

Polaris: a job-search agent for LLM and Agent internship seekers

Collects jobs and interview write-ups from career sites and job forums, searches them against your constraints (city, direction, days per week, months available), flags hard conflicts, pulls real interview questions from real posts, and watches a search goal over weeks. Every claim opens to its source; when nothing is found, it says so. The author uses it for their own internship search.

Code github.com/duoduo-cell/polaris Page duoduo-cell.github.io/polaris Release v0.1.0, Apache-2.0

18

agent tools; side effects need approval

801

backend tests; CI on every PR

3,800+

interview questions (author's DB, 10-25)

0.89

retrieval NDCG@10 (3 paths + rerank)



Synthetic demo (UI in Chinese): jobs under the user's constraints, hard requirements checked. Data and company names are invented.

What it does, and why you can trust it

- **Find jobs:** keyword, vector and graph recall, filtered by constraints first; unstated constraints are "to confirm".
- **Interview questions:** sourced and dated; asking for real questions forces a search before answering.
- **Citation check:** any id in the answer must appear in tool results, or it is marked unverified.
- **Follow-up cards:** watch a goal; changes ranked urgent / digest / quiet; reports built on rule-computed facts.
- **Competitiveness:** HR's three questions plus peers' public self-descriptions, honest about small samples.
- **Applications and memory:** funnel, drafts never auto-sent; bi-temporal memory that asks before overwriting.

Engineering cases (found by running the whole system; no offline test caught them)

Symptom	Cause	Fix	Result
Constrained search listed 1 job	Top 50 chunks from the whole corpus, then intersected with the scope: 40 candidates became 1 (post-filtering)	Pass in-scope ids into all three recall paths	1 → 32
A task took 265 s	6,000-character tool output, ~2,200 per job; the model paged through results	Slimmer list items, cap 16,000	77 s
Model requests failed at 30 s	A local TUN accelerator captured model traffic; handshakes hung	Narrow direct route, fallback on failure	103 → 2 s
Answer invented a "question from the database"	The model skipped the search; a prompt rule did not stop it	Evidence gate in orchestration	all sourced
Tests "passed" with the DB down	Skipped with exit code 0; 306 of 792 never ran	Fail in CI instead	PR #2

Stack

- LangGraph 1.2 / LangChain 1.4 with approval, call-limit, retry and evidence-gate middleware
- PostgreSQL 17 + pgvector + ParadeDB pg_search (BM25, Chinese tokenization), optional Neo4j; weighted RRF, rerank, MMR
- FastAPI, SQLAlchemy, Alembic, MCP; Next.js 16, React 19
- Qwen via Alibaba Cloud Bailian; every model call traced with tokens, latency, cost; GitHub Actions

Evaluation (module evaluations, 10-11 to 10-13)

- Retrieval NDCG@10: one path 0.73 → three paths 0.81 → reranked 0.89
- Cross-source alignment test F1 0.93 (labelled after rules were frozen)
- Answer quality: dev 24/24, held-out 13/14, zero fabricated sources
- Prompt injection: 0 of 15 attacks succeeded

Data boundaries: logged-in platforms are read only on the author's machine, with their own account, at a low rate; raw text never enters the repository or the demo; no automated messages or applications. Sources for every number are in the README and dev log.