Journal of Integrated VLSI, Embedded and Computing Technologies, ISSN: xxxx-xxxx Vol. 1, No. 1, 2024 (pp. 16-20)

RESEARCH ARTICLE ECEJOURNALS.IN

Energy-Efficient Algorithms for Machine Learning on Embedded Systems

M. Kavitha

Department of ECE, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, India

KEYWORDS:
Energy-efficient algorithms,
Machine learning,
Embedded systems,
Optimization

ARTICLE HISTORY:
Submitted: 14.02.2024
Revised: 22.03.2024
Accepted: 24.04.2024

DOI:
https://doi.org/10.31838/JIVCT/01.01.04

Abstract

This article delves into the domain of energy-efficient algorithms specifically designed for machine learning tasks executed on embedded systems. As the need for intelligent applications expands across various sectors, integrating machine learning capabilities into devices with limited resources becomes increasingly vital. However, the energy limitations of such systems present notable obstacles. Consequently, researchers have been devising algorithms optimized to operate with minimal energy consumption. This abstract provides a concise overview of the main themes and findings explored within the paper, offering insights into the challenges, strategies, and practical implementations of energy-efficient machine learning algorithms on embedded systems.

Author's e-mail: kavithavlsime@gmail.com

How to cite this article: Kavitha M, Energy-Efficient Algorithms for Machine Learning on Embedded Systems. Journal of Integrated VLSI, Embedded and Computing Technologies, Vol. 1, No. 1, 2024 (pp. 16-20).

Introduction

The integration of machine learning (ML) algorithms has brought transformative changes across industries, empowering intelligent decision-making and task automation [1]. However, deploying these algorithms on embedded systems with limited resources poses unique challenges. These systems, found in devices like smartphones and IoT sensors, require algorithms optimized for energy efficiency to ensure optimal performance without draining power. Figure 1 shows the sensors embedded in modern smartphones [2].Thus, the development of energy-efficient ML algorithms tailored for embedded systems has become increasingly important.

Figure 1. Smartphone sensors embedded in modern smartphones

Embedded systems, known for their compact size and constrained resources, are prevalent in modern technology. Adapting traditional ML algorithms designed for high-performance computing to run efficiently on these platforms is challenging. This adaptation involves balancing algorithm complexity with resource limitations to achieve accurate results while minimizing energy consumption. As such, there is growing interest in developing specialized ML algorithms specifically for embedded systems to address these challenges.

Efficiency in energy usage is paramount in embedded systems, especially those reliant on battery power. Conventional ML algorithms often demand significant computational resources, making them unsuitable for battery-powered devices with limited processing capabilities [3]. Energy-efficient ML algorithms aim to tackle this issue by optimizing computation, memory usage, and communication overhead. Techniques like model compression, quantization, and sparsity help reduce the computational load while maintaining prediction accuracy, making them suitable for resource-constrained environments.

Furthermore, energy-efficient ML algorithms facilitate edge computing, where data processing occurs close to the data source [4]. Figure 2 shows the general overview of edge computing architecture. Embedded systems serve as intelligent endpoints in edge computing, capable of real-time analysis and decision-making without relying on cloud-based services. This decentralized approach offers benefits like enhanced privacy, reduced network congestion, and improved reliability. Energy-efficient ML algorithms enable edge devices to process data efficiently while conserving power, thereby promoting the adoption of edge computing across various applications.

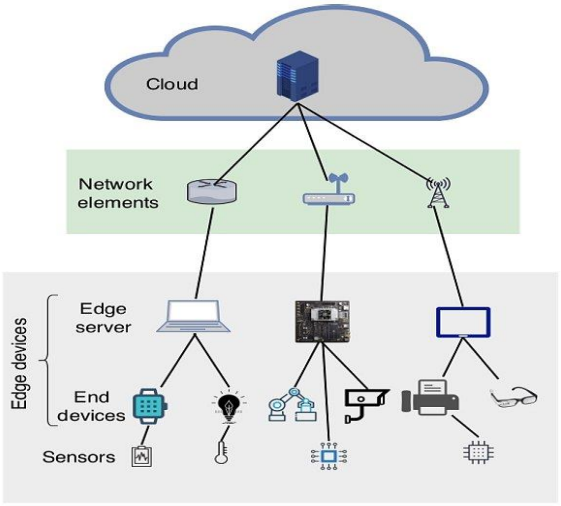

Figure 2. Overview of the edge computing architecture

Research efforts in energy-efficient ML algorithms for embedded systems focus on several key areas. Algorithm optimization explores novel techniques to reduce computational complexity and memory footprint while preserving model accuracy. Methods such as pruning and knowledge distillation have shown promise in reducing model size and computational overhead [5]. Additionally, hardware-aware optimization considers embedded platform characteristics, such as memory hierarchy and instruction set architecture, to design algorithms that utilize hardware resources efficiently.

In summary, energy-efficient ML algorithms are crucial for enabling intelligent systems on resource-constrained embedded platforms. These algorithms strike a balance between computational efficiency and prediction accuracy, making them suitable for applications where power consumption is a critical concern. As embedded systems continue to advance across various domains, the development of energy-efficient ML algorithms will play a pivotal role in driving innovation and sustainability in smart technologies.

Challenges in Implementing Machine Learning on Embedded Systems

Implementing machine learning (ML) on embedded systems presents several hurdles due to the limited resources inherent to such platforms. These systems typically have constrained processing power, memory, and energy, necessitating specialized adaptations of ML algorithms to ensure efficient operation without sacrificing performance. Overcoming these challenges requires addressing issues like computational complexity, memory constraints, power consumption, and the need for real-time processing capabilities.

One of the main challenges in embedding ML algorithms is managing their computational complexity. Traditional ML models often involve complex mathematical operations and large sizes, which can overwhelm the processing capabilities of embedded devices [6]. To tackle this, researchers focus on creating lightweight ML algorithms that prioritize simplicity and efficiency.

These algorithms employ techniques like shallow neural networks and decision trees, which require fewer computational resources while still delivering satisfactory results.

Memory limitations pose another significant obstacle for ML implementation on embedded systems. The restricted memory capacity restricts the size of ML models that can be deployed and affects both model complexity and dataset size. Additionally, the high memory bandwidth requirements of some ML algorithms can exceed the capabilities of embedded system memory architectures [7]. To address this, researchers explore methods such as model compression and quantization to reduce the memory footprint of ML models and optimize memory access.

Power consumption is a critical concern for embedded systems, especially for battery-powered devices or those operating in energy-constrained environments. ML algorithms often demand substantial computational resources, leading to increased power consumption and reduced battery life. To mitigate this, energy-efficient ML algorithms are developed, focusing on minimizing computational complexity and optimizing resource usage [8]. Techniques like sparsity and approximate computing help reduce energy consumption by minimizing unnecessary computations and exploiting data redundancy.

Real-time processing requirements present another challenge, particularly for applications where timely decision-making is essential. Traditional ML algorithms may exhibit high inference latency, making them unsuitable for real-time applications like autonomous systems and IoT devices. To address this, low-latency ML algorithms are developed, prioritizing computational efficiency and algorithmic simplicity. Techniques like model quantization and efficient memory access patterns are employed to reduce inference latency and enable real-time ML processing on embedded systems.

Energy-Efficient Algorithms for Machine Learning

Developing energy-efficient algorithms for machine learning (ML) is essential for overcoming the power limitations inherent in embedded systems and edge devices. These algorithms prioritize energy conservation while ensuring satisfactory performance, making them suitable for devices powered by batteries or operating in environments with limited power resources. Several strategies are employed to create energy-efficient ML algorithms, including model pruning, quantization, and approximation techniques.

Model pruning is a widely used method for reducing the computational complexity and memory usage of ML models. It involves eliminating redundant or unnecessary parameters and connections from the model, resulting in a more streamlined and efficient representation. By removing unimportant features and connections, pruned models require fewer computations during inference, leading to reduced energy consumption. Additionally, pruning allows ML models to be deployed on devices with restricted processing power and memory capacity.

Quantization is another effective approach for improving the energy efficiency of ML algorithms. It entails reducing the precision of model parameters and computations from floating-point to fixed-point representations [9]. By quantizing weights and activations to lower precision formats, such as 8-bit integers, the memory and computational demands of ML models are significantly decreased. This results in faster inference times and lower energy usage, making quantized models well-suited for deployment on embedded systems and edge devices.

Approximation techniques offer another avenue for achieving energy-efficient ML inference. These techniques involve replacing computationally intensive operations with simpler and faster approximations that yield acceptable levels of accuracy. For example, substituting costly matrix multiplications with low-rank approximations or utilizing piecewise-linear activation functions instead of complex nonlinear functions can substantially reduce the computational burden of ML models. While approximations may introduce some loss of accuracy, they enable significant energy savings without compromising performance in many applications.

Moreover, sparsity is a valuable characteristic that can be leveraged to enhance the energy efficiency of ML algorithms [10]. Sparse models contain a significant number of zero-valued parameters or connections, which can be exploited to reduce memory usage and computational complexity. Techniques like pruning, regularization, and sparse matrix representations are utilized to induce sparsity in ML models, resulting in more efficient inference and lower energy consumption. Sparse models are particularly advantageous for applications where memory and computational resources are limited, such as on-device AI inference and edge computing.

Optimization Techniques for Embedded Machine Learning

Optimization methods are pivotal for effectively implementing machine learning (ML) models on embedded systems that possess constrained computational capabilities. These methods concentrate on reducing the computational complexity and memory usage of ML models while maintaining their accuracy and efficacy. A variety of optimization approaches are utilized to strike this balance, encompassing model compression, hardware-aware optimization, and algorithmic enhancements.

Model compression strategies aim to shrink the size of ML models without compromising their performance. One prevalent technique is knowledge distillation, which involves training a smaller model to mimic the behavior of a larger, more intricate model [11]. This process transfers knowledge from the larger model to the smaller one, resulting in significant model

compression while preserving accuracy. Additionally, methods like weight pruning, parameter sharing, and low-rank factorization are employed to eliminate redundant parameters and connections from ML models, further reducing their size and memory requirements.

Hardware-aware optimization techniques tailor ML models to the specific hardware characteristics of embedded systems, optimizing their performance and energy efficiency [12]. These methods involve mapping ML operations to hardware accelerators, exploiting parallelism and vectorization instructions, and minimizing memory access and data movement. By leveraging hardware features such as SIMD (Single Instruction, Multiple Data) units and specialized instructions, hardware-aware optimization ensures efficient execution of ML algorithms on embedded platforms, maximizing performance while minimizing energy consumption.

Algorithmic optimizations focus on redesigning ML algorithms to enhance their efficiency and suitability for embedded deployment. These optimizations encompass algorithmic simplifications, approximation methods, and task-specific optimizations aligned with the requirements of embedded applications. For instance, replacing computationally intensive operations with simpler approximations or utilizing lightweight models with reduced complexity can notably alleviate the computational burden of ML algorithms. Moreover, task-specific optimizations adapt ML algorithms to the specific characteristics of the target application, optimizing performance and energy efficiency for real-world scenarios [13].

Furthermore, quantization and pruning techniques contribute to further optimizing ML models for deployment on embedded systems. Quantization reduces the precision of model parameters and computations, resulting in reduced memory footprint and computational demands. Pruning eliminates redundant parameters and connections from ML models, diminishing their size and complexity. By integrating these techniques with hardware-aware optimization and algorithmic enhancements, developers can craft highly efficient and compact ML models suitable for deployment on resource-limited embedded systems. These optimization strategies play a crucial role in extending the capabilities of embedded systems and facilitating the integration of intelligent algorithms into various IoT devices, wearables, and edge computing platforms.

Case Studies and Practical Implementations

Practical examples and real-world applications illustrate how energy-conscious algorithms and optimization strategies are effectively implemented for machine learning on embedded systems across different sectors. For instance, in agriculture, IoT devices equipped with embedded ML algorithms are utilized to monitor crop health and regulate irrigation schedules efficiently. By employing energy-efficient algorithms for image analysis and data processing, these devices accurately detect crop diseases and pests while consuming minimal computational resources. This optimized approach enables timely decision-making in agricultural management, leading to enhanced crop yields and resource utilization.

Similarly, healthcare benefits from wearable devices that integrate embedded ML capabilities for personalized health monitoring and disease management. Wearable biosensors, combined with energy-efficient ML algorithms, continuously monitor vital signs like heart rate and blood glucose levels. These lightweight ML models, tailored for low-power consumption, analyze physiological data in real-time, facilitating early detection of health issues and prompt interventions. By shifting computational tasks to edge devices such as smartwatches, embedded ML enhances healthcare delivery and empowers individuals to manage their well-being proactively.

In the realm of video surveillance and security systems, edge computing platforms leverage energy-efficient ML algorithms to enable intelligent monitoring and threat detection. Surveillance cameras and edge servers deploy lightweight ML models to perform real-time object detection and activity recognition while minimizing power usage. In smart cities, for instance, embedded ML algorithms analyze video feeds to detect traffic violations and manage traffic flow autonomously, reducing the reliance on centralized infrastructure and enhancing overall efficiency.

Moreover, in industrial settings, embedded ML algorithms are employed for equipment monitoring and predictive maintenance. By integrating energy-efficient ML models into edge devices embedded within machinery, manufacturers can conduct real-time condition monitoring and predict equipment failures. These embedded ML systems analyze sensor data to identify anomalies, allowing for proactive maintenance interventions to prevent downtime and production losses. This approach optimizes operational efficiency and prolongs equipment lifespan while reducing maintenance costs.

These practical implementations underscore the versatility and effectiveness of energy-conscious algorithms and optimization techniques for machine learning on embedded systems across various sectors. By embracing these approaches, organizations can harness the benefits of intelligent decision-making at the edge, leading to improved resource management, enhanced productivity, and sustainable operation across diverse applications.

Conclusion and Future Directions

In summary, the incorporation of energy-efficient algorithms and optimization strategies has significantly expanded the possibilities for integrating machine learning into various fields, transforming how data is processed and utilized at the edge. By tackling the inherent challenges of resource limitations and power constraints in embedded systems, these advancements

have facilitated the deployment of intelligent solutions across diverse sectors like agriculture, healthcare, surveillance, and industrial automation. Energy-efficient algorithms have played a crucial role in enhancing the performance of embedded machine learning models while simultaneously reducing power consumption, paving the way for smarter and more sustainable applications.

Looking forward, the trajectory of embedded machine learning shows considerable promise, with ongoing innovations expected to further refine efficiency and functionalities. One avenue for future exploration involves refining energy-efficient algorithms tailored explicitly for embedded systems, leveraging methods such as quantization, pruning, and model compression to streamline computational complexity without compromising accuracy. Moreover, advancements in hardware design, including the creation of specialized accelerators and low-power processors optimized for machine learning tasks, are poised to enhance the efficiency and performance of embedded systems even further.

Additionally, the growing prevalence of edge computing and the Internet of Things (IoT) is anticipated to propel the adoption of embedded machine learning across an even broader spectrum of applications. As the volume of data generated at the edge continues to surge, there arises a pressing need for intelligent solutions capable of processing and analyzing data locally, without the constant reliance on cloud connectivity. Embedded machine learning offers a compelling solution to this challenge, empowering real-time decision-making and actionable insights at the edge while minimizing latency and conserving bandwidth.

References

[1] Schmitt, Marc. "Automated machine learning: AI-driven decision making in business analytics." Intelligent Systems with Applications 18 (2023): 200188.

[2] Ashraf, Imran, Soojung Hur, and Yongwan Park. "Smartphone sensor based indoor positioning: Current status, opportunities, and future challenges." Electronics 9.6 (2020): 891.

[3] Jordan, Michael I., and Tom M. Mitchell. "Machine learning: Trends, perspectives, and prospects." Science 349.6245 (2015): 255-260.

[4] Satyanarayanan, Mahadev. "The emergence of edge computing." Computer 50.1 (2017): 30-39.

[5] Hinton, Geoffrey, Oriol Vinyals, and Jeffrey Dean. "Distilling the knowledge in a neural network, NIPS Deep Learning and Representation Learning Workshop." arXiv preprint arXiv:1503.02531 (2015).

[6] Han, Song, et al. "Learning both weights and connections for efficient neural network." Advances in neural information processing systems 28 (2015).

[7] Roth, Wolfgang, et al. "Resource-efficient neural networks for embedded systems." Journal of Machine Learning Research 25.50 (2024): 1-51.

[8] Fafoutis, Xenofon, et al. "Extending the battery lifetime of wearable sensors with embedded machine learning." 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). IEEE, 2018.

[9] Wu, Jiaxiang, et al. "Quantized convolutional neural networks for mobile devices." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[10] Hoefler, Torsten, et al. "Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks." Journal of Machine Learning Research 22.241 (2021): 1-124.

[11] Han, Song, Huizi Mao, and William J. Dally. "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding." arXiv preprint arXiv:1510.00149 (2015).

[12] Dave, Shail, et al. "Hardware acceleration of sparse and irregular tensor computations of ml models: A survey and insights." Proceedings of the IEEE 109.10 (2021): 1706-1752.

[13] Sarker, Iqbal H. "Machine learning: Algorithms, real-world applications and research directions." SN computer science 2.3 (2021): 160.