao, Y.; Yu, X.; Huang, K.; Zhang, Y.J.A.; Zhang, J. Green edge AI: A contemporary survey. *Proc. IEEE* **2024**, *112*, 880–911. [CrossRef]
44. Rodriguez, M.A.; Buyya, R. Container-based cluster orchestration systems: A taxonomy and future directions. *Softw. Pract. Exp.* **2019**, *49*, 698–719. [CrossRef]
45. Zhong, Z.; Xu, M.; Rodriguez, M.A.; Xu, C.; Buyya, R. Machine Learning-based Orchestration of Containers: A Taxonomy and Future Directions. *ACM Comput. Surv.* **2022**, *54*, 217. [CrossRef]
46. Casalicchio, E. Container Orchestration: A Survey. In *EAI/Springer Innovations in Communication and Computing*; Springer: Cham, Switzerland, 2019; pp. 221–235. ._14 [CrossRef]
47. Isah, H.; Abughofa, T.; Mahfuz, S.; Ajerla, D.; Zulkernine, F.; Khan, S. A survey of distributed data stream processing frameworks. *IEEE Access* **2019**, *7*, 154300–154316. [CrossRef]
48. Baltrušaitis, T.; Ahuja, C.; Morency, L.P. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 423–443. [CrossRef]
49. Barua, A.; Ahmed, M.U.; Begum, S. A Systematic Literature Review on Multimodal Machine Learning: Applications, Challenges, Gaps and Future Directions. *IEEE Access* **2023**, *11*, 14804–14831. [CrossRef]
50. Yin, S.; Fu, C.; Zhao, S.; Li, K.; Sun, X.; Xu, T.; Chen, E. A Survey on Multimodal Large Language Models. *arXiv* **2023**, arXiv:2306.13549. [CrossRef] [PubMed]
51. Bayoudh, K.; Knani, R.; Hamdaoui, F.; Mtibaa, A. A survey on deep multimodal learning for computer vision: Advances, trends, applications, and datasets. *Vis. Comput.* **2022**, *38*, 2939–2970. [CrossRef] [PubMed]
52. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2020**, *109*, 43–76. [CrossRef]
53. Liu, S.; Guo, B.; Fang, C.; Wang, Z.; Luo, S.; Zhou, Z.; Yu, Z. Enabling Resource-Efficient AIoT System With Cross-Level Optimization: A Survey. *IEEE Commun. Surv. Tutor.* **2023**, *26*, 389–427. [CrossRef]
54. Surianarayanan, C.; Lawrence, J.J.; Chelliah, P.R.; Prakash, E.; Hewage, C. A Survey on Optimization Techniques for Edge Artificial Intelligence (AI). *Sensors* **2023**, *23*, 1279. [CrossRef]
55. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Vincent Poor, H. Federated Learning for Internet of Things: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1622–1658. [CrossRef]
56. Wu, J.; Drew, S.; Dong, F.; Zhu, Z.; Zhou, J. Topology-aware Federated Learning in Edge Computing: A Comprehensive Survey. *J. ACM* **2023**, *37*, 35. [CrossRef]
57. Kar, B.; Yahya, W.; Lin, Y.D.; Ali, A. Offloading using Traditional Optimization and Machine Learning in Federated Cloud-Edge-Fog Systems: A Survey. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 1199–1226. [CrossRef]
58. Gama, J. A survey on learning from data streams: Current and future trends. *Prog. Artif. Intell.* **2012**, *1*, 45–55. [CrossRef]
59. Suárez-Cetrulo, A.L.; Quintana, D.; Cervantes, A. A survey on machine learning for recurring concept drifting data streams. *Expert Syst. Appl.* **2023**, *213*, 118934. [CrossRef]

60. Baruah, R.D.; Angelov, P. Evolving fuzzy systems for data streams: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2011**, *1*, 461–476. [CrossRef]
61. Krawczyk, B.; Minku, L.L.; Gama, J.; Stefanowski, J.; Woźniak, M. Ensemble learning for data stream analysis: A survey. *Inf. Fusion* **2017**, *37*, 132–156. [CrossRef]
62. Steinberger, J.; Schehlmann, L.; Abt, S.; Baier, H. Anomaly detection and mitigation at Internet scale: A survey. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 7943, pp. 49–60. [CrossRef]
63. Martí, J.; Queralt, A.; Gasull, D.; Barceló, A.; Costa, J.J.; Cortes, T. Dataclay: A distributed data store for effective inter-player data sharing. *J. Syst. Softw.* **2017**, *131*, 129–145. [CrossRef]
64. Gholami, A.; Kim, S.; Dong, Z.; Yao, Z.; Mahoney, M.W.; Keutzer, K. A Survey of Quantization Methods for Efficient Neural Network Inference. *arXiv* **2021**, arXiv:2103.13630. [CrossRef]
65. Zhao, H.X.; Magoulès, F. A review on the prediction of building energy consumption. *Renew. Sustain. Energy Rev.* **2012**, *16*, 3586–3592. [CrossRef]
66. Gans, W.; Alberini, A.; Longo, A. Smart meter devices and the effect of feedback on residential electricity consumption: Evidence from a natural experiment in Northern Ireland. *Energy Econ.* **2013**, *36*, 729–743. [CrossRef]
67. Rausser, G.; Strielkowski, W.; Štreimikienė, D. Smart meters and household electricity consumption: A case study in Ireland. *Energy Environ.* **2017**, *29*, 131–146. [CrossRef]
68. Cao, K.; Liu, Y.; Meng, G.; Sun, Q. An Overview on Edge Computing Research. *IEEE Access* **2020**, *8*, 85714–85728. [CrossRef]
69. Liu, Y.; Yu, W.; Dillon, T.; Rahayu, W.; Li, M. Empowering IoT Predictive Maintenance Solutions with AI: A Distributed System for Manufacturing Plant-Wide Monitoring. *IEEE Trans. Ind. Informat.* **2022**, *18*, 1345–1354. [CrossRef]
70. Babaghayou, M.; Chaib, N.; Lagraa, N.; Ferrag, M.A.; Maglaras, L. A Safety-Aware Location Privacy-Preserving IoV Scheme with Road Congestion-Estimation in Mobile Edge Computing. *Sensors* **2023**, *23*, 531. [CrossRef]
71. Maheshwari, S.; Zhang, W.; Seskar, I.; Zhang, Y.; Raychaudhuri, D. EdgeDrive: Supporting Advanced Driver AssistanceSystems using Mobile Edge Clouds Networks. In Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Paris, France, 29 April–2 May 2019.
72. Liu, L.; Zhao, M.; Yu, M.; Jan, M.A.; Lan, D.; Taherkordi, A. Mobility-Aware Multi-Hop Task Offloading for Autonomous Driving in Vehicular Edge Computing and Networks. *IEEE Trans. Intell. Transport. Syst.* **2022**, *24*, 2169–2182. [CrossRef]
73. Leroux, S.; Simoens, P.; Lootus, M.; Thakore, K.; Sharma, A. TinyMLOps: Operational Challenges for Widespread Edge AI Adoption. In Proceedings of the 2022 IEEE 36th International Parallel and Distributed Processing Symposium Workshops, Lyon, France, 30 May–3 June 2022; pp. 1003–1010 [CrossRef]
74. Symeonidis, G.; Nerantzis, E.; Kazakis, A.; Papakostas, G.A. MLOps—Definitions, Tools and Challenges. In Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference, Las Vegas, NV, USA, 26–29 January; pp. 453–460. [CrossRef]
75. Pustokhina, I.V.; Pustokhin, D.A.; Gupta, D.; Khanna, A.; Shankar, K.; Nguyen, G.N. An Effective Training Scheme for Deep Neural Network in Edge Computing Enabled Internet of Medical Things (IoMT) Systems. *IEEE Access* **2020**, *8*, 107112–107123. [CrossRef]
76. Sudharsan, B.; Patel, P.; Breslin, J.; Ali, M.I.; Mitra, K.; Dustdar, S.; Rana, O.; Jayaraman, P.P.; Ranjan, R. Toward Distributed, Global, Deep Learning Using IoT Devices. *IEEE Internet Comput.* **2021**, *25*, 6–12. [CrossRef]
77. Paganini, M.; Forde, J. Streamlining tensor and network pruning in pytorch. *arXiv* **2020**, arXiv:2004.13770. [CrossRef]
78. Xu, C.; Gao, W.; Li, T.; Bai, N.; Li, G.; Zhang, Y. Teacher-student collaborative knowledge distillation for image classification. *Appl. Intell.* **2022**, *53*, 1997–2009. [CrossRef]
79. Han, S.; Mao, H.; Dally, W.J. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. In Proceedings of the 4th International Conference on Learning Representations, ICLR 2016—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
80. Liu, Z.; Li, J.; Shen, Z.; Huang, G.; Yan, S.; Zhang, C. Learning Efficient Convolutional Networks through Network Slimming. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2755–2763. [CrossRef]
81. Jacob, B.; Kligys, S.; Chen, B.; Zhu, M.; Tang, M.; Howard, A.; Adam, H.; Kalenichenko, D. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2704–2713.
82. Choi, J.; Wang, Z.; Venkataramani, S.; Chuang, P.I.J.; Srinivasan, V.; Gopalakrishnan, K. PACT: Parameterized Clipping Activation for Quantized Neural Networks. *arXiv* **2018**, arXiv:1805.06085. [CrossRef]
83. Wang, K.; Liu, Z.; Lin, Y.; Lin, J.; Han, S. HAQ: Hardware-Aware Automated Quantization with Mixed Precision. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8604–8612. [CrossRef]
84. Hinton, G.; Vinyals, O.; Dean, J. Distilling the Knowledge in a Neural Network. *arXiv* **2015**, arXiv:1503.02531. [CrossRef]

85. Jain, A.; Bhattacharya, S.; Masuda, M.; Sharma, V.; Wang, Y. Efficient Execution of Quantized Deep Learning Models: A Compiler Approach. *arXiv* **2020**, arXiv:2006.10226. [CrossRef]
86. Jaderberg, M.; Vedaldi, A.; Zisserman, A. Speeding up convolutional neural networks with low rank expansions. *arXiv* **2014**, arXiv:1405.3866. [CrossRef]
87. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
88. Anubhav Singh, R.B. *Mobile Deep Learning with TensorFlow Lite, ML Kit and Flutter: Build Scalable Real-World Projects to Implement End-to-End Neural Networks on Android and iOS*; Packt Publishing Ltd.: Mumbai, India, 2020.
89. Krishnamoorthi, R. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv* **2018**, arXiv:1806.08342. [CrossRef]
90. Xiao, G.; Lin, J.; Seznec, M.; Wu, H.; Demouth, J.; Han, S. Smoothquant: Accurate and efficient post-training quantization for large language models. In Proceedings of the International Conference on Machine Learning, PMLR, Honolulu, HI, USA, 23–29 July 2023; pp. 38087–38099.
91. Frantar, E.; Ashkboos, S.; Hoefler, T.; Alistarh, D. OPTQ: Accurate quantization for generative pre-trained transformers. In Proceedings of the 11th International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
92. Lin, J.; Tang, J.; Tang, H.; Yang, S.; Dang, X.; Han, S. AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration. *arXiv* **2023**, arXiv:2306.00978. [CrossRef]
93. Chee, J.; Cai, Y.; Kuleshov, V.; De Sa, C.M. Quip: 2-bit quantization of large language models with guarantees. *Adv. Neural Inf. Process. Syst.* **2023 2015**, *36*, 4396–4429.
94. Sung, W.; Shin, S.; Hwang, K. Resiliency of Deep Neural Networks under Quantization. *arXiv* **2015**, arXiv:1511.06488.
95. Chen, G.; Choi, W.; Yu, X.; Han, T.; Chandraker, M. *Learning Efficient Object Detection Models with Knowledge Distillation*; Advances in Neural Information Processing Systems; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
96. Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [CrossRef] [PubMed]
97. Yim, J.; Joo, D.; Bae, J.; Kim, J. A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7130–7138. [CrossRef]
98. Fang, G.; Song, J.; Shen, C.; Wang, X.; Chen, D.; Song, M. Data-Free Adversarial Distillation. *arXiv* **2020**, arXiv:1912.11006. [CrossRef]
99. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
100. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *arXiv* **2020**, arXiv:2005.14165. [CrossRef]
101. Lee, S.; Song, B.C. Graph-based Knowledge Distillation by Multi-head Attention Network. In Proceedings of the 30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, 9–12 September 2019.
102. Polino, A.; Pascanu, R.; Alistarh, D. Model compression via distillation and quantization. In Proceedings of the 6th International Conference on Learning Representations, ICLR 2018—Conference Track Proceedings, Vancouver, BC, Canada, 30 April–3 May 2018.
103. Chuang, Y.S.; Su, S.Y.; Chen, Y.N. Lifelong Language Knowledge Distillation. In Proceedings of the EMNLP 2020—2020 Conference on Empirical Methods in Natural Language Processing, Online Event, 16–20 November 2020; pp. 2914–2924. [CrossRef]
104. Li, C.; Farkhoor, H.; Liu, R.; Yosinski, J. Measuring the intrinsic dimension of objective landscapes. *arXiv* **2018**, arXiv:1804.08838. [CrossRef]
105. Aghajanyan, A.; Zettlemoyer, L.; Gupta, S. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv* **2020**, arXiv:2012.13255. [CrossRef]
106. Xu, L.; Xie, H.; Qin, S.Z.J.; Tao, X.; Wang, F.L. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv* **2023**, arXiv:2312.12148.
107. Sculley, D.; Holt, G.; Golovin, D.; Davydov, E.; Phillips, T.; Ebner, D.; Chaudhary, V.; Young, M.; Crespo, J.F.; Dennison, D. In *Hidden Technical Debt in Machine Learning Systems*; Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2015.
108. Ruf, P.; Madan, M.; Reich, C.; Ould-Abdeslam, D. Demystifying mlops and presenting a recipe for the selection of open-source tools. *Appl. Sci.* **2021**, *11*, 8861. [CrossRef]
109. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge Computing: Vision and Challenges. *IEEE Internet Things J.* **2016**, *3*, 637–646. [CrossRef]
110. Raj, E.; Buffoni, D.; Westerlund, M.; Ahola, K. Edge MLOps: An Automation Framework for AIoT Applications. In Proceedings of the 2021 IEEE International Conference on Cloud Engineering, IC2E 2021, San Francisco, CA, USA, 22 November 2021; pp. 191–200. [CrossRef]

111. John, M.M.; Olsson, H.H.; Bosch, J. Towards MLOps: A Framework and Maturity Model. In Proceedings of the 2021 47th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2021, Palermo, Italy, 3 September 2021; pp. 334–341. [CrossRef]
112. Tamburri, D.A. Sustainable MLOps: Trends and Challenges. In Proceedings of the 2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2020, Timisoara, Romania, 1–4 September 2020; pp. 17–23. [CrossRef]
113. Kreuzberger, D.; Kühl, N.; Hirschl, S. Machine Learning Operations (MLOps): Overview, Definition, and Architecture. *IEEE Access* **2023**, *11*, 31866–31879. [CrossRef]
114. Zaharia, M.; Chen, A.; Davidson, A.; Ghodsi, A.; Hong, S.A.; Konwinski, A.; Murching, S.; Nykodym, T.; Ogilvie, P.; Parkhe, M.; et al. Accelerating the Machine Learning Lifecycle with MLflow. *IEEE Data Eng. Bull.* **2018**, *41*, 39–45.
115. Bisong, E. Kubeflow and kubeflow pipelines. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 671–685.
116. Bodor, A.; Hnida, M.; Najima, D. MLOps: Overview of current state and future directions. In *Proceedings of the International Conference on Smart City Applications*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 156–165.
117. Chaves, A.J.; Martín, C.; Díaz, M. The orchestration of Machine Learning frameworks with data streams and GPU acceleration in Kafka-ML: A deep-learning performance comparative. *Expert Syst.* **2023**, *41*, e13287. [CrossRef]
118. Klaise, J.; Van Looveren, A.; Cox, C.; Vacanti, G.; Coca, A. Monitoring and explainability of models in production. *arXiv* **2020**, arXiv:2007.06299. [CrossRef]
119. Woźniak, A.P.; Milczarek, M.; Woźniak, J. MLOps Components, Tools, Process, and Metrics: A Systematic Literature Review. *IEEE Access* **2025**, *13*, 22166–22175. [CrossRef]
120. Barry, M.; Bifet, A.; Billy, J.L. StreamAI: Dealing with Challenges of Continual Learning Systems for Serving AI in Production. In Proceedings of the 2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), Melbourne, Australia, 14 May 2023; pp. 134–137. [CrossRef]
121. Vela, D.; Sharp, A.; Zhang, R.; Nguyen, T.; Hoang, A.; Pianykh, O.S. Temporal quality degradation in AI models. *Sci. Rep.* **2022**, *12*, 11654. [CrossRef]
122. Lobo, J.L.; Laña, I.; Osaba, E.; Del Ser, J. On the Connection between Concept Drift and Uncertainty in Industrial Artificial Intelligence. *arXiv* **2023**, arXiv:2303.07940. [CrossRef]
123. Montiel, J.; Halford, M.; Mastelini, S.M.; Bolmier, G.; Sourty, R.; Vaysse, R.; Zouitine, A.; Gomes, H.M.; Read, J.; Abdessalem, T.; et al. River: Machine learning for streaming data in python. *J. Mach. Learn. Res.* **2021**, *22*, 4945–4952.
124. Ordóñez, S.A.C.; Samanta, J.; Suárez-Cetrulo, A.L.; Carbajo, R.S. Adaptive Machine Learning for Resource-Constrained Environments. In *Proceedings of the Discovering Drift Phenomena in Evolving Landscapes*; Piangerelli, M., Prenkaj, B., Rotalinti, Y., Joshi, A., Stilo, G., Eds.; Springer: Cham, Switzerland, 2025; pp. 3–19.
125. Yang, Q.; Liu, Y.; Cheng, Y.; Kang, Y.; Chen, T.; Yu, H. Introduction. In *Federated Learning*; Springer International Publishing: Cham, Switzerland, 2020; pp. 1–15. [CrossRef]
126. Li, T.; Sahu, A.K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; Smith, V. Federated Optimization in Heterogeneous Networks. *Proc. Mach. Learn. Syst.* **2020**, *2*, 429–450.
127. Arivazhagan, M.G.; Aggarwal, V.; Singh, A.K.; Choudhary, S. Federated Learning with Personalization Layers. *arXiv* **2019**, arXiv:1912.00818. [CrossRef]
128. Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečný, J.; Kumar, S.; McMahan, H.B. Adaptive Federated Optimization. *arXiv* **2021**, arXiv:2003.00295. [CrossRef]
129. Thapa, C.; Arachchige, P.C.M.; Camtepe, S.; Sun, L. SplitFed: When Federated Learning Meets Split Learning. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 8485–8493. [CrossRef]
130. Zhang, Z.; He, N.; Li, D.; Gao, H.; Gao, T.; Zhou, C. Federated transfer learning for disaster classification in social computing networks. *J. Saf. Sci. Resil.* **2022**, *3*, 15–23. [CrossRef]
131. Zhu, H.; Jin, Y. Multi-objective Evolutionary Federated Learning. *arXiv* **2019**, arXiv.1812.07478. [CrossRef]
132. Qi, L.; Hu, C.; Zhang, X.; Khosravi, M.R.; Sharma, S.; Pang, S.; Wang, T. Privacy-Aware Data Fusion and Prediction with Spatial-Temporal Context for Smart City Industrial Environment. *IEEE Trans. Ind. Informat.* **2021**, *17*, 4159–4167. [CrossRef]
133. Liu, X.; Shi, T.; Xie, C.; Li, Q.; Hu, K.; Kim, H.; Xu, X.; Li, B.; Song, D. UniFed: A Benchmark for Federated Learning Frameworks. *arXiv* **2022**, arXiv:2207.10308.
134. Kholod, I.; Yanaki, E.; Fomichev, D.; Shalugin, E.; Novikova, E.; Filippov, E.; Nordlund, M. Open-Source Federated Learning Frameworks for IoT: A Comparative Review and Analysis. *Sensors* **2020**, *21*, 167. [CrossRef] [PubMed]
135. He, C.; Li, S.; So, J.; Zeng, X.; Zhang, M.; Wang, H.; Wang, X.; Vepakomma, P.; Singh, A.; Qiu, H.; et al. FedML: A Research Library and Benchmark for Federated Machine Learning. *arXiv* **2020**, arXiv:2007.13518. [CrossRef]
136. Garcia, M.H.; Manoel, A.; Diaz, D.M.; Mireshghallah, F.; Sim, R.; Dimitriadis, D. FLUTE: A Scalable, Extensible Framework for High-Performance Federated Learning Simulations. *arXiv* **2022**, arXiv:2203.13789. [CrossRef]

137. Knott, B.; Venkataraman, S.; Hannun, A.; Sengupta, S.; Ibrahim, M.; van der Maaten, L. CrypTen: Secure Multi-Party Computation Meets Machine Learning. *arXiv* **2022**, arXiv:2109.00984.
138. Li Q, Zhaomin W, Cai Y, Yung CM, Fu T, He B. Fedtree: A federated learning system for trees. *Proc. Mach. Learn. Syst.* **2023**, *5*, 89–103.
139. Roth, H.R.; Cheng, Y.; Wen, Y.; Yang, I.; Xu, Z.; Hsieh, Y.T.; Kersten, K.; Harouni, A.; Zhao, C.; Lu, K.; et al. NVIDIA FLARE: Federated Learning from Simulation to Real-World. *arXiv* **2022**, arXiv:2210.13291.
140. Ziller, A.; Trask, A.; Lopardo, A.; Szymkow, B.; Wagner, B.; Bluemke, E.; Nounahon, J.M.; Passerat-Palmbach, J.; Prakash, K.; Rose, N.; et al. Pysyft: A library for easy federated learning. In *Federated Learning Systems: Towards Next-Generation AI*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 111–139.
141. Reddi, V.J.; Cheng, C.; Kanter, D.; Mattson, P.; Schmuelling, G.; Wu, C.J.; Anderson, B.; Breughe, M.; Charlebois, M.; Chou, W.; et al. Mlperf inference benchmark. In Proceedings of the 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA), 30 May 2020–3 June 2020; pp. 446–459.
142. Varghese, B.; Wang, N.; Bermbach, D.; Hong, C.H.; Lara, E.D.; Shi, W.; Stewart, C. A survey on edge performance benchmarking. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 66. [CrossRef]
143. Mattson, P.; Cheng, C.; Diamos, G.; Coleman, C.; Micikevicius, P.; Patterson, D.; Tang, H.; Wei, G.Y.; Bailis, P.; Bittorf, V.; et al. Mlperf training benchmark. *Proc. Mach. Learn. Syst.* **2020**, *2*, 336–349.
144. Saeed, E.; Coutinho, R.W. Performance evaluation of edge computing models for internet of things. In Proceedings of the 12th ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, Montreal, QC, Canada, 24–28 October 2022; pp. 63–69.
145. Tu, X.; Mallik, A.; Chen, D.; Han, K.; Altintas, O.; Wang, H.; Xie, J. Unveiling energy efficiency in deep learning: Measurement, prediction, and scoring across edge devices. In Proceedings of the Eighth ACM/IEEE Symposium on Edge Computing, Wilmington, DE, USA, 6–9 December 2023; pp. 80–93.
146. Fan, T.; Qiu, Y.; Jiang, C.; Wan, J. Energy aware edge computing: A survey. In Proceedings of the International Workshop on High Performance Computing for Advanced Modeling and Simulation in Nuclear Energy and Environmental Science, Beijing, China, 12 June 2018; pp. 79–91.
147. Aslanpour, M.S.; Gill, S.S.; Toosi, A.N. Performance evaluation metrics for cloud, fog and edge computing: A review, taxonomy, benchmarks and standards for future research. *Internet Things* **2020**, *12*, 100273. [CrossRef]
148. Caiazza, C.; Luconi, V.; Vecchio, A. Saving energy on smartphones through edge computing: An experimental evaluation. In Proceedings of the ACM SIGCOMM Workshop on Networked Sensing Systems for a Sustainable Society, Antipolis, France, 22 August 2022; pp. 20–25.
149. Reddi, V.J.; Cheng, C.; Kanter, D.; Mattson, P.; Schmuelling, G.; Wu, C.J. The vision behind mlperf: Understanding ai inference performance. *IEEE Micro* **2021**, *41*, 10–18. [CrossRef]
150. Vo, T.; Dave, P.; Bajpai, G.; Kashef, R. Edge, Fog, and Cloud Computing : An Overview on Challenges and Applications. *arXiv* **2022**, arXiv:2211.01863. [CrossRef]
151. Singh, P.P.; Khosla, P.K.; Mittal, M. Energy conservation in IoT-based smart home and its automation. *Stud. Syst. Decis. Control* **2019**, *206*, 155–177. [CrossRef]
152. Dong, P.; Ge, J.; Wang, X.; Guo, S. Collaborative Edge Computing for Social Internet of Things: Applications, Solutions, and Challenges. *IEEE Trans. Comput. Soc. Syst.* **2022**, *9*, 291–301. [CrossRef]
153. Alharbi, H.A.; Aldossary, M. Energy-Efficient Edge-Fog-Cloud Architecture for IoT-Based Smart Agriculture Environment. *IEEE Access* **2021**, *9*, 110480–110492. [CrossRef]
154. Rathi, V.K.; Rajput, N.K.; Mishra, S.; Grover, B.A.; Tiwari, P.; Jaiswal, A.K.; Hossain, M.S. An edge AI-enabled IoT healthcare monitoring system for smart cities. *Comput. Electr. Eng.* **2021**, *96*, 107524. [CrossRef]
155. Kamruzzaman, M.M. New Opportunities, Challenges, and Applications of Edge-AI for Connected Healthcare in Smart Cities. In Proceedings of the 2021 IEEE Globecom Workshops, GC Wkshps 2021, Madrid, Spain, 7–11 December 2021. [CrossRef]
156. Ullah, Z.; Al-Turjman, F.; Mostarda, L.; Gagliardi, R. Applications of Artificial Intelligence and Machine learning in smart cities. *Comput. Commun.* **2020**, *154*, 313–323. [CrossRef]
157. Fagnant, D.J.; Kockelman, K. Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Transp. Res. Part A Policy Pract.* **2015**, *77*, 167–181. [CrossRef]
158. Zonta, T.; da Costa, C.A.; da Rosa Righi, R.; de Lima, M.J.; da Trindade, E.S.; Li, G.P. Predictive maintenance in the Industry 4.0: A systematic literature review. *Comput. Ind. Eng.* **2020**, *150*, 106889. [CrossRef]
159. Sodhro, A.H.; Gurtov, A.; Zahid, N.; Pirbhulal, S.; Wang, L.; Rahman, M.M.U.; Imran, M.A.; Abbasi, Q.H. Toward Convergence of AI and IoT for Energy-Efficient Communication in Smart Homes. *IEEE Internet Things J.* **2021**, *8*, 9664–9671. [CrossRef]
160. Saad Al-Sumaiti, A.; Ahmed, M.H.; Salama, M.M. Smart Home Activities: A Literature Review. *Electr. Power Compon. Syst.* **2014**, *42*, 294–305. [CrossRef]
161. Zhang, S.; Zhang, H. A review of wireless sensor networks and its applications. In Proceedings of the IEEE International Conference on Automation and Logistics, ICAL, Zhengzhou, China, 15–17 August 2012; pp. 386–389. [CrossRef]

162. Ray, P.P. Internet of things for smart agriculture: Technologies, practices and future direction. *J. Ambient Intell. Smart Environ.* **2017**, *9*, 395–420. [CrossRef]
163. Chen, C.J.; Huang, Y.Y.; Li, Y.S.; Chen, Y.C.; Chang, C.Y.; Huang, Y.M. Identification of Fruit Tree Pests with Deep Learning on Embedded Drone to Achieve Accurate Pesticide Spraying. *IEEE Access* **2021**, *9*, 21986–21997. [CrossRef]
164. Liu, Y.; Wang, Y.S.; Xu, S.P.; Hu, W.W.; Wu, Y.J. Design and Implementation of Online Monitoring System for Soil Salinity and Alkalinity in Yangtze River Delta Tideland. In Proceedings of the 2021 IEEE International Conference on Artificial Intelligence and Industrial Design, Guangzhou, China, 28–30 May 2021; pp. 521–526. [CrossRef]
165. Sakthi, U.; Rose, J.D. Smart agricultural knowledge discovery system using IoT technology and fog computing. In Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT, Tirunelveli, India, 20–22 August 2020; pp. 48–53. [CrossRef]
166. Lynggaard, P.; Skouby, K.E. Deploying 5G-Technologies in Smart City and Smart Home Wireless Sensor Networks with Interferences. *Wirel. Pers. Commun.* **2015**, *81*, 1399–1413. [CrossRef]
167. Ullah, I.; Baharom, M.N.; Ahmad, H.; Wahid, F.; Luqman, H.M.; Zainal, Z.; Das, B. Smart Lightning Detection System for Smart-City Infrastructure Using Artificial Neural Network. *Wirel. Pers. Commun.* **2019**, *106*, 1743–1766. [CrossRef]
168. Neirotti, P.; De Marco, A.; Cagliano, A.C.; Mangano, G.; Scorrano, F. Current trends in Smart City initiatives: Some stylised facts. *Cities* **2014**, *38*, 25–36. [CrossRef]
169. Morello, R.; Mukhopadhyay, S.C.; Liu, Z.; Slomovitz, D.; Samantaray, S.R. Advances on sensing technologies for smart cities and power grids: A review. *IEEE Sens. J.* **2017**, *17*, 7596–7610. [CrossRef]
170. Mainetti, L.; Patrono, L.; Stefanizzi, M.L.; Vergallo, R. A Smart Parking System based on IoT protocols and emerging enabling technologies. In Proceedings of the IEEE World Forum on Internet of Things, WF-IoT, Milan, Italy, 14–16 December 2015; pp. 764–769. [CrossRef]
171. Bhardwaj, A.; Goundar, S. IoT enabled Smart Fog Computing for Vehicular Traffic Control. *EAI Endorsed Trans. Internet Things* **2019**, *5*, e3. [CrossRef]
172. Sapienza, M.; Guardo, E.; Cavallo, M.; La Torre, G.; Leombruno, G.; Tomarchio, O. Solving Critical Events through Mobile Edge Computing: An Approach for Smart Cities. In Proceedings of the 2016 IEEE International Conference on Smart Computing, SMARTCOMP, St. Louis, MO, USA, 18–20 May 2016. [CrossRef]
173. Sigwele, T.; Hu, Y.F.; Ali, M.; Hou, J.; Susanto, M.; Fitriawan, H. Intelligent and Energy Efficient Mobile Smartphone Gateway for Healthcare Smart Devices Based on 5G. In Proceedings of the 2018 IEEE Global Communications Conference, GLOBECOM, Abu Dhabi, United Arab Emirates, 9–13 December 2018. [CrossRef]
174. Varghese, B.; Wang, N.; Barbhuiya, S.; Kilpatrick, P.; Nikolopoulos, D.S. Challenges and Opportunities in Edge Computing. In Proceedings of the 2016 IEEE International Conference on Smart Cloud, SmartCloud, New York, NY, USA, 18–20 November 2016; pp. 20–26.
175. Haleem, A.; Javaid, M.; Singh, R.P.; Suman, R. Telemedicine for healthcare: Capabilities, features, barriers, and applications. *Sens. Int.* **2021**, *2*, 100117. [CrossRef]
176. Klonoff, D.C.; Aimbe, F. Fog Computing and Edge Computing Architectures for Processing Data From Diabetes Devices Connected to the Medical Internet of Things. *J. Diabetes Sci. Technol.* **2017**, *11*, 647–652. [CrossRef]
177. Hartmann, M.; Hashmi, U.S.; Imran, A. Edge computing in smart health care systems: Review, challenges, and research directions. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e3710. [CrossRef]
178. Kaur, A.; Jasuja, A. Health monitoring based on IoT using Raspberry PI. In Proceedings of the IEEE International Conference on Computing, Communication and Automation, ICCCA, Greater Noida, India, 5–6 May 2017; pp. 1335–1340. [CrossRef]
179. Zheng, H.; Paiva, A.R.; Gurciullo, C.S. Advancing from Predictive Maintenance to Intelligent Maintenance with AI and IIoT. *arXiv* **2020**, arXiv:2009.00351. [CrossRef]
180. Hafeez, T.; Xu, L.; McArdle, G. Edge intelligence for data handling and predictive maintenance in IIoT. *IEEE Access* **2021**, *9*, 49355–49371. [CrossRef]
181. Li, L.; Ota, K.; Dong, M. Deep Learning for Smart Industry: Efficient Manufacture Inspection System with Fog Computing. *IEEE Trans. Ind. Informat.* **2018**, *14*, 4665–4673. [CrossRef]
182. Cui, M.; Zhong, S.; Li, B.; Chen, X.; Huang, K. Offloading Autonomous Driving Services via Edge Computing. *IEEE Internet Things J.* **2020**, *7*, 10535–10547. [CrossRef]
183. Agarwal, P.; Mittal, M.; Ahmed, J.; Idrees, S.M. (Eds.) *Smart Technologies for Energy and Environmental Sustainability*; Springer: Berlin/Heidelberg, Germany, 2022. [CrossRef]
184. Su, X.; Liu, X.; Motlagh, N.H.; Cao, J.; Su, P.; Pellikka, P.; Liu, Y.; Petaja, T.; Kulmala, M.; Hui, P.; et al. Intelligent and Scalable Air Quality Monitoring with 5G Edge. *IEEE Internet Comput.* **2021**, *25*, 35–44. [CrossRef]
185. Moursi, A.S.; El-Fishawy, N.; Djahel, S.; Shouman, M.A. An IoT enabled system for enhanced air quality monitoring and prediction on the edge. *Complex Intell. Syst.* **2021**, *7*, 2923–2947. [CrossRef] [PubMed]

186. Bhardwaj, A.; Dagar, V.; Khan, M.O.; Aggarwal, A.; Alvarado, R.; Kumar, M.; Irfan, M.; Proshad, R. Smart IoT and Machine Learning-based Framework for Water Quality Assessment and Device Component Monitoring. *Environ. Sci. Pollut. Res.* **2022**, *29*, 46018–46036. [CrossRef]
187. Mei, G.; Xu, N.; Qin, J.; Wang, B.; Qi, P. A Survey of Internet of Things (IoT) for Geohazard Prevention: Applications, Technologies, and Challenges. *IEEE Internet Things J.* **2020**, *7*, 4371–4386. [CrossRef]
188. Ananthi, J.; Sengottaiyan, N.; Anbukaruppusamy, S.; Upreti, K.; Dubey, A.K. Forest fire prediction using IoT and deep learning. *Int. J. Adv. Technol. Eng. Explor.* **2022**, *9*, 246–256. [CrossRef]
189. Huang, X.R.; Chen, W.H.; Pai, W.Y.; Huang, G.Z.; Hu, W.C.; Chen, L.B. An AI Edge Computing-Based Robotic Automatic Guided Vehicle System for Cleaning Garbage. In Proceedings of the 2022 IEEE 4th Global Conference on Life Sciences and Technologies, Osaka, Japan, 7–9 March 2022; pp. 446–447. [CrossRef]
190. Morabito, R. Virtualization on internet of things edge devices with container technologies: A performance evaluation. *IEEE Access* **2017**, *5*, 8835–8850. [CrossRef]
191. Levis, P.; Madden, S.; Polastre, J.; Szewczyk, R.; Whitehouse, K.; Woo, A.; Gay, D.; Hill, J.; Welsh, M.; Brewer, E.; et al. TinyOS: An operating system for sensor networks. In *Ambient Intelligence*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 115–148. [CrossRef]
192. Manzalini, A.; Crespi, N. An edge operating system enabling anything-as-a-service. *IEEE Commun. Mag.* **2016**, *54*, 62–67. [CrossRef]
193. Li, B.; Fan, H.; Gao, Y.; Dong, W. ThingSpire OS: A WebAssembly-based IoT operating system for cloud-edge integration. In Proceedings of the MobiSys 2021 19th Annual International Conference on Mobile Systems, Applications, and Services, Virtual, 24 June–2 July 2021; pp. 487–488. [CrossRef]
194. Silva, M.; Cerdeira, D.; Pinto, S.; Gomes, T. Operating Systems for Internet of Things Low-End Devices: Analysis and Benchmarking. *IEEE Internet Things J.* **2019**, *6*, 10375–10383. [CrossRef]
195. Baccelli, E.; Gundogan, C.; Hahm, O.; Kietzmann, P.; Lenders, M.S.; Petersen, H.; Schleiser, K.; Schmidt, T.C.; Wahlisch, M. RIOT: An Open Source Operating System for Low-End Embedded Devices in the IoT. *IEEE Internet Things J.* **2018**, *5*, 4428–4440. [CrossRef]
196. Borgohain, T.; Kumar, U.; Sanyal, S. Survey of Operating Systems for the IoT Environment. *arXiv* **2015**, arXiv:1504.02517. [CrossRef]
197. Nakano, W.; Shinohara, Y.; Ishiura, N. Full Hardware Implementation of FreeRTOS-Based Real-Time Systems. In Proceedings of the IEEE Region 10 Annual International Conference, Auckland, New Zealand, 7–10 December 2021; pp. 435–440. [CrossRef]
198. Borges, M.; Paiva, S.; Santos, A.; Gaspar, B.; Cabral, J. Azure RTOS ThreadX Design for Low-End NB-IoT Device. In Proceedings of the 2020 IEEE 2nd International Conference on Societal Automation (SA) 2021, Madeira, Portugal, 9–11 September 2020; pp. 1–8. [CrossRef]
199. Sarwar Murshed, M.G.; Murphy, C.; Hou, D.; Khan, N.; Ananthanarayanan, G.; Hussain, F. Machine Learning at the Network Edge: A Survey. *ACM Comput. Surv.* **2022**, *170*, 54. [CrossRef]
200. Wang, S.; Tuor, T.; Salonidis, T.; Leung, K.K.; Makaya, C.; He, T.; Chan, K. When Edge Meets Learning: Adaptive Control for Resource-Constrained Distributed Machine Learning. In Proceedings of the IEEE INFOCOM, Honolulu, HI, USA, 15–19 April 2018; pp. 63–71. [CrossRef]
201. Gupta, C.; Suggala, A.S.; Goyal, A.; Simhadri, H.V.; Paranjape, B.; Kumar, A.; Goyal, S.; Udupa, R.; Varma, M.; Jain, P. ProtoNN: Compressed and Accurate kNN for Resource-scarce Devices. *Int. Conf. Mach. Learn.* **2017**, *70*, 1331–1340.
202. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical secure aggregation for privacy-preserving machine learning. In Proceedings of the ACM Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 1175–1191. [CrossRef]
203. Mankins, J.C. *Technology Readiness Levels*; NASA: Washington, DC, USA, 1995.
204. Benotmane, M.; Elhari, K.; Kabbaj, A. A review & analysis of current IoT maturity & readiness models and novel proposal. *Sci. Afr.* **2023**, *21*, e01748. [CrossRef]
205. Kosta, S.; Aucinas, A.; Hui, P.; Mortier, R.; Zhang, X. ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading. In Proceedings of the IEEE INFOCOM, Orlando, FL, USA, 25–30 March 2012; pp. 945–953. [CrossRef]
206. Chun, B.G.; Ihm, S.; Maniatis, P.; Naik, M.; Patti, A. CloneCloud. In Proceedings of the Sixth Conference on Computer Systems, New York, NY, USA, 10 April 2011 pp. 301–314. [CrossRef]
207. Li, H.; Ota, K.; Dong, M. Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing. *IEEE Netw.* **2018**, *32*, 96–101. [CrossRef]
208. Chen, X.; Zhang, H.; Wu, C.; Mao, S.; Ji, Y.; Bennis, M. Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning. *IEEE Internet Things J.* **2019**, *6*, 4005–4018. [CrossRef]
209. Lei, L.; Xu, H.; Xiong, X.; Zheng, K.; Xiang, W.; Wang, X. Multiuser Resource Control With Deep Reinforcement Learning in IoT Edge Computing. *IEEE Internet Things J.* **2019**, *6*, 10119–10133. [CrossRef]

210. Xu, X.; Li, H.; Xu, W.; Liu, Z.; Yao, L.; Dai, F. *Artificial Intelligence for Edge Service Optimization in Internet of Vehicles: A Survey*; Technical Report; Tsinghua University Press (TUP): Beijing, China, 2022. [CrossRef]
211. Chiu, T.C.; Shih, Y.Y.; Pang, A.C.; Wang, C.S.; Weng, W.; Chou, C.T. Semisupervised Distributed Learning with Non-IID Data for AIoT Service Platform. *IEEE Internet Things J.* **2020**, *7*, 9266–9277. [CrossRef]
212. Zhou, X.; Bilal, M.; Dou, R.; Rodrigues, J.J.P.C.; Zhao, Q.; Dai, J.; Xu, X. Edge Computation Offloading With Content Caching in 6G-Enabled IoV. *IEEE Trans. Intell. Transport. Syst.* **2023**, *25*, 2733–2747. [CrossRef]
213. Zhang, K.; Leng, S.; He, Y.; Maharjan, S.; Zhang, Y. Cooperative Content Caching in 5G Networks with Mobile Edge Computing. *IEEE Wirel. Commun.* **2018**, *25*, 80–87. [CrossRef]
214. Duan, S.; Zhang, D.; Wang, Y.; Li, L.; Zhang, Y. JointRec: A Deep-Learning-Based Joint Cloud Video Recommendation Framework for Mobile IoT. *IEEE Internet Things J.* **2020**, *7*, 1655–1666. [CrossRef]
215. Deng, Y.; Han, T.; Ansari, N. FedVision: Federated Video Analytics With Edge Computing . *IEEE Open J. Comput. Soc.* **2020**, *1*, 62–72. [CrossRef]
216. Zyrianoff, I.; Gigli, L.; Montori, F.; Sciullo, L.; Kamienski, C.; Di Felice, M. Cache-it: A distributed architecture for proactive edge caching in heterogeneous iot scenarios. *Ad Hoc Netw.* **2024**, *156*, 103413. [CrossRef]
217. Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Xu, K.; Wang, W.; Jiang, X.; Gao, B.; Lu, J. FedCache: A Knowledge Cache-Driven Federated Learning Architecture for Personalized Edge Intelligence. *IEEE Trans. Mobile Comput.* **2024**, *23*, 9368–9382. [CrossRef]
218. Davidson, A.; Goldberg, I.; Sullivan, N.; Tankersley, G.; Valsorda, F. Privacy pass: Bypassing internet challenges anonymously. *Proc. Priv. Enhancing Technol.* **2018**, *3*, 164–180. [CrossRef]
219. Gaber, M.M.; Zaslavsky, A.; Krishnaswamy, S. Mining data streams. *ACM SIGMOD Rec.* **2005**, *34*, 18–26. [CrossRef]
220. Cheng, B.; Papageorgiou, A.; Cirillo, F.; Kovacs, E. GeeLytics: Geo-distributed edge analytics for large scale IoT systems based on dynamic topology. In Proceedings of the IEEE World Forum on Internet of Things, WF-IoT 2015, Milan, Italy, 14–16 December 2015; pp. 565–570. [CrossRef]
221. Ananthanarayanan, G.; Bahl, P.; Bodik, P.; Chintalapudi, K.; Philipose, M.; Ravindranath, L.; Sinha, S. Real-Time Video Analytics: The Killer App for Edge Computing. *Computer* **2017**, *50*, 58–67. [CrossRef]
222. Dias de Assunção, M.; da Silva Veith, A.; Buyya, R. Distributed data stream processing and edge computing: A survey on resource elasticity and future directions. *J. Netw. Comput. Appl.* **2018**, *103*, 1–17. [CrossRef]
223. Garg, N. *Apache kafka*; Packt Publishing: Birmingham, UK, 2013.
224. Song, F.; Tomov, S.; Dongarra, J. Enabling and scaling matrix computations on heterogeneous multi-core and multi-GPU systems. In Proceedings of the International Conference on Supercomputing, Salt Lake City, UT, USA, 15 January 2013; pp. 365–375. [CrossRef]
225. Qureshi, K.N.; Iftikhar, A.; Bhatti, S.N.; Piccialli, F.; Giampaolo, F.; Jeon, G. Trust management and evaluation for edge intelligence in the Internet of Things. *Eng. Appl. Artif. Intell.* **2020**, *94*, 103756. [CrossRef]
226. Al-Rakhami, M.; Alsahli, M.; Hassan, M.M.; Alamri, A.; Guerrieri, A.; Fortino, G. Cost Efficient Edge Intelligence Framework Using Docker Containers. In Proceedings of the 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech), Athens, Greece, 12–15 August 2018; pp. 800–807. [CrossRef]
227. Mendez, J.; Bierzynski, K.; Cuéllar, M.P.; Morales, D.P. Edge Intelligence: Concepts, Architectures, Applications, and Future Directions. *ACM Trans. Embed. Comput. Syst. (TECS)* **2022**, *21*, 48. [CrossRef]
228. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the MCC'12 1st ACM Mobile Cloud Computing Workshop, Helsinki, Finland, 17 August 2012; pp. 13–15. [CrossRef]
229. Yang, J.; Yuan, Q.; Chen, S.; He, H.; Jiang, X.; Tan, X. Cooperative Task Offloading for Mobile Edge Computing Based on Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 3205–3219. [CrossRef]
230. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Honolulu, HI, USA, 21–26 July 2017; pp. 6848–6856. [CrossRef]
231. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. *arXiv* **2017**, arXiv:1807.11164.
232. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861. [CrossRef]
233. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
234. Rahmani, A.M.; Haider, A.; Moghaddasi, K.; Gharehchopogh, F.S.; Aurangzeb, K.; Liu, Z.; Hosseinzadeh, M. Self-learning adaptive power management scheme for energy-efficient IoT-MEC systems using soft actor-critic algorithm. *Internet Things* **2025**, *31*, 101587. [CrossRef]

235. Zhang, X.; Hou, D.; Xiong, Z.; Liu, Y.; Wang, S.; Li, Y. EALLR: Energy-aware low-latency routing data driven model in mobile edge computing. *IEEE Trans. Consum. Electron.* **2024**, *71*, 6612–6626. [CrossRef]
236. Moghaddasi, K.; Rajabi, S.; Gharehchopogh, F.S.; Ghaffari, A. An advanced deep reinforcement learning algorithm for three-layer D2D-edge-cloud computing architecture for efficient task offloading in the Internet of Things. *Sustain. Comput. Informatics Syst.* **2024**, *43*, 100992. [CrossRef]
237. Martin, C.; Garrido, D.; Diaz, M.; Rubio, B. From the edge to the cloud: Enabling reliable IoT applications. In Proceedings of the 2019 International Conference on Future Internet of Things and Cloud, FiCloud 2019, Istanbul, Turkey, 26–28 August 2019; pp. 17–22. [CrossRef]
238. Grover, J.; Garimella, R.M. Reliable and Fault-Tolerant IoT-Edge Architecture. In Proceedings of the IEEE SENSORS, New Delhi, India, 28–31 October 2018; pp. 1–4. [CrossRef]
239. Mohammadi, V.; Rahmani, A.M.; Darwesh, A.; Sahafi, A. Fault tolerance in fog-based Social Internet of Things. *Knowl. Based Syst.* **2023**, *265*, 110376. [CrossRef]
240. Tan, T.; Cao, G. FastVA: Deep learning video analytics through edge processing and NPU in mobile. In Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications, Virtual, 6–9 July 2020; pp. 1947–1956.
241. Indirli, F.; Ornstein, A.C.; Desoli, G.; Buschini, A.; Silvano, C.; Zaccaria, V. Layer-wise Exploration of a Neural Processing Unit Compiler's Optimization Space. In Proceedings of the 2024 10th International Conference on Computer Technology Applications, Vienna, Austria, 15–17 May 2024; pp. 20–26.
242. Rico, A.; Pareek, S.; Cabezas, J.; Clarke, D.; Ozgul, B.; Barat, F.; Fu, Y.; Münz, S.; Stuart, D.; Schlangen, P.; et al. Amd xdna™ npu in ryzen™ ai processors. *IEEE Micro* **2024**, *44*, 73–82. [CrossRef]
243. Lee, E.; Sung, M.; Jang, S.J.; Park, J.; Lee, S.S. Memory-centric architecture of neural processing unit for edge device. In Proceedings of the IEEE 2021 18th International SoC Design Conference (ISOCC), Jeju Island, Republic of Korea, 6–9 October 2021; pp. 240–241.
244. Shahid, A.; Mushtaq, M. A Survey Comparing Specialized Hardware and Evolution in TPUs for Neural Networks. In Proceedings of the 2020 23rd IEEE International Multi-Topic Conference, INMIC 2020, Bahawalpur, Pakistan, 5–7 November 2020. [CrossRef]
245. Jo, J.; Jeong, S.; Kang, P. Benchmarking GPU-accelerated edge devices. In Proceedings of the 2020 IEEE International Conference on Big Data and Smart Computing, BigComp 2020, Busan, Republic of Korea, 19–22 February 2020; pp. 117–120. [CrossRef]
246. Seshadri, K.; Akin, B.; Laudon, J.; Narayanaswami, R.; Yazdanbakhsh, A. An Evaluation of Edge TPU Accelerators for Convolutional Neural Networks. In Proceedings of the 2022 IEEE International Symposium on Workload Characterization, IISWC 2022, Austin, TX, USA, 6–8 November 2022; pp. 79–91. [CrossRef]
247. Alwahedi, F.; Aldhaheri, A.; Ferrag, M.A.; Battah, A.; Tihanyi, N. Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models. *Internet Things Cyber-Phys. Syst.* **2024**, *4*, 167–185. [CrossRef]
248. Cherbal, S.; Zier, A.; Hebal, S.; Louail, L.; Annane, B. Security in internet of things: A review on approaches based on blockchain, machine learning, cryptography, and quantum computing. *J. Supercomput.* **2024**, *80*, 3738–3816. [CrossRef]
249. Kang, P.; Somtham, A. An Evaluation of Modern Accelerator-Based Edge Devices for Object Detection Applications. *Mathematics* **2022**, *10*, 4299. [CrossRef]
250. Sufian, A.; You, C.; Dong, M. A deep transfer learning-based edge computing method for home health monitoring. In Proceedings of the IEEE 2021 55th Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA, 24–26 March 2021; pp. 1–6.
251. Yang, J.; Zou, H.; Cao, S.; Chen, Z.; Xie, L. MobileDA: Toward edge-domain adaptation. *IEEE Internet Things J.* **2020**, *7*, 6909–6918. [CrossRef]
252. Qian, J.; Gochhayat, S.P.; Hansen, L.K. Distributed active learning strategies on edge computing. In Proceedings of the 2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud), Paris, France, 21–23 June 2019; pp. 221–226.
253. Zhu, G.; Liu, D.; Du, Y.; You, C.; Zhang, J.; Huang, K. Toward an Intelligent Edge: Wireless Communication Meets Machine Learning. *IEEE Commun. Mag.* **2020**, *58*, 19–25. [CrossRef]
254. Wang, W.; Lv, Q.; Yu, W.; Hong, W.; Qi, J.; Wang, Y.; Ji, J.; Yang, Z.; Zhao, L.; Song, X.; et al. Cogvlm: Visual expert for pretrained language models. *arXiv* **2023**, arXiv:2311.03079.
255. Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X.V.; et al. Opt: Open pre-trained transformer language models. *arXiv* **2022**, arXiv:2205.01068. [CrossRef]
256. Chung, H.W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, Y.; Wang, X.; Dehghani, M.; Brahma, S.; et al. Scaling instruction-finetuned language models. *arXiv* **2022**, arXiv:2210.11416. [CrossRef]
257. Liu, H.; Li, C.; Wu, Q.; Lee, Y.J. Visual instruction tuning. *Adv. Neural Inf. Process. Syst.* **2023**, *36, 34892-34916.* .
258. Li, J.; Li, D.; Savarese, S.; Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv* **2023**, arXiv:2301.12597.
259. Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. The llama 3 herd of models. *arXiv* **2024**, arXiv:2407.21783. [CrossRef]

260. Lin, B.; Tang, Z.; Ye, Y.; Cui, J.; Zhu, B.; Jin, P.; Zhang, J.; Ning, M.; Yuan, L. Moe-llava: Mixture of experts for large vision-language models. *arXiv* **2024**, arXiv:2401.15947.
261. Li, S.; Tang, H. Multimodal Alignment and Fusion: A Survey. *arXiv* **2024**, arXiv:2411.17040. [CrossRef]
262. Li, J.; Selvaraju, R.; Gotmare, A.; Joty, S.; Xiong, C.; Hoi, S.C.H. Align before fuse: Vision and language representation learning with momentum distillation. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9694–9705.
263. Qin, J.; Xu, Y.; Lu, Z.; Zhang, X. Alternative Telescopic Displacement: An Efficient Multimodal Alignment Method. *arXiv* **2023**, arXiv:2306.16950. [CrossRef]
264. Cajas, S.A.; Restrepo, D.; Moukheiber, D.; Kuo, K.T.; Wu, C.; Chicangana, D.S.G.; Paddo, A.R.; Moukheiber, M.; Moukheiber, L.; Moukheiber, S.; et al. *A Multi-Modal Satellite Imagery Dataset for Public Health Analysis in Colombia*; PhysioNet: Online, 2024 [CrossRef]
265. Lahat, D.; Adali, T.; Jutten, C. Multimodal data fusion: An overview of methods, challenges, and prospects. *Proc. IEEE* **2015**, *103*, 1449–1477. [CrossRef]
266. Castanedo, F. A review of data fusion techniques. *Sci. World J.* **2013**, *2013*, 704504. [CrossRef]
267. Snoek, C.G.; Worring, M.; Smeulders, A.W. Early versus late fusion in semantic video analysis. In Proceedings of the 13th annual ACM international conference on Multimedia, Singapore, 6–11 November 2005; pp. 399–402.
268. Pereira, L.M.; Salazar, A.; Vergara, L. A comparative analysis of early and late fusion for the multimodal two-class problem. *IEEE Access* **2023**, *11*, 84283–84300. [CrossRef]
269. Akbari, H.; Yuan, L.; Qian, R.; Chuang, W.H.; Chang, S.F.; Cui, Y.; Gong, B. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 24206–24221.
270. Zhou, H.Y.; Yu, Y.; Wang, C.; Zhang, S.; Gao, Y.; Pan, J.; Shao, J.; Lu, G.; Zhang, K.; Li, W. A transformer-based representation-learning model with unified processing of multimodal input for clinical diagnostics. *Nat. Biomed. Eng.* **2023**, *7*, 743–755. [CrossRef]
271. Khader, F.; Kather, J.N.; Müller-Franzes, G.; Wang, T.; Han, T.; Tayebi Arasteh, S.; Hamesch, K.; Bressem, K.; Haarburger, C.; Stegmaier, J.; et al. Medical transformer for multimodal survival prediction in intensive care: Integration of imaging and non-imaging data. *Sci. Rep.* **2023**, *13*, 10666. [CrossRef] [PubMed]
272. Nguyen, H.H.; Blaschko, M.B.; Saarakkala, S.; Tiulpin, A. Clinically-inspired multi-agent transformers for disease trajectory forecasting from multimodal data. *IEEE Trans. Med Imaging* **2023**, *43*, 529–541. [CrossRef] [PubMed]
273. Garcia, J.; Masip-Bruin, X.; Giannopoulos, A.; Trakadas, P.; Cajas Ordoñez, S.A.; Samanta, J.; Suárez-Cetrulo, A.L.; Simón Carbajo, R.; Michalke, M.; Admela, J.; et al. ICOS An Intelligent MetaOS for the Continuum. In Proceedings of the MECC '25 2nd International Workshop on MetaOS for the Cloud-Edge-IoT Continuum, New York, NY, USA, 30 March–3 April 2025; pp. 53–59. [CrossRef]
274. Ren, J.; Yu, G.; Cai, Y.; He, Y. Latency optimization for resource allocation in mobile-edge computation offloading. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 5506–5519. [CrossRef]
275. Zhu, S.; Ota, K.; Dong, M. Energy-Efficient Artificial Intelligence of Things With Intelligent Edge. *IEEE Internet Things J.* **2022**, *9*, 7525–7532. [CrossRef]
276. Mao, W.; Zhao, Z.; Chang, Z.; Min, G.; Gao, W. Energy-Efficient Industrial Internet of Things: Overview and Open Issues. *IEEE Trans. Ind. Informat.* **2021**, *17*, 7225–7237. [CrossRef]
277. Jobin, A.; Ienca, M.; Vayena, E. The global landscape of AI ethics guidelines. *Nat. Mach. Intell.* **2019**, *1*, 389–399. [CrossRef]
278. Kumar, A.; Zhuang, V.; Agarwal, R.; Su, Y.; Co-Reyes, J.D.; Singh, A.; Baumli, K.; Iqbal, S.; Bishop, C.; Roelofs, R.; et al. Training language models to self-correct via reinforcement learning. *arXiv* **2024**, arXiv:2409.12917.
279. Hagendorff, T. The Ethics of AI Ethics: An Evaluation of Guidelines. *Minds Mach.* **2020**, *30*, 99–120. [CrossRef]
280. Tsouparopoulos, T.; Koutsopoulos, I. Explainability and Continual Learning meet Federated Learning at the Network Edge. *arXiv* **2025**, arXiv:2504.08536. [CrossRef]
281. Chen, T. All versus one: An empirical comparison on retrained and incremental machine learning for modeling performance of adaptable software. In Proceedings of the ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems, Montreal, QC, Canada, 25 May 2019; pp. 157–168. [CrossRef]
282. Aspis, M.; Ordónez, S.A.; Suárez-Cetrulo, A.L.; Carbajo, R.S. DriftMoE: A Mixture of Experts Approach to Handle Concept Drifts. *arXiv* **2025**, arXiv:2507.18464.