Yinxuan Fu

Pittsburgh, PA

(248) 480-6177 yinxuanf@andrew.cmu.edu github.com/fustinose linkedin.com/in/yinxuanfu/

Education

- **Carnegie Mellon University** **Pittsburgh**
  B.S. Computer Science (**GPA:** 3.86/4.00) *August 2018 – December 2021*

Work Experience

ByteDance **Shanghai**
Software Engineer Intern, AI Lab Speech and Audio Team, C++ *May 2021 – August 2021*

- Implemented the distributed virtual-consensus-based log system, and achieved near 400 *Append* QPS for 500KB log entries.
- Designed new system structure to make the existing distributed search system cloud native, and integrated the aforementioned log system into the search engine.
- Optimized the serialization and deserialization of snapshot files for inverted index, achieving 10x speedup on GBs of data.
- Developed a database service based on *RocksDB* that serves as the source of truth in the system.

TalkMeUp Inc. **Pittsburgh**
Software Engineer Intern, Backend Developer, JavaScript *March 2020 – October 2020*

- Led the design and implementation of backend schema relations with *Node.js* and *MongoDB* to handle inter-entity interaction and various user scenarios.
- Proposed the use of inverted index to leave room for potential metric analysis.
- Implemented service methods for the frontend webpage, including connecting frontend requests to back-end AI service module, data aggregation, and regression analysis.
- Optimized database I/O code and achieved 5x speedup.

Projects

Distributed System Course Work **Carnegie Mellon University**
Golang *August 2020 - January 2021*

- Implemented a distributed computing system with a master-worker structure and a customized network protocol, and used it to develop a pseudo bitcoin miner.
- Implemented the distributed consensus protocol *Raft*.
- Implemented a distributed key-value store with consistent hashing for sharding and distributed caching.

Ray Project **github.com/ray-project/ray**
Open Source Contributer, Python *January 2020 – May 2020*

- Benchmarked the performance of *spacy-ray* and diagnosed reasons for unsatisfying performance.
- Worked on *ray collective* feature with mentor, and implemented the multi-stream support for CUDA with inspiration from PyTorch.
- Implemented *collective actor* feature that provides users with Message Passing Interface in a ray-native programming experience.

Skills

Programming Languages: C++, Python, Java, Golang, JavaScript, SML, LaTeX
Frameworks and Softwares: RocksDB, MongoDB, AWS, Linux, PostgreSQL, Node.js, React
Knowledge Base: Distributed System, System Design, Database, Algorithm, Machine Learning