Leo Yao

Palo Alto, CA   leoyao@cmu.edu   ☎ +1 (650) 285-0101   leowo   LeoY20

Education

Carnegie Mellon University August 2024 – Present
B.S. in Statistics & Machine Learning and Computer Science *Pittsburgh, PA*

- **QPA:** 3.68/4.00
- **Relevant Coursework:** Introduction to Computer Systems, Introduction to Machine Learning, Principles of Functional Programming, Introduction to Computer Security, Principles of Imperative Computation, Probability and Statistical Inference I/II, Statistical Computing
- **Awards:** Dean's List, High Honors (Spring 2025, Fall 2025)

Experience

Research Assistant September 2025 - Present
FastML Lab *Pittsburgh, PA*

- Implemented a high-granularity quantization-aware training pipeline for FPGA-based GNNs used by CERN's CMS experiment with PyTorch, PyG, and HGQ2 while optimizing for microsecond-latency performance.
- Developing transformers for real-time rapid reconstruction of High-Granularity Calorimeter energy clusters.

Machine Learning Research Assistant June 2025 – August 2025
CMU SPICE Lab *Pittsburgh, PA*

- Engineered a data preprocessing pipeline using R Tidyverse, Pandas, NumPy, and SciKit-Learn to clean and normalize large-scale energy consumption datasets.
- Developed a neural network with PyTorch to forecast households' annual cooling energy, engineering a 75% accuracy improvement by implementing LightGBM-based feature selection and hyperparameter tuning (e.g. learning rate annealing, early stopping).

Projects

Cache Simulator June 2025

- Developed a write-back, write-allocate cache simulator in C with LRU eviction policy.
- Designed a flexible architecture to parse instruction traces and accurately model hit/miss/eviction rates across various cache geometries.

Dynamic Memory Allocator July 2025

- Developed a dynamic memory allocator using segregated free lists in C. Achieved 74.3% memory utilization and 15885 kilo-operations per second, ranking first in the class by performance.

Tiny Linux Shell July 2025

- Built a lightweight Linux shell in C featuring job control and I/O redirection.
- Improved performance and mitigated race conditions by implementing signal masks and custom signal handlers to manage background process synchronization.

Web Proxy Server July 2025

- Created a web proxy server in C. Optimized performance and reduced latency by introducing concurrency with POSIX Threads to handle individual client HTTP GET requests.

Shark File System July 2025

- Developed a thread-safe concurrent FAT-style file system in C with file creation, deletion, renaming, reading, and writing capabilities.
- Optimized performance and correctness with mutex synchronization to mitigate race conditions.

Skills

Programming Languages: Python, C, SQL, Java, SML, Dafny, x86-64 Assembly, R
Frameworks/Libraries: Pandas, NumPy, Scikit-Learn, PyTorch, Matplotlib, Tidyverse
Developer Tools: Git, Linux, PortSwigger Burp Suite, GDB, Valgrind