Jiaming Pei

jiamingpei16@gmail.com • +1-778-708-0116 • Vancouver, BC • LinkedIn • GitHub

EDUCATION

Northeastern University, Vancouver, BC *Sept 2024 – Aug 2027*
Master of Science in Computer Science — GPA: 3.85/4.0

Minzu University of China, Beijing *Sept 2020 – Jun 2024*
Bachelor of Economics in Finance

SKILLS

Languages: Python, Go, Java, TypeScript, SQL
Backend: Gin, REST APIs, Microservices, Saga, Event-Driven, RabbitMQ, testcontainers-go, Locust
Cloud / IaC: AWS (ECS Fargate, RDS, MQ, ALB, VPC, DynamoDB, ECR), Terraform, Docker, GitHub Actions, Linux
Observability: Prometheus, Grafana,
Data / ML: MySQL, MongoDB, PyTorch, INT8 Quantization

PROJECTS

Distributed Order Processing System
Go · Gin · MySQL · RabbitMQ · Terraform · AWS · Prometheus · Grafana · Locust

- Architected 3-microservice Saga platform (Order/Inventory/Payment) in Go/Gin with database-per-service isolation; persisted saga state to MySQL for crash-resume with compensation rollback
- Engineered optimistic-lock SKU reservation (version column + WHERE-guard, 50-retry budget) — testcontainers integration test: 50 concurrent goroutines on 10-unit stock yielded exactly 10 CONFIRMED + 40 FAILED, zero oversell
- Provisioned full AWS stack via Terraform IaC (ECS Fargate, RDS, Amazon MQ, ALB with path-based routing, ECR, Secrets Manager) and instrumented observability via zap structured logs + Prometheus collectors + auto-provisioned Grafana dashboards
- Resolved MySQL pool bottleneck via Prometheus go_sql_*; raised pool 25→50 cutting p95 21% (140→110ms), falsified row-lock and query-CPU hypotheses via processlist and confirmed fsync-bound 2PC commit serialization with pre-registered A/B toggling innodb_flush_log_at_trx_commit (+34.7% RPS)

VAD Model Compression — PyTorch Quantization Analysis
PyTorch · SpeechBrain · INT8 Quantization · Model Compression

- Benchmarked FP32/PTQ/QAT/PTQ+QAT on a 0.435 MB CRDNN VAD model with a two-stage warm-up protocol reporting median+P95; exposed a 6× P95/median spread in PTQ+QAT (327ms vs. 54ms) hidden by mean-based measurement
- Diagnosed a silent 8% F1 regression by tracing PyTorch's INT8 quantization coverage: only 1.2% of CRDNN params were in quantizable layers, so FakeQuant on Conv2d/GRU is dropped at inference — creating a train/test mismatch that explains why QAT underperformed PTQ

Distributed Book Recommendation System
Python · AWS · Terraform · Docker · boto3

- Owned data-pipeline + AWS-infra track end-to-end in a 4-person team; sole author of ETL, Terraform IaC, and DynamoDB batch loaders
- Built streaming Python ETL on 30 GB+ Open Library dumps; iteratively relaxed filters (3,574 → 50K books) and pre-computed `title_prefix` for A-Z GSI sharding
- Modeled 3-table DynamoDB schema with 3 GSIs via Terraform (PITR, SSE); loaded 50K records via boto3 BatchWriteItem with exponential backoff
- Quantified horizontal-scaling limits via Amdahl's Law on N=1/16/26 ECS shards using per-request phase headers (X-Phase-Parse/Aggregate/Total)

EXPERIENCE

Quantitative Developer Intern, Benefits Mutual Asset Management Co., Ltd. *Beijing, China · Jul 2023 – Sep 2023*

- Implemented and maintained Python-based macro trading strategy pipelines: ingested macroeconomic indicators across multiple asset classes, ran daily backtests, and persisted strategy metrics for downstream analysis
- Built monitoring tooling for live strategy performance: tracked positions, P&L, and risk exposures; generated automated daily summary reports flagging anomalies for PM review