Xuesen WEN

Durham, NC | 919-433-7962 | xuesenw98@gmail.com | LinkedIn

Education

Duke University

Master of Science, Software Engineering GPA : 3.78/ 4.0 **Aug 2020 – Dec 2023**

Related Courses: Operating System, Database System, Computer Network, Algorithms, Software Engineering, Engineering Robust Server Software, Computer and Information Security, Performance Optimization and Parallelism, Cloud Data Engineering

University of Nottingham, UK Campus

Bachelor of Engineering, Electronic & Computer Engineering GPA: 3.5/ 4.0 **Sep 2016 – Jun 2020**

SKILLS

Programming Languages: Java, C/C++, Python, Go, Verilog, JavaScript

Tools: Emacs, Git, Linux, Shell, Valgrind, GDB, Vim, Gradle, Junit, Bootstrap, MySQL, Docker, Postgres, jQuery, AWS S3, Hive, ClickHouse, Google protocol Buffer, Google Cloud, SpringBoot, SpringMVC, MyBatis, Kafka, Redis, ElasticSearch

Experience

DiDi (equivalent of Uber) | Software Engineering Intern – Beijing **Jun 2023 – Sep 2023**

- Constructed DiDi's new generation of dispute resolution system, moving opaque end-to-end machine learning model into derivation logic and basic capabilities (fine-grained machine learning models and rules) to improve interpretability behind the decision and reusability of judgments on driver and passenger, serving 5 million orders request per day.
- Developed a tool to visualize the decisions made by the system so algorithm engineers and business colleagues can refine machine learning models and rules' implementation as well as resolve specific customer cases from hour level to real-time.
- Migrated data storage for this system from **Hive** to **ClickHouse** to implement real-time query for online analysis.
- Utilized Alibaba **Ant Design** framework (template) to render judgment tree for online mode and forecast mode, integrated querying data from **Amazon S3** (forecast) and ClickHouse (online) via **Python Pandas** library.
- To continuously improve accuracy of the system, developed a simulation tool that determines if proposed machine learning model and rules meet precision and accuracy goals based on labeled data.
- Implemented a rule capability that integrated a cheat detection system into the overall dispute resolution derivation logic.

ePatentManager | Software Engineering Intern – Virginia Beach, VA **Jun 2021 – Aug 2021**

- Implemented an **Android** mobile app that can estimate patent's price based on research paper and patent.
- Integrated Android studio ORM to CRUD operation, used **PostgreSQL** as persistent storage.

Projects

Job-Hunting Community Web project（Java, SpringBoot, SpringMVC, MyBatis, Redis, Kafka）

- Design of a full modern web application, developed an open-source clone of the popular NowCoder.com website.
- Built set of **RESTful** API with **Spring Boot/MVC** to process post operation with **MyBatis** integrated with **MYSQL**.
- Utilize **Redis** to cache recently most popular posts to protect database from overload traffic, improve rending latency performance by 40%.
- Used **Kafka** to handle burst traffic, achieving sending notifications as an asynchronous job.
- Migrate Post search function from **MySQL** to **Elasticsearch,** enhanced search performance by 30%.
- Used **Quartz** to implement **task scheduling** functions such as calculating post scores regularly.
- Configured the microservices with **Spring Cloud** Gateway.

Mini-Amazon and Mini-UPS (C++, Python, Django, Google protocol Buffer, Postgres, Docker)

- Developed a delivery platform, simulating the whole process from buying products to getting package delivered.
- Utilized **Google Protocol Buffer** and time-out based retransmission mechanism to serialize data among Amazon, UPS and world-simulator.
- Utilized **Optimistic Concurrency Control** to **Postgres** operation to prevent oversold event**.**
- Managed **Docker images**/containers using **Dockerfile** and shell script. Utilized **Kubernetes** for efficient cluster orchestration.
- Deployed application to **CI/CD** pipelines with **Github Action**, automating builds, testing, and deployments.

Around: a Geo-index based social network (Go, Google Cloud)

- Built a scalable web service in Go to handle posts and deployed to **Google Cloud** (GAE flex) for better scaling.
- Utilized **ElasticSearch** (GCE) to provide **geo-location** based search functions such that users can search nearby posts within a distance.
- Used **Google Dataflow** to implement a daily dump of posts to **BigQuery** table for offline analysis.