Yunfan Zhang

Gothenburg, Sweden · +46 73 346 28 18 · yunfanteo@outlook.com
linkedin.com/in/yunfanz123 · github.com/44sunsetsf · yunfanteo.world

Profile

I am an MSc student at Chalmers looking for a spring 2027 thesis on LLM agents and how to evaluate them. At Huawei I helped build an MCP-based agent that 130+ engineers now use. I am curious, patient and a good listener, and I get along well with people from different backgrounds.

Experience

Software Development Engineer Intern — Huawei Technologies **Jun 2026 - Sep 2026**
Outstanding Intern award • *Python, FastMCP, Pydantic, SQLite, YAML, C++*

- Built core modules of PerfAgent (team project), an **MCP** server with a declarative skill system for LLM agents to analyse HarmonyOS traces beyond their context window; it grew from 20 to **130+** internal users.
- Designed its skill engine to reduce hallucination: the framework fills in tool parameters and checks each conclusion against the raw data; file-based context lets subagents run in parallel, and all 17 skills can resume after a failure.
- Built its query engine, which merges 5 parameter levels, keeps spikes in downsampling and orders 80+ metrics by dependency, so composite metrics are computed from their inputs.
- Built its testing and observability: decorators catch 4 agent failure modes, 27 golden samples catch silent regressions, and structured logs trace each session.
- Cut a dashboard API from 11.7 s to **0.93 s** by adding composite indexes and rewriting slow SQL.
- Implemented notification-type switching for OpenHarmony background tasks (C++), unit-testing all 9 tier transitions.

Education

Chalmers University of Technology — MSc in Computer Systems and Cybersecurity **Sep 2025 - Jun 2027**
IPOET Scholarship (75% tuition fee waiver)
Courses: Distributed Systems (5/5), Compiler Construction (5/5), Applied Machine Learning, Computer Security

Northwestern Polytechnical University — BEng in Information Security **Sep 2020 - Jun 2024**
GPA 86.4/100, ranked 10/94 (top 11%)

Projects

GoEuroOps | *Python, FastAPI, Redis, ChromaDB, Docker, GitHub Actions* **Mar 2026 - May 2026**
Multi-agent customer service for a study-abroad consultancy, deployed and live *Demo: yunfanteo.world/goeuroops | GitHub*

- Orchestrated **multiple agents** scoped by role contracts and tool whitelists, with two-stage routing to pick them.
- Fused LLM few-shot, embeddings and rules for intent detection: 91.7% accuracy (0.915 macro-F1) on 380 questions across 19 intents, including typos, slang and mixed Chinese and English.
- Designed an intent-gated **LLM wiki** that gives fully correct answers 95% of the time, against 88% for vector **RAG**.
- Hardened every tool call with Pydantic checks, timeouts and circuit breakers; prices come from code, not the LLM.
- Built **agent memory** on A-Mem: each user message becomes an instantly searchable vector note, later enriched with event times and links; a verifiable customer profile and per-session case notes let the agent follow up on open cases.
- Built dynamic skills that are injected by intent, keywords and semantic similarity, and reloaded without a restart.
- Built a scenario-based **LLM-as-a-judge** that scores quality and boundary compliance against a saved baseline.

VParser | *Go, Pydantic, gRPC, MySQL, Redis, Kafka, Qdrant, Vue* **May 2025 - Jul 2025**
Long-video analysis and Q&A agent, deployed and live *Demo: yunfanteo.world/vparser | GitHub*

- Built a Planner–Executor–Critic agent that answers questions about long videos with **RAG** and checks each timestamped claim against the transcript and on-screen text before showing it.
- Offloaded long tasks to a **Kafka** pipeline with tiered retries and a dead-letter queue, so task submission returns in under 50 ms instead of blocking for 60 s+.
- Enforced content-level deduplication and idempotent consumption with distributed locks and MD5 fingerprints, so the same video is never analysed twice.
- Implemented resumable uploads from the last chunk in Redis, and capped model spend with token-bucket rate limiting.

Skills and Languages

Programming	Python (Pydantic, LangChain, PyTorch) and Go, both proficient; C++, SQL, JavaScript
Backend	FastAPI, Gin, GORM, gRPC; MySQL, Redis, Kafka, MinIO; async design, idempotency, rate limiting
LLM agents	MCP servers, skill systems, multi-agent orchestration, agentic RAG, LLM wikis, agent memory, LLM judges
Tools	Git, GitHub Actions (CI/CD), Docker, Linux, CMake; AI coding tools such as Claude Code and OpenCode
Languages	Chinese – native; English – fluent