Zehao (Zach) Guan

zehaog@cs.cmu.edu | (412) 641-0276
linkedin.com/in/zach-guan | github.com/zach96guan

Education

Carnegie Mellon University — Pittsburgh, PA
M.S. in Artificial Intelligence and Innovation, School of Computer Science, GPA: 3.83/4.0 — *May 2020*

Zhejiang University — Hangzhou, China
B.E. in Information Engineering, School of Information and Electronic Engineering, GPA: 3.76/4.0 — *Jun. 2018*

Skills

Coursework: Distributed Systems, Principles of Software Construction, Intro to Computer Systems, Database Systems, Practical Data Science, Machine Learning, Machine Translation, Deep Learning(TA), Natural Language Processing(TA).
Programming Languages: Python, Java, Go, C/C++, JavaScript, SQL, Scala, Shell, MATLAB.
Tools and Frameworks: Linux, Git, AWS, Kubernetes, Docker, CUDA, Hive, Hadoop, Spark, MySQL, MongoDB, Django, Vue.js, TensorFlow, PyTorch.

Experience

Walmart Labs — Sunnyvale, CA
Software Engineer Intern, Personalization Team — *May 2019 – Aug. 2019*

- Designed a personalized recommender for implicit data, and developed latent factor model based collaborative filtering, including pointwise/pairwise non-negative matrix factorization and factorization machine methods.
- Aggregated million-scale Walmart.com transaction data by Hive, extracted and preprocessed raw data with PySpark.
- Visualized insights from sampled data by Tableau and utilized t-SNE and K-means to analyze latent representations.
- Applied ranking-based information retrieval metrics such as Hit Ratio, Mean Average Precision and Normalized Discounted Cumulative Gain in Python for graded relevance evaluation based on the customers preference.

University of Michigan — Ann Arbor, MI
Research Assistant, Electronic Engineering and Computer Science Dept. — *Jul. 2017 – Oct. 2017*

- Developed a prototype system integrated with wearable sensors and remote monitor display for U-M athletic swimmers.
- Assembled varied sensors, Bluetooth module and micro-controller on Arduino by C++ to collect real-time detected biometric data, and utilized Eagle to redesign schematics and circuit layouts to improve communication efficiency.
- Designed Android App to receive transmitted data and multi-thread programs to establish synchronous connections.

Nokia Siemens Networks Co., Ltd. — Hangzhou, China
Software Engineer Intern — *Aug. 2016 – Nov. 2016*

- Implemented image processing, segmentation and feature extraction by OpenCV in Python, and applied CNNs based model to build a human face recognition system in TensorFlow.
- Embedded the framework in Raspberry Pi with JSON-based front-end display interface to complete a human-machine interactive demo, which is capable of pushing customized information of recognized targets.
- Created an open source platform for mobile development on IoT and succeeded in application for Utility Model Patent.

Selected Projects

Chatbot for Interactive Voice Response System — *Dec. 2019 – Apr. 2020*

- Designed an interactive web application for phone trees of insurance company with Vue.js and DynamoDB.
- Provided RESTful APIs and built websocket connections for event-driven data transfer between server and clients.
- Utilized Twilio tools and Google Speech for real-time utterance transcription and sent it to backend with Flask.

Speech-to-Text Transcription System — *Feb. 2019 – Apr. 2019*

- Developed Attention-based deep neural networks in PyTorch, with a combination of RNNs, CNNs and Dense Nets to design an End-to-End system for speech utterance to corresponding text transcription.
- Evaluated on WSJ dataset with the Levenshtein distance and perplexity, and ranked top 5% on Kaggle leaderboard.

Intro to Computer Systems Labs — *May 2018 – Aug. 2018*

- Wrote a general-purpose cache simulator, and optimized a small matrix transpose kernel to minimize number of misses.
- Realized a simple Unix shell that supports basic commands of job control and implemented I/O redirection.
- Implemented a dynamic memory allocator by designing segregated free list and manipulating bits in header/footer with LIFO and best-fit policy to improve both space utilization and throughput.
- Built a concurrent HTTP proxy that caches recently accessed web objects to handle multiple client requests in parallel.