Yiru Wu

ML Engineer (Reinforcement Learning / Robotics)
Tel: 608-209-7452|E-mail: yw8174@nyu.edu
Work Authorization: F-1 STEM OPT (36 months). CPT available May/Jun 2026

PROFILE

Machine Learning Engineer with focused expertise in reinforcement learning and robotics, demonstrated through end-to-end development of large-scale, GPU-accelerated training systems. Experienced in building vectorized simulation environments (Isaac Gym, MuJoCo) with 4,096+ parallel actors, optimizing PPO and actor-critic algorithms to achieve 100x speed improvements and 30% faster convergence. Proven ability to design reward functions, ensure full reproducibility via MLOps practices (Docker, Hydra, W&B), and deploy robust controllers for real-world robotic locomotion. Strong engineering foundation in Python, PyTorch, and scalable ML infrastructure, complemented by applied research in continuous control and computer vision.

EDUCATION

New York University - Courant Institute of Mathematical Sciences New York, NY
MSc in Computer Science Sep 2025-May 2027 (Expected)

- Specializing in Machine Learning Systems, Reinforcement Learning, Optimization.

University of Wisconsin - Madison Madison, WI
BSc in Computer Science Aug 2023-May 2025

- Core modules: ML, AI, Probability & Information Theory, Optimization, Discrete Math, Statistics, Operating Systems.

The Chinese University of Hong Kong, Shenzhen Shenzhen, China
BEng in Electrical & Computer Engineering (Transfer Coursework) Sep 2021-May 2023

PROFESSIONAL EXPERIENCE

UW - Madison: Large-Scale Reinforcement Learning for Continuous Control (Quadruped Case Study) Madison, WI
Research Intern | Supervisor: Prof. Xiaobin Xiong Jul 2025 - Aug 2025

Tech Stack: Python, PyTorch, Isaac Gym/MuJoCo (batched physics), Docker, Git/GitHub Actions

- Engineered a large-scale, **vectorized simulation environment** in Isaac Gym with 4,096 parallel actors, optimizing memory layout and batched physics in PyTorch to achieve a **100x speedup (90,000 env steps/s) over CPU baselines.**
- Designed and implemented a real-time control pipeline using PyTorch to predict **desired ground-reaction forces** and map them to actuator torques, enabling stable and precise low-level control for a quadrupedal agent.
- Innovated on the PPO algorithm by integrating **dynamics-aware features** into the critic network, enhancing sample efficiency and **boosting performance by 30% faster convergence** and **10% higher episode return** across all random seeds.
- Productionized the ML research lifecycle using Weights & Biases and TensorBoard for tracking, enforced reproducibility via **deterministic seeding** and Docker, automated testing with CI/CD including throughput KPIs for regression detection.

UW - Madison: Data-Driven RL Pipeline for Bipedal Control (2D Strider) Madison, WI
Research Intern | Supervisor: Prof. Xiaobin Xiong Sep 2023-May 2024

Tech Stack: Python, PyTorch, MuJoCo, GitHub Actions

- Engineered a **high-throughput reinforcement learning pipeline** in Python using PyTorch, implementing batched rollouts, pinned memory, and asynchronous GPU transfers to significantly reduce wall-clock training time.
- Designed and implemented a specialized curriculum learning strategy with reward shaping for smooth stand-to-walk transitions in MuJoCo, successfully reducing fall rates while achieving target locomotion speed and energy efficiency metrics.
- Developed a comprehensive evaluation framework featuring extensive seed sweeps and automated metrics export, ensuring full reproducibility and enabling detailed ablation studies for algorithm improvement.
- Established automated CI/CD workflows using GitHub Actions to maintain code quality and experimental integrity throughout the research cycle, from development to final evaluation.

Tsinghua University: AI Methods for Power-System Operation (Forecasting & Optimization) Beijing, China
Research Intern | Supervisor: Prof. Chenye Wu Jun 2022-Dec 2022

Tech Stack: Python, NumPy/SciPy, Gurobi/Ipopt, GAMS

- Developed and implemented **Gaussian Process regression models** in Python using NumPy/SciPy for accurate power generation forecasting, effectively reducing prediction error and lowering offline economic dispatch costs compared to rule-based baselines.
- Formulated a market equilibrium problem as a Mathematical Program with Equilibrium Constraints **(MPEC)** and implemented it in GAMS, leveraging Gurobi and Ipopt solvers to enhance solution stability and reduce runtime on standard benchmark cases.
- Engineered a full-stack analytical pipeline that integrated statistical forecasting with large-scale optimization, translating model predictions into actionable operational insights to improve power-system decision-making.

Ultrapower Beijing, China
Software Engineer Intern Jun 2022-Jul 2022

Tech Stack: Java, Python, Linux, Maven, Git

- Stabilized a critical **carrier-grade monitoring system** for China Mobile by diagnosing and resolving complex JAR dependency conflicts in Maven, eliminating recurring build failures and ensuring system reliability.
- Enhanced team development workflows by introducing standardized Git branching strategies and implementing automated testing, which significantly reduced QA cycle time and improved overall productivity.
- Prototyped a predictive **RNN-based forecasting model** in Python to anticipate system demand, achieving a measurable improvement in Mean Absolute Percentage Error (MAPE) over the existing rule-based baseline.

SELECTED PROJECTS

UW-Madison / Autonomous Robotics: Maze Navigation Robot (Particle Filter + A*) Mar 2025-Apr 2025

Tech Stack: Python, Webots (robotics simulator), A* (path planning), Particle Filter (sensor denoising)

- Developed a **maze navigation robot** system in Python using the Webots simulator, implementing a **particle filter** to denoise LiDAR/radar data and maintain accurate state estimation under sensor uncertainty.
- Engineered the motion planning stack by integrating a **global A path planning* algorithm** with a local feedback controller for differential drive kinematics, enabling reliable and collision-free maze traversal.
- Designed and validated a full-stack robotic controller that successfully met all performance metrics, including navigating unknown maze environments and reliably reaching target goals within a 5-minute timeframe.

UW-Madison / Deep Learning in CV: Center - based One-Stage Object Detector (FCOS) in PyTorch Mar 2025-Apr 2025

Tech Stack: PyTorch, CNN, FPN (Feature Pyramid Network), Anchor-free Detection (FCOS), NMS

- Implemented an end-to-end **FCOS** (Fully Convolutional One-Stage) object detector from scratch in PyTorch, designing the network architecture with a CNN backbone, Feature Pyramid Network (FPN), and centerness prediction head.
- Engineered the complete detection pipeline, including anchor-free bounding box regression, classification, and Non-Maximum Suppression (NMS)-based inference to achieve robust multi-scale object detection.
- Trained and validated the model on standard datasets, rigorously testing the pipeline to ensure correctness, reproducibility, and a deep understanding of modern one-stage detection methodologies.

UW-Madison / Machine Learning in Healthcare: Breast Cancer Classification with Scikit-Learn Mar 2025-Apr 2025

Tech Stack: Python 3.8, scikit-learn (Decision Tree/KNN/Logistic Regression), Matplotlib, Linux Ubuntu

- Developed and evaluated **three supervised learning models** - Decision Tree, KNN, and Logistic Regression using scikit-learn - on the Wisconsin Breast Cancer Dataset (569 samples, 30 features) to classify tumors as benign or malignant.
- Performed feature importance analysis to identify and select the top **three most predictive features**, refining the dataset to improve model interpretability and generalization performance.
- Trained and validated all models under a rigorous train-test split, achieving competitive performance metrics with both **Logistic Regression and KNN**, while highlighting the trade-off between accuracy and explainability in Decision Trees.
- Visualized results and model performance using Matplotlib, including confusion matrices and feature importance plots, to communicate findings effectively and support clinical interpretability.

SKILLS

Programming	Python, PyTorch, NumPy, Pandas, Gurobi, Ipopt, GAMS
ML/RL	PPO, Actor-Critic, Gaussian Processes, CNNs, Reward Design, Ablation Studies
MLOps & Eng	Hydra, Weights & Biases, TensorBoard, Docker, GitHub Actions, Deterministic Training
Simulation	Isaac Gym (vectorized), MuJoCo, Parallel Data Collection (4k+ actors), GPU Optimization
Systems	Performance Profiling (wall-clock/throughput), Regression Detection, Metric Instrumentation
Languages	English (Professional), Mandarin (Native)