

Tailai Ying

385-256-3856 | tty6@cornell.edu | [tailaiying32.github.io](https://github.com/tailaiying32) | [linkedin.com/in/tailai-ying-099041260](https://www.linkedin.com/in/tailai-ying-099041260) | github.com/tailaiying32

EDUCATION

College of Engineering, Cornell University
B.S. in Computer Science

Ithaca, NY
Expected May 2027

TECHNICAL SKILLS

Languages: Python, Java, C/C++, Go, JavaScript, TypeScript, SQL, OCaml, HTML/CSS, XML

Frameworks/Libraries: React, Next.js, Node.js, Flask, Spring Boot, Tailwind CSS

Tools/Databases: Docker, SLURM, Linux, Git, GitHub Actions, PostgreSQL, SQLite, Prisma

EXPERIENCE

Research Assistant

May 2025 – Present

EmPRISE Lab

Ithaca, NY

- Owned theoretical and practical implementation for active learning core of personalized caregiving robotics framework, learning user reachability with 14% more accuracy than current methods.
- Engineered extensive distributed data analysis and visualization pipelines to validate performance, ensuring data integrity and fault tolerance.
- Enhanced framework with CUDA support, batched processing, distributed execution, and hyperparameter search with SLURM and shell scripts, resulting in over 10× model efficiency and over 50× reduction in model training time.
- Validated theoretical/real-world performance via 20+ ablation studies and 10-user study. Achieved statistically significant results in human-robot engagement ($p < 0.001$), and co-authored (co-2nd author) paper submitted to RSS 2026.

Machine Learning Intern

May 2025 – Aug 2025

Aria Lab

Salt Lake City, UT

- Built a high-fidelity 3D swarm simulation with modular controllers supporting diverse multi-agent configurations.
- Optimized simulation throughput by parallelizing agent updates, enabling the simultaneous evaluation of 100+ swarm configurations across HPC nodes.
- Engineered an automated data analysis pipeline using novelty search and clustering to programmatically discover and classify 10+ previously unknown agent behaviors.

SELECTED PROJECTS

Borglite - Distributed Task Scheduler | *Go, gRPC, Protobuf*

Jan 2026 – Feb 2026

- Designed and implemented a distributed task scheduler capable of managing tasks across multiple worker nodes by engineering a Master-Agent architecture with gRPC and Protobuf for efficient RPC communication.
- Achieved fault tolerance and system resilience by implementing a heartbeat monitoring system that automatically detects node failures within 5 seconds and re-schedules affected tasks using a custom retry mechanism.
- Optimized resource utilization across the cluster by developing a custom First-Fit scheduling algorithm that dynamically assigns tasks to agents based on real-time CPU and memory availability.
- Streamlined cluster management by creating a user-friendly CLI tool that allows users to submit jobs, monitor real-time status updates, and inspect agent health with sub-second latency.

Jarvis - Multi-modal Voice Assistant | *C++, llama.cpp, sherpa-onnx*

Dec 2025 – Jan 2026

- Prototyped a high-fidelity audio-first voice assistant using Qwen3, Whisper, and Sherpa-ONNX, optimizing for natural speech flow and low-latency user interaction.
- Engineered an intelligent phrase-boundary detection system to optimize real-time inference, reducing end-to-end latency to sub-1s to ensure a seamless consumer experience.
- Orchestrated concurrent model execution and audio pre-buffering to maintain pipeline performance and scalability.

CritterEvo - Artificial Life Simulator | *Java*

Dec 2024 – Feb 2025

- Engineered evolutionary ecosystem simulator in Java, featuring procedural world generation, neural network controllers, and genetic algorithms to simulate emergent behavior and natural selection.
- Optimized performance with multi-threading and lazy loading, increasing simulation throughput by over 80%.
- Optimized entity navigation by implementing an A* pathfinding engine with state caching, significantly reducing computational overhead during high-density simulations.
- Achieved 95% test coverage with comprehensive JUnit suite, ensuring robust functionality across edge cases.

Lockd - Smart Lock System | *Raspberry Pi, React Native, Flask*

Oct 2024

- Developed an IoT smart lock system featuring remote control and vibration/sound anomaly detection with under 200ms alert latency from edge device to client.
- Awarded Finalist and Beginner's Prize at BigRedHacks (135+ hackers) for technical implementation and system design.
- Implemented real-time push and email notification services to ensure immediate user response to security threats.