On-Device and Cloud-Based Learning for Next-Generation AI Applications

Jenifer Nadine *, Liam Chen , Sara Ahmed , Michael Tran , Aisha Okafor

Posted Date: 5 August 2025

doi: 10.20944/preprints202508.0352.v1

Keywords: collaborative learning; edge computing

 doi:10.20944/preprints202508.0352.v1

Article

On-Device and Cloud-Based Learning for Next-Generation AI Applications

Jenifer Nadine [1,*], Liam Chen [2], Sara Ahmed [3], Michael Tran [2] and Aisha Okafor [4]

[1] University of Warwick, UK
[2] Massachusetts Institute of Technology, Cambridge, MA, USA
[3] ETH Zurich, Zurich, Switzerland
[4] University of Toronto, Toronto, ON, Canada
* Correspondence: jenifer.nadine@warwick.ac.uk

Abstract

The widespread deployment of deep learning in real-world applications has prompted a paradigm shift toward collaborative learning between resource-constrained edge devices and powerful cloud-based infrastructures. Traditional deep learning architectures, typically optimized for centralized cloud environments, often fall short in scenarios where latency, privacy, energy efficiency, and real-time responsiveness are critical. Conversely, purely on-device models are limited in capacity and accuracy due to stringent computational and memory constraints. To address these challenges, a hybrid approach has emerged, where lightweight on-device models collaborate with large, high-capacity models hosted in the cloud, enabling the seamless integration of low-latency inference and high-accuracy computation. This survey provides a comprehensive examination of collaborative learning frameworks that bridge the gap between on-device and cloud-based models, including federated learning, split learning, model offloading, and knowledge distillation. We analyze the theoretical foundations of each approach, explore their mathematical formulations, and discuss their practical trade-offs in terms of communication overhead, privacy guarantees, learning efficiency, and robustness to heterogeneous environments. The survey further explores the optimization of collaborative inference workflows, where inference is partitioned between devices and the cloud to minimize latency and energy consumption while maximizing model accuracy. Techniques such as dynamic model partitioning, early exit strategies, quantization, and sparsification are discussed in detail, along with system-level co-design considerations that align learning objectives with hardware and network capabilities. We highlight key applications across diverse domains, including healthcare, autonomous systems, smart cities, and personalized AI, demonstrating how collaborative learning enables responsive, context-aware, and privacy-preserving AI services. Through in-depth case studies, we illustrate how these systems are implemented in practice, shedding light on architectural decisions, model deployment strategies, and real-time performance outcomes. Moreover, the abstract delves into emerging trends that are shaping the future of collaborative learning, such as privacy-enhancing technologies, edge AI hardware acceleration, continual learning, and adaptive collaboration mechanisms. The intersection of these advances is poised to redefine how AI is deployed at scale, enabling intelligent systems that are not only accurate and efficient but also secure, autonomous, and adaptable to dynamic environments. We conclude by identifying key open challenges, including the need for standardized benchmarks, scalable learning protocols, and trust frameworks that ensure responsible AI deployment in collaborative settings. This survey aims to serve as both a foundational reference for researchers entering the field and a strategic guide for practitioners designing next-generation AI systems that leverage the full potential of collaborative learning across the edge-cloud continuum.

Keywords: collaborative learning; edge computing; cloud computing; on-device AI; Federated learning; split learning; knowledge distillation; edge-cloud inference; privacy-preserving AI; distributed deep learning; adaptive inference; resource-constrained devices; deep neural networks; AI systems engineering

 doi:10.20944/preprints202508.0352.v1

1. Introduction

In recent years, the rapid advancement of deep learning has brought about transformative changes across a wide spectrum of application domains, ranging from computer vision, natural language processing, and speech recognition to autonomous driving, healthcare, and smart Internet-of-Things (IoT) environments. Traditionally, state-of-the-art deep learning models have required immense computational resources and large-scale datasets for both training and inference. These models, typically comprising millions or even billions of parameters, such as BERT, GPT, and ResNet, are predominantly trained and deployed in centralized cloud data centers where abundant computational resources and storage capabilities are available. However, the proliferation of mobile devices, edge computing infrastructures, and embedded systems has introduced a new paradigm in deep learning: the desire and necessity to perform inference, and to some extent training, on resource-constrained devices at the edge of the network [1]. This shift is driven by multiple compelling factors [2]. First, there is an increasing demand for real-time, low-latency inference in applications such as augmented reality, robotics, and smart homes, which cannot tolerate the round-trip latency incurred by transmitting data to remote cloud servers [3]. Second, concerns regarding data privacy, security, and compliance with regulations such as GDPR have motivated on-device processing to minimize the exposure of sensitive user data. Third, the sheer volume of data generated at the edge makes it impractical and cost-prohibitive to transmit all data to the cloud for processing, thus necessitating more intelligent and distributed computing paradigms [4]. Nonetheless, the limited computational power, energy constraints, and memory limitations of edge devices pose significant challenges to the deployment of large-scale deep learning models directly on-device. To address these constraints, various techniques such as model quantization, pruning, knowledge distillation, and architecture search for efficient models (e.g., MobileNet, TinyBERT) have been proposed to enable the design and deployment of small deep learning models that can operate within the capabilities of edge hardware [5]. Despite these advances, small models often suffer from reduced accuracy and generalization performance compared to their large-scale cloud counterparts, thereby creating a trade-off between efficiency and performance [6]. This confluence of cloud and edge computing capabilities has given rise to the concept of collaborative learning, a hybrid learning paradigm where small models on edge devices and large models in the cloud cooperate to optimize inference and learning tasks. Collaborative learning seeks to harness the complementary strengths of both edge and cloud: the proximity, privacy, and responsiveness of on-device models, and the accuracy, computational richness, and data aggregation abilities of cloud-based models. The overarching objective is to design synergistic frameworks that allow seamless interaction between these heterogeneous models while addressing challenges related to communication overhead, model consistency, system heterogeneity, and dynamic environmental conditions [7]. Several collaborative learning frameworks have been proposed, including but not limited to, split computing (also known as split learning), where a deep neural network is partitioned between the edge and cloud; federated learning, where decentralized models are trained collaboratively without sharing raw data; and knowledge distillation-based approaches, where large cloud models distill knowledge to small edge models either offline or dynamically during inference. Moreover, hybrid techniques that integrate multiple paradigms, such as federated distillation or split-federated learning, have emerged to exploit the advantages of each method. Beyond the methodological developments, the deployment of collaborative learning systems introduces a range of system-level challenges, including network variability, energy management, model synchronization, data heterogeneity, and security threats such as model inversion and poisoning attacks. Consequently, the design of robust, efficient, and secure collaborative learning architectures demands a cross-disciplinary approach, integrating insights from machine learning, systems engineering, communication networks, and security [8]. This survey provides a comprehensive and structured review of the emerging field of collaborative learning between on-device small models and cloud-based large deep learning models. Our aim is to elucidate the fundamental principles, technical challenges, and current state-of-the-art methodologies in this domain [9]. We categorize and analyze existing approaches, highlight key design trade-offs,

 doi:10.20944/preprints202508.0352.v1

and identify promising directions for future research. Specifically, we address the following aspects in this survey:

- The motivation and rationale for collaborative learning, with an emphasis on application scenarios and system constraints [10].
- A taxonomy of collaborative learning frameworks, including split learning, federated learning, and knowledge distillation, and their adaptations for heterogeneous model collaboration.
- Technical challenges inherent to collaborative learning, such as latency, energy consumption, model heterogeneity, and security, along with potential mitigation strategies.
- An overview of real-world systems and platforms that support collaborative learning, including hardware, software, and network considerations.
- Open research questions and future directions, including the role of foundation models, personalized learning, adaptive collaboration strategies, and standardization efforts [11].

In summary, as the demand for intelligent edge applications continues to grow, and the limitations of both standalone cloud and edge computing become more pronounced, collaborative learning offers a promising pathway towards scalable, efficient, and privacy-preserving deep learning systems. This survey seeks to equip researchers, practitioners, and system designers with a deep understanding of the landscape of collaborative learning and to foster the development of next-generation intelligent systems that seamlessly blend the capabilities of on-device small models with those of cloud-based large models [12].

2. Problem Formulation

Let us formally define the problem of collaborative learning between an on-device small model and a cloud-based large deep learning model. We denote the input space by $\mathcal{X} \subseteq \mathbb{R}^d$ and the corresponding output or label space by $\mathcal{Y}$. Given a data distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, the goal of a learning system is to find a function $f : \mathcal{X} \to \mathcal{Y}$ that minimizes a loss function $\mathcal{L}(f(x), y)$, where $(x, y) \sim \mathcal{D}$. In the context of collaborative learning, this function f is not implemented by a single monolithic model, but rather decomposed into two (or more) components that are distributed across the edge device and the cloud infrastructure [13]. Let us denote the on-device small model by $f_s : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{Z} \subseteq \mathbb{R}^k$ represents an intermediate feature space, and the cloud-based large model by $f_l : \mathcal{Z} \to \mathcal{Y}$ [14]. The complete model is then given by the composition $f(x) = f_l(f_s(x))$. The optimization objective is to minimize the expected loss:

$$\min_{f_s, f_l} \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(f_l(f_s(x)), y)]. \tag{1}$$

In practice, only a finite dataset $\{(x_i, y_i)\}_{i=1}^n$ sampled from $\mathcal{D}$ is available. Moreover, this dataset may be distributed across edge devices and cloud servers, denoted by $\mathcal{D}_e$ and $\mathcal{D}_c$ respectively. In some cases, data cannot be directly shared between devices and the cloud due to privacy constraints, leading to the need for distributed optimization strategies [15]. Let us denote the empirical loss over the dataset by:

$$\hat{\mathcal{L}}(f_s, f_l) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f_l(f_s(x_i)), y_i). \tag{2}$$

The problem then becomes minimizing $\hat{\mathcal{L}}$ subject to system constraints:

$$\begin{aligned} \min_{f_s, f_l} \quad & \hat{\mathcal{L}}(f_s, f_l) \\ \text{subject to} \quad & C(f_s, f_l) \leq \tau, \end{aligned} \tag{3}$$

where $C(f_s, f_l)$ represents a composite cost function encompassing computation time, energy consumption, communication latency, and privacy leakage, and τ is a user-defined threshold. One of the primary challenges in collaborative learning lies in the optimal partitioning of the model and the

 doi:10.20944/preprints202508.0352.v1

coordination between f_s and f_l. For instance, in split learning, the partition point π determines the layer at which the model is split between the edge and the cloud [16]. Let $f(x;\pi) = f_l^{[\pi+1:L]}(f_s^{[1:\pi]}(x))$, where L denotes the total number of layers in the complete model. Finding an optimal partition point requires balancing the computational cost $C_e(\pi)$ on the edge and $C_c(\pi)$ on the cloud, as well as the communication cost $C_{comm}(\pi)$ incurred in transmitting the intermediate activation $z = f_s^{[1:\pi]}(x)$ from the edge to the cloud. Therefore, the partition optimization problem can be formalized as:

$$\min_{\pi} \quad C_e(\pi) + C_c(\pi) + C_{comm}(\pi), \quad \text{s.t.} \quad \mathcal{L}(f(x;\pi), y) \leq \epsilon, \tag{4}$$

where ϵ denotes an acceptable loss tolerance. In knowledge distillation-based collaborative learning, the objective shifts towards minimizing a distillation loss $\mathcal{L}_{KD}$ between the output of the small model f_s and a large teacher model f_T deployed in the cloud [17]. The distillation objective is typically formulated as:

$$\mathcal{L}_{KD} = \alpha \cdot \mathcal{L}_{CE}(f_s(x), y) + (1-\alpha) \cdot \mathcal{L}_{KL}(f_s(x), f_T(x)), \tag{5}$$

where $\mathcal{L}_{CE}$ denotes the cross-entropy loss, $\mathcal{L}_{KL}$ is the Kullback–Leibler divergence between the soft outputs of f_s and f_T, and $\alpha \in [0,1]$ is a hyperparameter balancing the two terms. The goal is to train the on-device small model to mimic the predictions of the large cloud model while retaining acceptable computational efficiency. In federated learning scenarios, let us consider a set of M devices, each with a local dataset $\mathcal{D}_m$, $m = 1, \ldots, M$ [18]. The collaborative objective is to minimize the global loss:

$$\min_{f_s} \sum_{m=1}^{M} \frac{|\mathcal{D}_m|}{n} \hat{\mathcal{L}}_m(f_s), \tag{6}$$

where $\hat{\mathcal{L}}_m(f_s) = \frac{1}{|\mathcal{D}_m|} \sum_{(x_i,y_i)\in\mathcal{D}_m} \mathcal{L}(f_s(x_i), y_i)$. The large cloud model f_l may serve as an aggregator or teacher, distilling knowledge to the federated edge models via a server-based coordination protocol [19]. In summary, the mathematical formulation of collaborative learning encompasses diverse optimization objectives and constraints, dependent on the chosen collaboration paradigm—split learning, federated learning, knowledge distillation, or hybrid models [20]. Each paradigm introduces unique computational and communication trade-offs, necessitating multi-objective optimization approaches to balance performance, efficiency, and privacy [21]. Additionally, dynamic environmental factors such as network bandwidth fluctuations, device heterogeneity, and user mobility introduce further complexity into this optimization landscape, making adaptive and robust solutions critical to real-world deployment. This section sets the mathematical foundation upon which subsequent discussions of specific methodologies and systems will be built [22].

3. System Architecture

A fundamental aspect of collaborative learning between on-device small models and cloud-based large models lies in the design and implementation of an efficient, scalable, and adaptable system architecture [23]. This architecture must orchestrate the interactions between edge devices and cloud servers to facilitate joint model inference and training while satisfying constraints related to latency, privacy, energy, and communication bandwidth [24]. Figure 1 presents a vertical overview of a typical collaborative learning architecture [25]. At the bottom layer are Edge Devices, which include smartphones, IoT sensors, autonomous vehicles, and wearable devices. Each device contains limited computational resources—CPU, GPU, or specialized AI accelerators—and stores a lightweight model component f_s, along with a local dataset $\mathcal{D}_e$ [26]. These devices perform initial data processing and model inference, and optionally contribute to local training [27]. Above the edge lies the Communication Layer, responsible for managing data transmission between edge and cloud. This layer accounts for variable network conditions, latency constraints, and security protocols such as encryption and differential privacy. It also supports compression and quantization of intermediate representations to minimize communication overhead. At the top is the Cloud Infrastructure, which hosts the large

 doi:10.20944/preprints202508.0352.v1

deep learning model f_l, trained on massive datasets $\mathcal{D}_c$. The cloud has high computational capacity, enabling intensive tasks such as fine-tuning, centralized aggregation, and large-scale inference [28]. It also coordinates collaborative protocols, maintains global model versions, and performs knowledge distillation to enhance the performance of edge models.

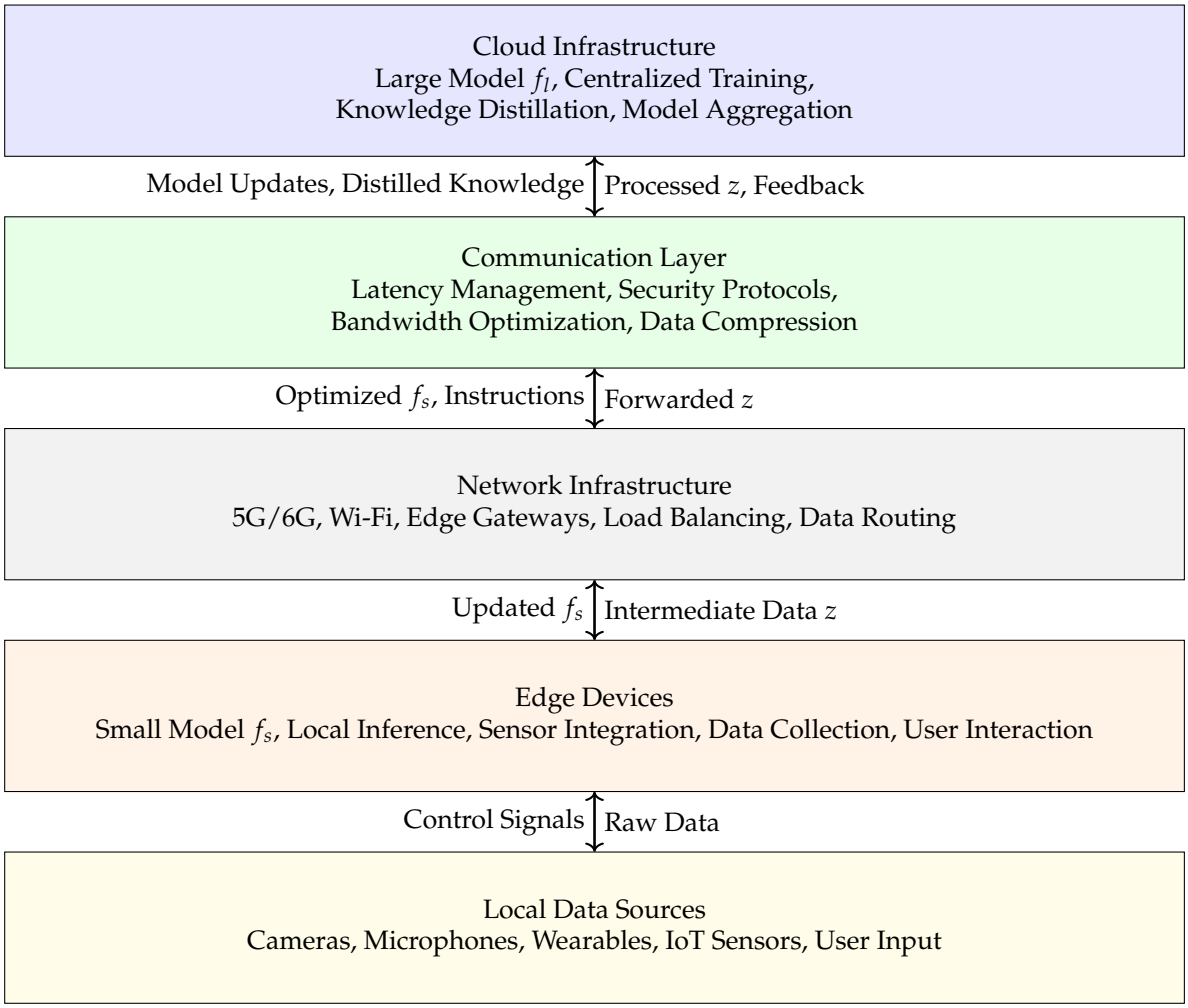

Figure 1. Extended vertical system architecture for collaborative learning between on-device small models and cloud-based large models. The figure illustrates a full-stack view from data sources to the cloud, emphasizing communication and model update flows.

The edge-cloud collaboration depicted in Figure 1 supports various operational modes [29]. In inference mode, raw input x is first processed by the small model f_s on the device to generate intermediate features $z = f_s(x)$ [30]. These features are transmitted through the communication layer to the cloud, where the large model f_l completes the inference, producing $\hat{y} = f_l(z)$ [31]. This division reduces latency while leveraging the accuracy of the large model [32]. In training mode, either federated learning or knowledge distillation may be used. In federated learning, each edge device trains f_s on local data and periodically sends model updates (e.g., gradients or weights) to the cloud, which aggregates them into a global model. In distillation, the cloud model f_l sends distilled outputs or soft labels to the edge, enhancing the training of f_s without exposing raw data. A critical component of this architecture is the communication protocol [33]. Due to bandwidth constraints and energy limitations on edge devices, it is essential to compress the transmitted data z without significantly degrading model performance [34]. Techniques such as sparsification, quantization, and entropy coding are commonly used [35]. Additionally, privacy-preserving mechanisms such as homomorphic encryption, secure multi-party computation, or differential privacy can be applied to z to protect sensitive user information [36]. Furthermore, the architecture must adapt to heterogeneous

 doi:10.20944/preprints202508.0352.v1

edge devices with varying capabilities [37]. Dynamic model partitioning allows the system to select the optimal split point π in the model, based on real-time monitoring of network latency, energy availability, and computational load. This requires a runtime decision engine, potentially driven by reinforcement learning, to balance the trade-offs between local computation and remote processing [38]. In conclusion, the system architecture for collaborative learning must holistically integrate edge computation, communication efficiency, and cloud intelligence. The interplay between small and large models across this architecture enables a continuum of intelligence from edge to cloud, empowering applications that demand responsiveness, privacy, and high accuracy simultaneously [39]. Subsequent sections will delve into the specific methodologies that instantiate this architecture in practical systems.

4. Comparison of Collaborative Learning Paradigms

Collaborative learning between on-device small models and cloud-based large models can be implemented through several distinct paradigms, each with its own operational principles, advantages, and limitations [40]. Among the most widely adopted frameworks are Split Learning, Federated Learning, and Knowledge Distillation, along with emerging hybrid models that combine elements from these approaches [41]. A systematic comparison of these paradigms is crucial for understanding the design trade-offs involved and for selecting the most appropriate strategy for a given application scenario. Table 1 provides a comparative summary of the key characteristics of the major collaborative learning paradigms. The comparison covers aspects such as the structure of model partitioning, data locality, communication patterns, privacy implications, computational load distribution, and typical use cases [42]. Each paradigm is tailored to address specific constraints and goals, and their suitability varies depending on the context, such as the heterogeneity of edge devices, network conditions, and privacy requirements.

Table 1. Comparison of Collaborative Learning Paradigms.

Aspect	Split Learning	Federated Learning	Knowledge Distillation
Model Partitioning	Model is split between edge and cloud at an intermediate layer	Full model resides on each edge device; no split	Small model on device, large teacher in cloud
Data Locality	Raw data stays on edge; only intermediate activations sent	Raw data stays on edge; only model updates sent	Raw data may stay on edge; soft labels or logits exchanged
Communication Pattern	Frequent bidirectional transfer of activations and gradients per sample/batch	Periodic upload of model updates; occasional downloads	Irregular transfer of predictions or distilled knowledge
Privacy Preservation	Moderate; activations may leak some data	High; only updates transmitted	Variable; depends on distillation method used
Edge Computation Load	Low to moderate; partial forward/backward pass	High; full training/inference on device	Low; primarily forward pass
Cloud Computation Load	High; completes forward/backward pass for each sample	Low to moderate; model aggregation or central training	High; teacher model training and distillation
Latency Sensitivity	High; real-time communication needed for inference/training	Low to moderate; training is asynchronous	Low; distillation can be scheduled flexibly
Typical Use Cases	Real-time inference, privacy-sensitive settings	Large-scale collaborative training, personalization	Model compression, continual learning, personalization

Split Learning offers a fine-grained division of computation between edge and cloud, making it particularly suitable for real-time inference in latency-sensitive applications [43]. In this paradigm, the edge device processes the input x through the early layers of the model, generating intermediate activations z, which are sent to the cloud. The cloud completes the inference or backpropagation and may return gradients or output predictions [44]. While this approach minimizes data exposure and

 doi:10.20944/preprints202508.0352.v1

supports dynamic model partitioning, it requires low-latency, high-bandwidth connections and can be communication-intensive during training. Federated Learning decentralizes the training process by distributing full copies of a small model to each participating edge device. Devices perform local training on their own data and periodically transmit model updates, such as weight deltas or gradients, to a central server [45]. The server aggregates these updates to produce a global model, which is redistributed [46]. This method excels in privacy preservation and supports massive scalability, but imposes significant computation on edge devices and may struggle with data heterogeneity (non-IID data) and stragglers in asynchronous environments. Knowledge Distillation enables indirect collaboration by having a powerful cloud model (teacher) generate softened labels or feature representations, which are then used to train a smaller model (student) on the edge. This can occur either offline or online, and may involve continual distillation as the student model encounters new data. This approach is flexible in terms of communication frequency and can significantly compress large models for edge deployment [47]. However, its effectiveness depends on the alignment between teacher and student models and may require repeated cloud access for optimal performance [48]. Ultimately, the choice of paradigm depends on the specific application requirements, such as the need for privacy, latency constraints, and the computational capacity of edge devices. Hybrid models, such as split-federated learning or federated distillation, are gaining popularity as they attempt to combine the strengths of different approaches [49]. For example, federated learning may be augmented with distillation to reduce communication overhead, or split learning may be integrated with privacy-preserving techniques to enhance security. In conclusion, understanding the comparative landscape of collaborative learning paradigms allows system designers and researchers to better align technical capabilities with application demands [50]. As edge-cloud ecosystems become more diverse and pervasive, the ability to flexibly adopt and combine different paradigms will be crucial for building robust, efficient, and intelligent systems.

5. Challenges and Research Opportunities

While collaborative learning between on-device small models and cloud-based large models presents immense potential, realizing its full benefits requires overcoming several fundamental and practical challenges. These challenges span multiple dimensions, including system-level constraints, algorithmic limitations, privacy-preserving mechanisms, and real-world deployment concerns. Addressing these issues not only enables more robust and efficient systems but also opens up fertile ground for impactful research and innovation. One of the foremost challenges is the optimization of communication efficiency between edge and cloud [51]. As collaborative learning often involves frequent exchange of intermediate data, model parameters, or gradients, the communication cost can become a bottleneck, especially in bandwidth-constrained or latency-sensitive environments [52]. In split learning, for instance, transmitting high-dimensional feature maps z at each forward pass imposes significant load on the network, which may not be sustainable for real-time applications or in regions with limited connectivity. Advanced techniques such as adaptive compression, sparse representation learning, and progressive transmission of features are actively being explored to mitigate this issue. Additionally, dynamically selecting the model partition point π based on current network conditions and computational load is a promising strategy, but it requires sophisticated orchestration mechanisms and real-time monitoring. Another critical issue is data heterogeneity and non-IID distribution across edge devices [53]. In federated and distillation-based learning paradigms, each device typically collects data unique to its environment and user behavior, leading to statistically diverse local datasets $\mathcal{D}_e^{(i)}$ [54]. This non-IID nature of data introduces challenges in achieving convergence and generalization, as model updates from different devices may be conflicting or biased. Consequently, aggregation methods such as weighted averaging in federated learning (e.g., FedAvg) may lead to suboptimal performance [55]. Research is increasingly focusing on personalized federated learning, meta-learning, and domain adaptation techniques that can account for data heterogeneity and tailor the global model to diverse local conditions without compromising overall performance. Privacy and

 doi:10.20944/preprints202508.0352.v1

security concerns are also paramount in collaborative learning scenarios. Although raw data typically remains on the edge, intermediate features or model updates may still leak sensitive information. For example, in split learning, it has been shown that activations z can be exploited through inversion attacks to reconstruct input data. Similarly, in federated learning, gradients may reveal private data if not properly sanitized [56]. Techniques such as differential privacy, secure multi-party computation (SMPC), homomorphic encryption, and trusted execution environments (TEE) offer potential solutions, but each comes with trade-offs in terms of computational cost, latency, and scalability [57]. A key research direction involves developing lightweight, scalable privacy-preserving mechanisms that can be deployed efficiently across heterogeneous devices with limited resources. From an algorithmic perspective, model co-design poses a unique challenge [58]. Collaborative learning necessitates designing models that are not only accurate but also modular, with clearly defined interfaces between the small model f_s and large model f_l [59]. This requires innovations in neural network architectures that support efficient partitioning, distillation, or transfer of knowledge [60]. Techniques such as neural architecture search (NAS), modular neural networks, and efficient transformer variants (e.g., MobileBERT, TinyViT) are being adapted to facilitate this co-design [61]. Moreover, dynamically reconfigurable models that adjust their complexity or depth based on available resources can offer additional flexibility, but they require reliable runtime control mechanisms and robust performance guarantees. A further challenge lies in the evaluation and benchmarking of collaborative learning systems. Traditional metrics such as accuracy and inference time do not fully capture the multi-dimensional trade-offs involved, including energy consumption, privacy risk, communication overhead, and system robustness. Standardized benchmarks, datasets, and simulation environments that reflect realistic edge-cloud scenarios are urgently needed. Such benchmarks should account for diverse device capabilities, network conditions, and user behaviors to enable fair comparison and reproducibility of research outcomes. Finally, scalability and deployment at scale remain significant hurdles. Real-world applications involve a large number of edge devices, each with varying hardware, software, and network configurations [62]. Efficient orchestration, fault tolerance, and update propagation in such heterogeneous environments are complex yet essential for practical adoption. Emerging technologies such as 5G, edge AI accelerators, and distributed computing frameworks (e.g., Kubernetes, TensorFlow Federated) offer infrastructure support, but their integration into collaborative learning workflows remains an open research area. In conclusion, while collaborative learning between on-device small models and cloud-based large models holds transformative potential, realizing this vision necessitates overcoming substantial challenges across communication, computation, privacy, and scalability [63]. These challenges, however, simultaneously represent rich research opportunities. Future work that integrates advances in machine learning, systems engineering, and privacy-enhancing technologies will be crucial in making collaborative learning not only feasible but also ubiquitous across diverse domains such as healthcare, autonomous systems, smart cities, and personal AI.

6. Applications and Case Studies

The collaborative learning paradigm—where on-device small models work in tandem with powerful cloud-based deep learning models—has begun to transform a wide range of application domains. This transformation is driven by the growing need for systems that balance the often conflicting requirements of responsiveness, accuracy, privacy, and energy efficiency [64]. In this section, we explore several prominent application areas and case studies that demonstrate the tangible benefits and practical considerations of deploying collaborative learning systems in real-world environments [65]. One of the most impactful areas is healthcare and personalized medicine, where patient data is highly sensitive and privacy regulations such as HIPAA and GDPR impose strict data protection requirements. In scenarios such as wearable health monitoring or mobile diagnostics, edge devices such as smartwatches or smartphones continuously collect physiological data (e.g., heart rate, blood oxygen levels, ECG signals). On-device models provide immediate feedback, detecting anomalies or triggering alerts in real-time. However, due to the limited capacity of edge models, complex diagnostic tasks or

 doi:10.20944/preprints202508.0352.v1

longitudinal pattern recognition may necessitate cloud-based analysis. Through collaborative learning—often via split learning or federated learning—the edge can process raw signals locally and send compact, privacy-preserving representations to the cloud, which refines the diagnosis using advanced models trained on large-scale medical datasets. Case studies have demonstrated this approach in early detection of arrhythmias, diabetic retinopathy screening, and even mental health monitoring via speech and behavioral analysis, offering both high accuracy and robust privacy [66]. In the realm of autonomous vehicles and intelligent transportation systems, real-time decision-making is critical, yet full reliance on cloud services is infeasible due to latency and connectivity limitations. Autonomous vehicles must process sensor data (e.g., LiDAR, camera, radar) rapidly to perform tasks such as object detection, path planning, and collision avoidance. On-board models offer low-latency inference but are constrained in complexity due to energy and space limitations [67]. Collaborative learning enables these systems to offload computationally intensive tasks, such as complex scene understanding or global route optimization, to the cloud [68]. For example, a vehicle might detect objects locally but send semantically rich representations to the cloud for high-level decision support, receiving back optimized navigation strategies. Furthermore, federated learning can be used to aggregate driving experiences from multiple vehicles without exposing raw sensor data, leading to continual improvement of both edge and cloud models [69]. These systems also benefit from real-time cloud updates, enabling vehicles to adapt quickly to new traffic patterns or hazardous conditions [70]. Smart cities and IoT infrastructure represent another fertile domain for collaborative learning [71]. In such environments, a multitude of distributed sensors and edge devices collect data related to energy consumption, air quality, traffic flow, and public safety. These devices, typically constrained in power and compute, require efficient local inference to enable immediate responses, such as triggering alarms or adjusting environmental controls [72]. However, the aggregate data across a city provides valuable insights for long-term planning and policy-making. Collaborative learning enables both immediate local responsiveness and comprehensive global analysis. For instance, in energy management systems, local devices predict energy demand and adjust consumption autonomously, while cloud-based models forecast city-wide energy trends and optimize grid operations [73]. Split learning facilitates secure, real-time interaction between local and centralized intelligence, while federated learning ensures that sensitive user data remains on-premise. In the field of natural language processing (NLP) and personal AI assistants, collaborative learning is crucial for enabling personalized yet privacy-preserving interactions. Virtual assistants like Siri, Alexa, or Google Assistant must respond quickly to voice commands, often without reliable network connectivity [74]. On-device models handle basic commands and wake-word detection, while cloud-based large language models provide more sophisticated understanding and dialog generation [75]. Through distillation and collaborative inference, the on-device models can continually improve from cloud-based models without exposing user conversations [76]. Moreover, federated learning allows models to learn from user interactions across millions of devices while ensuring that individual voice data never leaves the device. Recent studies have shown that such approaches can significantly improve the quality and personalization of dialog systems while maintaining strong privacy guarantees. Finally, in augmented reality (AR) and mobile gaming, collaborative learning enables rich, immersive experiences on resource-constrained devices. AR applications require real-time processing of video, depth, and spatial data to render virtual content accurately and responsively. On-device models perform rapid pose estimation and environment mapping, while cloud models handle complex tasks such as object recognition, scene understanding, or multiplayer coordination [77]. Collaborative inference ensures that latency-critical operations are kept local, while cloud intelligence enhances realism and consistency across devices [78]. Case studies in mobile AR gaming and industrial AR support systems highlight how hybrid model deployment enables scalable, low-latency experiences that were previously only possible on high-end hardware [79]. In summary, collaborative learning between on-device small models and cloud-based large models is being actively deployed across diverse application domains, each with unique requirements and challenges. The key to successful deployment lies in carefully balancing the division of labor between edge and cloud, optimizing com-

 doi:10.20944/preprints202508.0352.v1

munication, and ensuring privacy and robustness. As hardware capabilities continue to evolve and networks become more pervasive, the reach and impact of collaborative learning systems are poised to expand dramatically, enabling smarter, more responsive, and more privacy-respecting technologies across all facets of daily life.

7. Future Directions and Emerging Trends

As collaborative learning continues to gain traction in both academic research and industrial deployment, several emerging trends and future directions are shaping its evolution [80]. These trends are influenced not only by advances in machine learning algorithms but also by innovations in hardware, networking, privacy technologies, and system-level design. The future of collaborative learning will be characterized by increasing intelligence and autonomy at the edge, more seamless and efficient integration with the cloud, and greater personalization, security, and adaptability of AI systems across diverse application contexts. One significant future direction lies in the development of adaptive and dynamic collaboration mechanisms [81]. Current collaborative learning systems often rely on static configurations, where the division of computation and learning responsibilities between edge and cloud is fixed or predetermined. However, real-world environments are inherently dynamic, with fluctuating network bandwidth, varying computational resources, and evolving user requirements [82]. Future systems must be capable of dynamically adjusting the model partitioning point π, selectively offloading tasks based on current resource availability, latency constraints, or privacy concerns. Such systems require real-time monitoring, predictive analytics for resource forecasting, and reinforcement learning-based strategies that can learn optimal collaboration policies over time. Additionally, dynamic model reconfiguration, such as elastic neural networks that adjust their size or depth on-the-fly, will play a key role in enabling fluid, context-aware collaborative intelligence [83]. Another critical area of growth is the integration of privacy-enhancing technologies (PETs) into collaborative learning workflows [84]. While methods such as differential privacy, homomorphic encryption, and secure multi-party computation have seen increasing adoption, future systems will need to embed these technologies more deeply and efficiently into learning pipelines [85]. Lightweight cryptographic protocols, efficient privacy budget management, and user-controllable privacy settings will become essential, particularly in consumer-facing applications such as personal AI assistants and health monitoring [86]. Moreover, the intersection of PETs with federated and split learning opens up new possibilities, such as privacy-preserving split learning or federated analytics, where both raw data and intermediate representations remain secure. The design of PET-aware model architectures, communication protocols, and optimization algorithms is an emerging field that will significantly influence the deployment of trustworthy AI systems [87]. The rise of specialized edge AI hardware represents another transformative trend. Dedicated AI accelerators, such as Tensor Processing Units (TPUs), Neural Processing Units (NPUs), and custom ASICs, are becoming increasingly prevalent in smartphones, IoT devices, and embedded systems. These accelerators enable low-latency, energy-efficient execution of inference and training tasks on the edge, thus expanding the scope of collaborative learning. Future research will focus on co-designing algorithms and hardware to maximize performance within stringent power and thermal envelopes. Techniques such as quantization-aware training, hardware-aware neural architecture search (HW-NAS), and compiler optimizations for AI inference (e.g., TVM, XLA) will be critical enablers of efficient collaboration [88]. Additionally, the emergence of neuromorphic computing and in-memory processing holds promise for ultra-low-power edge intelligence, further enhancing the feasibility of real-time, on-device learning [89]. Another promising direction is the advancement of continual and lifelong learning in collaborative settings. In many real-world scenarios, data is not static but evolves over time, reflecting changes in user behavior, environment, and system requirements [90]. Collaborative learning systems must be capable of incremental learning, continually adapting to new data without catastrophic forgetting [91]. This requires the development of algorithms that support on-device continual learning, coupled with cloud-assisted memory consolidation and knowledge transfer [92]. Techniques such as elastic weight consolidation, experience replay, and meta-learning can

 doi:10.20944/preprints202508.0352.v1

be adapted to collaborative settings, enabling personalized and adaptive AI experiences that improve over time [93]. Moreover, decentralized learning paradigms, where learning occurs in peer-to-peer networks without centralized cloud orchestration, may emerge as viable alternatives in privacy-critical or resource-constrained environments. Finally, the future of collaborative learning will be shaped by the emergence of standardized frameworks, benchmarks, and platforms that facilitate experimentation, evaluation, and deployment. Current research is often siloed, with limited comparability due to the lack of common datasets, metrics, or system configurations [94]. Open-source platforms such as TensorFlow Federated, PySyft, and Flower are beginning to address this gap, but there is a pressing need for comprehensive toolchains that integrate model design, privacy controls, communication optimization, and deployment orchestration. Benchmarks that reflect realistic edge-cloud scenarios—including heterogeneous devices, dynamic network conditions, and adversarial threats—will enable more robust evaluation of proposed methods [95]. Furthermore, the development of simulation environments and digital twins for collaborative learning will facilitate rapid prototyping and stress-testing of systems under diverse conditions [96]. In conclusion, the future of collaborative learning lies at the intersection of machine learning, systems engineering, privacy science, and hardware design. The trajectory of research and development will be shaped by the need for intelligent, secure, and adaptable systems that seamlessly integrate the strengths of edge and cloud. As collaborative learning matures, it has the potential to unlock transformative capabilities across sectors, from personalized healthcare and autonomous systems to smart environments and beyond [97]. Continued interdisciplinary collaboration and innovation will be essential to realize this vision and to address the profound technical and societal challenges it presents.

8. Conclusions

The proliferation of intelligent devices and the exponential growth of deep learning capabilities have converged to create both opportunities and challenges in the deployment of artificial intelligence across diverse environments. At the heart of this convergence lies the paradigm of collaborative learning between on-device small models and cloud-based large models—a paradigm that seeks to combine the immediacy, privacy, and autonomy of edge computing with the power, scalability, and holistic intelligence of the cloud [98]. This survey has examined the multifaceted aspects of this collaborative learning landscape, encompassing fundamental methodologies, technical challenges, practical applications, and emerging research directions [99]. One of the core themes that emerges from this survey is the intricate balance that collaborative learning seeks to strike among competing system objectives. On-device models are constrained by limited compute, storage, and energy resources, yet they offer unique advantages such as low-latency inference, enhanced privacy, and real-time responsiveness [100]. In contrast, cloud-based models leverage vast computational resources and massive datasets, enabling sophisticated analysis, continual model updates, and global context-awareness [101]. Collaborative learning frameworks—including split learning, federated learning, knowledge distillation, and hybrid approaches—provide flexible mechanisms to distribute computational and learning workloads across the edge-cloud continuum, dynamically adapting to system constraints and user needs. The design and deployment of such systems, however, is far from trivial [102]. Communication efficiency, privacy preservation, robustness to heterogeneous environments, and the ability to personalize models without compromising scalability are persistent challenges [103]. The interplay between model design and system architecture is particularly critical—requiring co-optimization of algorithms, network protocols, and hardware acceleration [104]. The survey has outlined how recent advances in edge AI hardware, privacy-enhancing technologies, and dynamic model orchestration are beginning to address these challenges, but significant research and engineering work remains. Moreover, the applications of collaborative learning are rapidly expanding across domains [105]. From real-time health monitoring and autonomous driving to smart infrastructure and natural language interfaces, collaborative learning systems are enabling intelligent functionalities that were previously unattainable in resource-constrained settings. These applications not only demonstrate the practical feasibility of

 doi:10.20944/preprints202508.0352.v1

collaborative learning but also highlight its societal impact—in improving accessibility, enhancing safety, protecting privacy, and delivering personalized services at scale. Looking forward, the future of collaborative learning promises even greater integration and intelligence. Emerging trends such as adaptive learning architectures, privacy-centric AI governance, continual and lifelong learning, and standardized benchmarking will drive the next wave of innovation. Collaborative learning is poised to become a foundational element in the architecture of ubiquitous AI—permeating every aspect of human-computer interaction, embedded intelligence, and distributed computing [106].

In conclusion, the collaborative learning of on-device small models and cloud-based large models represents a transformative shift in the design of intelligent systems. It transcends traditional boundaries between centralized and decentralized computing, offering a unified approach that leverages the best of both worlds. Realizing the full potential of this paradigm will require sustained, interdisciplinary efforts that bridge machine learning, systems engineering, privacy science, and user-centric design. As these efforts mature, collaborative learning will not only enhance the capabilities of AI systems but also ensure that these capabilities are delivered in ways that are efficient, equitable, secure, and aligned with human values.

References

1. Don-Yehiya, S.; Venezian, E.; Raffel, C.; Slonim, N.; Choshen, L. ColD Fusion: Collaborative Descent for Distributed Multitask Finetuning. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Rogers, A.; Boyd-Graber, J.; Okazaki, N., Eds., Toronto, Canada, 2023; pp. 788–806. https://doi.org/10.18653/v1/2023.acl-long.46.
2. Givón, T. Verb serialization in Tok Pisin and Kalam: A comparative study of temporal packaging. *Melanesian Pidgin and Tok Pisin* **1990**, pp. 19–55.
3. Kearns, M.J. Computational Complexity of Machine Learning. PhD thesis, Department of Computer Science, Harvard University, 1989.
4. Lee, J.; Kang, S.; Lee, J.; Shin, D.; Han, D.; Yoo, H.J. The hardware and algorithm co-design for energy-efficient DNN processor on edge/mobile devices. *IEEE Transactions on Circuits and Systems I: Regular Papers* **2020**, *67*, 3458–3470.
5. Zniyed, Y.; Nguyen, T.P.; et al. Enhanced network compression through tensor decompositions and pruning. *IEEE Transactions on Neural Networks and Learning Systems* **2024**.
6. Barisin, T.; Horenko, I. On entropic sparsification of neural networks. *Pattern Recognition Letters* **2025**.
7. Schick, T.; Schütze, H. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 2021; pp. 2339–2352. https://doi.org/10.18653/v1/2021.naacl-main.185.
8. Langer, M.; He, Z.; Rahayu, W.; Xue, Y. Distributed training of deep learning models: A taxonomic perspective. *IEEE Transactions on Parallel and Distributed Systems* **2020**, *31*, 2802–2818.
9. Shomron, G.; Gabbay, F.; Kurzum, S.; Weiser, U. Post-training sparsity-aware quantization. *Advances in Neural Information Processing Systems* **2021**, *34*, 17737–17748.
10. Niven, T.; Kao, H.Y. Probing Neural Network Comprehension of Natural Language Arguments. In Proceedings of the Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 4658–4664.
11. Shrotri, A.A.; Narodytska, N.; Ignatiev, A.; Meel, K.S.; Marques-Silva, J.; Vardi, M.Y. Constraint-driven explanations for black-box ML models. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2022, Vol. 36, pp. 8304–8314.
12. Author, N.N. Suppressed for Anonymity, 2021.
13. Nalisnick, E.; Smyth, P. Stick-breaking variational autoencoders. In Proceedings of the International Conference on Learning Representations (ICLR), 2017.
14. Hayou, S.; Ton, J.F.; Doucet, A.; Teh, Y.W. Robust pruning at initialization. *arXiv preprint arXiv:2002.08797* **2020**.
15. Zhang, H.; Liu, Y. BFP: Balanced Filter Pruning via Knowledge Distillation for Efficient Deployment of CNNs on Edge Devices. *Neurocomputing* **2025**, p. 130946.
16. Pansare, N.; Katukuri, J.; Arora, A.; Cipollone, F.; Shaik, R.; Tokgozoglu, N.; Venkataraman, C. Learning compressed embeddings for on-device inference. *Proceedings of Machine Learning and Systems* **2022**, *4*, 382–397.

 doi:10.20944/preprints202508.0352.v1

17. Wang, Z. Sparsert: Accelerating unstructured sparsity on gpus for deep learning inference. In Proceedings of the Proceedings of the ACM international conference on parallel architectures and compilation techniques, 2020, pp. 31–42.
18. Durbin, J. airoboros: Customizable implementation of the self-instruct paper. https://github.com/jondurbin/airoboros, 2024.
19. Wang, H.; Sayadi, H.; Mohsenin, T.; Zhao, L.; Sasan, A.; Rafatirad, S.; Homayoun, H. Mitigating cache-based side-channel attacks through randomization: A comprehensive system and architecture level analysis. In Proceedings of the 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 2020, pp. 1414–1419.
20. Qi, M.; Wang, D.; Yang, W.; Liu, B.; Wang, F.; Chen, Z. Fine-grained hierarchical singular value decomposition for convolutional neural networks compression and acceleration. *Neurocomputing* **2025**, *636*, 129966.
21. Beck, T.; Bohlender, B.; Viehmann, C.; Hane, V.; Adamson, Y.; Khuri, J.; Brossmann, J.; Pfeiffer, J.; Gurevych, I. Adapterhub playground: Simple and flexible few-shot learning with adapters. *arXiv preprint arXiv:2108.08103* **2021**.
22. Passalis, N.; Raitoharju, J.; Tefas, A.; Gabbouj, M. Efficient adaptive inference for deep convolutional neural networks using hierarchical early exits. *Pattern Recognition* **2020**, *105*, 107346.
23. Yu, L.; Xiang, W. X-pruner: explainable pruning for vision transformers. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 24355–24363.
24. Aribandi, V.; Tay, Y.; Schuster, T.; Rao, J.; Zheng, H.S.; Mehta, S.V.; Zhuang, H.; Tran, V.Q.; Bahri, D.; Ni, J.; et al. ExT5: Towards Extreme Multi-Task Scaling for Transfer Learning. In Proceedings of the International Conference on Learning Representations, 2022.
25. Zhao, Z.; Gan, L.; Wang, G.; Hu, Y.; Shen, T.; Yang, H.; Kuang, K.; Wu, F. Retrieval-Augmented Mixture of LoRA Experts for Uploadable Machine Learning. *arXiv preprint arXiv:2406.16989* **2024**.
26. Aggarwal, S.; Binici, K.; Mitra, T. Chameleon: Dual memory replay for online continual learning on edge devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2023**.
27. Liu, H.; Simonyan, K.; Yang, Y. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055* **2018**.
28. Gross, W.J.; Meyer, B.H.; Ardakani, A. Hardware-aware design for edge intelligence. *IEEE Open Journal of Circuits and Systems* **2020**, *2*, 113–127.
29. Elkerdawy, S.; Elhoushi, M.; Singh, A.; Zhang, H.; Ray, N. One-shot layer-wise accuracy approximation for layer pruning. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP). IEEE, 2020, pp. 2940–2944.
30. Peng, H.; Pappas, N.; Yogatama, D.; Schwartz, R.; Smith, N.; Kong, L. Random Feature Attention. In Proceedings of the International Conference on Learning Representations, 2021.
31. Huang, C.; Liu, Q.; Lin, B.Y.; Pang, T.; Du, C.; Lin, M. LoraHub: Efficient Cross-Task Generalization via Dynamic LoRA Composition, 2024, [arXiv:cs.CL/2307.13269].
32. Gadosey, P.K.; Li, Y.; Yamak, P.T. On pruned, quantized and compact cnn architectures for vision applications: an empirical study. In Proceedings of the Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing, 2019, pp. 1–8.
33. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial intelligence and statistics. PMLR, 2017, pp. 1273–1282.
34. Michalski, R.S.; Carbonell, J.G.; Mitchell, T.M., Eds. *Machine Learning: An Artificial Intelligence Approach, Vol. I*; Tioga: Palo Alto, CA, 1983.
35. Wang, Q.; Van Hoof, H. Model-based meta reinforcement learning using graph structured surrogate models and amortized policy search. In Proceedings of the International Conference on Machine Learning. PMLR, 2022, pp. 23055–23077.
36. Ansell, A.; Ponti, E.M.; Pfeiffer, J.; Ruder, S.; Glavaš, G.; Vulić, I.; Korhonen, A. MAD-G: Multilingual Adapter Generation for Efficient Cross-Lingual Transfer. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, 2021, pp. 4762–4781. https://doi.org/10.18653/v1/2021.findings-emnlp.410.
37. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep learning*; Vol. 1, MIT Press, 2016.
38. Zhu, Y.; Peng, H.; Fu, A.; Yang, W.; Ma, H.; Al-Sarawi, S.F.; Abbott, D.; Gao, Y. Towards robustness evaluation of backdoor defense on quantized deep learning models. *Expert Systems with Applications* **2024**, *255*, 124599.

 doi:10.20944/preprints202508.0352.v1

39. Workshop, B.; Scao, T.L.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A.S.; Yvon, F.; et al. BLOOM: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100* **2022**.
40. Sun, T.; Shao, Y.; Li, X.; Liu, P.; Yan, H.; Qiu, X.; Huang, X. Learning sparse sharing architectures for multiple tasks. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2020, Vol. 34, pp. 8936–8943.
41. Wu, X.; Zhang, Y.; Shi, M.; Li, P.; Li, R.; Xiong, N.N. An adaptive federated learning scheme with differential privacy preserving. *Future Generation Computer Systems* **2022**, *127*, 362–372.
42. Nagel, M.; Fournarakis, M.; Bondarenko, Y.; Blankevoort, T. Overcoming oscillations in quantization-aware training. In Proceedings of the International Conference on Machine Learning. PMLR, 2022, pp. 16318–16330.
43. Chevalier-Boisvert, M.; Bahdanau, D.; Lahlou, S.; Willems, L.; Saharia, C.; Nguyen, T.H.; Bengio, Y. BabyAI: First Steps Towards Grounded Language Learning With a Human In the Loop. In Proceedings of the International Conference on Learning Representations, 2019.
44. Liu, J. LlamaIndex, a data framework for your LLM applications. https://github.com/run-llama/llama_index, 2024.
45. Yang, Q.; Liu, Y.; Chen, T.; Tong, Y. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* **2019**, *10*, 1–19.
46. Stich, S.U. Local SGD converges fast and communicates little. *arXiv preprint arXiv:1805.09767* **2018**.
47. Ayyat, M.; Nadeem, T.; Krawczyk, B. ClassyNet: Class-Aware Early Exit Neural Networks for Edge Devices. *IEEE Internet of Things Journal* **2023**.
48. Kessler, S.; Nguyen, V.; Zohren, S.; Roberts, S. Hierarchical Indian Buffet Neural Networks for Bayesian Continual Learning. *arXiv preprint arXiv:1912.02290* **2019**.
49. Newell, A.; Rosenbloom, P.S. Mechanisms of Skill Acquisition and the Law of Practice. In *Cognitive Skills and Their Acquisition*; Anderson, J.R., Ed.; Lawrence Erlbaum Associates, Inc.: Hillsdale, NJ, 1981; chapter 1, pp. 1–51.
50. Gim, I.; Ko, J. Memory-efficient DNN training on mobile devices. In Proceedings of the Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services, 2022, pp. 464–476.
51. Chronopoulou, A.; Pfeiffer, J.; Maynez, J.; Wang, X.; Ruder, S.; Agrawal, P. Language and Task Arithmetic with Parameter-Efficient Layers for Zero-Shot Summarization. *arXiv preprint arXiv:2311.09344* **2023**.
52. Perez, E.; Strub, F.; De Vries, H.; Dumoulin, V.; Courville, A. FiLM: Visual reasoning with a general conditioning layer. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2018, Vol. 32.
53. Bengio, Y.; LeCun, Y. Scaling Learning Algorithms Towards AI. In *Large Scale Kernel Machines*; MIT Press, 2007.
54. Zhong, Z.; Bao, W.; Wang, J.; Zhu, X.; Zhang, X. Flee: A hierarchical federated learning framework for distributed deep neural network over cloud, edge, and end device. *ACM Transactions on Intelligent Systems and Technology (TIST)* **2022**, *13*, 1–24.
55. Dekhovich, A.; Tax, D.M.; Sluiter, M.H.; Bessa, M.A. Continual prune-and-select: class-incremental learning with specialized subnetworks. *Applied Intelligence* **2023**, *53*, 17849–17864.
56. Wang, Z.; Tsvetkov, Y.; Firat, O.; Cao, Y. Gradient Vaccine: Investigating and Improving Multi-task Optimization in Massively Multilingual Models. In Proceedings of the International Conference on Learning Representations, 2021.
57. Liu, J.; Huang, J.; Zhou, Y.; Li, X.; Ji, S.; Xiong, H.; Dou, D. From distributed machine learning to federated learning: A survey. *Knowledge and Information Systems* **2022**, *64*, 885–917.
58. Ravi, S. Efficient on-device models using neural projections. In Proceedings of the International Conference on Machine Learning. PMLR, 2019, pp. 5370–5379.
59. Rajendran, J.; Prasanna, P.; Ravindran, B.; Khapra, M.M. Attend, Adapt and Transfer: Attentive Deep Architecture for Adaptive Transfer from multiple sources in the same domain. In Proceedings of the International Conference on Learning Representations, 2017.
60. Zhao, Z.; Wallace, E.; Feng, S.; Klein, D.; Singh, S. Calibrate Before Use: Improving Few-shot Performance of Language Models. In Proceedings of the Proceedings of the 38th International Conference on Machine Learning; Meila, M.; Zhang, T., Eds. PMLR, 18–24 Jul 2021, Vol. 139, *Proceedings of Machine Learning Research*, pp. 12697–12706.
61. Zhou, H.; Lan, J.; Liu, R.; Yosinski, J. Deconstructing lottery tickets: Zeros, signs, and the supermask. In Proceedings of the Advances in Neural Information Processing Systems, 2019, pp. 3597–3607.

 doi:10.20944/preprints202508.0352.v1

62. Duan, Z.; Zhang, H.; Wang, C.; Wang, Z.; Chen, B.; Zhou, M. EnsLM: Ensemble language model for data diversity by semantic clustering. In Proceedings of the Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 2954–2967.
63. Velliangiri, S.; Alagumuthukrishnan, S.; et al. A review of dimensionality reduction techniques for efficient computation. *Procedia Computer Science* **2019**, *165*, 104–111.
64. Bouzidi, H.; Odema, M.; Ouarnoughi, H.; Al Faruque, M.A.; Niar, S. HADAS: Hardware-aware dynamic neural architecture search for edge performance scaling. In Proceedings of the 2023 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 2023, pp. 1–6.
65. Dong, R.; Mao, Y.; Zhang, J. Resource-constrained edge ai with early exit prediction. *Journal of Communications and Information Networks* **2022**, *7*, 122–134.
66. Wang, Y.; Mishra, S.; Alipoormolabashi, P.; Kordi, Y.; Mirzaei, A.; Arunkumar, A.; Ashok, A.; Dhanasekaran, A.S.; Naik, A.; Stap, D.; et al. Benchmarking Generalization via In-Context Instructions on 1,600+ Language Tasks, 2022. https://doi.org/10.48550/ARXIV.2204.07705.
67. Standley, T.; Zamir, A.; Chen, D.; Guibas, L.; Malik, J.; Savarese, S. Which tasks should be learned together in multi-task learning? In Proceedings of the International conference on machine learning. PMLR, 2020, pp. 9120–9132.
68. Bingel, J.; Søgaard, A. Identifying beneficial task relations for multi-task learning in deep neural networks. *arXiv preprint arXiv:1702.08303* **2017**.
69. Zhong, Z.; Friedman, D.; Chen, D. Factual Probing Is [MASK]: Learning vs. Learning to Recall. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 2021; pp. 5017–5033. https://doi.org/10.18653/v1/2021.naacl-main.398.
70. Li, M.; Gururangan, S.; Dettmers, T.; Lewis, M.; Althoff, T.; Smith, N.A.; Zettlemoyer, L. Branch-train-merge: Embarrassingly parallel training of expert language models. *arXiv preprint arXiv:2208.03306* **2022**.
71. Lazarevich, I.; Kozlov, A.; Malinin, N. Post-training deep neural network pruning via layer-wise calibration. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 798–805.
72. Hui, D.Y.T.; Chevalier-Boisvert, M.; Bahdanau, D.; Bengio, Y. BabyAI 1.1. *arXiv preprint arXiv:2007.12770* **2020**.
73. Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.M.; Chen, W.; et al. Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models. *arXiv preprint arXiv:2203.06904* **2022**.
74. Sutton, R.S. Temporal credit assignment in reinforcement learning. PhD thesis, University of Massachusetts Amherst, 1984.
75. Guo, X.; Wang, W.S.; Zhang, J.; Gong, L.S. An Online Growing-and-Pruning Algorithm of a Feedforward Neural Network for Nonlinear Systems Modeling. *IEEE Transactions on Automation Science and Engineering* **2024**.
76. Vu, T.; Wang, T.; Munkhdalai, T.; Sordoni, A.; Trischler, A.; Mattarella-Micke, A.; Maji, S.; Iyyer, M. Exploring and predicting transferability across NLP tasks. *arXiv preprint arXiv:2005.00770* **2020**.
77. Chen, S.; Yu, D.; Zou, Y.; Yu, J.; Cheng, X. Decentralized wireless federated learning with differential privacy. *IEEE Transactions on Industrial Informatics* **2022**, *18*, 6273–6282.
78. Lu, Z.; Fan, C.; Wei, W.; Qu, X.; Chen, D.; Cheng, Y. Twin-Merging: Dynamic Integration of Modular Expertise in Model Merging. *arXiv preprint arXiv:2406.15479* **2024**.
79. Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457* **2018**.
80. Chen, J.; Ran, X. Deep learning with edge computing: A review. *Proceedings of the IEEE* **2019**, *107*, 1655–1674.
81. Caccia, L.; Ponti, E.; Su, Z.; Pereira, M.; Roux, N.L.; Sordoni, A. Multi-Head Adapter Routing for Cross-Task Generalization, 2023, [arXiv:cs.AI/2211.03831].
82. Tyagi, S.; Swany, M. ScaDLES: Scalable Deep Learning over Streaming data at the Edge. In Proceedings of the 2022 IEEE International Conference on Big Data (Big Data). IEEE, 2022, pp. 2113–2122.
83. Zhang, Y.; Zeng, D.; Luo, J.; Fu, X.; Chen, G.; Xu, Z.; King, I. A survey of trustworthy federated learning: Issues, solutions, and challenges. *ACM Transactions on Intelligent Systems and Technology* **2024**, *15*, 1–47.
84. Baciu, V.E.; Braeken, A.; Segers, L.; Silva, B.d. Secure Tiny Machine Learning on Edge Devices: A Lightweight Dual Attestation Mechanism for Machine Learning. *Future Internet* **2025**, *17*, 85.

 doi:10.20944/preprints202508.0352.v1

85. Ye, S.; Kim, D.; Jang, J.; Shin, J.; Seo, M. Guess the Instruction! Flipped Learning Makes Language Models Stronger Zero-Shot Learners. In Proceedings of the The Eleventh International Conference on Learning Representations, 2023.
86. Choudhary, T.; Mishra, V.; Goswami, A.; Sarangapani, J. A comprehensive survey on model compression and acceleration. *Artificial Intelligence Review* **2020**, *53*, 5113–5155.
87. Bragman, F.J.; Tanno, R.; Ourselin, S.; Alexander, D.C.; Cardoso, J. Stochastic filter groups for multi-task cnns: Learning specialist and generalist convolution kernels. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1385–1394.
88. Qin, C.; Zhao, H.; Wang, L.; Wang, H.; Zhang, Y.; Fu, Y. Slow learning and fast inference: Efficient graph similarity computation via knowledge distillation. *Advances in Neural Information Processing Systems* **2021**, *34*, 14110–14121.
89. Lester, B.; Al-Rfou, R.; Constant, N. The Power of Scale for Parameter-Efficient Prompt Tuning, 2021, [arXiv:cs.CL/2104.08691].
90. Reimers, N.; Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* **2019**.
91. Teh, Y.W.; Görür, D.; Ghahramani, Z. Stick-breaking Construction for the Indian Buffet Process. In Proceedings of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS 2007), 2007, pp. 556–563.
92. Bourechak, A.; Zedadra, O.; Kouahla, M.N.; Guerrieri, A.; Seridi, H.; Fortino, G. At the confluence of artificial intelligence and edge computing in iot-based applications: A review and new perspectives. *Sensors* **2023**, *23*, 1639.
93. Wang, Q.; Xu, M.; Jin, C.; Dong, X.; Yuan, J.; Jin, X.; Huang, G.; Liu, Y.; Liu, X. Melon: Breaking the memory wall for resource-efficient on-device machine learning. In Proceedings of the Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services, 2022, pp. 450–463.
94. Goyal, A.; Lamb, A.; Gampa, P.; Beaudoin, P.; Levine, S.; Blundell, C.; Bengio, Y.; Mozer, M. Object files and schemata: Factorizing declarative and procedural knowledge in dynamical systems. *arXiv preprint arXiv:2006.16225* **2020**.
95. Zhu, Z.; Shi, Y.; Luo, J.; Wang, F.; Peng, C.; Fan, P.; Letaief, K.B. Fedlp: Layer-wise pruning mechanism for communication-computation efficient federated learning. In Proceedings of the ICC 2023-IEEE International Conference on Communications. IEEE, 2023, pp. 1250–1255.
96. Blakeney, C.; Li, X.; Yan, Y.; Zong, Z. Parallel blockwise knowledge distillation for deep neural network compression. *IEEE Transactions on Parallel and Distributed Systems* **2020**, *32*, 1765–1776.
97. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* **2017**.
98. Tang, C.; Ouyang, K.; Wang, Z.; Zhu, Y.; Ji, W.; Wang, Y.; Zhu, W. Mixed-precision neural network quantization via learned layer-wise importance. In Proceedings of the European Conference on Computer Vision. Springer, 2022, pp. 259–275.
99. Liu, H.; Tam, D.; Muqeeth, M.; Mohta, J.; Huang, T.; Bansal, M.; Raffel, C. Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning, 2022.
100. Shi, Y.; Yuan, L.; Chen, Y.; Feng, J. Continual learning via bit-level information preserving. In Proceedings of the Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, 2021, pp. 16674–16683.
101. Sung, Y.L.; Nair, V.; Raffel, C. Training Neural Networks with Fixed Sparse Masks. In Proceedings of the Advances in Neural Information Processing Systems; Beygelzimer, A.; Dauphin, Y.; Liang, P.; Vaughan, J.W., Eds., 2021.
102. Clune, J.; Mouret, J.B.; Lipson, H. The evolutionary origins of modularity. *Proceedings of the Royal Society B: Biological sciences* **2013**, *280*.
103. Sukhbaatar, S.; Golovneva, O.; Sharma, V.; Xu, H.; Lin, X.V.; Rozière, B.; Kahn, J.; Li, D.; Yih, W.t.; Weston, J.; et al. Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM. *arXiv preprint* **2024**, arXiv:2403.07816.
104. Bacon, P.L.; Harb, J.; Precup, D. The option-critic architecture. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2017, Vol. 31.

 doi:10.20944/preprints202508.0352.v1

105. Ha, D.; Dai, A.; Le, Q.V. HyperNetworks. In Proceedings of the Proceedings of the International Conference on Learning Representations 2017, 2017.
106. Dun, C.; Garcia, M.H.; Zheng, G.; Awadallah, A.H.; Sim, R.; Kyrillidis, A.; Dimitriadis, D. FedJETs: Efficient Just-In-Time Personalization with Federated Mixture of Experts, 2023. http://arxiv.org/abs/2306.08586.