

Conner Babinchak

Software Engineer | AI Agents & Eval's

✉ conner@connerbabinchak.com 📍 Philadelphia, PA 🌐 github.com/babinchak

🌐 linkedin.com/in/connerbabs

SUMMARY

Backend engineer with 5 years shipping production Python/AWS systems, now building and rigorously evaluating agentic AI.

PROFESSIONAL EXPERIENCE

Software Engineer II

11/2021 – Present | Philadelphia, PA

Comcast

- Building an internal agent that answers questions across Comcast's commerce platform, from simple lookups to multi-step root-cause analysis correlating Elasticsearch logs with backend APIs; designed and own the eval harness that scores tool selection and answer accuracy across its full capability set.
- Built a benchmarking suite for a multi-agent sales assistant on Google ADK, comparing LLMs on latency, per-agent performance, and tool-call behavior; enriched business-intelligence signals with an LLM for agent consumption.
- Primary owner of a Python/FastAPI/GraphQL microservice powering Comcast purchase flows and offer-eligibility logic, handling ~3M requests/day.
- Introduced Redis-backed caching and async downstream fetches for offer and eligibility queries, enabling paginated results across requests and reducing latency by ~40%.

Software Engineer Intern

06/2020 – 08/2020 | Seattle, WA

Amazon

- Prototyped a fuzzy-matching service on Lambda, DynamoDB, and S3 to flag fraudulent Kindle accounts.

INDEPENDENT PROJECTS

Hindsight · MongoDB Hackathon NYC, Sept 2026 · built solo in 6.5 hours 🌐

- Built a temporal memory benchmark for coding agents from real session history (47 validated cases, 2,835 runs): mined memory raised pass@1 from 4% to 44%, and a self-tuning harness reached 67% on a held-out split vs. 40% for the best hand-designed config.

PEICO-Bench (peicobench.com) · live model leaderboard 🌐

- Built a benchmark for a multi-turn tool-calling customer sales agent at a fictional insurance company.
- Designed a scoring system that pairs deterministic database diff assertions with a rubric LLM judge.
- Published results for 11 model configurations across Anthropic, OpenAI, and Google; scores spread 58% to 96%.

Minerva Reader (minervareader.com) 🌐

Next.js · TypeScript · Supabase/Postgres · pgvector · AWS Lambda (Go / TypeScript) · OpenAI · LangGraph

- Built an agentic RAG assistant that answers questions about a single book or across a user's entire library, with clickable in-answer references that navigate the reader to the exact quoted passage.

EDUCATION

B.S. Computer Engineering

2021 | Pittsburgh, PA

University of Pittsburgh

GPA: 3.81

SKILLS

Languages

Python, Go, TypeScript, JavaScript, SQL, Bash / Shell

Infra / Data

AWS, Postgres, Redis, Docker, Terraform, ELK Stack, CloudWatch, CI/CD, GitHub Actions, Lambda

Frontend

React, Next.js, Tailwind, TanStack Query, Vercel, Vite

Backend

FastAPI, GraphQL, REST, asyncIO, microservices, gRPC, Pydantic, OAuth / JWT

AI / LLM

agentic systems, tool calling, multi-agent orchestration, evals, LLM-as-judge, Arize Phoenix, context engineering, agent memory, RAG, vector search, prompt engineering, LangChain / LangGraph, Google ADK, MCP