

# Dahua Feng

Website / GitHub

wwh8us@virginia.edu

## Education

---

**B.S. in Information and Computing Sciences, Peking University**

Beijing, China

*School of Electronics Engineering and Computer Science*

Sep. 2020 - Jul. 2024

- **GPA:** 3.700/4.000 (87.4/100)
- **Thesis:** *A Simulator Design for the Storage of Mobile Devices* (supervised by Prof. Jie Zhang)

## Publications

---

1. **Profiling Apple Silicon Performance for ML Training**, Preprint, 2025

Dahua Feng\*, Zhiming Xu\*, Rongxiang Wang, Felix Xiaozhu Lin

## Research Experience

---

**SEPT Lab, University of Southern California**

Jun. 2025–Present

*Research Assistant*

Advisor: Prof. Mengyuan Li

Project: CPU-GPU communication overhead in confidential computing environments

- Investigated CPU-GPU communication overheads introduced by confidential computing (CC) for LLM inference and fine-tuning; identified the CC authentication protocol as a serialization bottleneck.
- Implemented a block-indexed parallel authentication scheme and a lazy-decryption mechanism to avoid unnecessary decrypting/re-encrypting of unused data.
- Conducted a literature survey and theoretical analysis, implemented the end-to-end system for the scheme, and evaluated its performance.

**XSEL Lab, University of Virginia**

Aug. 2024–Dec. 2024

*Research Assistant*

Advisor: Prof. Felix Lin

Project: Apple Silicon for machine learning

- Focused on evaluating machine learning training performance on Apple Silicon to support affordable, democratized training and fine-tuning; profiled and analyzed performance gaps with NVIDIA GPUs, emphasizing LLM workloads.
- Executed end-to-end training experiments on state-of-the-art generative models and micro-benchmarked BLAS kernels across multiple hardware platforms.
- Consolidated and analyzed profiling data, producing insights and actionable recommendations for practitioners.

**Picasso Lab, University of California, Santa Barbara**

Jul. 2023–Dec. 2023

*Research Intern*

Advisor: Prof. Yufei Ding

Project: CXL-based memory-disaggregated system for DLRM

- Designed and evaluated a CXL-backed memory disaggregation architecture to accelerate DLRM training by sharding embedding tables across GPUs and CXL memory expanders.
- Formally analyzed embedding-table placement and proposed load-balanced memory-allocation algorithms to minimize remote accesses and contention across GPUs.
- Surveyed ML-based and heuristic sharding methods (including RL) and identified practical improvements to reduce communication overhead and improve utilization.

Project: DLRM embedding lookup with NVSHMEM communication

- Integrated NVSHMEM as an inter-GPU communication layer to support asymmetric embedding-table placement and reduce remote lookup latency.
- Collected realistic kernel-level profiling data via CUDA kernels to quantify the impact of placement and batching strategies.
- Built a latency-prediction model for batched embedding lookups and used it to guide placement and batching decisions for improved throughput.

**School of Computer Science, Peking University**

*Research Intern*

Mar. 2022–Jan. 2023

Advisor: Prof. Zhi Yang

Project: DNN acceleration via graph optimization

- Designed and implemented graph-based scheduling and resource-allocation techniques to accelerate DNN computation. Developed a hybrid genetic algorithm combining BFS and dynamic programming to optimize operator ordering and assignments.
- Analyzed Roller and TVM timing/evaluation source code to identify optimization opportunities and benchmarking behavior.
- Implemented resource-allocation support in the IOS scheduler to improve operator throughput and hardware utilization.

*Teaching Experience*

---

**Teaching Assistant of Computer Architectures**

Fall 2023

*School of Electronic Engineering and Computer Science, Peking University*

- Assisted Prof. Jie Zhang in designing and refining course projects; led Q&A sessions; prepared and administered quizzes; contributed to final exam design and grading.

*Skills*

---

**Programming Languages & Software:** C, C++, Python, CUDA C

**Python Packages:** torch, scikit-learn

**Languages:** Mandarin (native), English (proficient), Korean (intermediate)