10.48047/jocaaa.2025.34.11.53

Cost-informed Model Choice for On-device AI Applications

Reeshav Kumar
Independent Researcher, USA

Abstract

On-device AI deployment presents a set of unique challenges that are fundamentally different from those of cloud-based systems. These challenges necessitate specialized optimization approaches. The deployment must strike a balance between user experience and stringent resource constraints, including privacy preservation, energy efficiency, thermal management, and memory limitations. This article introduces a cost-informed decision framework for selecting and optimizing AI models for mobile and edge computing environments. We use a systematic scorecard methodology to evaluate models across six critical dimensions: task complexity scoring, compute resource utilization, energy budget analysis, memory footprint efficiency, latency performance, and throughput scalability. The framework addresses the fundamental challenge of determining optimal model architectures and configurations to deliver maximum user value per unit of energy while operating on resource-limited devices. Through practical case studies spanning wake-word detection, keyboard prediction, assistive vision, and on-device co-pilot applications, the article demonstrates how different optimization strategies, including neural network quantization, knowledge distillation, early exit mechanisms, and specialized hardware acceleration, can be systematically applied to achieve deployment viability across diverse computational and energy constraints. The implementation strategies encompass model architecture optimization, dynamic resource allocation, memory and storage optimization, hardware acceleration integration, energy management, and quality assurance techniques that collectively enable sustainable on-device AI deployment while maintaining optimal task performance and user experience quality across varying operational conditions.

Keywords: On-Device AI, Neural Network Quantization, Mobile AI Optimization, Energy-Efficient Inference, Edge Computing Constraints

[I] Introduction

The proliferation of AI capabilities on mobile and edge devices has fundamentally transformed user expectations for intelligent, responsive applications. Unlike cloud-based AI systems, which can leverage a large pool of computational resources, on-device AI operates within stringent physical and economic constraints that directly affect model selection decisions. The challenge extends beyond simply fitting models within available memory or compute budgets. On-device deployment for real-time applications necessitates optimizing for the intricate interplay among user experience, energy efficiency, thermal management, and privacy preservation. For example, the MobileNets architecture demonstrates this challenge through its innovative use of depthwise separable convolutions, which reduces computational cost by a factor of 9X compared to standard convolutions while maintaining comparable accuracy for mobile vision applications [1].

Traditional model selection approaches that prioritize accuracy metrics alone are insufficient for on-device deployment scenarios. The MobileNets framework addresses this limitation by introducing a width multiplier parameter that enables models to trade accuracy for latency, achieving up to a 4.2% reduction in accuracy while delivering a 2.6X speedup on ARM processors [1]. The unique constraints of mobile

10.48047/jocaaa.2025.34.11.53

hardware necessitate a holistic evaluation framework that considers the total cost of ownership for AI models throughout their operational lifecycle. Pruning and quantization techniques have emerged as critical optimization strategies, with magnitude-based pruning capable of achieving 90% sparsity while maintaining model performance within acceptable bounds [2]. On-device inference cost encompasses not only computational resources but also battery drain, thermal impact, and the opportunity cost of allocating limited device resources to specific AI tasks. This complexity underscores the need for a comprehensive evaluation framework.

Modern neural network optimization demonstrates significant potential for resource efficiency through systematic compression approaches. Quantization methods can reduce model size by 75% through the use of 8-bit integer representations, while maintaining competitive accuracy across various deep learning tasks [2]. These compression techniques directly address the fundamental constraint of mobile deployment, where storage limitations and memory bandwidth restrictions create bottlenecks for model execution. The economic implications extend beyond immediate computational costs to include the broader system-level impact of AI workloads on device performance and user experience quality.

Our cost-informed model choice framework for on-device deployment addresses this fundamental question: given the constraints of a target device and the requirements of a specific AI task, what model architecture and configuration will deliver optimal user value per unit of energy consumed? We present a systematic scorecard methodology that evaluates models across six critical dimensions, incorporating insights from efficient architectures, such as MobileNets, and optimization techniques, including pruning and quantization. The framework can be applied across diverse on-device AI workloads, with a key focus on maintaining the essential balance between computational efficiency and task performance requirements. This balance is crucial for ensuring that user experience is not compromised while the system remains sustainable.

++

[II] On-device AI Constraints and Requirements

On-device AI deployment introduces a unique set of constraints that differ significantly from those of cloud-based or server-side AI systems. Understanding these constraints is crucial for developing effective model selection frameworks that strike a balance between user experience and system sustainability. The deployment of efficient neural networks on mobile platforms requires careful consideration of computational overhead and model optimization techniques that can maintain performance while reducing resource requirements. These fundamental differences necessitate specialized optimization approaches that address the multi-dimensional constraint space of mobile computing environments, where traditional cloud-based optimization strategies prove inadequate for resource-constrained scenarios. Our framework is designed to maintain this balance, ensuring that user experience is not compromised while the system remains sustainable.

Privacy and data locality requirements are a primary driver for on-device AI deployment, eliminating the need to transmit sensitive user data to external servers and addressing growing concerns about privacy and regulatory requirements. This constraint mandates that models be sufficiently capable of performing inference locally while maintaining acceptable accuracy levels. Neural network quantization emerges as a critical technique in this context, with 8-bit quantization capable of reducing model size by approximately four times while maintaining inference performance within 1% of that of full-precision models across various deep learning architectures [4]. The privacy benefit comes at the cost of reduced model

10.48047/jocaaa.2025.34.11.53

complexity, as devices cannot leverage large-scale distributed computing resources or real-time model updates from cloud services. Modern quantization approaches demonstrate particular effectiveness in addressing these constraints, with post-training quantization methods enabling deployment of complex models on resource-limited devices without significant accuracy degradation.

Offline availability and network independence create additional architectural constraints that significantly influence model design decisions. Unlike cloud-based systems, on-device AI must function reliably in scenarios with limited or no network connectivity, requiring self-contained models that incorporate all necessary knowledge and processing capabilities locally. This requirement influences model architecture decisions, favoring approaches that can deliver robust performance without relying on external dependencies. Effective data augmentation techniques become crucial for improving model generalization in offline scenarios, as augmentation strategies achieve accuracy improvements of 5-15% across various computer vision tasks, while reducing overfitting tendencies that could compromise offline performance [3]. The offline constraint also influences the choice between specialized single-task models versus more general-purpose architectures that must handle diverse scenarios without external knowledge bases.

Thermal management and sustained performance constraints create dynamic operational challenges that directly impact the execution characteristics of models. Mobile devices implement aggressive thermal throttling to prevent overheating, which can dramatically reduce available computational resources during extended AI workloads. Models must be designed to maintain acceptable performance even when processing units are operating at reduced frequencies, requiring architectural choices that gracefully degrade under resource constraints. Quantization techniques offer substantial benefits in thermal management scenarios, as reduced-precision arithmetic operations generate significantly less heat while maintaining computational throughput, thereby enabling sustained inference performance under thermal constraints [4]. This constraint particularly affects sustained inference tasks, such as real-time language processing or continuous vision applications, where thermal buildup can lead to significant performance degradation over time.

Energy budget and battery life impact considerations represent critical constraints that influence every aspect of on-device AI design decisions. Battery capacity remains a fundamental limitation for mobile devices, making energy efficiency a primary concern for on-device AI applications. The energy cost of AI inference directly competes with other device functions, necessitating careful optimization to prevent a negative impact on overall user experience. Quantization techniques offer substantial energy benefits by reducing memory bandwidth requirements and simplifying arithmetic operations, with 8-bit integer operations consuming approximately 25% of the energy required for equivalent 32-bit floating-point computations [4]. Memory hierarchy and storage constraints create complex optimization challenges that affect both model architecture and deployment strategies. Compressed models enable faster loading times and reduced memory footprint requirements, which are essential for multi-application scenarios.

10.48047/jocaaa.2025.34.11.53

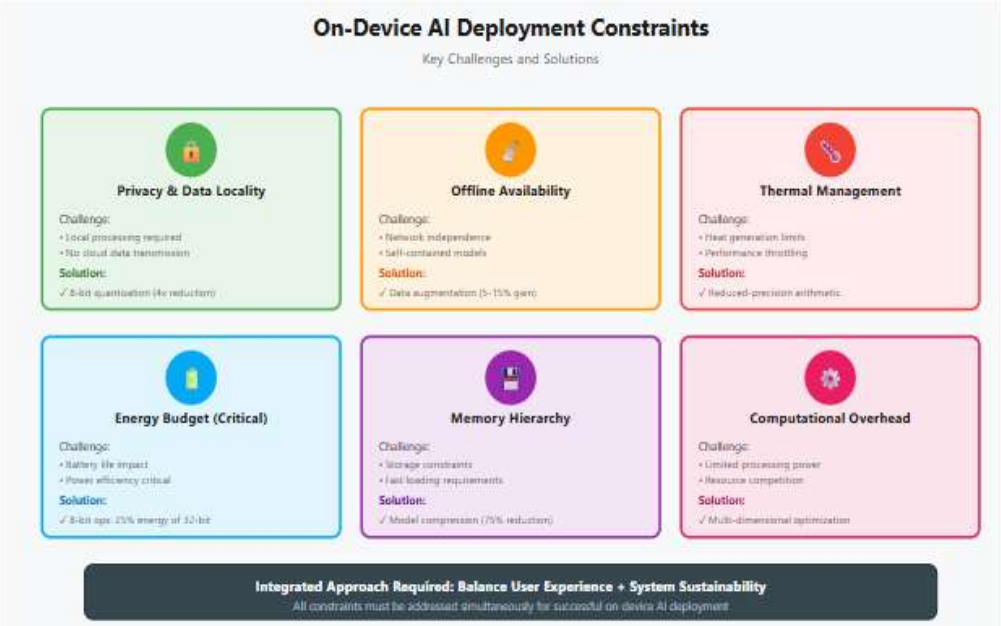

Fig 1: On-Device AI deployment constraints

Constraint Type	Primary Challenge	Key Requirements	Optimization Strategy	Impact Level
Privacy & Data Locality	Local Processing	Maintain accuracy without cloud	Neural Network Quantization	High
Offline Availability	Network Independence	Self-contained models	Data Augmentation	Medium
Thermal Management	Heat Generation	Sustained performance	Reduced-precision arithmetic	High
Energy Budget	Battery Life	Power efficiency	8-bit integer operations	Critical
Memory Hierarchy	Storage Constraints	Fast loading & reduced footprint	Model compression	High
Computational Overhead	Resource Limitations	Balanced performance	Multi-dimensional optimization	Critical

Table 1: On-device AI Constraint Categories and Optimization Strategies [3, 4]

[III] Multi-dimensional Scorecard Framework

The scorecard framework evaluates on-device AI models across six critical dimensions that collectively determine their suitability for specific deployment scenarios. Each dimension captures a fundamental aspect of on-device performance and resource utilization, enabling systematic comparison and optimization of model choices. The framework addresses the basic challenge of optimizing deep neural networks for resource-constrained environments, where traditional accuracy-focused evaluation metrics prove insufficient for real-world deployment decisions. Contemporary mobile AI deployment necessitates sophisticated evaluation methodologies that consider the intricate interplay between computational

10.48047/jocaaa.2025.34.11.53

efficiency, energy consumption, and user experience quality across various hardware configurations and usage scenarios.

Task complexity scoring represents the foundational dimension that evaluates both the inherent difficulty of the AI task and the quality requirements for acceptable user experience. This dimension assesses whether a model's capability level appropriately matches the complexity of the target application, with quantization techniques demonstrating remarkable effectiveness in maintaining task performance while reducing computational requirements. Recent advances in neural network quantization have demonstrated that 8-bit quantization can achieve model size reductions of up to 75% while maintaining accuracy degradation of less than 2-3% for most computer vision and natural language processing tasks [5]. For simple tasks, such as wake-word detection, lightweight models may provide adequate performance. In contrast, complex tasks like real-time language translation require more sophisticated architectures that can benefit from advanced quantization strategies. The scoring methodology incorporates accuracy requirements, robustness to input variations, and the consequences of model errors on user experience. Mixed-precision quantization approaches enable fine-grained optimization that balances accuracy preservation with resource efficiency.

Compute resource utilization evaluation focuses on how efficiently models utilize available processing resources across the heterogeneous computing environment of modern devices. The assessment considers FLOPS requirements, memory bandwidth utilization, and the ability to leverage specialized hardware accelerators, with quantization providing substantial computational efficiency improvements through reduced-precision arithmetic operations. Models are scored based on their computational density and their ability to maintain consistent performance across different hardware configurations and power states. Neural network quantization techniques enable significant reductions in computational overhead, with integer-only inference achieving a 3- 5x speedup compared to floating-point implementations while consuming substantially less energy per operation [6]. The framework particularly emphasizes the importance of hardware-aware optimization strategies that can effectively leverage specialized processing units, with quantized models demonstrating superior compatibility with mobile neural processing units and edge computing accelerators.

Energy budget analysis encompasses both immediate power consumption during inference and the cumulative impact on battery life over typical usage patterns. This evaluation examines dynamic power consumption across various processing units, idle power overhead for maintaining models in memory, and the energy cost associated with model loading and initialization. The framework accounts for the non-linear relationship between computational intensity and power consumption, particularly relevant for thermally constrained scenarios where sustained AI workloads can significantly impact device performance. Quantization techniques provide substantial energy efficiency benefits, with 8-bit quantized models demonstrating 4-8x reduction in energy consumption compared to 32-bit floating-point implementations across various neural network architectures [5]. Advanced quantization strategies enable sustainable on-device AI deployment by minimizing the energy cost per inference while maintaining acceptable task performance levels.

Memory footprint and loading efficiency scoring evaluate both static model size and dynamic memory usage during inference, with quantization enabling dramatic storage and bandwidth reductions essential for mobile deployment scenarios. Latency and real-time performance evaluation extend beyond simple inference time to consider complete response pipelines, with quantized models achieving inference speedups of up to four times that of full-precision implementations [6]. Throughput and scalability assessment consider concurrent processing capabilities. At the same time, composite scoring

10.48047/jocaaa.2025.34.11.53

methodology combines individual dimension scores using a weighted aggregation that reflects specific application priorities. It enables systematic optimization decisions across the multi-dimensional constraint space.

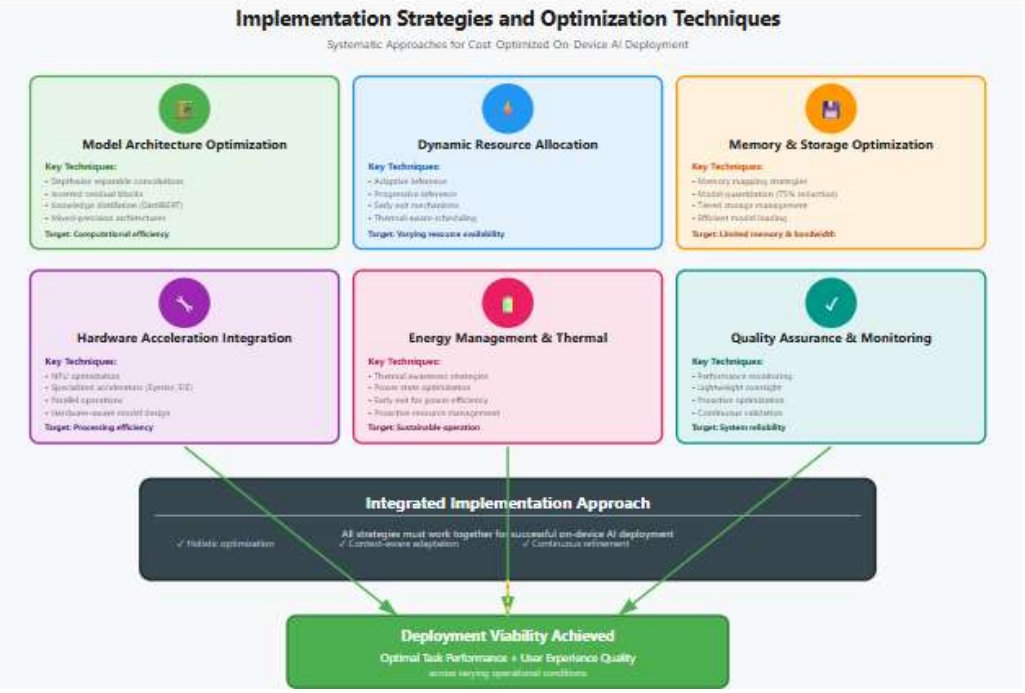

Table 2: Implementation Strategies and Optimization Techniques

Dimension Category	Primary Focus	Key Assessment Criteria	Optimization Technique	Expected Performance Range
Task Complexity Scoring	Capability Matching	Accuracy, Robustness, Error Impact	Mixed-precision Quantization	2-3% accuracy degradation
Compute Resource Utilization	Processing Efficiency	FLOPS, Memory Bandwidth, Hardware Acceleration	Integer-only Inference	3-5x speedup
Energy Budget Analysis	Power Consumption	Dynamic Power, Idle Overhead, Loading Cost	8-bit Quantization	4-8x energy reduction
Memory Footprint	Storage Efficiency	Static Size, Dynamic Usage, Loading Time	Model Compression	Up to 75% size reduction
Latency Performance	Response Time	Inference Time, Pipeline Overhead	Quantized Operations	2-4x speedup
Throughput & Scalability	Concurrent Processing	Multi-task Handling, Resource Sharing	Optimized Architectures	Enhanced scalability

Table 2: Scorecard Framework Evaluation Dimensions and Optimization Strategies [5, 6]

10.48047/jocaaa.2025.34.11.53

[IV] Application Case Studies

This section demonstrates the practical application of our scorecard framework across four representative on-device AI workloads, each presenting distinct constraint profiles and optimization priorities that illustrate the framework's versatility and effectiveness. The case studies emphasize the crucial importance of tailored optimization strategies for various application domains, with energy-efficient accelerator architectures and compressed neural networks proving essential for achieving deployment viability across diverse computational and energy constraints. Each application scenario presents unique trade-offs among accuracy, latency, energy consumption, and memory utilization, necessitating specialized model selection and optimization approaches.

Wake-word detection systems must operate continuously with minimal battery impact while maintaining high accuracy for activation phrases and low false positive rates for background audio. Our scorecard analysis reveals that task complexity requirements are moderate, focusing on temporal pattern recognition in audio signals rather than complex semantic understanding. Energy-efficient accelerator architectures, such as Eyeriss, demonstrate remarkable power efficiency for convolutional operations, achieving 67.2 GOP/s/W with a power consumption of 278 mW, making them highly suitable for always-on audio processing applications [7]. Energy budget considerations dominate the scoring, as wake-word detection systems must operate continuously without significantly impacting battery life. The analysis reveals that specialized compact models with dataflow optimizations provide optimal value, with Eyeriss-style architectures enabling continuous operation through minimized memory access costs and optimized energy consumption patterns, which are essential for battery-powered wake-word detection systems.

Keyboard prediction applications require real-time text generation with minimal latency to maintain natural typing flow while adapting to individual user patterns without compromising privacy. The scorecard framework reveals that task complexity is moderate but requires personalization capabilities that benefit from efficient inference architectures. Latency requirements are critical, with acceptable performance requiring sub-100ms response times even under thermal throttling conditions. Sparse neural network architectures demonstrate significant advantages for text prediction tasks, with the EIE accelerator achieving 102 GOP/s/W efficiency through the exploitation of weight sparsity and compressed storage formats that reduce memory bandwidth requirements [8]. Memory footprint considerations favor models that can efficiently cache user-specific patterns while maintaining reasonable startup times. Compressed neural networks enable faster loading and reduce memory access overhead, which is essential for responsive keyboard prediction systems.

Assistive vision applications, such as real-time scene description or object recognition for accessibility purposes, present significant computational challenges while requiring high accuracy for user safety and utility. Task complexity scoring reflects the need for robust visual understanding across diverse environmental conditions and lighting scenarios. The Eyeriss architecture demonstrates exceptional suitability for convolutional neural networks used in vision applications, delivering 67.2 GOP/s/W energy efficiency through optimized dataflow that minimizes energy consumption per operation [7]. Compute resource utilization becomes a primary bottleneck, with specialized accelerators enabling efficient use of processing resources while maintaining consistent performance under thermal constraints. The energy budget impact is substantial due to continuous camera operation and intensive image processing requirements, requiring architectures that can sustain high throughput with minimal power consumption.

On-device copilot applications integrate natural language understanding, context awareness, and task execution capabilities while maintaining privacy and offline functionality. These applications present the most complex scorecard profile, with high task complexity requirements spanning multiple AI domains.

10.48047/jocaaa.2025.34.11.53

The EIE architecture's support for compressed sparse neural networks provides significant advantages for copilot applications, achieving 189x and 13x improvements in energy efficiency and area efficiency, respectively, compared to conventional processors [8]. The memory footprint becomes critical as copilot systems must maintain multiple specialized models. Compressed neural networks enable the deployment of sophisticated AI capabilities within resource constraints through efficient sparse matrix operations and optimized memory access patterns.

Application Type	Task Complexity	Primary Constraint	Latency Requirement	Energy Priority	Memory Optimization Strategy
Wake-word Detection	Moderate	Energy Budget	Continuous operation	Critical	Dataflow optimization
Keyboard Prediction	Moderate	Latency	Sub-100ms response	High	Compressed storage
Assistive Vision	High	Compute Resource	Real-time processing	Substantial	Optimized dataflow
On-device Copilots	Very High	Memory Footprint	Variable workload	Critical	Sparse matrix operations

Table 3: Application Case Studies Constraint Profiles and Optimization Priorities [7, 8]

[V] Implementation Strategies and Optimization Techniques

The successful deployment of cost-optimized on-device AI models requires systematic implementation approaches that address the multidimensional constraints identified in our scorecard framework. This section presents practical strategies for optimizing model performance across the identified dimensions while maintaining an acceptable level of user experience quality. The implementation challenges of on-device AI necessitate sophisticated optimization techniques that can deliver efficient inference while operating within the stringent resource constraints of mobile and edge computing environments.

Model architecture optimization represents the foundation of efficient on-device AI deployment, requiring a careful balance between representational capacity and computational efficiency. Depth-wise separable convolutions and inverted residual blocks have proven particularly effective for vision tasks, reducing parameter counts while maintaining high-quality feature extraction. For natural language processing tasks, knowledge distillation techniques can compress large transformer models into smaller variants that retain substantial performance while dramatically reducing computational requirements. DistilBERT architectures demonstrate exceptional effectiveness for text analysis applications, achieving remarkable compression ratios while maintaining high accuracy levels for various natural language processing tasks, including text classification and generation detection [9]. Mixed-precision architectures enable fine-grained optimization strategies that use higher precision for critical computations while leveraging quantized operations for routine processing tasks, achieving optimal trade-offs between accuracy and efficiency for diverse on-device applications.

Dynamic resource allocation strategies enable on-device AI systems to adapt intelligently to varying resource availability, which can be influenced by thermal throttling, concurrent application usage, and changing power states. Adaptive inference techniques allow models to dynamically adjust computational intensity based on available resources, potentially reducing model complexity during resource-

10.48047/jocaaa.2025.34.11.53

constrained periods while maintaining acceptable performance levels. Progressive inference approaches provide preliminary results quickly, while continuing to refine them when resources permit, thereby improving perceived responsiveness for interactive applications. Early exit mechanisms in deep neural networks demonstrate significant potential for computational efficiency, with network architectures achieving substantial inference speedup through strategic placement of intermediate decision points that enable early termination when confidence thresholds are met [10]. These dynamic optimization strategies prove essential for maintaining a consistent user experience across varying operational conditions.

Memory and storage optimization techniques address the critical constraints of limited device memory and storage bandwidth through sophisticated model management strategies. Efficient memory utilization requires careful consideration of model loading strategies, particularly for applications that switch between multiple specialized models during operation. Memory mapping techniques can reduce loading times by maintaining frequently accessed model components in faster storage tiers while streaming less critical parameters as needed. Model quantization beyond inference optimization can significantly reduce storage requirements, with post-training quantization techniques achieving substantial compression ratios with minimal accuracy degradation for many on-device applications.

Hardware acceleration integration strategies leverage the diverse processing capabilities of modern mobile devices through specialized optimization approaches tailored to different processing units. Neural processing units typically excel at highly parallel operations with consistent data patterns, making them ideal for convolutional operations and linear transformations that benefit from specialized acceleration. Energy management and thermal awareness strategies ensure sustainable on-device AI operation through proactive resource management that considers both immediate power consumption and long-term thermal impacts. Early exit mechanisms provide additional benefits for power efficiency by reducing unnecessary computations [10]. Quality assurance and performance monitoring systems provide essential oversight mechanisms for maintaining optimal system performance, with lightweight monitoring enabling proactive optimization without significantly impacting system resources.

Strategy Category	Primary Focus	Key Techniques	Target Constraint	Implementation Complexity
Model Architecture Optimization	Computational Efficiency	DistilBERT, Mixed-precision	Model Size & Accuracy	High
Dynamic Resource Allocation	Adaptive Performance	Early Exit, Progressive Inference	Thermal & Power States	Medium
Memory & Storage Optimization	Resource Management	Memory Mapping, Quantization	Storage & Bandwidth	Medium
Hardware Acceleration Integration	Processing Efficiency	NPU Optimization, Parallel Operations	Compute Resources	High

10.48047/jocaaa.2025.34.11.53

Energy Management	Power Sustainability	Thermal Awareness, Early Exit	Battery Life	Medium
Quality Assurance	Performance Monitoring	Lightweight Monitoring	System Reliability	Low

Table 4: Implementation Strategy Categories and Optimization Focus Areas [9, 10]

[VI] Conclusion

This article presents a comprehensive framework for cost-informed model selection in on-device AI applications, successfully addressing the complex trade-offs between computational efficiency, energy consumption, and user experience quality inherent in resource-constrained mobile computing environments. The multi-dimensional scorecard article provides systematic evaluation capabilities that enable quantitative comparison and optimization of model choices across diverse deployment scenarios, with validation through representative case studies demonstrating the framework's practical applicability across wake-word detection, keyboard prediction, assistive vision, and copilot applications. The implementation strategies emphasize the essential role of integrated, holistic optimization methods that combine model architecture tuning, dynamic resource management, dedicated hardware acceleration, and proactive energy control to provide sustainable AI functionality within the limitations of mobile devices. The findings of the research determine that on-device AI deployment should be accomplished through advanced optimization methods such as neural network quantization, knowledge distillation, early exit strategies, and hardware-conscious optimization techniques that together facilitate the deployment of intricate AI functionalities without violating inherent boundaries on battery capacity, thermal control, memory hierarchy, and computational resources. The cost-aware model selection is a significant contribution to making AI functionality accessible on the mobile domain for everyone, ensuring that user privacy, system performance, and sustainable battery life are not compromised. This approach offers key guidance to researchers and practitioners designing future intelligent mobile applications that are resource-efficient within the tight constraints of edge computing.

10.48047/jocaaa.2025.34.11.53

References

[1] Andrew G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," ResearchGate, April 2017. https://www.researchgate.net/publication/316184205_MobileNets_Efficient_Convolutional_Neural_Networks_for_Mobile_Vision_Applications

[2] Tailin Liang et al., "Pruning and Quantization for Deep Neural Network Acceleration: A Survey," ResearchGate, January 2021. https://www.researchgate.net/publication/348757169_Pruning_and_Quantization_for_Deep_Neural_Network_Acceleration_A_Survey

[3] Prem Enkvetchakul & Olarik Surinta, "Effective Data Augmentation and Training Techniques for Improving Deep Learning in Plant Leaf Disease Recognition," ResearchGate, June 2021. https://www.researchgate.net/publication/346607345_Effective_Data_Augmentation_and_Training_Techniques_for_Improving_Deep_Learning_in_Plant_Leaf_Disease_Recognition

[4] Markus Nagel et al., "A White Paper on Neural Network Quantization," ResearchGate, June 2021. https://www.researchgate.net/publication/352425391_A_White_Paper_on_Neural_Network_Quantization

[5] Lu Wei et al., "Advances in the Neural Network Quantization: A Comprehensive Review," ResearchGate, August 2024. https://www.researchgate.net/publication/383437678_Advances_in_the_Neural_Network_Quantization_A_Comprehensive_Review

[6] Olivia Weng et al., "Neural Network Quantization for Efficient Inference: A Survey," ResearchGate, December 2021. https://www.researchgate.net/publication/357014029_Neural_Network_Quantization_for_Efficient_Inference_A_Survey

[7] Yu-Hsin Chen et al., "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," ResearchGate, February 2016. https://www.researchgate.net/publication/292869497_Eyeriss_An_Energy-Efficient_Reconfigurable_Accelerator_for_Deep_Convolutional_Neural_Networks

[8] Song Han et al., "Retrospective: EIE: Efficient Inference Engine on Sparse and Compressed Neural Network," ResearchGate, June 2023. https://www.researchgate.net/publication/371684585_Retrospective_EIE_Efficient_Inference_Engine_on_Sparse_and_Compressed_Neural_Network

[9] Pranay Kumar BV et al., "DistilBERT: A Novel Approach to Detect Text Generated by Large Language Models (LLM)," ResearchGate, January 2024. https://www.researchgate.net/publication/377909511_DistilBERT_A_Novel_Approach_to_Detect_Text_Generated_by_Large_Language_Models_LLM

[10] Joao Simioni et al., "An Early Exit Deep Neural Network for Fast Inference Intrusion Detection," ResearchGate, May 2025. https://www.researchgate.net/publication/391753698_An_Early_Exit_Deep_Neural_Network_for_Fast_Inference_Intrusion_Detection