

Akaash R. Parthasarathy

US Citizen | (470) 871-9690 | akaashrp@gmail.com | linkedin.com/in/akaashrp | cs.cmu.edu/~arpartha | Mountain View, CA

EDUCATION

MS in Machine Learning, Carnegie Mellon University

Dec 2026 | GPA: **4.28/4.0**

BS in Computer Science, Georgia Institute of Technology

May 2025 | GPA: **4.0/4.0**

EXPERIENCE

Deep Learning Intern

Sept 2026 – Present

NVIDIA (FlashInfer)

- Shipped Rubin MoE megakernel backends in FlashInfer (NVFP4, MXFP8, MXFP4) and validated on 2–8 GPUs
- Wrote a CuTe DSL normalization kernel for GDN prefill matching cuDNN and FLA across 25 shapes on B200
- Characterizing the crossover between mega and split MoE kernels to set FlashInfer’s autotuner policy

Machine Learning Intern

May 2026 – Aug 2026

Google (TorchTPU)

- Wrote fused Pallas kernels (conv1d, chunked SSD scan) for Mamba-2 prefill, achieving 10× speedup over unfused XLA
- Scaled Nemotron-3 to 16 v7x chips with tensor, expert, and fully sharded data parallelism, overlapping all-gathers
- Improved StableHLO lowerings for `ragged_dot` and `repeat_interleave` based on profiling analysis with XProf

Machine Learning Compilation Researcher

Aug 2025 – Present

Catalyst Group, CMU Machine Learning Department (Prof. Tianqi Chen)

- Leading development of [WebLLM](#) (19K+ GitHub stars, 460K monthly npm downloads), an in-browser LLM engine
- Decoupled token readback from WebGPU submission, improving decode speed 2.7–5.5× to within 5% of native Metal
- Added KV-cache checkpointing to TVM for crash-resumable generation, recovering progress 33–46% faster than replay
- Made weight loading zero-copy from WASM to WebGPU, cutting peak renderer memory at initialization by up to 48%
- Built `mlc-web`, a compiler and runtime for multi-stage WebGPU models, achieving speeds within 8% of native kernels
- Invited speaker at Google Web AI Summit 2026, JSNation US 2026, and WebGPU Community Meetup

Efficient Inference Researcher

Aug 2025 – Present

OPAL Lab, CMU ECE Department (Prof. Gauri Joshi)

- Built a latency-aware vLLM router and TTFT estimator, cutting p99 TTFT 96% at higher response quality
- Achieved 70% TTFT-SLO attainment under peak load, where latency-aware baselines degrade to between 9 and 28%
- Co-developed on-policy attention linearization method, retaining 100% of teacher long-context retrieval and 83–93% of math accuracy at 4× lower KV/state memory and 2× vLLM throughput

Machine Learning Intern

Jun 2025 – Aug 2025

Millennium Management

- Built a multi-agent system resolving ambiguous natural-language queries over SSAS tabular trading models
- Wrote a DAX parser using ANTLR to feed structured errors into the correction loop, raising accuracy from 35% to 96%

SELECTED PUBLICATIONS (* equal contribution)

C Ruan, Y Qin*, **AR Parthasarathy*** et al. [WebLLM: A High-Performance In-Browser LLM Inference Engine](#). *Submitted*.

S Patel*, **AR Parthasarathy***, A Mallick, G Joshi. [Beyond Accuracy and Cost: Latency-Aware LLM Query Routing for Dynamic Workloads](#). *Submitted*.

R Lai*, H Kang*, H Tang, **AR Parthasarathy**, Z Yu, J Shao, TC Mowry, C Xiong, T Chen. [PithTrain: A Compact and Agent-Native MoE Training System](#). *NeurIPS 2026*.

A Raje*, A Nayak*, A Fei, **AR Parthasarathy**, M Abdelfattah, G Joshi. [On-Policy Attention Linearization](#). *Submitted*.

AR Parthasarathy, T Chen. WebLLM 2: WebGPU-Resident Decoding, Resumable Inference, and Capability-Specialized Kernels in the Browser. *In Preparation*.

SKILLS

Languages: Python, C++, C, WGSL, CUDA

Kernels/Compilers: Pallas, CUTLASS/CuTeDSL, Apache TVM, XLA/StableHLO

Frameworks/Libraries: PyTorch, JAX, vLLM, SGLang, FlashInfer, MLC-LLM, NCCL, MPI

Profiling: Nsight Systems, Nsight Compute, XProf, IKET