Eric Liu

Email: eliu4913@usc.edu · Cell: (650) 865-8068 · eliu4913.github.io/
Research Interests: Efficient AI, Trustworthy AI, ML Systems

EDUCATION

University of Southern California, Viterbi School of Engineering – Los Angeles, CA Aug. 2020 – Dec. 2025
B.S. in **Computer Science & Business Administration** | **GPA 3.92/4.0**
M.S. in **Computer Science (Progressive Degree Program)** | **GPA 4.0/4.0**

- **Relevant Courses**: Optimization for Machine Learning, Deep Learning and its Applications, Machine Learning, Applied NLP, Computer Systems, Professional C++, Analysis of Algorithms *(Received A's in all listed courses)*
- **Honor**: Dean's Honor List in all semesters.

PUBLICATIONS

Conference Papers

1. Yunfan Li, **Eric Liu**, Lin Yang.*Trajectory Graph Learning: Aligning with Long Trajectories in Reinforcement Learning Without Reward Design*, NeurIPS 2025 **(Spotlight).**
2. Xiaokang Ye, Jiawei Ren, Yan Zhuang, Xuhong He, Yiming Liang, Yiqing Yang, Mrinaal Dogra, Xianrui Zhong, **Eric Liu**, Kevin Benavente, Rajiv Mandya Nagaraju, Dhruv Vivek Sharma, Ziqiao Ma, Tianmin Shu, Zhiting Hu, Lianhui Qin. *SimWorld: An Open-ended Simulator for Agents in Physical and Social Worlds*, NeurIPS 2025 **(Spotlight).**
3. Qingran Zhou, **Eric Liu**, Alessio Brini. *Graph Neural Networks for Bridge Swap Link Prediction in Uniswap v3*, ACM ACAIF 2025 **(Best Student Paper).**

Preprints and Manuscripts

1. Zheng Wang, **Eric Liu**, Linan Jiang, Zhongkai Yu, Zaifeng Pan, Yue Guan, Yuke Wang, Yufei Ding. *FlashCP: Load-Balanced Communication-Efficient Context Parallelism for LLM Training*, Under Review at MLSys 2026.

RESEARCH EXPERIENCE

Research Assistant May 2025 – Present
University of Southern California

- Leading a research project with Prof. Sai Praneeth Karimireddy on the robust evaluation of Large Language Model (LLM) unlearning.
- Designing adversarial evaluation pipelines to elicit "forgotten" knowledge using prompt optimization, in-context examples, fine-tuning and parameter perturbations.
- Investigating and quantifying the stability of unlearning and the effective "unlearning basin" by analyzing model sensitivity to noise and fine-tuning on unrelated data.

Research Assistant March 2025 – Present
University of California, San Diego, PICASSO Lab

- Worked with Prof. Yufei Ding and PICASSO Lab on *FlashCP*, a context parallelism framework that addresses workload imbalance in long-context training by introducing a "Whole-Doc" sharding strategy and sharding-aware communication.
- Implemented SOTA baselines (Per-Sequence and Per-Document context parallelism), and co-developed core components of *FlashCP*.
- Conducted large-scale experiments on 8×H100 GPUs and AWS clusters with models up to 70B parameters across three public datasets, showing avg. 1.63× end-to-end speed-ups over existing state-of-the-art context parallelism frameworks.
- The resulting manuscript is currently under review for **MLSys 2026**.

Research Assistant March 2025 – Present
University of California, San Diego, SimWorld Team

- Worked with Prof. Lianhui Qin on *SimWorld*, an Unreal Engine 5 simulator that generates large-scale, open-ended urban environments for embodied-AI research.
- Engineered the experimental pipeline for the LLM-driven delivery agent benchmark, including implementing task variants, configuration scripts, and baseline ablation setups.
- Generated and summarized experimental results in the manuscript for **NeurIPS 2025 (accepted as Spotlight)**.

Research Assistant May 2024 – Present

University of California, Los Angeles

- Worked with Prof. Lin Yang on *Trajectory Graph Learning*, a novel framework that formulates policy alignment as a maximum weight independent set (MWIS) problem on a trajectory conflict graph to bypass manual reward design.
- Implemented the core algorithm which leverages UCB-style exploration to compute optimal policies in polynomial time under structural assumptions (e.g., bounded realizability).
- Built the full experimental pipeline to benchmark TGL against standard imitation learning baselines, demonstrating superior performance in preserving long-horizon behavioral coherence.
- Co-authored and submitted the resulting manuscript to **NeurIPS 2025 (accepted as Spotlight)**.

TEACHING EXPERIENCE

Course Producer (CSCI 270 Introduction to Algorithms and Theory of Computing) Jan. 2024 – Dec. 2024

University of Southern California, Viterbi School of Engineering

- Conducted weekly office hours and exam review to assist 150+ students with concepts such as Greedy Algorithms, Divide and Conquer, Dynamic Programming, NP-Complete, and Network Flow.
- Proctored exams, graded assignments, and answered questions from students on discussion forums.

ACADEMIC PROJECTS

ZeRO-Offload Stages and Model Size Impact Jan. 2025 – Apr. 2025

- Reproduced the implementation of ZeRO-Offload stages to transfer optimizer states, gradients, and model parameters from GPU VRAM to CPU memory.
- Conducted experiments on different ZeRO-Offload strategies and evaluated metrics such as peak and average memory use, final accuracy, workload tradeoffs (data transfer vs. forward/backward computations), and total throughput.
- Identified optimal strategies to balance training efficiency and scalability for large-scale transformer models.

Impact of Object Detection Methods for Game-Playing Agents Aug. 2024 – Dec. 2024

- Developed a game-playing agent that leverages advanced object detection pipelines (e.g., YOLOv8, detection transformers) to inform decision-making, balancing the tradeoff between detection accuracy and real-time inference latency.
- Designed and executed experiments that evaluated precision, recall, mAP, and latency, correlating these metrics with in-game performance to quantify trade-offs in detection methods.

Virtual Machine and Compiler (C++, Bison, x86 Assembly) Oct. 2023 – Dec. 2023

- Created a Virtual Machine in C++ with fifteen 32-bit integer registers, 1 KB of stack space, and 3-bit color graphics that support turtle graphics by using template metaprograms, exceptions, and smart pointers.
- Designed a compiler that enabled the generation of assembly instructions from an Abstract Syntax Tree that streamlined code compilation and increased efficiency by 15% compared to open-source solutions.
- Optimized register usage and improved execution efficiency of VM operations by implementing implementing Linear Scan Register Allocation algorithm to map virtual registers to real registers.

SKILLS

- Programming Languages and Frameworks: Python (PyTorch, TensorFlow), CUDA, JAX, C++, C, and Unreal Engine.
- Language: Bilingual in Mandarin and English.