Benchmarking Emerging Deep Learning Quantization Methods for Energy Efficiency

Saurabhsingh Rajput and Tushar Sharma
Dalhousie University
Canada
{saurabh, tushar}@dal.ca

***Abstract*—In the era of generative artificial intelligence (AI), the quest for energy-efficient AI models is increasing. The increasing size of recent AI models has led to quantization techniques that reduce large models' computing and memory requirements. This study aims to compare the energy consumption of five quantization methods, *viz.* Gradient-based Post-Training Quantization (GPTQ), Activation-aware Weight Quantization (AWQ), GPT-Generated Model Language (GGML), GPT-Generated Unified Format (GGUF), and Bits and Bytes (BNB). We benchmark and analyze the energy efficiency of these commonly used quantization methods during inference. This preliminary exploration found that GGML and its successor GGUF were the most energy-efficient quantization methods. Our findings reveal significant variability in energy profiles across methods, challenging the notion that lower precision universally improves efficiency. The results underscore the need to benchmark quantization techniques from an energy perspective beyond just model compression. Our findings could guide the selection of models using quantization techniques and the development of new quantization techniques that prioritize energy efficiency, potentially leading to more environmentally friendly AI deployments.**

***Index Terms*—Quantization, Energy Consumption, Green AI, Energy Efficiency**

I. Introduction

Artificial Intelligent (AI)-based applications have become ubiquitous, touching almost all aspects of modern human life and extending their influence to diverse domains. However, the environmental sustainability of AI-based systems, especially during the inference phase, has become a pressing concern. Recent studies reveal the staggering energy consumption and carbon emissions associated with deploying and running large AI models for inference [1], [2].

In response, techniques such as weight quantization [3] are gaining traction. Quantization compresses models by reducing numerical precision, and in turn, reduces computing and memory requirements at the inference time to limit energy usage. Researchers have proposed several quantization methods including Activation-aware Weight Quantization (AWQ) [4], Gradient-based Post-Training Quantization (GPTQ) [5], Bits and Bytes (BNB) [6], GPT-Generated Model Language (GGML) [7] and GPT-Generated Unified Format (GGUF) [8]. However, their comparative energy effectiveness is under-explored.

Past studies for energy-aware quantization [9]–[11] have focused primarily on model compression, reducing computational complexity and memory usage by lowering weight and activation precision. While these methods have shown reductions in model size and FLOPs, their relative energy efficiency remains unclear. Most existing techniques do not explicitly optimize energy consumption, and their hardware-dependent savings are difficult to compare. This poses challenges when selecting among multiple methods that compress models to the same size.

Our study addresses this gap by exploring state-of-the-art quantization techniques through comparative energy benchmarking of leading same-size quantized models (LLAMA-2-7B). By uniformly evaluating prominent 4-bit quantization techniques, we reveal energy profile variability that compression rates do not explain. Our findings underscore the need to develop quantization optimized holistically for accuracy, model size, and energy efficiency. By comparing their energy consumption at the matched quantization levels, we can characterize the energy efficiency of each method, fostering innovations further in the pursuit of energy-efficient AI [12], [13].

II. Background

A. Quantization Methods

We briefly discuss common quantization methods and their key characteristics.

Activation-aware Weight Quantization (AWQ): AWQ [4] is a post-training quantization technique that leverages the observation that not all weights in a large language model are equally important. By identifying and preserving just a small fraction (0.1%–1.0%) of the most salient weights, AWQ can significantly reduce quantization error. Interestingly, in AWQ, the weight salience is determined by the magnitude of the corresponding activation values rather than the weight values themselves.

Gradient-based Post-Training Quantization (GPTQ): GPTQ [5] is a post-training quantization method that aims to find the optimal quantized weights to minimize the difference between the outputs of the full precision and quantized models. It formulates the layer-wise compression problem as a least squares optimization, which is solved using a variant of the Optimal Brain Surgeon algorithm [14].

Bits and Bytes (BNB): BNB [6] is a unified framework for quantization-aware training that compresses model weights to 4 bits of precision, significantly reducing memory footprint

while maintaining performance close to the full precision. It leverages a novel data type called 4-bit *NormalFloat* (NF4), which is theoretically optimal for representing normally distributed weights. BNB stores the quantized weights in 4-bit precision but performs computations in 16-bit or 32-bit precision to ensure numerical stability and computational efficiency.

GPT-Generated Model Language (GGML): GGML [7] is a binary format for distributing quantized transformer language models. It uses techniques like low-precision quantization to reduce model size and computational requirements. The format consists of the model's hyperparameters, vocabulary, and quantized weight tensors grouped into layers. By exploiting reduced precision, GGML enables running large models on consumer hardware.

GPT-Generated Unified Format (GGUF): GGUF [8] is a successor to now deprecated GGML, designed to address its limitations by creating a generalized file format that maintains backward compatibility and avoids frequent breaking changes. GGML faced challenges in incorporating extra model information and introducing new features without causing compatibility issues, requiring users to modify complex settings manually. GGUF overcomes these obstacles by allowing the addition of new features while maintaining compatibility with older models, simplifying the transition to newer versions [15]. It incorporates quantization-aware kernel optimization techniques and extensibility features. It offers advantages such as single-file deployment, faster loading and saving, a user-friendly design, and comprehensive information storage, contributing to a more streamlined and accessible process for working with large language models.

B. Energy Consumption Calculation

To accurately assess the energy efficiency of the quantization methods, it is essential to distinguish between power consumption and energy consumption. Power consumption represents the rate of energy transfer or consumption, typically measured in Watts (W), while energy consumption refers to the total amount of energy consumed over a specific period or operation, often expressed in Watt-hours (Wh) or Joules (J); one Joule is equivalent to 1 Watt-sec. This study calculates and reports energy consumption values by considering both the power consumption and the duration of the inference operation.

III. Methodology

This study employs CodeCarbon [16], an open-source tool, to measure and compare the energy consumption of prominent quantization techniques during inference. CodeCarbon estimates the energy usage and associated carbon emissions for computing tasks based on the underlying hardware specifications.

We evaluate five leading quantization methods—GPTQ, AWQ, GGML, GGUF, and BNB. We quantize a transformer-based model LLAMA-2-7B [17] down to a uniform bandwidth (*i.e.,* 4 bits) for uniform comparison. We use *wikitext-2* [18] dataset to benchmark inference across all considered quantization methods. Energy consumption is estimated for running inference on this dataset in milliWatt-hour (mWh).

A. Experimental Setup

The hardware used for benchmarking is a single NVIDIA 3070 Ti GPU coupled with an Intel(R) Xeon(R) Gold 5317 CPU. We conduct five experiments per method to account for randomness in energy consumption. We use the same hyperparameter settings and prompt selection for all quantization methods. Specifically, we use a batch size of 1, a sequence length of 512, and no beam search or other sampling techniques during inference.

We randomly sample a set of 100 prompts from the wikitext-2 dataset. The same prompts are used across all quantization methods to ensure a fair comparison. Regarding inference throughput, we measure it in terms of tokens per second without any constraints or caps applied. This means that the models are allowed to generate as many tokens as needed to complete the prompt without any limitations. This approach ensures that the measured throughput accurately reflects the computational expense of each quantization method without introducing any potential biases.

Our methodology enables an empirical comparison of leading quantization methods from an energy consumption standpoint. By isolating the effects of quantization from other factors, we can identify the most energy-efficient strategies to inform Green AI research and practice.

IV. Results: Is Lower Bitwidth Alone Sufficient for Selecting an Energy-Efficient Model?

Table I presents the average energy consumption and throughput measurements for prominent 4-bit quantization techniques benchmarked in this study. We executed text generation inference for standardized prompts from wikitext-2 [18] dataset using each quantization method. We measure the energy usage in milliwatt-hours (mWh) across three hardware components—GPU, CPU, and RAM—for each quantization technique during the inference stage. We also report the token generation throughput in tokens per second for each method. Furthermore, we use perplexity (PPL) [19] to evaluate the quality of the generated text, where lower PPL indicates better output quality. Similar to the Hugging Face optimum benchmark (*optimum/llm-perf-leaderboard*) [20], we use tokens generated per unit energy consumed (t/mWh) as a metric, which represents the cost of each token generated with respect to energy consumption. This metric provides insights into the energy efficiency of each quantization technique in terms of the number of tokens generated per milliwatt-hour of energy used.

Our preliminary investigation comparing the 4-bit quantization methods has revealed significant variability in energy efficiency, challenging the notion that reduced model size universally translates to energy savings. While techniques like GPTQ, AWQ, and BNB enable 4-bit quantization with

minimal accuracy loss, their energy consumption profiles diverge considerably. Our initial benchmarks demonstrate GGML and GGUF can offer over 200% energy savings relative to GPTQ despite matching 4-bit precision. This suggests that the quantization approach drives energy efficiency, not merely lower bitwidth.

Energy advantages are highly dependent on model architecture and layer distribution. No single technique optimizes all scenarios. Energy-aware quantization demands accounting for model internals and hardware interactions, not just model size. **Our study reveals that GGML and it's successor GGUF technique are the most energy-efficient quantization methods**, consuming just 308 and 318 mWh respectively, combined for GPU, CPU, and RAM, compared to $1,123$ mWh for GPTQ, 528 mWh for Bits and Bytes, 809 mWh for AWQ. Moreover, GGUF achieves the best tokens per milliwatt-hour (t/mWh), throughput and perplexity score, making it the most energy-efficient choice for generating tokens without compromising the output's quality.

TABLE I
COMPARISON OF QUANTIZATION TECHNIQUES' ENERGY CONSUMPTION.

Model	Energy cons. (mWh)			Throughput (t/sec)	Energy (t/mWh)	PPL
	GPU	CPU	RAM			
GPTQ	732	191	200	32.6	0.47	6.09
GGML	**224**	**40**	**44**	290.77	2.16	7.54
GGUF	232	41	45	**342.11**	**3.42**	**5.96**
AWQ	560	118	131	51.41	0.46	6.02
BNB	339	90	99	24.3	0.37	7.90

We envision the community moving beyond accuracy-centric notions like model size for quantization space exploration and embracing energy consumption as a first-class optimization objective. The move will pave the way for novel quantization techniques optimized for efficiency across diverse models, not one-size-fits-all compression.

V. RESULTS: DO WE USE CPU OR GPU TO DEPLOY A QUANTIZED MODEL?

Our study comparing the energy consumption of 4-bit quantization methods has significant implications for hardware selection when deploying quantized models. The choice between using a CPU or GPU can greatly impact the performance and efficiency of the quantized model.

CPU considerations: Using a CPU for inference with a quantized model has some potential trade-offs, as listed below.

- GGML and GGUF are optimized for CPUs, providing native support for features like 16-bit floats and integer quantization down to 4-bits. They utilize CPU vectorization instructions for improved performance [7].
- CPUs tend to be more cost-effective, especially for upgrading to support larger models. The expense of high-end GPUs with ample VRAM may be prohibitive. However, CPUs can struggle with parallel processing workloads required by larger models. An older CPU may take substantially longer to generate tokens compared to newer CPUs.
- CPUs generally consume less power than GPUs during inference. But slower run times may offset this benefit [21].

GPU considerations: GPUs also have trade-offs for quantized model inference.

- GPUs offer much faster inference thanks to massively parallel architecture, which is important for real-time applications [21].
- Although originally designed for CPUs, techniques like GGML now also support GPU offloading. Users can control the amount of computation dispatched to GPUs while retaining the remainder on CPUs by adjusting the number of offloaded layers.
- Some quantization methods such as GPTQ and AWQ provide optimizations for GPUs, potentially improving speed and lowering memory usage [4], [5].
- While GPUs draw more power, their speed can result in lower total energy consumption compared to slower but lower-power CPUs [21].

Our findings suggest that developing quantization methods optimized for specific hardware can enable more efficient deployments. Understanding the energy consumption patterns on different hardware also empowers practitioners to balance performance and efficiency [21].

VI. DISCUSSION

Our findings underscore the promising potential of specialized quantization formats such as GGML and GGUF for achieving superior energy efficiency compared to general-purpose techniques such as GPTQ, AWQ, and BNB. A key contributor could be GGUF's design focus on optimizing transformer-based language model inference. By tailoring quantization strategies and data layouts to the LLM use case, GGUF may leverage hardware more effectively than general techniques. This hardware-aware specialization aligns with co-designing efficient AI algorithms and systems. Additionally, GGUF prioritized deployment on resource-constrained devices from inception, influencing architectural choices favouring reduced energy consumption over other metrics.

Looking ahead, extending GGUF's efficiency to activation quantization and mixed-precision arithmetic is promising, potentially drawing from AWQ's activation-aware principles. Combining GGUF with complementary techniques such as pruning or distillation could further push performance-efficiency frontiers. From a systems perspective, continued co-design of efficient sparse and quantized compute kernels tailored for LLMs will unlock further speedups and savings by leveraging hardware sparse tensor support, custom DMA(Direct Memory Access) engines, and quantized matrix multiply units. Furthermore, while this study focused on inference, quantization enabling efficient LLM fine-tuning and lifelong learning is an intriguing future direction as rapid task adaptation becomes crucial. Finally, evaluating quantization across diverse hardware platforms beyond GPUs and CPUs is

also important, as architectural traits could favour different compression strategies.

VII. Related Work

A. Energy-Efficient Deep Learning

The growing concern over the environmental impact of large-scale deep learning models has motivated research into improving the energy efficiency of AI systems. Strubell *et al.* [1] highlighted the significant energy consumption and carbon emissions associated with training large language models, spurring efforts to develop more sustainable AI practices, including techniques to reduce the computational and energy requirements of deep learning models. Luccioni *et al.* [2] provided a comprehensive overview of the power consumption challenges in AI deployment, underscoring the need for holistic optimization of models, hardware, and software for energy efficiency.

B. Model Quantization

One prominent approach to improving the energy efficiency of deep learning models is model quantization, which reduces the numerical precision of model parameters and activations. This can lead to significant reductions in model size, memory usage, and computational complexity. Existing quantization techniques, such as Gradient-based Post-Training Quantization (GPTQ) [5], Activation-aware Weight Quantization (AWQ) [4], Bits and Bytes (BNB) [6], GPT-Generated Model Language (GGML) [7], and GPT-Generated Unified Format (GGUF) [8], have demonstrated the ability to compress models while maintaining accuracy, but their comparative energy efficiency has not been extensively explored.

Several studies have explored the use of quantization techniques for compressing large deep learning models, enabling efficient deployment on resource-constrained devices [22]. Wei *et al.* [23] investigated the impact of quantization on code generation tasks, demonstrating that quantization can compress large models without significant accuracy or robustness degradation, allowing the deployment of a 6B model on a regular laptop.

C. Energy-Aware Quantization

Prior studies have investigated the energy implications of model quantization but have primarily focused on reducing computational complexity and memory usage without explicitly optimizing for energy consumption [24]. Moons *et al.* [10] proposed a minimum energy quantized neural network approach, while Wang *et al.* [11] developed a hardware-aware automated quantization method. However, these studies did not provide a comprehensive comparison of leading quantization techniques from an energy efficiency perspective.

D. Green AI Practices

The field of Green AI has gained attention, with researchers advocating for energy efficiency as an important evaluation criterion alongside accuracy [25]. Yarally *et al.* [26] explored energy-efficient practices in deep learning training, such as hyperparameter tuning strategies and model complexity's impact on energy consumption. Researchers have also explored energy-efficient AI systems and practices, although not specifically focused on quantization methods. Stecyk and Enescu [27] presented a comprehensive review of Collaborative Energy Optimization Platforms (CEOP), which leverage AI algorithms for optimizing energy systems. Zhou *et al.* [28] introduced EfficientBioAI, a toolbox for compressing bioimaging AI models to reduce energy cost and inference time without compromising accuracy.

Despite these advancements in Green AI, there remains a lack of comprehensive benchmarking studies that compare the energy efficiency and computational performance of different quantization methods on standardized deep learning models and datasets. Our work addresses this gap by conducting a detailed benchmarking of prominent 4-bit quantization methods, evaluating their energy consumption and computational performance on a standardized deep learning model and dataset. This allows for a more nuanced understanding of the energy efficiency trade-offs associated with different quantization approaches, which can inform the development of future energy-optimized quantization techniques.

VIII. Implications and conclusions

Our study comparing the energy efficiency of prominent quantization techniques has several key implications for research and practice towards sustainable AI systems. First, the variability in energy profiles among methods that achieve similar model compression highlights the need to make energy optimization an explicit design criterion alongside accuracy. Our results imply that future quantization techniques consider minimizing energy and carbon footprint in addition to model size and FLOPs. Second, the superior energy efficiency of methods such as GGUF optimized for CPU hardware underscores the importance of co-developing software and hardware. This suggests value in profiling algorithms across diverse hardware devices to guide optimized codesign. Third, our findings reveal opportunities to select the most efficient quantization scheme based on model architecture, hardware platform, and use case constraints. This empirical data can inform greener model development, training, and deployment patterns tailored to specific applications. Finally, looking ahead, further efforts in benchmarking emerging methods on larger models, newer hardware, and across full training cycles could provide more robust insights. Developing standard energy efficiency metrics and reporting practices is critical to enable transparent model comparisons.

References

[1] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for modern deep learning research," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 09, 2020, pp. 13 693–13 696.

[2] A. S. Luccioni, Y. Jernite, and E. Strubell, "Power Hungry Processing: Watts Driving the Cost of AI Deployment?" no. arXiv:2311.16863, Nov. 2023, arXiv:2311.16863 [cs]. [Online]. Available: http://arxiv.org/abs/2311.16863

[3] M. Nagel, M. v. Baalen, T. Blankevoort, and M. Welling, "Data-free quantization through weight equalization and bias correction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1325–1334.

[4] J. Lin, J. Tang, H. Tang, S. Yang, X. Dang, C. Gan, and S. Han, "AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration," 2023.

[5] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, "GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers," 2023.

[6] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient Finetuning of Quantized LLMs," 2023.

[7] "GGML," 2024, accessed: 2024-01-25. [Online]. Available: https://github.com/ggerganov/ggml

[8] "GGUF," 2024, accessed: 2024-01-25. [Online]. Available: https://github.com/ggerganov/ggml/blob/master/docs/gguf.md

[9] C.-Y. Chang, K.-C. Chou, Y.-C. Chuang, and A.-Y. Wu, "E-UPQ: Energy-Aware Unified Pruning-Quantization Framework for CIM Architecture," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 13, no. 1, pp. 21–32, 2023.

[10] B. Moons, K. Goetschalckx, N. Van Berckelaer, and M. Verhelst, "Minimum energy quantized neural networks," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2017, pp. 1921–1925.

[11] K. Wang, Z. Liu, Y. Lin, J. Lin, and S. Han, "Haq: Hardware-aware automated quantization with mixed precision," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8612–8620.

[12] S. K. Esser, J. L. McKinstry, D. Bablani, R. Appuswamy, and D. S. Modha, "Learned step size quantization," *arXiv preprint arXiv:1902.08153*, 2019.

[13] P. Wang, Q. Hu, Y. Zhang, C. Zhang, Y. Liu, and J. Cheng, "Two-step quantization for low-bit neural networks," in *Proceedings of the IEEE Conference on computer vision and pattern recognition*, 2018, pp. 4376–4384.

[14] B. Hassibi, D. G. Stork, and G. J. Wolff, "Optimal brain surgeon and general network pruning," in *IEEE international conference on neural networks*. IEEE, 1993, pp. 293–299.

[15] D. Team, "GGUF vs. GGML: Why GGUF Is a Better File Format — Deci," 2024, accessed: 2024-03-25. [Online]. Available: https://deci.ai/blog/ggml-vs-gguf-comparing-formats-amp-top-5-methods-for-running-gguf-files/

[16] "Code Carbon," 2023, accessed: 2024-02-07. [Online]. Available: https://github.com/mlco2/codecarbon

[17] "LLAMA Model," 2023, accessed: 2024-02-25. [Online]. Available: https://huggingface.co/meta-llama/Llama-2-7b

[18] "Wikitext 2 dataset," 2021, accessed: 2024-01-25. [Online]. Available: https://paperswithcode.com/dataset/wikitext-2

[19] "Perplexity," 2024, accessed: 2024-03-25. [Online]. Available: https://huggingface.co/docs/transformers/en/perplexity

[20] "Huggingface LLM Perf Optimum Leaderboard," 2024, accessed: 2024-03-25. [Online]. Available: https://huggingface.co/spaces/optimum/llm-perf-leaderboard

[21] "CPU vs. GPU: Which One is Right for Your Workload? - DRex Electronics," 2023, accessed: 2024-03-25. [Online]. Available: https://www.icdrex.com/cpu-vs-gpu-which-one-is-right-for-your-workload/

[22] O. Weng, "Neural network quantization for efficient inference: A survey," *arXiv preprint arXiv:2112.06126*, 2021.

[23] X. Wei, S. Gonugondla, W. Ahmad, S. Wang, B. Ray, H. Qian, X. Li, V. Kumar, Z. Wang, Y. Tian *et al.*, "Greener yet powerful: Taming large code generation models with quantization," *arXiv preprint arXiv:2303.05378*, 2023.

[24] K. Vasquez, Y. Venkatesha, A. Bhattacharjee, A. Moitra, and P. Panda, "Activation density based mixed-precision quantization for energy efficient neural networks," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021, pp. 1360–1365.

[25] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green AI," *Communications of the ACM*, vol. 63, no. 12, pp. 54–63, 2020.

[26] T. Yarally, L. Cruz, D. Feitosa, J. Sallou, and A. Van Deursen, "Uncovering energy-efficient practices in deep learning training: Preliminary steps towards green ai," in *2023 IEEE/ACM 2nd International Conference on AI Engineering–Software Engineering for AI (CAIN)*. IEEE, 2023, pp. 25–36.

[27] A. Stecyk and I. Miciuła, "Harnessing the power of artificial intelligence for collaborative energy optimization platforms," *Energies*, vol. 16, no. 13, p. 5210, 2023.

[28] Y. Zhou, J. Sonneck, S. Banerjee, S. Dörr, A. Grüneboom, K. Lorenz, and J. Chen, "Efficientbioai: Making bioimaging ai models efficient in energy, latency and representation," *arXiv preprint arXiv:2306.06152*, 2023.