Chengshuo (Danny) Jiang

Email: jdanny@umich.edu | Phone: (678) 559 6020 | Atlanta, GA

EDUCATION

University of Michigan–Ann Arbor, Ann Arbor, MI — Sept 2022 – May 2025
- **Computer Science** (BSE) | GPA: 3.95/4.00

Georgia Institute of Technology — Sept 2025 – May 2027
- **Computer Science** (MS)

SKILLS

- Programming Languages: Python, Java, C++, JavaScript, TypeScript, Go, PHP, HTML/CSS, Bash, SQL
- Software & Frameworks: AWS, Azure, React, Flask, REST API, Linux, Docker, Kubernetes, Git, CI/CD
- Portfolio: https://danny-jiang.com/

PROFESSIONAL EXPERIENCE

YIWEI Medical Technology Co., Ltd, Shenzhen, China — Aug 2024 – Jan 2025
Deep Learning Research Engineer (Remote)
- Developed a hybrid LSTM-CNN model using TensorFlow to predict autism spectrum disorder (ASD), integrating temporal patterns from time-series eye-tracking data and spatial features from scan path images, achieving enhanced diagnostic accuracy
- Built a Java-based backend (Spring Boot) to manage API requests and data flow, paired with a Next.js frontend to deliver an intuitive interface for healthcare professionals, advancing early ASD diagnostics
- Preprocessed diverse eye-tracking datasets using Pandas and NumPy through normalization, augmentation, and feature extraction, ensuring compatibility for multimodal model input
- Deployed the model within a scalable web application on Alibaba Cloud using Kubernetes for container orchestration, ensuring high availability across a multi-node cluster

SUSE Linux s.r.o, Prague, CZ — Jun 2023 – Aug 2023
Software Engineer Intern
- Developed and optimized containerized Python scripts for Project Iguana on the Factory Team, creating an initial ramdisk to automate system environment setup across diverse hardware using container orchestration
- Designed Docker containers for disk partitioning, tested across 5 devices with 5 configurations each, ensuring cross-hardware reliability via the company's CI/CD pipeline
- Reduced partitioning runtime by 8% through regex input parsing optimizations, enhancing performance and efficiency of the setup process
- Collaborated with a team of 4 engineers to scale the solution for 100+ deployments, ensuring consistent performance and reliability across production environments

PROJECT EXPERIENCE

Distributed Paxos Key/Value Service — Mar 2025 – May 2025
- Developed a fault-tolerant distributed key/value store in Go, integrating the Paxos consensus algorithm to guarantee data consistency across nodes, overcoming challenges like network partitions and node failures
- Harnessed Go's concurrency model with goroutines and channels to orchestrate the Paxos roles (proposer, acceptor, learner), enabling efficient parallel execution and deadlock-free communication under high load
- Boosted system performance by optimizing message passing in Paxos, cutting latency by 30% and increasing throughput via streamlined network protocols and in-memory state caching
- Performed extensive testing with over 100 test cases, simulating failures like network partitions, node crashes, and message delays, achieving data consistency and availability across a wide range of scenarios

Social Media Web App — Mar 2024 – Apr 2024
- Developed a full-stack social media application using React for dynamic front-end interactions and Python with Flask for a responsive back-end API, deployed on AWS to ensure scalability and high availability.
- Implemented secure user authentication in Python, leveraging bcrypt for password salting and Flask-Session for session management, enhancing data security and user convenience.
- Designed and managed a relational database with SQL and SQLAlchemy in Python, optimizing data storage and retrieval for 50 users and 2,000 posts.
- Validated system performance through extensive testing, utilizing Cypress for front-end functionality and Pytest for back-end reliability, ensuring a seamless user experience under load.