Xuyang Wang

📞 (213) 245-4899 | ✉ xuyangwang825@gmail.com | in linkedin.com/in/xuyang-wang | Los Angeles, CA

EDUCATION

University of Southern California   Los Angeles, CA
Master of Science in Applied Data Science   *Aug. 2021 – May 2023*

Beijing University of Technology   Beijing, China
Bachelor of Engineering in Software Engineering   *Sep. 2017 – July 2021*

SKILLS

Programming Languages: C/C++, Java, Python, Golang, SQL, JavaScript, HTML & CSS
Web Development: Front-End (Vue.js, React), Back-End (Node.js, Django), DataBase (MySQL, MongoDB, Redis)
Machine Learning PyTorch, TensorFlow, ML Algorithms, CV(Object Detection & Image Segmentation)
DevOps CI/CD (Kubernetes, Docker), Cloud(AWS, GCP, Azure), Configuration(Terraform, Ansible)
Data Processing: Big Data(Hadoop, Spark), Stream Processing(Kafka, Flink)

WORK EXPERIENCE

Meta (Facebook)   Menlo Park, CA
Production Engineer Intern, Stream Processing Team in Data Infrastructure Org.   *May. 2022 – Aug. 2022*

- **Operator Push Server**: Implemented an asynchronous operator push thrift service by Python for Meta's next generation stream processing framework; The peak traffic of stream processing service can reach 12.8TB/s.
- **Stateless Push**: Optimized the service design to resume a push using metadata in request body instead of storing it locally so that the service can run in distributed environment, which greatly increased service's reliability & scalability; Deployed new push service on 5 clusters to divert push which reduce processing time by 70%.
- **Metrics Dashboard**: Created a metrics dashboard including more than 20 performance metrics for monitoring and alerting; Included request count, error rate, request latency and resource usage.
- **Pipeline Migration**: Migrated more than 500 stream processing pipelines for XFN users, such as Meta Ads ML Infra team & IG Reels team. Built out library and wrote document for future user to easily deploy pipeline.

SenseTime   Beijing, China
Software Engineer Intern, ML Model Production Platform Team   *Apr. 2021 – July 2021*

- **Back-End Service**: Implemented features on an industrial quality inspection platform to conduct defect detection based on AI model; Applied Gin & ProtoBuf & gRPC-Gateway to provide both RESTful & gRPC API; Adopted MySQL, MongoDB, Redis to store different type of data; Deployed services based on Docker, Kubernetes.
- **ML Ops**: Refactored more than 20 APIs in ML Ops service which reduced model running latency by 12%.

Meituan   Beijing, China
Software Engineer in Test Intern, Community Group Buying Product Team   *Feb. 2021 – Apr. 2021*

- **Primary Workflow**: Participated in requirements & technical reviews; Formulated test plans; Designed test cases; Tracked, analyzed and promoted the resolution of problems in the testing process and online problems.
- **Automated Test**: Integrated automated functional testing, interface testing and stress testing into CI/CD environment. Updated automated tests to maintain interface and code coverage reach 97%.

PROJECT

[Web Application] Instant Messaging System —— *Vue.js, Node.js, Express, Axios, MongoDB, Socket.io*

- **Leadership**: Coordinated a team of 10 in designing an IM system based on B/S arch and MVVM design pattern.
- **Tech Stack**: Implemented front-end service with Vue.js; Built a micro back-end development by Node.js & Express; Performed HTTP requests using Axios; Stored different types of data in Mysql/MongoDB/redis; Communicated between instances by Socket.io; Applied Bcrypt & JWT on user information security.

[Computer Vision] Medical Image Segmentation —— *Pytorch, OpenCV, Dlib, LabelMe*

- **Model Design**: Co-designed a Multi-Encoder Parse-Decoder Network (MEPDNet) for medical image (CT) segmentation. Optimized traditional encoder-decoder model to multi- stages which make whole process robust.
- **Dataset Collection**: Assisted in labeling and analyzing tens of thousands Lumbar-CT grayscale images provided by PUMCH. Built a new Lumbar-CT dataset for image segmentation.
- **Ablation Experiment**: Conducted ablation experiments on different public datasets to verify model performance.