Peterson Guo

petersonguo@gmail.com | petersonguo.com | PetersonGuo | PetersonGuo

Education

University of Waterloo — Waterloo, ON
BASc. in Honours Electrical Engineering — *09/2023 - 05/2028*

Skills

Languages: C/C++, Python, MLIR, JS, Java, Bash, SQL, Terraform
Software: Git, LLVM, PyTorch, Tensorflow, CUDA, ROCm, NumPy, Pandas, Spark, Docker, AWS, GCP
Others: ASICs, Drivers, Operating Systems, Linux, LLMs, CNNs, LSTMs, Neural Networks, Kernel Debugging

Experience

Compiler Engineer Intern — 05/2025 – 12/2025
Nvidia — *Toronto, ON*

- Developed and integrated custom **AI kernels** for a **Python**-based **ML compiler** into the **LLM** inference path, delivering up to **2.67x** speedups by reducing memory traffic and kernel-launch overhead
- Enhanced **AI kernel** fusion passes by introducing new fusion algorithm and graph patterns, securing an additional **0.7%** inference speedup on multi-billion parameter **LLM** models
- Extended the **MLIR**-based CUDA dialect, adding **20+** ops with result-handling semantics, lowerings, and tests, extending **CUDA**, enabling **Python** DSL operators to use advanced features on **Rubin, Feynman, and Blackwell GPUs**
- Validated CUTLASS/Collective IR kernels in **MLIR** to ensure correct lowering and codegen, exercising tensor-compute and data-movement pipelines across next-gen GPU architectures.
- Decomposed end-to-end **LLM** graph latency with Nsight Systems/Compute to isolate kernel, scheduler, and memory stall contributors for targeted remediation
- Writing custom **AI** kernels for zero-day **LLM** models, reducing launch-time performance bottlenecks for next-gen deployments

Software Engineer Intern — 09/2024 – 12/2024
AMD — *Markham, ON*

- Proposed and prototyped a lightweight **ML-based** upscaling PoC using **PyTorch**, **ROCm**, and **Python**, which was later extended into an internal tech conference shaping future display pipeline research
- Developed kernel drivers in **C** and **C++** for next-gen AMD graphics units, improving **hardware** compatibility and performance between the GPU/APU frame buffer and display
- Resolved **25+** kernel-level GPU issues such as crashes, hangs, stability, and performance, leveraging WinDbg, crash dumps, hardware register analysis, ETL traces, and firmware traces, enhancing system responsiveness and stability for the Navi4x and Ryzen AI APU launch achieving the most stable GPU software release, validated by QA testing
- Partnered with Microsoft OS team to optimize OS-GPU interactions in **5+** tickets, analyzing **firmware** and **memory** dumps, resolving initialization failures and performance bottlenecks to ensure full-stack stability
- Contributed to **Linux's** open-source AMD display driver, addressing visual corruption, color calibration, frame synchronization, DSC, and display pipeline issues

Security Developer Co-op — 01/2024 – 5/2024
eSentire — *Waterloo, ON*

- Solely designed and delivered an AI threat analytics dashboard, building full-stack data pipelines in Snowflake, **Python**, and Vue3, that enabled enterprise clients to visualize real-time security threats
- Dashboard success gained executive visibility, leading to VP-level strategy discussions on advanced analytics adoption.
- Cut ingestion latency by **50%+** by optimizing JSON parsing and Snowflake queries, improving automation and responsiveness for high-volume threat data.
- Boosted log-processing speed by **400%** through algorithmic optimizations in **Python**, improving scalability of the analytics platform
- Enhanced analyst productivity by extending an open-source PCAP scrubber with 10+ new features like, GUI, autosave, checksum validation, and **multi-tasking**

Projects

ML Upscaling | ROCm, CUDA, Machine Learning, CNNs, PyTorch, Python

- Prototyped a real-time ML upscaling model for **AMD graphics cards**, leveraging **transformer** based **super sampling** to enhance visual fidelity and performance for potential display pipeline adoption

InvestIQ | Python, LSTMs, IBKR API, CUDA

- Built **CUDA**-accelerated **LSTM** trading system with real-time market data using IBKR API, incorporating **Monte Carlo** volatility forecasting

Bionic Evo | C/C++, Assembly, Neural Networks, TensorFlow, CUDA, STM32

- Engineered a humanoid arm prototype for amputees by utilizing **STM32** and **EMG** sensors, integrating **pattern recognition** to achieve **precise gesture classification** and seamless arm control