Aryan Ashta

+1(321)450-5200 | aashta2@illinois.edu | github.com/aryan-ashta | linkedin.com/in/aryan-ashta | aryanashta.me

Education

University of Illinois Urbana-Champaign May 2028 (Expected Graduation)
Bachelor of Science, Mathematics and Computer Science *Champaign, IL*

- **Relevant Coursework:** Algorithms & Models of Computation, Intro to Higher Mathematics, Data Structures, Probability and Statistics, Discrete Mathematics, Multivariable Calculus

Projects

Tiny Tensor Compiler June 2026 — August 2026

- Engineered an end-to-end tensor compiler in Python translating quantized PyTorch computational DAGs into standalone C kernels for ARM Cortex-M0+ (RP2040 Pico SDK), validating bit-exact parity across 2,000+ test vectors.
- Implemented a static buffer allocator using interval live-range analysis and memory aliasing, reducing peak activation SRAM footprint by 46% (388B to 208B) with zero dynamic allocation overhead.
- Formally proved bit-exact equivalence between reference and assembly-optimized execution paths using the Z3 SMT solver over a 2^{144} discrete state space, eliminating numerical drift and rounding discrepancies.

Fused RMSNorm Kernel May 2026 — June 2026

- Authored fused forward and backward RMSNorm GPU kernels in Triton from scratch, fusing row-wise reduction, variance computation, and affine scaling into single-pass SRAM operations.
- Benchmarked memory bandwidth across variable batch/hidden dimensions, achieving 4.0x (forward) and 2.7x (backward) effective DRAM bandwidth improvements over eager PyTorch while matching torch.compile (Inductor) throughput.
- Automated thread block tiling, warp sizing, and memory layout configurations using triton.autotune to maximize memory bus saturation across non-power-of-two matrix geometries.

Hierarchical Adapter Fusion for Continual LLM Adaptation August 2025 — April 2026

- Designed a variational hypernetwork to dynamically generate rank-8 LoRA adapter weights in a single forward pass, eliminating per-task gradient updates and reducing adaptation latency 15x relative to standard LoRA fine-tuning.
- Built an embedding retrieval pipeline using FAISS flat index lookups to condition weight synthesis on historical task representations, mitigating catastrophic forgetting across sequential reasoning benchmarks.
- Optimized candidate weight search under NF4 (4-bit) quantization on a single 16GB GPU, evaluating output generation stability across GSM8K and OpenCodeInstruct datasets.

Competitions

Mathworks Math Modelling Challenge February 2026

- Formulated a 9-state time-varying Markov transition system to simulate credit delinquency and default cascades, introducing dynamic hazard multipliers to model loss-chasing feedback loops (Top 15% international finish).
- Ingested and calibrated state transition matrices against empirical credit datasets from the Federal Reserve (G.19), NY Fed Consumer Credit Panel, PACER court filings, and Bank of England lending series to tailor model to both US and UK environments.

Skills

- **Languages:** Python, C, CUDA
- **Systems & Tools:** Triton, Z3 SMT Solver, Git
- **Machine Learning & Modelling:** PyTorch, Transformers, Quantization, LoRA, FAISS