Gan Xu

6640 Washington Ave, St. Louis, MO 63130 | 984-888-6951 | gan.xu@icloud.com | ganxu.science | Github

EDUCATION

- MS, Computer Science, Washington University in St. Louis, GPA: 3.53 09/2018- 05/2021
- BS, Computer Science & Mathematics, University of North Carolina at Chapel Hill, Major GPA: 3.64 08/2015- 12/2017

TECHNICAL SKILLS

- **Programming languages:** Python, Java, C/C++, Shell, JavaScript, HTML, CSS, Go
- **Framework and Tools:** Git, Kafka, Docker, Maven, Flask, Spring Boot/MVC, MySQL, MongoDB, React
- **Related Coursework**: Algorithms, Data Structures, Databases, Operating System, Computer Networks, High Performance Computer System, System Security, Artificial Intelligence, Machine Learning, Bayesian Methods in Machine Learning etc.

WORK EXPERIENCE

Washington University **Saint Louis, MO**
Graduate Research Assistant *Feb 2019 - Nov 2020*

- Proposed innovative method to improve the **communication performance** over unreliable networks for distributed multi-agent algorithms, including message split and reconstruction, customized RUDP protocol, forward error correction and etc.
- Collaborated with **Raytheon** BBN Technologies on DAPRA funded **distributed AI** project, details available upon approval.

University of North Carolina **Chapel Hill, NC**
Assistant Bioinformatic Analyst - Full-time *Feb 2018 - Aug 2018*

- Introduced scripts to pull sequencing data(GB per entry) from public biological databases(e.g NCBI).
- Set up work environment on cluster with SLURM workload scheduler. Migrated old workflows from **LSF** platform to **SLURM**.
- Designed and optimized workflow pipelines for I/O and CPU heavy job, reduced **50%** idle time for some experiments.

PROJECTS

Distributed Agent Workflow Scheduling
Java, Maven, Kafka, Jenkins, SLF4J *May 2019 - Nov 2020*

- Mapped scheduling problems for distributed agents to be solved by distributed constraint optimization(**DCOP**) framework.
- Built a real-time messaging system for distributed agents to coordinate with each other based on **Apache Kafka**.
- Deployed maximum gain messaging(**MGM**) algorithm, an anytime algorithm allowing agents get valid even if interrupted.
- Created APIs based on the need of other modules in the project to access optimization functions and results.

Multi-Room Chat Server(Web Application) **Github**
JavaScript, Node.js, HTML, CSS, MongoDB, Socket.IO *Jun 2020 - Aug 2020*

- Designed a real-time multi-room chat server using **Node.JS** and **Socket.IO**.
- Implemented both client-server and chat-server to realize the functions, saved chat history with **MongoDB**.
- Automated system deployment with **Docker**, and operated the online application on an **AWS EC2** Instance to improve the performance and make good management of the application.

Smart Pet Feeder **Github**
Assembly, Shell, Python, C, AWS IoT/EC2, Raspberry Pi *Sep 2019 - Dec 2019*

- Designed and prototyped an automated pet food dispenser based on low power programmable wireless devices.
- Deployed **AWS IoT** to receive data, send instructions and allow easy scheduling and dispensing of pet food from cloud.
- Implemented facial recognition with **SVM** algorithm for pets identification. On RPi 3, the system is able to train model with limited sized samples within minutes and distinguish pets identity within **1s** with onboard CPU with trained models.
- Designed machine learning algorithms with IoT sensors to monitor pet feeding habits and detect abnormal situations.

Pysbatch **Github**
Python, SLURM, UNIX, Linux, Twine *Aug 2017 to Dec 2017*

- Implemented a python library wrapping UNIX/Linux system calls and **SLURM** command. The library enables users to set up complicated pipeline workflows using only python functions and avoid Shell script.
- Provided simplified options for users to set job dependency and limit concurrent jobs by pre-set user quota.
- Packaged and released on PyPI and conda-forge platforms, downloaded over **2000** times.