JUNSHANG JIA (*GreenCardHolder*)

junshanj@andrew.cmu.edu ⋄ 412-403-0254 ⋄ Linkedin ⋄ Github ⋄ Pittsburgh, PA ⋄ github.io/Junshang-Jia/

EDUCATION

Carnegie Mellon University　　　　　　　　　　　　　　　　Aug 2023 - Expected Dec 2024
Master of Science in Information Networking, GPA: 4.0/4.0
Relevant Coursework: 15213 Computer Systems(C), 10601 Machine Learning(Python), 15440 Distributed Systems(Go)

University of Pittsburgh　　　　　　　　　　　　　　　　　　Aug 2019 - May 2023
Bachelor of Science in Computer Science, GPA: 3.9/4.0
Relevant Coursework: Operating Systems(C), Artificial Intelligence(Python), Functional Programming(Haskell)

SKILLS

Programming Languages: Java,C/C++,Python(PyTorch,Pandas/Numpy), JavaScript,TypeScript,SQL,HTML/CSS, Go
Web Technologies: React.js, Node.js, Angular, NoSQL, MySQL, MongoDB, PostgreSQL,Spring Boot, Flask
Tools: Linux, Git, Bash/Shell, AWS, Azure, Google Cloud, VS Code, Docker,Github Actions, CI/CD

WORK EXPERIENCE

Software Developer Intern　　　　　　　　　　　　　　　　　Jun 2021 - Sep 2021
Ericsson　　　　　　　　　　　　　　　　　　　　　　　　　　　*Beijing, China*

- Conducted and accomplished a cutting-edge, web-based management system to optimize product development lifecycle of 5G radio base stations and increase overall product quality by leveraging **Java**.
- Designed an ER diagram based on thorough product documentation, accomplishing a **MySQL** database with optimized indexing and schema design, resulting in a **40%** increase in database query performance.
- Developed a dynamic web application leveraging **Spring Boot** framework and **RESTful APIs** allowed users to manage millions of data by CRUD operations.
- Engineered highly interactive frontend visual interfaces by **React** and Antd framework,incorporating dynamic features to enrich user interactions in search and information checking behaviors.

PROJECTS

Opensoundscape

- Contributed an **open-source machine learning library**, OpenSoundscape, that allows users to select the magnitude or power of the Spectrogram and generate specific formats of annotation files employing **Python**.
- Constructed a comprehensive set of meticulous test cases to ensure thorough **testing** of software functionalities and identify any potential bugs or issues may arise during development process.
- Collaborated with team members in development process by posting pull requests and engaging in discussions to review and enhance code quality and project outcomes.

Raft Protocol

- Implemented the leader election mechanism as per the Raft protocol, ensuring that the system could automatically select a leader to coordinate and manage data replication by **Go**.
- Designed and implemented the log replication process, ensuring uniform data consistency across all cluster nodes and committing log entries upon acknowledgment by a majority of nodes.
- Establish clear and structured **RPC** procedures for leader election, log replication, and heartbeat communication.

Memory Allocator

- Designed and implemented a memory allocator from scratch for efficient memory management using **C**.
- Applied various strategies(segregated free list, explicit lists, and memory alignment) to mitigate fragmentation.
- Created a robust and efficient memory allocation algorithm outperformed existing solutions by improving throughput memory utilization by an impressive **70%**.

Bitcoin Miner

- Designed and developed a distributed system using a custom Live Sequence Protocol (LSP) to efficiently perform a compute-intensive task – Bitcoin mining by **Go**.
- Introduced a time-triggered epoch system, bolstering robustness through periodic transmission of heartbeat messages.
- Optimized a **load balancing** system to ensure both efficiency and equitable task allocation, taking into account parameters such as request size and order of arrival.