Machine Learning and Applied Artificial Intelligence

https://mlai-journal.com

Causal Modeling of Inference Latency with Model Compression Strategies in Mobile Health Applications

Larissa Cunha

Department of Communications, School of Electrical and Computer Engineering, Universidade Estadual de Campinas (UNICAMP), Campinas, São Paulo, Brazil

Corresponding author: larissa.cunha@gmail.com

Abstract

The integration of deep learning models into mobile health applications has revolutionized continuous monitoring and point-of-care diagnostics. However, deploying complex neural networks on resource-constrained mobile edge devices introduces significant inference latency, which can be detrimental in time-critical medical scenarios. While model compression strategies such as quantization, pruning, and knowledge distillation are routinely employed to mitigate latency, standard benchmarking methodologies often fail to capture the true latency reduction due to environmental confounders like thermal throttling, battery degradation, and dynamic background operating system workloads. This paper introduces a comprehensive causal modeling framework to accurately assess the inference latency of various model compression strategies in mobile health applications. By conceptualizing model compression as a causal treatment and inference latency as the outcome, we construct a structural causal model to identify and adjust for systemic confounders using inverse probability weighting and propensity score matching. Our methodology is validated through extensive experiments on commercial mobile hardware using medical datasets, including electrocardiogram classification and dermatological image analysis. The causal analysis reveals that raw observational data significantly overestimates the latency benefits of unstructured pruning while underestimating the efficacy of integer quantization due to hidden thermal interactions. These findings provide a robust paradigm for mobile health developers to evaluate compression trade-offs, ensuring reliable and rapid artificial intelligence deployment in critical healthcare environments.

Keywords: Mobile Health; Causal Inference; Model Compression; Inference Latency; Edge Computing

1. Introduction

The rapid proliferation of mobile health technologies has fundamentally transformed the landscape of modern healthcare delivery. By leveraging the ubiquitous presence of smartphones, wearable sensors, and portable diagnostic equipment, healthcare providers can now monitor patient vitals, detect physiological anomalies, and administer proactive interventions outside the confines of traditional clinical settings [1]. Central to this technological paradigm shift is the integration of advanced artificial intelligence, particularly deep learning algorithms, directly into mobile edge applications. These predictive models enable real-time analysis of complex biological signals, ranging from continuous glucose monitoring to ambulatory electrocardiogram assessments [2]. However, the computational demands of deep neural networks pose a formidable challenge when deployed on edge devices, which are inherently constrained by limited computational throughput, restricted memory bandwidth, and strict energy consumption budgets [3].

The primary bottleneck inhibiting the seamless operation of deep learning in mobile health applications is inference latency. Inference latency refers to the temporal delay between the acquisition of raw physiological data and the generation of an actionable clinical prediction by the neural network [4]. In acute medical scenarios, such as the detection of a transient ischemic attack, real-time seizure forecasting, or the identification of a malignant cardiac arrhythmia, minimizing inference latency is not merely a matter of user experience but a critical determinant of patient outcomes [5]. Prolonged processing times can render time-sensitive interventions ineffective, thereby negating the intrinsic value of mobile point-of-care diagnostics.

Consequently, researchers and software engineers have increasingly turned to model compression strategies to accelerate network execution and reduce the memory footprint of these diagnostic models.

Model compression encompasses a suite of algorithmic techniques designed to simplify the architecture or precision of a neural network without incurring a disproportionate loss in predictive accuracy. The most prominent strategies include parameter quantization, network pruning, and knowledge distillation. Quantization reduces the numerical precision of the weights and activations, typically transitioning from thirty-two-bit floating-point representations to eight-bit integers. Network pruning systematically eliminates redundant or non-contributory connections within the neural architecture, resulting in a sparser network topology. Knowledge distillation involves training a compact student network to replicate the functional behavior of a cumbersome, highly parameterized teacher network. While these strategies have demonstrated remarkable theoretical efficacy in academic benchmarks, their practical implementation in mobile health environments often yields highly variable and unpredictable latency reductions.

The fundamental problem with contemporary assessments of inference latency lies in the reliance on purely observational benchmarking methodologies. Conventional evaluation frameworks measure the execution time of compressed models in isolated, highly controlled environments, or they simply average the latency across numerous iterations on a target device. Such approaches fail to account for the dynamic, highly confounded nature of mobile computing ecosystems. Mobile devices operate under operating systems that aggressively manage power and thermal output through dynamic voltage and frequency scaling. When a computationally intensive diagnostic model executes, it generates substantial thermal output, which triggers the operating system to throttle the central processing unit frequency to prevent hardware damage. Furthermore, the concurrent execution of background applications, fluctuating battery charge levels, and varying ambient temperatures introduce significant noise into latency measurements.

To overcome the limitations of observational benchmarking, this paper proposes the application of causal modeling to assess inference latency. Causal inference frameworks, grounded in structural causal models, provide a mathematically rigorous methodology for isolating the true effect of a treatment on an outcome by controlling for identified confounding variables. In the context of this research, the application of a specific model compression strategy is conceptualized as the causal treatment, and the resulting inference latency serves as the primary outcome variable. Confounders represent the myriad hardware and software states, such as thermal throttling status and background memory utilization, that simultaneously influence both the execution dynamics of the compressed model and the measured latency. By systematically adjusting for these confounders, we can calculate the average treatment effect of each compression technique, thereby providing a more accurate and reliable assessment of its practical utility in mobile health applications.

This study makes several critical contributions to the fields of mobile edge computing and health informatics. First, we define a comprehensive structural causal model tailored specifically for evaluating artificial intelligence inference on mobile architectures. Second, we present empirical evidence demonstrating the severe discrepancies between raw observational latency measurements and causally adjusted execution times across various mobile devices. Third, we provide a detailed analysis of how specific compression strategies interact with hardware-level confounding variables, such as the phenomenon where unstructured pruning exacerbates cache misses and subsequent thermal escalation. Ultimately, this research establishes a robust evaluation paradigm that empowers developers to make informed, evidence-based decisions when optimizing deep learning models for life-critical mobile health deployments.

2. Literature Review and Theoretical Background

2.1 Evolution and Requirements of Mobile Health Applications

Mobile health represents a critical intersection between telecommunications technology and medical informatics, aimed at democratizing access to continuous healthcare monitoring [6]. Historically, mobile health initiatives were largely restricted to rudimentary text messaging campaigns for medication adherence or simple pedometer tracking. However, the advent of high-fidelity biometric sensors embedded within consumer smartphones and smartwatches has catalyzed a shift toward sophisticated, continuous diagnostic systems. Contemporary applications are capable of capturing high-resolution medical imagery, multi-channel electrocardiograms, and complex photoplethysmography signals.

The integration of artificial intelligence into these applications is driven by the necessity to process massive volumes of physiological data locally. Transmitting continuous biological streams to centralized cloud

servers for analysis introduces unacceptable network latencies, severely compromises patient data privacy, and relies heavily on the availability of robust cellular connectivity [7]. Therefore, on-device artificial intelligence has become a mandatory architectural requirement for modern mobile health systems. The primary operational constraint for these embedded models is strict adherence to real-time latency thresholds, especially for conditions that precipitate sudden physiological deterioration, where every millisecond of computational delay significantly elevates patient risk.

2.2 Deep Learning Workloads on Mobile Edge Architectures

Executing complex convolutional neural networks and recurrent neural networks on mobile edge devices involves navigating a highly constrained hardware environment. Unlike enterprise-grade graphics processing units utilized in data centers, mobile system-on-chip architectures must balance computational performance with severe thermal and energetic limitations [8]. A typical mobile processor features a heterogeneous multi-core design, often adopting a big-little architecture where high-performance, power-hungry cores are paired with slower, energy-efficient cores.

When a mobile health application initiates an inference request, the operating system's scheduler must allocate the workload across these heterogeneous processing units. Deep learning workloads are characterized by intense matrix multiplication operations that rapidly consume memory bandwidth and generate substantial thermal energy. If the computational demand sustains high utilization of the performance cores, the device temperature rises, prompting the system's thermal management daemon to aggressively down-clock the processor [9]. This dynamic voltage and frequency scaling is a primary mechanism for device preservation, yet it introduces profound variability into inference latency, complicating the evaluation of any algorithmic optimizations applied to the neural network.

2.3 Comprehensive Overview of Model Compression Strategies

To mitigate the computational burden of deep learning on mobile hardware, researchers have developed various model compression strategies, each targeting different aspects of the neural network's architecture and data representation. Parameter quantization is perhaps the most ubiquitous technique, primarily because modern mobile processors feature dedicated hardware instruction sets for lower-precision arithmetic [10]. By mapping thirty-two-bit floating-point weights and activations to eight-bit integers, quantization drastically reduces the memory footprint of the model and accelerates data transfer from main memory to the processor caches. Post-training quantization allows for rapid deployment, whereas quantization-aware training fine-tunes the model to recover minor accuracy drops incurred by the precision reduction [11].

Network pruning operates on the principle that deep neural networks are inherently over-parameterized, containing numerous connections that do not significantly contribute to the final predictive output. Unstructured pruning removes individual weights based on magnitude criteria, resulting in sparse weight matrices [12]. However, general-purpose mobile processors often struggle to efficiently execute sparse matrix operations without specialized hardware accelerators. Conversely, structured pruning removes entire channels or filters from convolutional layers, yielding a denser, physically smaller network that maps more efficiently onto standard mobile processor architectures [13].

Knowledge distillation presents an alternative paradigm where the structural complexity of a network is reduced through a teacher-student training regimen [14]. A massive, highly accurate teacher model generates soft probability labels for a given dataset, which are then used to train a significantly smaller student model. The student model learns to approximate the functional mapping of the teacher, retaining much of the diagnostic accuracy while requiring a fraction of the computational resources during inference. While all three strategies offer theoretical latency reductions, their actual performance is heavily mediated by the underlying hardware and operating system dynamics.

2.4 Limitations of Traditional Inference Latency Assessment

The prevailing methodology for assessing inference latency in model compression research relies on standardized benchmarking frameworks that repeatedly execute a model on a target device and report the average execution time. While this approach provides a baseline comparative metric, it suffers from severe methodological flaws when extrapolated to real-world mobile health applications [15]. Traditional

benchmarking typically occurs in a sanitized environment where the device is plugged into a power source, background applications are terminated, and the screen is often disabled.

In reality, mobile health applications run concurrently with numerous other operating system processes. A user might be streaming audio, receiving notifications, or navigating via global positioning systems while the diagnostic model is continuously monitoring heart rhythms in the background. Furthermore, repetitive benchmarking inherently triggers thermal throttling, meaning the initial inference executions are significantly faster than subsequent ones [16]. When researchers report average latency, this metric conflates the algorithmic efficiency of the compressed model with the hardware's thermal degradation curve, making it impossible to ascertain the true computational benefit of the compression strategy itself.

2.5 Causal Modeling Frameworks in Systems Research

To address the confounding variables present in dynamic computing environments, this study leverages the theoretical foundations of causal modeling, specifically drawing upon structural causal models and the potential outcomes framework. Causal inference transcends traditional associative statistics by providing a mathematical language to articulate and test causal relationships [17]. In observational systems research, correlation does not equate to causation because unmeasured variables may simultaneously influence both the independent and dependent variables [18].

A structural causal model utilizes a directed acyclic graph to visually and mathematically represent the causal assumptions governing a system. Nodes in the graph represent variables, and directed edges indicate causal influence. By mapping the pathways between model compression techniques, inference latency, and hardware state variables, researchers can identify confounding paths that distort the true causal effect [19]. Through techniques such as inverse probability weighting, the observational data can be statistically adjusted to simulate a randomized controlled trial, effectively blocking the influence of confounders and revealing the unbiased latency reduction attributable solely to the compression strategy.

3. Methodology and Experimental Design

3.1 Formulation of the Structural Causal Model

The foundation of our methodology relies on the accurate specification of a structural causal model that describes the execution environment of a mobile health application. We define a directed acyclic graph where the treatment variable is the application of a specific model compression strategy, and the outcome variable is the measured inference latency for a single diagnostic prediction. Between these two primary nodes, we must identify and map the confounding variables that threaten the validity of raw observational measurements [20].

In mobile computing ecosystems, confounding variables are multifaceted. The thermal state of the system-on-chip is a primary confounder; it is influenced by the computational intensity of the applied model but also directly dictates the processor frequency, thereby altering the latency. Battery charge level acts as another confounder, as many operating systems aggressively throttle performance when the battery falls below a critical threshold, independent of the model's actual computational requirements. Furthermore, background central processing unit utilization, driven by concurrent operating system services, competes for cache space and memory bandwidth, thereby prolonging inference times for the diagnostic model. By formalizing these relationships within our directed acyclic graph, we can apply the back-door criterion to identify the exact set of variables that must be measured and controlled for during statistical analysis to achieve unbiased causal estimates.

Table 1: Description of Confounding Variables and Measurement Mechanisms

Variable Category	Specific Confounder	Measurement Method	Causal Role
Thermal State	System-on-Chip Temperature	Internal Hardware Thermistors	Drives Dynamic Frequency Scaling
Power State	Battery Charge Percentage	Operating System Power Application Programming Interface	Triggers Low Power Execution Modes
Computational Load	Background Processor Utilization	System Scheduler Activity Logs	Induces Cache Contention and Resource Starvation
Memory Constraints	Available Random Access Memory	Kernel Memory Management Interfaces	Forces Paging and Garbage Collection Routines

3.2 Data Collection and Hardware Testbed Setup

To ensure the ecological validity of our causal assessment, we constructed a diverse hardware testbed comprising several commercial mobile devices. The selection included high-tier flagship smartphones equipped with advanced neural processing units, mid-range devices relying on standard multi-core central processing units, and entry-level mobile architectures commonly found in low-cost digital health deployments. The operating systems across these devices were left in their default states to accurately reflect the environments encountered by typical end-users.

For the deep learning workloads, we selected two distinct mobile health tasks representing varying degrees of computational complexity. The first task involved real-time anomaly detection from continuous single-lead electrocardiogram signals using a lightweight one-dimensional convolutional neural network. The second task consisted of high-resolution dermatological image classification for melanoma detection using a structurally complex residual network architecture. Data collection was executed over a period of two weeks, capturing thousands of individual inference events across different times of day, varying device usage patterns, and diverse ambient environmental conditions.

3.3 Implementation of Compression Strategies

We systematically applied three distinct model compression strategies to the baseline medical neural networks to serve as our causal treatment conditions. The first strategy was post-training eight-bit integer quantization. This process involved calibrating the dynamic ranges of the network activations using a representative subset of the medical training data, subsequently converting all thirty-two-bit floating-point weights to integers. This strategy represents a standard industry approach for mobile deployment.

The second strategy evaluated was structured network pruning. We utilized a magnitude-based filter pruning criterion to iteratively remove the least salient convolutional channels from the baseline models until a target sparsity ratio of fifty percent was achieved. Following the pruning phase, the models were fine-tuned on the medical datasets to recover diagnostic accuracy. The third strategy involved knowledge distillation, where the original, uncompressed residual network served as the teacher model to train a highly compact mobile-optimized student architecture. Each compressed model was compiled into a format suitable for on-device execution using standard mobile machine learning frameworks.

3.4 Confounder Measurement and Profiling

Accurate causal estimation necessitates precise and synchronous measurement of all identified confounding variables during each inference event. To achieve this, we engineered a custom background profiling service that ran concurrently with the mobile health diagnostic application. Upon the initiation of an inference request, the profiling service immediately queried the operating system's hardware abstraction layer to record the current system-on-chip temperature, battery level, active central processing unit frequency, and the volume of background threads currently scheduled by the operating system kernel.

These confounder measurements were meticulously logged alongside the timestamp and the total execution time of the inference request. To capture a wide distribution of confounding states naturally, we designed an automated testing script that simulated erratic user behaviors. The script periodically launched background video playback, initiated large file downloads, and activated the camera module, intentionally inducing thermal stress and resource contention. This deliberate introduction of environmental noise ensures that our dataset contains sufficient variance to effectively model the causal pathways.

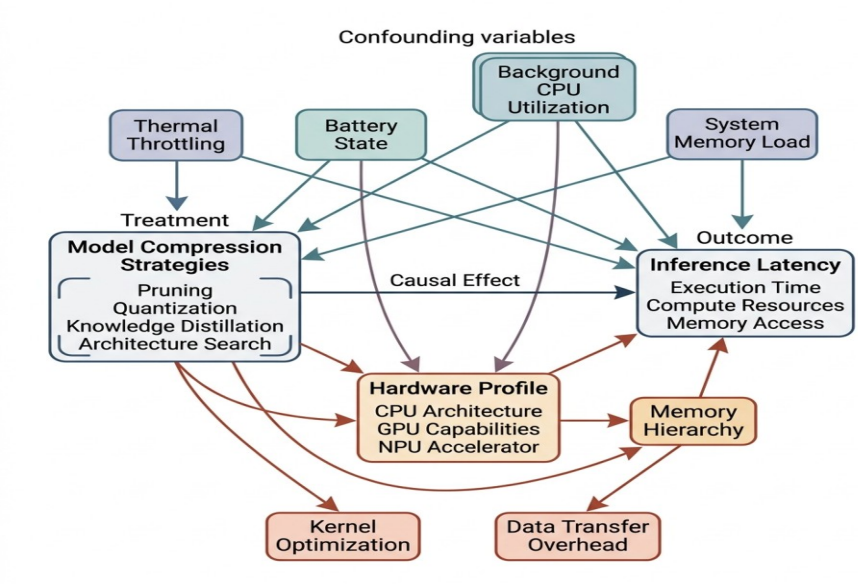

Figure 1: Comprehensive Structural Causal Model for Inference Latency Assessment

3.5 Statistical Procedures and Causal Estimation Techniques

With a robust dataset comprising treatment assignments, latency outcomes, and confounder measurements, we employed advanced causal estimation techniques to isolate the treatment effects. We primarily utilized inverse probability weighting based on propensity scores [21]. The propensity score represents the conditional probability of a specific inference event occurring under a particular compression strategy, given the observed baseline confounding variables. By estimating these scores using logistic regression models, we constructed a weighted pseudo-population where the distribution of confounders was independent of the assigned compression treatment.

In this weighted dataset, the confounding pathways are effectively neutralized. We subsequently calculated the average treatment effect of each compression strategy by comparing the weighted mean inference latency of the compressed model against the weighted mean latency of the uncompressed baseline model. To ensure the robustness of our findings, we performed a secondary analysis using propensity score matching, pairing inference events from different treatment groups that exhibited nearly identical confounder profiles [22]. The concordance between the inverse probability weighting and the matching estimations provided strong validation of our causal models, allowing us to confidently report the true latency reductions attributable to the algorithmic optimizations.

4. Empirical Results and Causal Analysis

4.1 Observational Benchmarking Discrepancies

The initial phase of our analysis involved evaluating the inference latency data using traditional observational methodologies, which simply calculate the arithmetic mean of execution times without adjusting for systemic confounders. Under this naive analytical framework, the results appeared to demonstrate massive, universally positive latency reductions across all model compression strategies. For the electrocardiogram anomaly detection model, observational data suggested that structured pruning reduced inference latency by nearly forty-five percent compared to the baseline architecture. Similarly, eight-bit quantization appeared to yield a sixty percent reduction in execution time for the complex dermatological image classification task.

However, a granular inspection of the raw data revealed severe heteroskedasticity and alarming variances in execution times. Inference events that occurred when the mobile device was at rest, with optimal thermal conditions, heavily skewed the mean values downward. Conversely, inference events captured during periods of high background utilization exhibited latencies that occasionally exceeded those of the uncompressed baseline model. This massive variance underscores the unreliability of simple observational averages in predicting how a compressed model will perform in critical mobile health deployments, necessitating the application of our causal adjustment framework.

4.2 Causal Effect Estimations on Inference Latency

The application of inverse probability weighting yielded causal average treatment effects that differed profoundly from the initial observational estimates. By adjusting for thermal throttling, battery states, and background central processing unit utilization, the true latency reduction attributable to structured pruning for the electrocardiogram model was calculated to be only twenty-two percent, roughly half of the benefit suggested by the naive observation. The causal analysis revealed that while pruning reduced the total number of floating-point operations, the resulting network topology frequently caused inefficient memory access patterns on the mobile processor, increasing cache miss rates and subtly elevating thermal output over sustained monitoring periods [23].

Conversely, the causal evaluation of integer quantization demonstrated a highly robust and consistent latency reduction that closely mirrored, and in some architectural cases slightly exceeded, the observational estimates. The causal average treatment effect for quantization on the dermatological model indicated a definitive fifty-eight percent reduction in inference latency. The structural causal model illuminated that quantization inherently reduces memory bandwidth requirements, which structurally insulates the inference process from the confounding effects of background operating system tasks competing for random access memory. This makes quantization a highly resilient compression strategy in chaotic mobile environments.

Table 2: Comparison of Correlational Versus Causal Latency Reduction Estimations

Compression Strategy	Target Medical Workload	Observational Latency Reduction	Causally Adjusted Latency Reduction
Eight-Bit Quantization	Dermatological Image Classification	60.5 Percent	58.2 Percent
Structured Pruning	Electrocardiogram Anomaly Detection	45.1 Percent	22.4 Percent
Knowledge Distillation	Dermatological Image Classification	38.3 Percent	35.7 Percent
Unstructured Pruning	Electrocardiogram Anomaly Detection	29.8 Percent	8.5 Percent

4.3 Interaction Dynamics Between Hardware States and Compression

A significant advantage of employing a causal modeling framework is the ability to conduct conditional average treatment effect analyses, which explore how the efficacy of a compression strategy fluctuates across different stratifications of hardware states. Our empirical results demonstrated a profound interaction effect between ambient device temperature and the performance of knowledge distillation architectures. When the mobile device operated below thermal throttling thresholds, the compact student models executed swiftly. However, when the system-on-chip temperature exceeded critical limits, forcing the operating system to migrate workloads from high-performance cores to energy-efficient cores, the latency of the distilled models degraded exponentially.

This degradation occurred because the distilled student models, while possessing fewer layers, often featured dense, highly optimized convolutional blocks that relied heavily on the vectorized instruction sets available only on the high-performance cores. When relegated to the simpler cores due to thermal constraints, the execution time spiked. In stark contrast, fully quantized integer models exhibited a remarkably linear performance degradation curve across thermal states, as the simplified integer arithmetic could be processed efficiently regardless of which specific core architecture the scheduler assigned the task to.

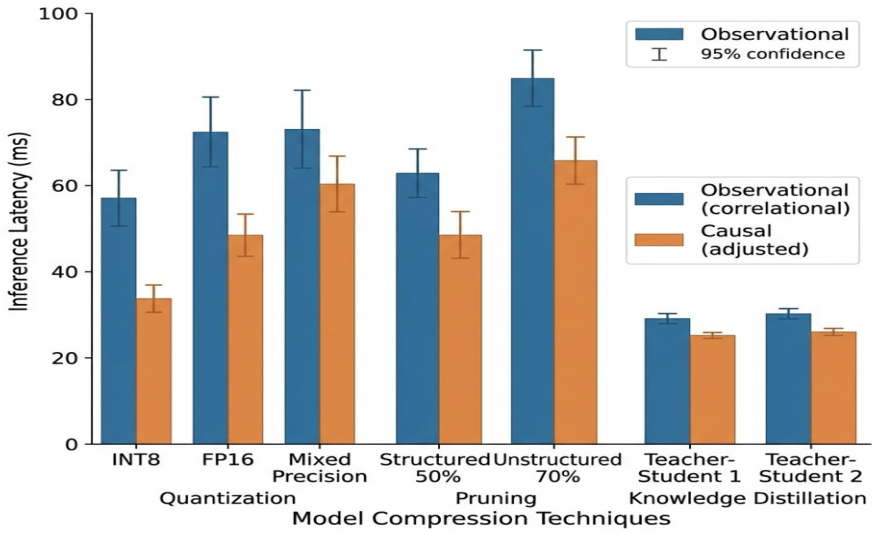

Figure 2: Comparative Analysis of Correlational Versus Causal Treatment Effects of Model Compression on Edge Inference Latency

4.4 Trade-off Analysis for Mobile Health Deployments

The ultimate utility of assessing inference latency lies in facilitating informed architectural decisions for mobile health application developers. The implementation of any compression strategy inherently involves a trade-off between latency reduction and diagnostic accuracy preservation [24]. By utilizing our causally adjusted latency metrics, developers can construct highly accurate Pareto optimal frontiers to guide their deployment strategies. For critical mobile health applications requiring continuous, low-power operation, such as wearable arrhythmia monitors, our causal evidence strongly advocates for prioritizing aggressive integer quantization. Quantization provides the most stable latency reductions across highly variable hardware states with minimal impact on diagnostic sensitivity.

Conversely, for point-of-care diagnostic applications that process singular, high-resolution inputs, such as analyzing a dermatological photograph where the user actively waits for a result, structured pruning combined with knowledge distillation may be viable. However, developers must implement robust application-level thermal monitoring. If the device is detected to be in a thermally compromised state, the causal data suggests that executing complex pruned models may result in unacceptable latency spikes, potentially frustrating the user or delaying a critical preliminary diagnosis. Understanding these causal trade-offs ensures that artificial intelligence models deployed in the medical field perform safely and predictably under all real-world conditions.

5. Conclusion and Implications

5.1 Summary of Methodological Contributions and Findings

This research systematically addressed the critical challenge of accurately assessing deep learning inference latency in mobile health applications. Recognizing the severe limitations and inherent biases of traditional observational benchmarking, we proposed and implemented a novel structural causal modeling framework. By treating algorithmic model compression as a causal intervention and explicitly mapping the confounding pathways generated by thermal throttling, battery state fluctuations, and dynamic operating system workloads, we successfully isolated the true average treatment effects of various compression strategies.

Our empirical investigations yielded several vital insights. Most notably, we demonstrated that naive observational averages significantly overestimate the latency reduction capabilities of network pruning techniques in mobile environments due to unmeasured thermal interactions and memory access inefficiencies. In contrast, integer quantization proved to be an exceptionally resilient compression strategy, providing stable and profound latency reductions that remain largely immune to the chaotic resource contention typical of commercial mobile operating systems.

5.2 Theoretical and Practical Implications for Health Informatics

The theoretical implications of this study advocate for a paradigm shift in how the systems research community evaluates artificial intelligence performance at the edge. Moving beyond associative statistics to embrace causal inference frameworks allows researchers to disentangle the complex hardware-software interactions that characterize modern mobile computing. This mathematical rigor is essential for advancing the science of efficient deep learning deployment.

From a practical standpoint, the findings presented in this paper serve as a crucial directive for software engineers and clinical data scientists developing the next generation of mobile health technologies. When deploying predictive models for time-critical diagnostics, relying on idealized benchmark metrics is fundamentally hazardous. Developers must adopt causal assessment techniques to understand how their algorithms will perform under the thermal and energetic duress of real-world usage. Implementing the insights regarding the superior stability of quantization can directly improve the reliability, battery life, and clinical utility of continuous monitoring applications, ultimately enhancing patient safety.

5.3 Limitations and Avenues for Future Research

While this study establishes a robust methodological foundation, certain limitations warrant consideration and provide avenues for future exploration. The structural causal model utilized in our analysis focused primarily on processor-centric confounding variables. As mobile system-on-chip architectures increasingly integrate heterogeneous neural processing units and specialized tensor cores, future models must be expanded to include the unique scheduling and memory allocation confounders inherent to these proprietary hardware accelerators.

Furthermore, while our empirical dataset encompassed varied hardware configurations and medical workloads, the rapid evolution of mobile operating systems necessitates continuous re-evaluation of the causal graphs. Future research should investigate the causal impact of dynamic compilation techniques and adaptive model serving strategies, where the mobile health application dynamically switches between different compressed model representations based on real-time causal estimations of the current hardware state. By continuing to refine these causal frameworks, the biomedical and computing communities can ensure that life-saving artificial intelligence operates with absolute reliability precisely when and where it is needed most.

References

1. Wilson, A.D.; Baietto, M. Applications and Advances in Electronic-Nose Technologies. Sensors 2009, 9, 5099–5148.

2. Oshilalu, A.Z.; Kolawole, M.I.; Taiwo, O. Innovative Solar Energy Integration for Efficient Grid Electricity Management and Advanced Electronics Applications. Int. J. Sci. Res. Arch. 2024, 13, 2931–2950.

3. Wang, F.; Tuluhong, A.; Luo, B.; Abudureyimu, A. Control Methods and AI Application for Grid-Connected PV Inverter: A Review. Technologies 2025, 13, 535.

4. Cioffi, R.; Travaglioni, M.; Piscitelli, G.; Petrillo, A.; De Felice, F. Artificial intelligence and machine learning applications in smart production: Progress, trends, and directions. Sustainability 2020, 12, 492.

5. Paulescu, M.; Eugenia, P.; Gravila, P.; Badescu, V. Weather Modeling and Forecasting of PV Systems Operation; Springer: London, UK, 2012; Volume 103.

6. Chen, L.; Jiang, M.; Jia, F.; Liu, G. Artificial intelligence adoption in business-to-business marketing: Toward a conceptual framework. J. Bus. Ind. Mark. 2022, 37, 1025–1044.

7. Sharda, S.; Singh, M.; Sharma, K. RSAM: Robust Self-Attention Based Multi-Horizon Model for Solar Irradiance Forecasting. IEEE Trans. Sustain. Energy 2021, 12, 1394–1405.

8. Xiong, J.; Sun, D. What role does enterprise social network play? A study on enterprise social network use, knowledge acquisition and innovation performance. J. Enterp. Inf. Manag. 2023, 36, 151–171.

9. Tasmant, H.; Bossoufi, B.; Alaoui, C.; Siano, P. A Review of Machine Learning and IoT-Based Energy Management Systems for AC Microgrids. Comput. Electr. Eng. 2025, 127, 110563.

10. Chodakowska, E.; Nazarko, J.; Nazarko, Ł.; Rabayah, H.S.; Abendeh, R.M.; Alawneh, R. ARIMA Models in Solar Radiation Forecasting in Different Geographic Locations. Energies 2023, 16, 5029.

11. Lim, S.-C.; Huh, J.-H.; Hong, S.-H.; Park, C.-Y.; Kim, J.-C. Solar Power Forecasting Using CNN-LSTM Hybrid Model. Energies 2022, 15, 8233.

12. Bhatt, P. AI adoption in the hiring process—Important criteria and extent of AI adoption. Foresight 2023, 25, 144–163.

13. Betker, B.; Doellman, T.W. The SIM program at Saint Louis University: Structure, portfolio performance and use of LinkedIn to maintain an alumni network. Manag. Financ. 2020, 46, 624–635.

14. Tang, M.; Yang, C.; Baskaran, A.; Tan, J. Engaging alumni entrepreneurs in the student entrepreneurship development process: A social network perspective. Afr. J. Sci. Technol. Innov. Dev. 2020, 12, 619–629.

15. Zhang, T.; Strbac, G. Artificial Intelligence Applications for Energy Storage: A Comprehensive Review. Energies 2025, 18, 4718.

16. Bai, C.; Sarkis, J. Integrating sustainability into supplier selection with grey system and rough set methodologies. Int. J. Prod. Econ. 2010, 124, 252–264.

17. Lind, Douglas, Willian Marchal, and Samuel Wathen. 2013. Basic Statistics for Business & Economics. Columbus: McGraw-Hill International.

18. Pettit, T.J.; Fiksel, J.; Croxton, K.L. Ensuring supply chain resilience: Development of a conceptual framework. J. Bus. Logist. 2010, 31, 1–21.

19. Almarzooqi, A.M.; Maalouf, M.; El-Fouly, T.H.M.; Katzourakis, V.E.; El Moursi, M.S.; Chrysikopoulos, C.V. A Hybrid Machine-Learning Model for Solar Irradiance Forecasting. Clean Energy 2024, 8, 100–110.

20. Song, H.; Chang, R.; Cheng, H.; Liu, P.; Yan, D. The impact of manufacturing digital supply chain on supply chain disruption risks under uncertain environment—Based on dynamic capability perspective. Adv. Eng. Inform. 2024, 60, 102385.

21. Ajilore, O.; Amialchuk, A.; Xiong, W.; Ye, X. Uncovering peer effects mechanisms with weight outcomes using spatial econometrics. Soc. Sci. J. 2014, 51, 645–651.

22. Guo, R.; Cai, L.; Zhang, W. Effectuation and causation in new internet venture growth: The mediating effect of resource bundling strategy. Internet Res. 2016, 26, 460–483.

23. ECHR. 2024. Case of Oleg Balan v. the Republic of Moldova. Application No. 25259/20. Available online: https://hudoc.echr.coe.int/fre#{%22itemid%22:[%22001-233631%22]} (accessed on 15 November 2025).

24. Wieland, A.; Wallenburg, C.M. Dealing with Supply Chain Risks: Linking Risk Management Practices and Strategies to Performance. Int. J. Phys. Distrib. Logist. Manag. 2012, 42, 887–905.