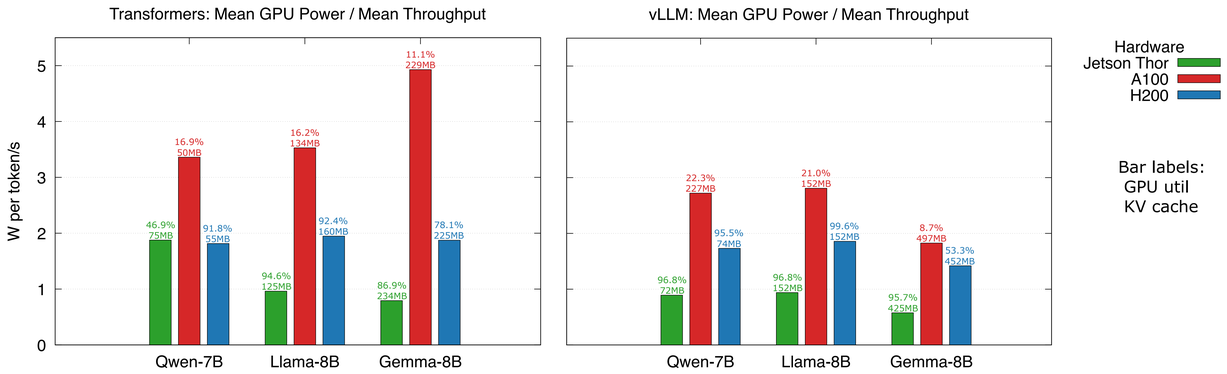
Characterizing GPU Performance in AI Inference with Power Metrics

Sean Tai[1,†], Mohammad Atif[2,‡], Sankha Dutta[2], Meifeng Lin[2]
[1]Queensborough Community College, Bayside, NY [2]Brookhaven National Laboratory, Upton, NY

Abstract

Large language model (LLM) inference is increasingly limited not only by raw compute capability, but also by memory bandwidth, runtime behavior, and GPU power efficiency. This project studies how LLM inference performance and energy use vary across hardware platforms and inference frameworks, with a focus on comparing an edge system-on-module against datacenter GPUs. Experiments are conducted on NVIDIA Jetson Thor, NVIDIA A100, and NVIDIA H200 GPUs using two inference frameworks: vLLM and Hugging Face Transformers. Three open-weight models are evaluated: Meta Llama-3.1-8B-Instruct, Google Gemma-4-E4B-it, and Alibaba Qwen2.5-7B-Instruct. For each hardware, framework, and model combination, inference metrics are collected with PyNVML, including tokens per second, GPU power, GPU utilization, KV-cache usage, generation time, and power-per-throughput. Pareto and histogram analyses are used to study the trade-off between throughput and energy efficiency. To connect inference behavior to hardware limits, BabelStream benchmark's triad kernel is used to measure GPU-memory bandwidth. In addition, roofline analysis separates prefill and decode phases to show when performance is compute-bound or memory-bound. These measurements help explain why the Jetson Thor can be energy-efficient at lower absolute throughput, while A100 and H200 provide higher throughput but consume more power.

Mean Power Consumption (W) per Throughput (tokens/s) [= Joules/token]



- Model dependence: Gemma behaves differently from Qwen and Llama. The larger KV footprint likely makes Gemma more sensitive to memory traffic and cache behavior, which helps explain why its efficiency does not follow the same simple utilization (higher utilization => smaller W per token/s) trend as Qwen/Llama.
- A100 shows poor efficiency for Qwen and Llama under vLLM despite low-to-moderate utilization.This suggests the GPU is drawing substantial power without being kept productively occupied.
- H200 is not the most energy-efficient here, but it is more consistently utilized than A100. This supports the interpretation that H200 is better for high-throughput serving, while Jetson is better for energy-efficient inference at smaller scale.
- Jetson Thor has the lowest power-per-throughput – thus is much more energy-efficient, even though its absolute throughput is lower (see Figure 1).
- vLLM is especially favorable on Jetson Thor. For Llama, the two are closer, but vLLM still remains slightly better. This indicates vLLM converts Jetson's high utilization into useful token generation more effectively.
- vLLM improves A100's relative behavior for Llama and Qwen, but not enough to beat Jetson or H200. In the Transformers plot, A100's utilization remains low, but its power-per-throughput is comparable to or worse than H200. This points to underutilization and likely poor batching/kernel occupancy rather than raw hardware limitation.

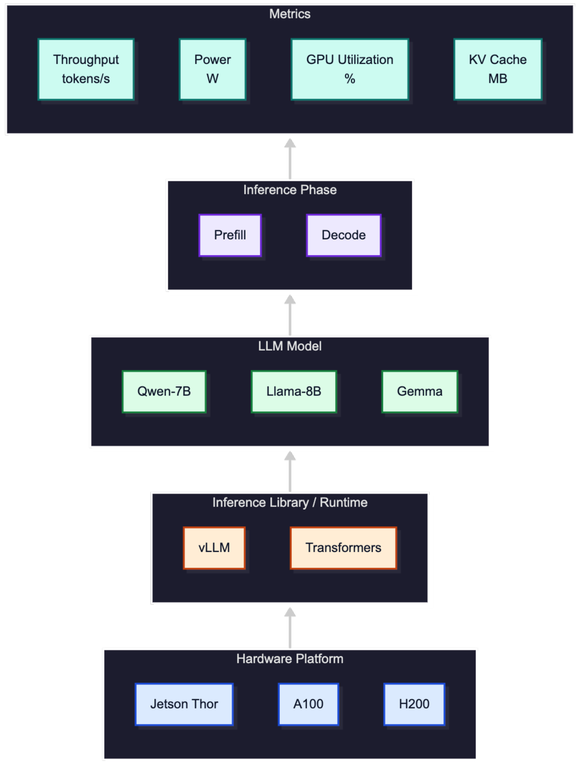
Acknowledgements

- LDRD program at Brookhaven National Laboratory, which is sponsored by the US Department of Energy, Office of Science, under Contract Number DE-SC0012704.
- Sean was supported by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Community College Internships (CCI) Program.
- Zhihua Dong for providing seamless access to computers at BNL.

Future Work

- Extend the study to additional LLMs, quantization formats, and longer context lengths to understand how model architecture and KV-cache growth affect energy efficiency.
- Evaluate production-oriented serving configurations such as CUDA graphs, continuous batching, larger batch sizes, and concurrent requests to compare roofline behavior with real deployment performance.
- Inspect deeper hardware-counters for kernel bottlenecks.
- Study end-to-end application workloads with realistic prompt distributions and response lengths to determine the observed efficiency trends under real user-serving conditions.

Method: Stack Diagram



Inference Efficiency Frontier

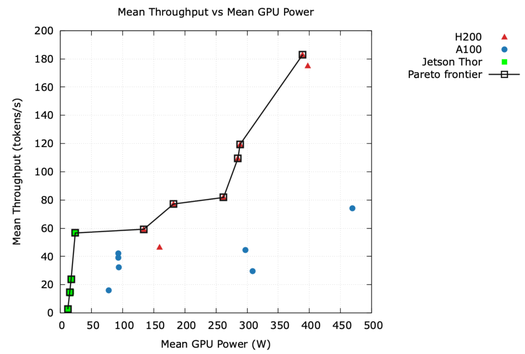
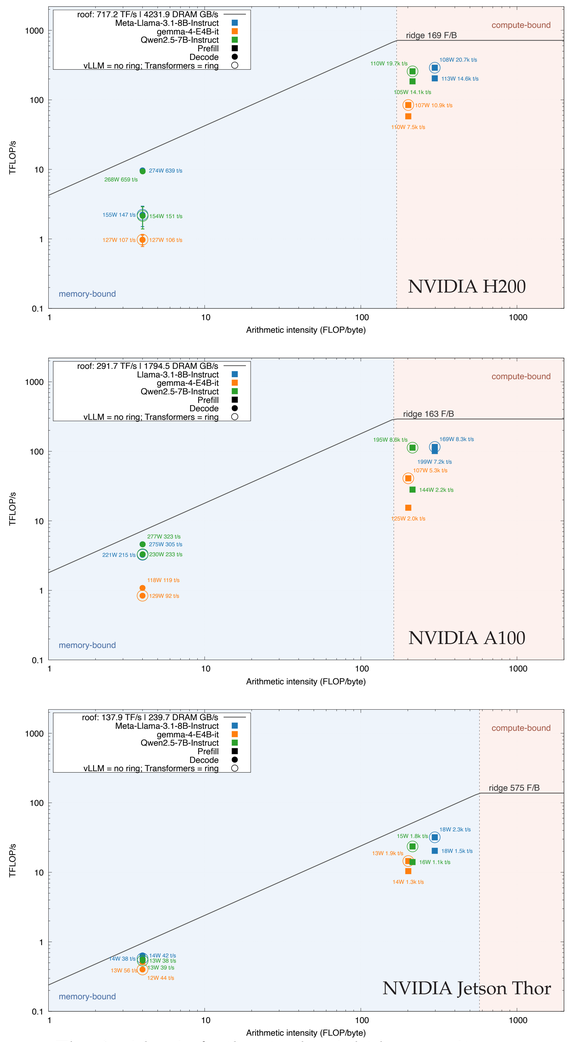


Figure 1: Power-throughput Pareto frontier. Thor defines the low-power end and H200 the high-throughput end.
A Pareto frontier is the set of configurations where no other point is better in both objectives at once: here, lower GPU power and higher tokens/sec.
Jetson appears on the frontier because it achieves very low power while still producing useful throughput, so neither of the higher-throughput server GPUs dominate it at the same power level.

Roofline Plots with Power and Throughput



- Thor's ridge is further to the right because its compute-to-memory-bandwidth ratio is higher, i.e., more compute cores relative to its available memory bandwidth.
- Optimize memory-bound for memory traffic and batching, optimize compute-bound for tensor-core/kernel efficiency.
- Gemma's decode is relatively healthy on Thor, but poor on A100/H200.
- Transformers prefill is consistently better than vLLM, while vLLM is better for decode.
- To optimize decode, larger batch size and vLLM scheduler limits need to be investigated.
- Thor – where prefills are compute bound – should benefit from smaller KV cache and quantizations which will reduce memory traffic.

U.S. DEPARTMENT OF ENERGY
www.bnl.gov
Brookhaven National Laboratory