

Chaitanya Deshpande

✉ dcchaitanyadeshpande@gmail.com  [chaitanya0305](#)  [LinkedIn](#)

Professional Summary

Building production AI systems taught me that an agent is only as good as the context it retrieves and the evidence it is judged on. With multi-agent systems shipped across four client products and a Springer publication, I work where applied engineering meets evaluation research. I build systems that hold up in production while measuring where they fall short: ungrounded retrieval, unverifiable claims, and the gap between a fluent answer and a correct one.

Education

Bachelor of Engineering

Thadomal Shahani Engineering College, University of Mumbai, India

2021 – 2025

CGPA: 7.02

Experience

AI Engineer

TheAgentic.AI

Jul 2025 – Present

San Francisco, California

- Architected the AI layer of an AI COO platform for startup founders: ~60 PydanticAI agents and LLM features across investor discovery over 150k investors, fit scoring, a revenue CRM, outreach automation and data-room RAG, built on pgvector semantic search, human-in-the-loop approvals and Gmail, Outlook and Calendar integrations.
- Designed an LLM investor-fit assessor with an offline evaluation harness (1,200 planted-truth pairs), lifting name-only discrimination (AUC) from 0.508 to 0.822, and built the platform's multi-provider LLM layer: per-feature model routing through OpenRouter, batched background workloads and zero-LLM prefilters.
- Architected a multi-agent automotive repair assistant that turns a vehicle, complaint and fault codes into ranked root causes, repair procedures, wiring schematics, recalls and service bulletins through four flows: an OEM repair-manual flow for pre-2014 vehicles, a knowledge-graph flow for OEM groups ingested into Neo4j, a web deep-research flow (root-cause agent plus parallel solution agents) for every other vehicle, and Quick Assist, an intent-routed conversational agent with real-time SSE streaming.
- Co-engineered vehicle-scoped retrieval over the knowledge graph: Brand → Model → Document → Chunk traversal capped at depth 2, scope-first exact cosine with filtered-ANN fallback, lexical-first routing for fault-code queries and a document tag plane (make, year range, engine codes), eliminating off-vehicle results (30/30 on-vehicle precision).
- Optimized retrieval across the other flows: hybrid cosine + BM25 ranking over a precomputed index of 70K+ OEM manuals (32–65× faster; foreign-vehicle pages cut from 60% to 4.4%), a pre-built manual taxonomy (vehicle lookup 30.6 s → 0.1 ms), and LLM query expansion with trusted-source filtering that returned 4.2× more service-bulletin results.
- Architected the multi-agent orchestrator of an AI football scouting platform that routes natural-language questions across player discovery, market valuation, counterfactual transfer-impact and pre-match tactical agents, using guarded text-to-SQL over a 572K-player PostgreSQL warehouse and live web search.
- Built its evaluation and data backbone: a backtesting harness that replays 106 historical transfers with point-in-time cut-offs and two-stage LLM critics (verdict alignment 53.3% → 71.3%, co-built), an injury-data pipeline of 45K+ observations, and server-side PDF report export.
- Built the data-protection and connector layer of an LLM incident root-cause-analysis agent (LangGraph) for Kubernetes estates: PII/PAN redaction across every agent seam and verified connectors for 14 telemetry providers; the agent scored 15/15 on SREGym-Lite live fault injection (8/12 on held-out faults).
- Tech stack: Python, FastAPI, PydanticAI, LangGraph, PostgreSQL + pgvector, Neo4j, FAISS, React/TypeScript, Docker, Kubernetes, GitHub Actions.

Python Intern

Imperative Business Ventures

Dec 2023 – Feb 2024

Mumbai, India

- Built an AI reconciliation tool that matched debit and credit balances and flagged discrepancies, plus complaint sentiment scoring and document chatbots over CSV, Excel and text.

Projects

LiveGraph: Live Multi-Agent Research Canvas

Oct 2026

- Architected a steerable multi-agent research system: a planner agent draws the workflow graph live and 8 agents (research, critic, prosecutor/defender debate, judge, editor) stream a cited brief; users can steer, edit or veto any node, and only affected nodes recompute, so local edits cost 38–59% of a fresh run.
- Built an event-sourced real-time engine (Postgres event log, WebSocket fan-out, timeline replay) and cut end-to-end runs from 373s to 154s through provider pinning and prompted-JSON outputs.
- Designed a pre-registered evaluation harness measuring factual precision, citation precision and recall, and calibration, with five ablations and bootstrap confidence intervals to isolate each agent's contribution.
- Tech stack: FastAPI, PydanticAI, Postgres + pgvector, Next.js, React Flow, Docker.

Research & Publications

Sneaker Resale Price Prediction using Ensemble Learning Techniques

L. Doshi, J. Samberkar, **C. Deshpande**, K. Saxena (2024). *ICT Analysis and Applications: Proceedings of ICT4SD 2024*, Vol. 7, Springer. [\[Link\]](#)

- Compared Linear, Lasso and Random Forest regressors with bagging and stacking ensembles to predict sneaker resale prices in a market of about \$13 billion; a stacked Lasso ensemble performed best.

Technical Skills

Languages: Python, TypeScript, JavaScript, SQL, Java, C, HTML/CSS

Agent Systems: Multi-agent orchestration, custom tool-use implementations, typed structured outputs, human-in-the-loop approval workflows, custom evaluation frameworks (backtesting, LLM critics and judges), production RAG pipelines, knowledge-graph retrieval, agent guardrails and PII redaction, real-time streaming (SSE, WebSocket)

Infrastructure & Cloud: Docker, Kubernetes (DigitalOcean), CI/CD (GitHub Actions), AWS (CloudWatch, SNS), Linux

Frameworks & Tools: PydanticAI, LangGraph, FastAPI, SQLAlchemy, Alembic, React, Next.js, Node.js, Tailwind CSS, Playwright, pytest, Pydantic Logfire, Streamlit

Data Systems: PostgreSQL, pgvector, pg_trgm, Neo4j, knowledge graphs, MongoDB, MySQL, Redis, Kafka, FAISS, vector databases

Other: Data Structures & Algorithms, Machine Learning, scikit-learn, ensemble methods, NLP, Data Analysis & Visualization, pandas, Matplotlib, Seaborn, Git, GitHub