

# Xi Chen

1578 Neil Ave ◊ Columbus, Ohio 43210  
chen.10183@buckeyemail.osu.edu  
cx441000319.github.io

## EDUCATION

---

### The Ohio State University

*Columbus, OH, USA*

Ph.D. Candidate in Computer Science and Engineering

*2020 – Present*

- Advisors: Prof. Andrew Perrault and Prof. Zhihui Zhu
- RESEARCH INTERESTS: **deep reinforcement learning (RL)**; **machine learning interpretability**; **large language models (LLMs)**

### Xidian University

*Xi'an, Shaanxi, China*

Bachelor in Software Engineering

*2016 – 2020*

- GPA: 3.9/4.0; Rank: 15/445 (Top 3%)
- Class of Software Engineering for Talented (40 out of 445 students)

## PUBLICATIONS

---

- Understanding Learned Representations and Action Collapse in Visual Reinforcement Learning  
**Xi Chen**, Zhihui Zhu, and Andrew Perrault  
*RLC 2025*
- The Distributional Reward Critic Framework for Reinforcement Learning Under Perturbed Rewards  
**Xi Chen**, Zhihui Zhu, and Andrew Perrault  
*AAAI 2025*
- Using Reinforcement Learning for Multi-Objective Cluster-Level Optimization of Non-Pharmaceutical Interventions for Infectious Disease  
Xueqiao Peng, Jiaqi Xu, **Xi Chen**, Dinh Song An Nguyen, Andrew Perrault  
*ML4H 2023*

## CURRENT RESEARCH PROJECTS

---

**Motivation.** In current Reinforcement Learning with Verifiable Rewards (RLVR) tasks, RL has not fully showcased its strengths for LLMs.

**Ongoing.** (1) Conducting controlled experiments to reveal the limitations of today's RL fine-tuning setups for LLMs; (2) analyzing the observed failure modes to introduce a new RL paradigm for LLMs.

## AWARDS

---

**Eleanor Quinlan Graduate Teaching Award** — CSE Department, The Ohio State University 2025

## ACADEMIC SERVICES

---

<b>AAAI</b>	Student Volunteer	2025
<b>IEEE VIS</b>	Student Volunteer	2021

## TECHNICAL SKILLS

---

<b>Programming</b>	Python; C/C++; Bash; (Java; MATLAB; SQL)
<b>ML/LLM Frameworks</b>	PyTorch (primary); TensorFlow/Keras (familiar)
<b>RL &amp; LLM</b>	PPO; RLHF; DPO; RLVR/GRPO
<b>Systems &amp; Distributed</b>	Linux; Slurm; PyTorch DDP (via Accelerate); CUDA