Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

OPTIMIZING COMPUTING IN DISTRIBUTED AI SYSTEMS USING EDGE COMPUTING

Aram Andreasyan
IT Specialist, Los Angeles, USA

Abstract:
This article discusses approaches to optimizing computational processes in distributed artificial intelligence systems based on the Edge concept. Computing. This article analyzes architectural models, load balancing methods, model compression algorithms, and computation orchestration mechanisms between the cloud and edge nodes. Particular attention is paid to reducing latency, conserving network resources, and improving the energy efficiency of AI infrastructures.

Keywords: Edge Computing, distributed AI, optimization computing, federated learning, model compression, IoT, latency.

Introduction

The scientific novelty of the study lies in the systematization and structuring of methods for optimizing distributed AI computing at the edge level, taking into account architectural, algorithmic, and infrastructural factors.

The rapid development of artificial intelligence, the Internet of Things (IoT), and cyber-physical systems is accompanied by exponential growth in the volume of generated data. Smart cities, autonomous vehicles, industrial automation, and video surveillance systems generate continuous information flows that require rapid processing and analysis. The traditional centralized cloud computing model faces a number of limitations in such environments, including high data transfer latency, network infrastructure overload, and increased risks of confidential information leaks [1].

Cloud Architecture Computing involves transmitting data from endpoints to remote processing centers where artificial intelligence models are trained and inferred. However, when working with latency-sensitive applications (real-time) in the context of advanced technologies (analytics, autonomous control,

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

telemedicine), this approach is becoming ineffective. Even minor network delays can impact service quality and system security [2].

In response to these challenges, the Edge paradigm was formed. Computing, which involves moving computing resources closer to data sources - to the network edge. This approach enables preprocessing, filtering, and inference directly on edge nodes, reducing the load on the cloud and improving system response times [3].

From a scientific point of view Edge Computing is considered a key architectural component of distributed AI systems. It enables horizontal scaling of computations, support for mobile and IoT devices, and infrastructure resilience to network failures. Research shows that integrating AI algorithms at the network edge can significantly reduce latency and improve the efficiency of streaming data processing [4].

One of the most important areas of Edge AI development is optimizing the distribution of computing tasks between infrastructure layers: devices, edge servers, and the cloud. Computation methods are used for this purpose. offloading, dynamic resource orchestration and intelligent load balancing [2].

Energy efficiency and network efficiency in distributed AI systems adds further relevance to this topic. Transferring large amounts of data to the cloud requires significant energy and bandwidth resources, while local processing on edge nodes minimizes network traffic and reduces operating costs [5].

Along with this, distributed learning methods are actively developing, in particular federated learning, which allows training artificial intelligence models without centralized aggregation of source data. This is especially important in the context of growing requirements for privacy and personal information protection [6].

Thus, optimization of computations in distributed AI systems based on Edge Computing is a pressing scientific and technical challenge aimed at improving the performance, scalability, and energy efficiency of intelligent infrastructures.

Architecture of distributed artificial intelligence systems using Edge Computing is built on a multi-tiered principle that ensures a balance between computing power, processing latency, and network load. The classic model includes three interconnected tiers: the device tier, the edge tier, and the cloud tier [1].

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

Device level (Device Layer) is represented by end data sources: IoT sensors, mobile devices, industrial controllers, CCTV cameras, and autonomous systems. At this level, primary data collection is performed, as well as basic pre-processing (filtering, normalization, feature extraction). In some cases, lightweight inference using compact AI models (TinyML) [3].

Peripheral level (Edge Layer) includes edge servers, gateways, and micro-data centers located in close proximity to data sources. This layer plays a key role in optimizing computing, providing: local inference of AI models; data aggregation and buffering; pre-training and retraining of models; and orchestration of computing tasks. Offloading computing to the edge significantly reduces processing latency and decreases the volume of traffic transferred to the cloud [2].

Cloud level (Cloud Layer) provides centralized data storage, large-scale model training, long-term analytics, and global orchestration of distributed infrastructure. The cloud has virtually unlimited computing resources, making it an optimal environment for training deep neural networks and managing the lifecycle of AI models [4].

An important feature of the Edge -AI architecture is the flexible distribution of tasks between layers. Latency-sensitive operations (real-time inference, streaming analytics) are performed on edge nodes, while resource-intensive training and storage processes are offloaded to the cloud. This hybrid approach ensures scalability, fault tolerance, and energy efficiency distributed AI systems [1]. Additionally, the architecture can include mechanisms for containerization, virtualization, and orchestration (e.g., edge clustering), which simplifies the deployment and management of AI services in a heterogeneous environment.

So, the multi-tiered Edge architecture Computing forms the technological foundation for optimizing computing in distributed artificial intelligence systems, ensuring efficient interaction between devices, edge nodes, and cloud infrastructure.

Optimizing Computing Processes in Distributed AI Systems Using Edge Computing aims to reduce processing latency, reduce network load, improve energy efficiency, and rationally utilize the limited resources of peripheral

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

devices. Algorithmic, architectural, and infrastructural optimization methods are used to achieve these goals [1].

One of the basic optimization mechanisms is computation Offloading is the dynamic redistribution of tasks between devices, edge nodes, and the cloud. Decisions to offload computations are made based on: data transfer latency; CPU/GPU load; device power consumption; and network bandwidth. Dynamic offloading algorithms minimize response times and improve the overall performance of distributed AI infrastructure [2].

The limited resources of edge devices require methods to reduce the computational complexity of neural network models. The most common approaches include:

- Pruning - removal of excess weight;
- Quantization - reducing the bit depth of parameters;
- Knowledge Distillation - transfer of knowledge of a compact model;
- Lightweight architectures (MobileNet, SqueezeNet).

Compression methods can significantly reduce memory and energy requirements while maintaining acceptable model accuracy [5].

Federated Learning is a distributed approach to training models, where data remains on local devices, and only weight updates are transmitted to the cloud. Advantages of this method include: data privacy protection; reduced network traffic; training scalability; and the ability to work in heterogeneous environments. This approach is particularly effective for IoT ecosystems and mobile AI applications.

Optimizing distributed AI computing is impossible without resource management systems. Containerization (Docker), edge clustering, and Kubernetes are used for this purpose at Edge; serverless approaches. Orchestration ensures automatic scaling of services, load balancing, and infrastructure fault tolerance [1].

An additional optimization mechanism is local caching and pre-filtering of data at edge nodes. This allows for: reducing the volume of transmitted information; reducing network latency; and accelerating inference processes. Filtering is particularly effective when working with streaming video data and sensor systems.

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

Table 1 - Methods for optimizing computations in Edge AI systems

Method	The essence of the approach	Main advantages	Restrictions
Computation Offloading	Migrating computing between devices , edge , and cloud	Latency reduction , load balancing	Network dependence
Model compression	Pruning , quantization , distillation	Reduced power consumption and memory	Possible loss of accuracy
Federated Learning	Distributed learning without data transfer	Privacy, traffic savings	Aggregation complexity
Resource orchestration	Containerization, Kubernetes , serverless	Scalability, fault tolerance	Management overhead
Caching and filtering	Local processing and data selection	Reducing network load	Edge memory limitations

Methods for optimizing computations in distributed Edge -AI systems are complex and cover both the algorithmic level (compression, federated Learning) and infrastructure (offloading, orchestration). Their integration ensures high performance, energy efficiency , and scalability of intelligent services.

One of the key benefits of using Edge Computing in distributed AI systems is about significantly reducing data processing latency and improving the overall performance of the computing infrastructure. In the traditional cloud model, data is transmitted to remote processing centers, which increases response times due to network latency and communication congestion. Offloading computation to edge nodes allows processing to be performed closer to the data source, minimizing transfer time and ensuring near-real -time performance. processing). Reducing latency is especially critical for latency-sensitive applications, including autonomous vehicles, industrial automation systems, telemedicine, and intelligent video surveillance. Research shows that running inference procedures on edge nodes can reduce latency severalfold compared to centralized cloud processing [3].

Further performance gains are achieved by distributing the computing load across infrastructure layers, using data caching, local flow filtering, and lightweight AI

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

models. This approach reduces network traffic and increases overall system throughput [1].

Therefore, Edge integration Computing in distributed AI architectures not only accelerates data processing but also improves service resilience, quality of service (QoS), and the scalability of intelligent applications.

Energy efficiency is a key factor in optimizing distributed AI systems, especially in the context of the widespread adoption of IoT devices and edge computing nodes. The traditional cloud model requires the constant transfer of significant volumes of data to data centers, resulting in high energy costs for network infrastructure and data processing.

Using Edge Computing significantly reduces the volume of transmitted traffic through local processing, filtering, and aggregation of data at the network edge. Only relevant or aggregated results are transmitted to the cloud, reducing the load on backbone communication channels and lowering data center energy consumption [2].

Additional resource savings are achieved through the use of lightweight AI models, compression algorithms, and dynamic computational allocation. This is especially important for autonomous and mobile devices with limited power capabilities.

Thus, the integration of Edge Computing in distributed AI architectures enables more efficient use of network and energy resources, increasing the sustainability and economic efficiency of intelligent systems.

Despite Edge's significant advantages in computing, the practical implementation of this approach to optimizing distributed AI computing is associated with a number of technological and infrastructural limitations.

One of the key challenges is the limited computing resources of edge devices. Unlike cloud data centers, edge nodes have lower CPU/GPU performance, limited memory, and power resources, complicating the execution of resource-intensive AI algorithms. Infrastructure heterogeneity remains a significant challenge. The edge environment combines devices with different architectures, operating systems, and communication protocols, complicating the deployment and orchestration of AI services. Cybersecurity poses additional risks . Edge nodes are more susceptible to physical access, network attacks, and data

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

compromise, requiring specialized security mechanisms. Complexities of scaling, managing distributed models, and ensuring data consistency during federated training also pose limiting factors. High orchestration and synchronization overhead can reduce the overall optimization effect.

Table 2 - Problems and limitations of Edge AI systems

Problem	The essence of the restriction	Impact on AI systems	Possible solutions
Limited resources	Low CPU/GPU power, memory	in inference speed	Model compression, TinyML
Heterogeneity of the environment	Different OS, protocols, architectures	Deployment complexity	Containerization, standardization
Cybersecurity	Edge node vulnerability	Risk of leaks and attacks	Encryption, Zero-Trust
Scalability	Growing number of devices	Orchestration overload	Edge clustering
Data synchronization	Distributed learning	Loss of coherence	Improved FL algorithms

So, despite the high efficiency of Edge Computing V In distributed AI systems, its implementation requires addressing resource constraints, security, standardization, and distributed infrastructure management.

The study examines approaches to optimizing computations in distributed AI systems based on Edge. Computing. It is shown that offloading data processing to edge nodes can significantly reduce latency, reduce network load, and improve energy efficiency in intelligent infrastructure. Key optimization methods, including computation, are analyzed. offloading, model compression, federated learning and resource orchestration. It is noted that Edge integration Computing forms the technological foundation for scalable and resilient real-time AI systems. However, limitations related to resource constraints on edge devices, security issues, and the complexity of managing distributed environments have been identified. Therefore, further development of optimization algorithms, lightweight models, and orchestration mechanisms will improve the efficiency of distributed AI systems and expand the practical applications of edge intelligence.

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 01, Issue 01, April, 2025
Website: usajournals.org

References

1. Shi W., Cao J., Zhang Q., Li Y., Xu L. Edge Computing: Vision and Challenges // IEEE Internet of Things Journal. – URL: https://ieeexplore.ieee.org/document/7488250
2. Mao Y., You C., Zhang J., Huang K., Letaief K. Mobile Edge Computing: Survey and Research Outlook // IEEE Communications Surveys & Tutorials. – URL: https://ieeexplore.ieee.org/document/8103676
3. Satyanarayanan M. The Emergence of Edge Computing // Computer. – URL: https://ieeexplore.ieee.org/document/7807186
4. Mach P., Becvar Z. Mobile Edge Computing: A Survey on Architecture and Computation Offloading // IEEE Communications Surveys & Tutorials. – URL: https://ieeexplore.ieee.org/document/7937606
5. Han S., Mao H., Dally W. Deep Compression: Compressing Deep Neural Networks // arXiv . – URL: https://arxiv.org/abs/1510.00149
6. McMahan B. et al. Communication-Efficient Learning of Deep Networks from Decentralized Data // arXiv . – URL: https://arxiv.org/abs/1602.05629