Shiba Wang

s2259wan@uwaterloo.ca | Shibawang.ca | LinkedIn | GitHub

Education

University of Waterloo Expected 2027

Bachelor of Mathematics Co-op, Computational Mathematics · GPA 3.72/4.00 *Waterloo, ON*

Experience

Fountain Health Technologies Inc. Apr 2026 – Present

Founding Engineer *Waterloo, ON*

- Shipped a production AI care-navigation platform end-to-end, now live across 20+ clinics: a Flutter iOS app (on the App Store), a NestJS/PostgreSQL API, and React clinic and admin dashboards at 99% uptime.
- Built an LLM intake pipeline that parses free-text patient descriptions into schema-validated symptom data, blocking downstream routing until typed validation and confidence checks pass.
- Designed a safety layer where the LLM proposes and a deterministic engine decides: 50+ clinical rules route each case to four acuity levels (911 / clinic / pharmacist / self-care) at 95% accuracy, so no model output goes unchecked.
- Engineered a tamper-evident, hash-chained PHIPA audit ledger on an append-only store isolated from the operational DB, capturing every patient-data access, consent, and disclosure event.

University of Waterloo, Centre for Extended Learning Sept 2025 – Dec 2025

Software Engineer Co-op *Waterloo, ON*

- Designed and built a multi-reviewer academic-review platform from scratch: a React front end, a .NET API, and a SQL data model powering faculty and admin review workflows.
- Engineered its real-time collaboration layer with optimistic concurrency, queued writes, and atomic transactions, so reviewers edit live without lost updates or conflicts.
- Secured the platform with a role- and status-based authorization model across client and API, hardened by a 57-test CI gate over the full 5-role x 6-status matrix.

Nanjing Xitai Trading Co., Ltd. Jun 2024 – Aug 2024

Software Engineer Intern *Nanjing, China*

- Cut a daily 2-hour manual reporting task to under 10 minutes by building a Python pipeline (pandas, PostgreSQL) that validated and loaded the sales and inventory data automatically.
- Built a React and FastAPI dashboard that gave the team self-serve access to regional sales and inventory metrics over REST, retiring the manual weekly Excel reports.

Projects

Qwen Serve — Production LLM Inference Service Live Demo | GitHub

Serves an open LLM through a cheap, OpenAI-compatible API that costs almost nothing when idle

- Served Qwen2.5-7B (AWQ INT4) on an NVIDIA L4 via vLLM behind an OpenAI-compatible FastAPI gateway (auth, rate limiting, SSE streaming, cold-start handling), scaled to zero on Cloud Run.
- Cut serving cost 33x ($14.28 to $0.43 per 1M tokens) and raised throughput 34x (16.5 to 556 tok/s, 6.6s p95) via continuous batching and AWQ INT4, with no GSM8K accuracy loss.
- Gated every deploy on k6 load tests in a GitHub Actions pipeline that builds a multi-stage CUDA image, with Prometheus, OpenTelemetry, and Langfuse tracing streamed to Grafana Cloud.

Can I Work? — Agentic RAG Assistant with Evaluation Harness Live Demo | GitHub

Tells international students what work they're allowed to do in the U.S. and Canada, every answer cited

- Built an agentic RAG pipeline: PII and prompt-injection screening, jurisdiction routing, hybrid retrieval and cross-encoder reranking in Qdrant, and self-correction that abstains when ungrounded.
- Cut wrong-jurisdiction answers from 8/56 to 0/56 (95% CI 0–6%) and raised faithfulness 0.85 → 0.98 on a 56-question Ragas + LLM-judge eval harness.
- Enforced verbatim citations (each quote snapped to its source or dropped) over a provider-agnostic LiteLLM layer adding RPM/TPM pacing, retry backoff, and on-disk response caching.

Skills

Languages: Python, C/C++, Java, TypeScript/JavaScript, SQL, C#, Dart

AI / LLM: agentic RAG (hybrid search, reranking), evaluation (Ragas, LLM-as-judge), guardrails (prompt-injection, PII), structured outputs, LangGraph, vLLM, Qdrant, LLM APIs (Gemini, Groq, Cerebras, OpenAI-compatible)

Infrastructure & Tools: Docker, AWS, GCP Cloud Run, FastAPI, NestJS, CI/CD (GitHub Actions, k6 load-gating), Prometheus/Grafana, OpenTelemetry, Langfuse, PostgreSQL, Supabase, React, Next.js