Mengyu Zhang

Seattle, Washington, United States
6264384723 | sam.zhang.069@gmail.com

SUMMARY

Result-driven developer with experience and diverse background in software development, machine learning, designing and implementing **high-performance, scalable systems** across industries. Proficient in **Java**, **Python**, **AWS**, and **microservices architecture** in Agile environments. Demonstrated success and experienced in building end-to-end systems, developing distributed systems, optimizing LLM models, and deploying AI solutions for real-world impact. Committed to leveraging **data-driven strategies** and **collaborative problem-solving** to deliver scalable, efficient, and innovative solutions.

CORE SKILLS

Programming: Java, JavaScript, TypeScript, Python (NumPy, Pandas, PyTorch, TensorFlow), C++ (CUDA)
Backend & Cloud: AWS (EC2, ECS, Lambda, EKS, CloudFormation, RDS, DynamoDB), Spring Boot, Flask, RESTful APIs, microservices, Terraform, Docker, Kubernetes, serverless architecture
Performance: Multithreading, multiprocessing, caching (Redis, Memcached), distributed systems, high-throughput APIs
Data: PostgreSQL, MySQL, Redshift, MongoDB, DynamoDB, Apache Kafka, AWS Kinesis, ETL automation
Machine Learning: ML model deployment, inference optimization, feature engineering, GPU acceleration (CUDA, TensorRT)
Other: Jenkins, CircleCI, Maven, Gradle, observability (Grafana, AWS CloudWatch), batch & real-time data pipelines

PROFESSIONAL EXPERIENCE

Software Engineer | *Amazon*

Seattle, WA| Jan. 2024 - Present

- Optimized ML inference pipeline for Alexa Music Expert, increasing API invocation accuracy from 85% to 95%, reducing response latency by 500ms, and improving model token efficiency, resulting in smoother user experience.
- Spearheaded development of **LLM-driven** recommendation and search features for Alexa Radio, enabling dynamic audio discovery and personalized station suggestions based on real-time user interaction and content metadata.
- Drove performance optimizations across Alexa Radio services, implementing rate-limiting, request deduplication, and caching strategies, reducing API call redundancies and improving response times by 35%.
- Migrated legacy batch communication system for price change announcements to modern architecture using **AWS CDK**, and reducing operational execution time by 90%, while improving system reliability and scalability.
- Designed and developed interactive UI control page using **React** and **AWS Amplify**, enhancing system observability with **AWS X-Ray** and **CloudWatch**, and providing a guided interface to manage batch communications, reducing manual intervention by 30%.
- Redesigned traffic control mechanisms with TPS throttling, idempotency strategies, and real-time traffic monitoring, utilizing **AWS Step Functions** and **SQS**, increasing throughput by 150% while stabilizing downstream services under load.
- Automated third-party radio station name update workflow for Alexa Radio, enabling comprehensive harness testing for newly added stations, decreasing manual effort and improving operational efficiency.

Software Engineer | *Open Innovation LLC (Self-Employed)*

Seattle, WA| May. 2023 – Jan. 2024

- Developed distributed web application leveraging **Node.js** and **Java microservices architecture**, ensuring scalability and modularity.
- Designed a seamless login and authentication system with **React** and **JWT**, integrating **OAuth** to provide secure Google and email-based authentication, facilitating smooth user session handling across app.
- Integrated Stable Diffusion into application, enabling real-time text-to-image generation with **AWS Lambda** for serverless execution and **EC2** to handle concurrent high-load requests, significantly enhancing performance scalability.
- Utilized **Redis** as distributed caching solution and **Elasticsearch** to optimize data retrieval times for images and other user inputs, reducing latency and improving user experience.
- Orchestrated containerized microservices using **Docker** and deployed them via **Kubernetes**, enabling automatic horizontal scaling, fault isolation, and streamlined deployment across environments.
- Established CI/CD pipelines with **CircleCI**, automated testing using **JUnit5**, **Mockito**, and **Jest** for both unit and integration tests, ensuring continuous integration and maintaining code quality throughout development lifecycle.

Software Engineer - AI/ML| *Inspur Group*

Beijing, China| May. 2020 – Jul. 2021

- Developed and deployed ML-powered OCR pipeline with scalable inference serving by integrating YOLOv4 for object detection and Tesseract OCR for text extraction, enabling automated document recognition for enterprise clients.

• Developed and deployed **RESTful APIs** using **Flask** and **FastAPI**, exposing OCR services to external applications and handling high-throughput requests.
• Containerized and deployed OCR pipeline using **Docker** on **Huawei Cloud Bastion Host**, ensuring scalability**,** fault tolerance, and GPU-accelerated inference for production workloads.
• Optimized data pipeline with asynchronous task queues using **Celery** and **Redis**, reducing latency by 20% and handling high-throughput OCR requests efficiently.
• Preprocessed images using **OpenCV** (grayscale conversion, noise reduction, adaptive thresholding) to improve OCR accuracy, achieving **87.4% F1-score**, a 10% improvement over baseline.
• Automated deployment workflows using **Jenkins** and **GitLab CI/CD**, ensuring seamless model updates, version control, and smooth integration of OCR model improvements into production.
• Integrated OCR capabilities into an existing microservices architecture built with **Spring Boot**, implementing **gRPC** and **RESTful APIs** for efficient communication with other backend services and external applications.

Data Science Intern | *Sikka Software* San Jose, CA| May. 2022 – Aug. 2022

• Developed ETL Pipelines with **Spark** and **Boto3** to update and maintenance over 1 billion patients' data stored on **AWS S3**, ensuring efficient data ingestion and preparation for downstream analytics.
• Integrated **AWS Athena** with **PyAthena** and **SQLAlchemy** to perform data management and preview on collected data.
• Built **BiLSTM** model with **PyTorch** on **AWS EC2** to predict health indicators for evaluating patients' dental health.
• Analyzed medical notes with **AWS Comprehend Medical**, performing **exploratory data analysis (EDA)** to extract key medical entities, uncovering insights into sequential procedure patterns.
• Demonstrated feasibility of **progressive disease prediction** through time-series modeling, enabling improved patient treatment plans and early intervention strategies.

PROJECT EXPERIENCE

Automated Short-Term Options Trading Bot Sep. 2022 – Jan. 2024

• Designed and implemented Python-based trading bot for 1-day SPY/QQQ options, achieving average monthly ROI of **20%+** using personal funds in **live market conditions** with consistent profitability.
• Integrated **Robinhood API** for real-time market data retrieval and automated order execution, implementing **fail-safes**, **rate-limiting**, and **retry logic mechanisms** to handle dynamic market volatility efficiently.
• Deployed bot on **AWS EC2** instances with **Docker**, enabling horizontal scaling to handle concurrent trades, live analytics, and complex data processing seamlessly.
• Built **React-based dashboard** to visualize PnL, trade performance, and live positions, integrating automated Slack/email alerts for unusual price movements or system errors efficiently.

Campus Second-Hand Platform for Students Mar. 2023 – Jul. 2023

• Developed distributed e-commerce system based on **Spring Boot** and **Spring Cloud** backend Microservice architecture.
• Created login functionality with **React** using **Firebase**, allowed **Google Authentication** and email sign-in methods.
• Integrated application with **Redis** as distributed cache for optimizing system performance, **Elasticsearch** for faster product retrieval time, **Spring Session** for session data sharing**, thread pool** and asynchronous task for stability.
• Implemented registration/configuration center with **Nacos**, **Gateway** as gateway, and **Feign** for remote calls.
• Initialized **CI/CD Pipelines** on **Circles** and designed unit test/integration test processes with **Junit5**, **Mockito**, **Jest**.

Scholarly Text Institution Name Disambiguation |JHU - IEEE Capstone Dec. 2021 – Jun. 2022

• Implemented Python script with **Psycopg** to update database for IEEE's data lake of 5.4 million papers in **AWS Redshift**.
• Performed Named Entity Recognition with **spaCy**, created geolocator with **GeoPy** to impute geographical features.
• Applied **LightGBM** Classifier with Python to perform supervised pairwise comparison and classification.
• Deployed **Hierarchical Agglomerative Clustering** (HAC) to center and standardize clusters.
• Published algorithms and data with Sphinx documentation on AWS EC2, achieved 81.4% accuracy.

EDUCATION

Johns Hopkins University, Master of Science in Engineering, Data Science May. 2023
University of California, Los Angeles, Bachelor of Science, Applied Mathematics Major, Statistics Minor Dec. 2020
UCLA Extension, Certificate Program, Data Science Mar. 2021