NOOR MKDAD

AI/ML Infrastructure Engineer | LLM Serving, GPU Optimization & Quantization

Email: noorhayatywadam@gmail.com • Location: Orange County, CA (Remote, US) • U.S. Citizen, no sponsorship
LinkedIn: linkedin.com/in/noor-mkdad-6b0a77293 • GitHub: https://github.com/DrMkdaddy • Live Platform: https://stanzaapi.com

PROFESSIONAL SUMMARY

LLM inference engineer working on one 16 GB AMD GPU: 114.3 tok/s on an 8B model with bit-exact speculative decoding, 495 tok/s aggregate at eight slots after a Vulkan kernel patch now open upstream in llama.cpp, and a 59% memory cut for a 4.2% quality cost. Builds the serving layer around the model — admission control, KV paging, prefix caching, quantization — and measures every claim against a committed benchmark artifact.

Public work: llama.cpp kernel patch open upstream (#31022) · 5 inference repos with benchmark artifacts · 14 CC0 datasets on Hugging Face (7,499 downloads)

TECHNICAL SKILLS

Inference Serving: llama.cpp, OpenAI-compatible gateway, continuous batching, admission control, preemption, readiness gates
KV Cache & Scheduling: block allocation, prefix caching, priority and WFQ queues, TTFT and goodput under overload
Speculative Decoding: draft acceptance, break-even analysis, bit-exact verification
Quantization & Evaluation: GGUF (Q4_K_M, Q5_K_M, Q6_K, Q8_0, IQ4_XS), perplexity, KL divergence, VRAM accounting
GPU & Kernels: AMD RDNA4 (RX 9070 XT, 16 GB), Vulkan, K-quant dequantization, bit-exact evaluation, C/C++ (llama.cpp patch)
Languages & Systems: Python, Go, TypeScript, SQL, Bash; Linux, Proxmox VE (LXC, KVM), ZFS

PROFESSIONAL EXPERIENCE

Platform & Systems Engineer, Stanza API — live at stanzaapi.com

Remote (US) · January 2025 – Present

- **Built** the official MCP server (@stanzaapi/mcp) with stdio and HTTP/SSE transports, exposing 12 production APIs as agent-callable tools with fail-closed authentication.
- **Built and operate** 12 production APIs on Cloudflare Workers for claims, payments, and supply-chain data.
- **Built** a zero-regex streaming parser that checks 255,000 records in 14.2 seconds inside a 25 MB V8 old-space cap — 30% less wall time than the closest alternative. github.com/StanzaAPI/benchmark-harness

Inference & Infrastructure Engineer, Independent

Orange County, California · June 2024 – Present

- **Optimized** llama.cpp's Vulkan quantized matvec kernel: 437 to 495 tok/s (+13.4%) at eight slots, single-stream speed unchanged, bit-exact greedy output. Patch open upstream: ggml-org/llama.cpp#31022
- **Built** a self-hosted inference stack on a 16 GB AMD RX 9070 XT: 101 tok/s generation, 2,400 tok/s prefill, 286 tok/s across four concurrent slots at 9.9 GB VRAM, behind an OpenAI-compatible gateway with cost-aware routing at 19 to 41 cents per million tokens.
- **Implemented** prefix caching for repeated prompts, cutting cached-prefix latency from 405 ms cold to 161 ms warm.

PROJECTS

Kept an LLM server answering every request under overload: 79/79 completed at 4 req/s, premium traffic at 74 ms TTFT p99 while background work was preempted 14 times, using admission control and preemption in a Go scheduler in front of llama.cpp. github.com/DrMkdaddy/kv-serving-scheduler
Made an 8B model 13% faster with byte-identical output using speculative decoding (114.3 vs 101.3 tok/s, 40/40 exact, 64.7% acceptance against a 50.9% break-even); the same draft loses 12.5% on a 4B target, which marks the boundary. github.com/DrMkdaddy/speculative-decoding-lab
Let one GPU serve 1.7x more concurrent requests by paging working memory into shared blocks, with output proven identical; offloading that memory to Valkey only wins above roughly 4k-token prompts (1.28x at 6.1k, 0.90x at 2k). github.com/DrMkdaddy/valkey-kv-l2
Cut model memory from 6.1 GiB to 2.5 GiB for a 4.2% quality cost by benchmarking five GGUF quantization levels against an uncompressed reference. github.com/DrMkdaddy/quant-quality-harness
Benchmarked serving configs on RDNA4: KV Q4_0 held 474 tok/s on 31% less VRAM (7.8 vs 11.2 GB) and cut p99 to 107 ms from 194 ms, while disabling flash attention collapsed p99 to 3.6 s. github.com/DrMkdaddy/llama-backend-compare

EDUCATION

Independent study — ML systems, inference optimization, and GPU kernels (2024 – Present)