Renyuan Liu

rliu23@gmu.edu | LinkedIn | Google Scholar

AI Infrastructure · LLM Inference · GPU Systems · ML Systems

SUMMARY

Ph.D. candidate in Computer Science at George Mason University (expected graduation: May 2027) with 5+ years of experience building high-performance ML systems for **LLM inference and serving**. Expertise in **GPU runtime scheduling**, asynchronous I/O, memory management, and **custom GPU kernel** optimization using CUDA, Triton, and TVM. Built systems achieving up to **2.1**× higher LLM token-generation throughput, **7.8**× GPU operator speedup, and **22.9**× training-memory reduction, with publications at MobiCom, EuroSys, MobiSys, and SenSys.

AI INFRASTRUCTURE & ML SYSTEMS

LeanStream *GPU-Centric LLM Runtime for Compute–I/O Scheduling* 2026

Built a GPU-centric LLM inference runtime for models whose weights do not fit in GPU memory. The runtime continuously refines weight priorities from partial GPU results, asynchronously streams high-priority weights from storage, and executes loaded weights as they arrive to overlap I/O with computation. On edge SoCs, implemented fine-grained CPU–GPU coordination using **unified memory** and **custom GPU kernels**, achieving up to **2.1**× higher token-generation throughput and **7.5**× lower memory usage. Prototyping a server-side extension using **persistent GPU kernels** for remote weight streaming.

TokenFlow *Responsive LLM Serving via Preemptive Scheduling* 2026

Built a responsive LLM serving system with **buffer-aware preemptive scheduling** on SGLang. The scheduler tracks each request's buffered tokens and target consumption rate, pausing requests that are overproducing and resuming those running low to maintain responsive streaming under bursty workloads. Added proactive **KV cache migration** between GPU and CPU memory and overlapped cache movement with computation, improving effective throughput by up to **82.5%** and reducing P99 TTFT by up to **80.2%**.

DAF *Dynamic-Quantization Training Runtime* 2025

Built a memory-efficient DNN training runtime that dynamically adjusts activation precision based on runtime importance. Used **kernel fusion** to reduce quantization and memory-movement overhead, and designed a **page-based memory manager** for variable-sized quantized activations, reducing training memory usage by up to **22.9**×.

DynaSpa *Dynamic Sparse GPU Runtime* 2024

Built a dynamic sparse GPU execution framework for convolution and attention. The runtime selects custom GPU kernel variants for input-dependent sparsity patterns and uses **sparsity-aware tiling** to group irregular active regions into GPU-friendly work tiles, improving shared-memory/L1 reuse and reducing global-memory traffic. Achieved up to **7.8**× operator speedup.

EDUCATION

George Mason University, Fairfax, VA, USA — Aug. 2021 – Expected May 2027
Ph.D. in Computer Science.

Institut d'Optique Théorique et Appliquée, Paris, France — Sep. 2017 – Nov. 2019
M.S. in Automation and Signal.

Huazhong University of Science and Technology, Wuhan, China — Sep. 2013 – Jun. 2017
B.E. in Optoelectronic Information Science and Engineering.

SELECTED PUBLICATIONS

[**MobiCom'26**] **Renyuan Liu**, et al. *LeanStream: A Speculate-and-Refine Streaming Framework for Efficient On-Device LLM Inference.*

[**EuroSys'26**] J. Chen, C. Du, **Renyuan Liu**, et al. *TokenFlow: Responsive LLM Text Streaming Serving under Request Burst via Preemptive Scheduling.*

[**MobiSys'25**] **Renyuan Liu**, et al. *DAF: An Efficient End-to-End Dynamic Activation Framework for On-Device DNN Training.*

[**SenSys'24**] **Renyuan Liu**, et al. *DynaSpa: Exploiting Spatial Sparsity for Efficient Dynamic DNN Inference on Devices.* **Best Paper Award Nominee**

SKILLS

Programming & GPU: Python, C/C++, CUDA, OpenCL
ML Systems & Compilers: SGLang, vLLM, TVM, Triton, PyTorch, TensorFlow
Systems & Performance: LLM Inference, LLM Serving, GPU Runtime Scheduling, GPU Kernel Optimization, Asynchronous I/O, KV Cache Management, Memory Management, CUDA Streams, Tensor Parallelism, Pipeline Parallelism, Linux, Nsight Systems