Runguo Li

Ph.D. Applicant, Fall 2028
M.S. Student, University of Illinois Urbana-Champaign | Champaign, IL, USA
runguo.ai@gmail.com | runguol2@illinois.edu
runguoli.com | github.com/LiRunGuo | Google Scholar | x.com/Runguo_Li

Research Objective

I build efficient and trustworthy systems for long-context language models. My work combines sparse attention, KV-cache compression, and low-bit quantization with explicit error guarantees, then validates these methods through kernel-, compiler-, and serving-level analysis on real GPU stacks. I aim to make long-context inference fast, predictable, and correct at production scale.

Education

University of Illinois Urbana-Champaign (UIUC) Aug 2026 – Present
M.S. in Information Management (expected Jul 2028) | *[Transcript]* Urbana-Champaign, IL, USA

Shanghai University of Finance and Economics (SUFE) Sep 2022 – Jun 2026
B.S. in Business Analytics, GPA: 3.71/4.0 | *[Transcript]* Shanghai, China

Relevant Coursework: *Undergraduate*: Python Programming, Algorithms for Big Data, Data Mining, Mathematical Statistics, Linear Algebra, Mathematical Analysis, Advanced Mathematics, Stochastic Processes, Regression Analysis. *Graduate*: Computer Architecture, Applied Parallel Programming, Distributed Systems, Natural Language Processing, Data Structures & Algorithms, Large Language Models, Multimodal Machine Learning, Database Design & Prototyping.

Research Experience

University of Illinois Urbana-Champaign Aug 2026 – Present
Research Intern, advised by Prof. Minjia Zhang Urbana-Champaign, IL, USA

- Developed certified sparse attention with per-head retrieval-error guarantees; evaluated against 15 baseline families at 16K–128K tokens, reducing tail error by 4.6–7.9× [1].
- Studied fixed-byte KV-cache compression and showed that allocating budget across token count and precision beats eviction alone in 17 of 18 long chain-of-thought settings [2].
- Built reproducible GPU benchmarking and analysis workflows for attention, KV-cache, and quantization experiments.

Shanghai Jiao Tong University Mar 2026 – Jul 2026
Research Intern, advised by Prof. Xiaodong Gu Shanghai, China

- Co-first authored **VeriBRT**, a plan-guided framework that generates evidence-backed bug-reproduction tests for repository-level issues.
- Built the SWT-Bench Verified evaluation harness and reproduced 341 of 433 issues from execution evidence.
- Analyzed failure cases across repository, test, and agent traces to improve reproduction reliability.

Tencent Youtu AI Lab Oct 2025 – May 2026
Research Intern Shanghai, China

- Co-first authored **RVCFT**, a multimodal chain-of-thought fine-tuning method that supervises reasoning-relevant and visually sensitive tokens.
- Built adversarial data-synthesis and multilingual content-safety moderation pipelines.
- Evaluated model behavior across multilingual and adversarial settings to identify reasoning and safety failure modes.

Selected Publications and Preprints

(* equal contribution)

[1] Zhi Yang*, **Runguo Li***, et al. **FinVault: Benchmarking Financial Agent Safety in Execution-Grounded Environments**. *arXiv preprint arXiv:2601.07853, 2026.* [arXiv] [Code]
Execution-grounded safety benchmark for financial agents (31 sandbox scenarios, 963 test cases).

[2] Haibo Jin, Peng Kuang, Xiaopeng Yuan, ..., **Runguo Li**, ..., Haohan Wang. **The Last Mile to the Real World: Agent-Native Primitives as Reusable Building Blocks for Solving Real-World Tasks**. *arXiv preprint arXiv:2611.07461, 2026.* [arXiv] [PDF]
Builds task-specific harnesses from reusable agent-native primitives; raises Terminal-Bench 4.0 resolution from 37.3% to 55.5% on the same backbone.

[3] **Runguo Li**. **How Much Does Random Rotation Help GPTQ? Sharp Worst-Case Rate Gaps in the High-Rate Regime**. *arXiv preprint arXiv:2611.01669, 2026.* [arXiv] [PDF]
Sharp worst-case rate gap between randomly rotated GPTQ and WaterSIC, answering an open question.

[4] **Runguo Li**, Dezhen Lu. **Weight Placement Is Not Enough: Synchronization in Multi-Node CPU Decoding of Ternary LLMs**. *arXiv preprint arXiv:2610.28620, 2026.* [arXiv] [PDF]
Shows node-local weight placement is insufficient on multi-socket CPUs; synchronization-aware decoding is 1.5–2.1× faster than the fastest baseline.

[5] **Runguo Li**. **Certified Sparse Attention via Perturbed Retrieval**. *arXiv preprint arXiv:2610.24991, 2026.* [arXiv] [PDF]
Sparse attention with per-head error certificates; 4.6–7.9× lower tail error than the best of 15 baseline families at 16K–128K tokens.

[6] **Runguo Li**. **Spend Bytes on Breadth: Precision–Count Trade-offs for Decode-Time KV Compression in Long Chain-of-Thought Reasoning**. *arXiv preprint arXiv:2610.05685, 2026.* [arXiv] [PDF]
Splits a fixed KV byte budget between token count and precision; beats eviction alone in 17 of 18 settings.

Additional arXiv preprints (Google Scholar-indexed): What Sets the Serving Speed of Looped Language Models; Where KV-Cache Error Lives; Prefix Keys and Values Matter Late; ScaleRLT.

Open-Source Contributions

30 merged pull requests across 17 upstream projects (private repositories excluded; project names link to merged PRs):

- **vLLM-Omni**: block-sparse attention in the Qwen2.5-Omni Token2Wav DiT, 3.5–6.4× lower per-request latency on H200 (#7975, #7959).
- **vLLM**: fixed int4 per-token-head KV-cache quantization for non-power-of-two head sizes (#56198).
- **FlashInfer**: normalized strided attention-plan metadata before planning and kernel use, preventing illegal memory access and wrong output across ragged, paged, and block-sparse backends (#5385).
- **DeepSpeed**: fixed ZeRO-3 rollout synchronization and early-stop deadlocks, made Hybrid Engine fall back on unsupported policies, and kept GPT-NeoX QKV layout changes through ZeRO-3 repartitioning (#8264, #8265, #8390, #8392).
- **FlashAttention**: block-sparse tiling and causal-mask skipping in the SM90 forward kernel (#2903, #2904).

Also merged into: Kubernetes, LLVM, Triton, JAX, Apache TVM, verl, colibri, LMCache, SGLang, Megatron-LM, UCCL, and Liger-Kernel.

Teaching and Service

Teaching Assistant: Data Mining (Fall 2025) | Python Programming (Spring 2026)
Academic Service: MLSys 2027 | AAAI 2027 | CVPR 2027

Honors and Awards

National Encouragement Scholarship (2024, 2025) | People's Scholarship (2024)
Shanghai Municipal Scholarship (2025)

Skills

Languages: Python, C/C++, CUDA, Triton, SQL.
ML systems: PyTorch, JAX, vLLM, SGLang, DeepSpeed, Megatron-LM; distributed training, inference serving, KV-cache optimization, quantization.
Performance and correctness: Nsight Systems/Compute, NCCL, profiling, benchmarking, CI/testing, compiler and kernel debugging.
Compilers and hardware: MLIR, LLVM; NVIDIA and AMD GPUs.