Chengyang Shi

chengysh132@icloud.com · Homepage · Google Scholar · LinkedIn · GitHub

Education

University of Michigan, Computer Science & Mathematics, *Undergraduate* Aug. 2025 - Present

Shanghai Jiaotong University, Electrical & Computer Eng., *Undergraduate* Sept. 2023 - Present

Core Courses: Data Structures & Algorithms (A+), Machine Learning (A), Software Design (A+), Comp. Finance (A+), Stochastic Processes (A), Combinatorics & Graph Theory (A), Probabilistic Methods (A)

Professional Experience

Student Researcher | ARA Labs *(Mentor: Dr. Jiachen Liu)* Jun. 2026 - Sep. 2026

- **Agent-Native Research Artifact (ARA):** Core contributor (top committer) to the open-source ARA protocol (680+ GitHub stars, 50+ forks), which replaces the narrative paper with a machine-executable research package for AI scientists; featured in *IEEE Spectrum*. Built the core agent skills: the *compiler* (papers/repos/logs → ARA), *research-manager* (live provenance recording), *visualizer* (interactive research-trajectory viewer), and *research-foresight* (queryable world model), with which a frozen LLM agent cleared all 7 levels of an unseen ARC-AGI-3 game (∼1% frontier-LLM pass rate).
- **Open-Endedness Bench (OEB)** *(Under review at ICLR 2027, arXiv, Blog)*: Built a benchmark-agnostic methodology that scores a research agent's *epistemic process* (how it forms, tests, and revises hypotheses) from its execution record alone, never reading a reference answer or outcome score.
  - **Epistemic Event Graph:** Compiled raw agent logs into a graph linking stated propositions to the executed actions that test them; an LLM judge only answers closed 3-vote questions over code-verified excerpts.
  - **Measurement:** Scored four competence axes (evidence, experiment, revision, no reward hacking) and six persona traits under one rule (opportunities taken / offered); median rescoring drift is 0.04.
  - **Findings:** On 119 runs over 12 tasks from 3 benchmarks (LLM post-training, chip design, nanoGPT speedrun), only 16–29% of improvements agents claim are real; winning runs keep testing new ideas late (9 of 10 tasks); and the model, not the task, sets an agent's research persona (43% vs. 7% of variance).
- **Agentic Auto-Research is Fuzz Testing** *(arXiv)*: Framed auto-research as greybox fuzzing guided by dense epistemic-progress signals; our agent finds a hidden physical law that score-optimizing baselines miss.

Research Experience

WebEvolve: Version-Aware Benchmark for Web Agents Advisor: Prof. Honglak Lee

- **Drift Taxonomy & Capability Mapping:** Proposed a seven-layer taxonomy of website drift (surface to runtime) mapped to five agent capability dimensions, enabling failure diagnosis beyond aggregate success rates.
- **Version-Aware Benchmark Pipeline:** Built the framework and Linkding/SimpleWiki branches: multi-version sites as reproducible Docker environments; version-affected tasks found via knowledge-graph diffing.
- **Grounding-Aware Evaluation:** Evaluated Qwen3-VL-8B on hundreds of tasks across four evolving platforms with an evidence-verification protocol, revealing failures concentrated in specific drift categories.

Automatic Detection of Accessibility Navigation Failures Advisor: Prof. Yue Jiang

Under review at CHI 2027 (arXiv)

- **LLM-Synthesized Test Environments:** Built an agentic pipeline (Claude Code + GPT-5.6 Sol) that synthesizes 31 interactive websites modeled on real-world sites and tasks as the checker benchmark.
- **Trace-Level Verification:** Formalized 13 categories of screen-reader navigation failures as executable LTL properties, monitored at runtime over exploration traces (accessibility tree, DOM, focus).
- **LLM-Agent Baseline & Evaluation:** Implemented a TaskAudit-style LLM-agent baseline; our verifier confirmed 274 of 309 reported failures (88.7% precision) vs. 158 at 57.2% for the agent, and out-detected Axe/WAVE.

Skills

- **Machine Learning:** LLM agents & agentic workflows, deep learning, NLP, reinforcement learning
- **Evaluation & Research:** benchmark design, LLM-as-judge evaluation, ablation studies, statistical analysis
- **Frameworks & Tools:** PyTorch, HuggingFace Transformers, vLLM, Claude Code, Codex, MCP, Pandas, Docker, Git, Linux
- **Programming:** Python, C/C++