Allan Zhang

Last updated in July 2026

Los Angeles, CA allanzhang440@gmail.com
609-943-8429 allanyz.com giyushino

Education

University of California, Los Angeles *BS in Computational Mathematics* *Sept 2024 – June 2028*

- **Relevant Coursework:** Multivariable Calculus, Linear Algebra, Differential Equations, Discrete Structures, Probability Theory, Data Structures and Algorithms, Large Scale Machine Learning (graduate coursework)

Experience

Research Intern | DatologyAI *Jan 2026 - Present*

- Built a reinforcement learning framework for VLMs from scratch to generate and evaluate high-quality reasoning traces; optimized weight synchronization between inference workers and trainers
- Developed checkpoint resumption system with optimizer state loading for VLM training, enabling fault-tolerant recovery and flexible CPT launch configurations
- Created an end-to-end pipeline for filtering synthetic VLM reasoning traces, leveraging Flyte for orchestration, Ray for distributed compute, and Spark for data processing

Undergraduate Research Assistant | BigML @UCLA *Nov 2024 – Present*

- Conducting research on data-efficient training for LLMs, focusing on data selection for reinforcement learning and synthetic data generation. Advised by Prof. Baharan Mirzasoleiman
- Co-designed a data-efficient variant of GRPO, achieving 30% faster convergence and a 40% reduction in required training samples relative to vanilla GRPO

Projects

Torchure (Pure PyTorch Training Stack) *Ongoing*

- Built an LLM pretraining stack from scratch in pure PyTorch: Qwen3 0.6B implementation (GQA, RoPE, SwiGLU), streaming dataloader with sequence packing, CUDA prefetching, and stateful checkpointing
- Improved single-GPU throughput 52% (35.8k → 54.5k tokens/sec) and cut peak memory 20% via per-block torch.compile and a custom fused linear + cross-entropy kernel that avoids materializing 152k-vocab logits
- Implemented distributed primitives from scratch: N-dimensional device mesh and collectives (all-reduce, reduce-scatter, all-to-all, ring send/recv), validated on Gloo/NCCL as groundwork for FSDP2 and tensor parallelism

Thinking Constrained GRPO *grpo-thinking-budget*

- Implemented constrained reasoning for LLMs by enforcing thinking token budgets during GRPO training, enabling fine-grained control over model inference costs
- Designed a two-phase generation system with vLLM that separates thinking and response phases, with automatic delimiter injection when thinking budget is exhausted
- Created fast inference pipeline using vLLM with two-phase generation to control thinking vs. response token allocation for evaluations

Publications

Beyond What Seems Necessary: Hidden Gains from Scaling Training-Time Reasoning Length under Outcome Supervision *Under review, NeurIPS 2026*
Yihao Xue, Allan Zhang, Jianhao Huang, Amit Sahai, Baharan Mirzasoleiman

20/20 Vision Language Models: A Prescription for Better VLMs through Data Curation Alone *Technical report, 2026*
DatologyAI

Skills

Programming Languages and Frameworks: Python, PyTorch, JAX, NumPy, Matplotlib, TensorFlow, scikit-learn, OpenCV, Hugging Face, LaTeX, C++, Bash, vLLM, SGLang

Languages: English, Korean