SHAURYA GULATI

(412) 240-9813 | i.shauryagulati@gmail.com | linkedin.com/in/shauryagulati | github.com/Shauryagulati

EDUCATION

Carnegie Mellon University | Pittsburgh, Pennsylvania August 2024 - December 2025
Master's in Information Systems Management (Business Intelligence and Data Analytics)
Coursework: Computational Data Science, Machine Learning, Applications of NLP and LLMs, Distributed Systems, Applied AI

Chandigarh University | Mohali, India July 2018 - June 2022
Bachelor's of Engineering in Computer Science and Engineering- Specialization in Artificial Intelligence and Machine Learning

SKILLS

Languages & Databases: Python, SQL, MySQL
AI & LLM: Claude Agent SDK, OpenAI API, LangChain, MCPs, RAG, LLM-as-judge, evals, prompt engineering and caching, Agents
Retrieval: pgvector, FAISS, hybrid search, BM25, RRF fusion, BGE cross-encoder reranking, query decomposition
Backend: Python, FastAPI, PostgreSQL, Redis, SQLAlchemy, Alembic, asyncio, SSE streaming
Observability / Infra: Langfuse, OpenTelemetry, Docker, GitHub Actions, Git

WORK EXPERIENCE

Lunon - Founding AI Engineer February 2026 - May 2026
AI-native firm building automated commercial due diligence for private-equity firms (scoped for initial AI pipeline and infra)
- Built a 5 Phase core AI pipeline: a **multi-agent system for PE**, with quality gates, automated correction loops.
- Built the retrieval stack, document ingestion, hybrid search **(pgvector + lexical with RRF fusion)**, query decomposition, and cross-encoder reranking.
- Built the analyst copilot: a ~20-tool agent with streaming chat, a **propose-then-confirm flow** for edits, and citation verification against the source corpus.
- Traced and fixed a **quality-gate failure loop** blocking a client demo, added warn-mode so runs completed instead of stalling.
- Set up **Langfuse observability** for tracing and cost/latency monitoring across the agent pipeline.

Carnegie Mellon University - AI Research Assistant June 2025 - December 2025
- Engineered hybrid RAG system combining dense retrieval (E5 embeddings, ChromaDB) with sparse BM25 and BGE cross-encoder reranking across 1,000+ regulatory documents.
- Built a PDF ingestion pipeline with page parsing, header detection, and overlapping chunking, with data-lineage tracking for accurate vector-store population.
- Evaluated system across 50+ query sets, achieving **94% accuracy** with grounded citations.

YMGrad - Software Development Engineer June 2022 - July 2024
- Built analytics APIs and optimized Redis caching and MySQL indexing, cutting load times by ~25% and query time by ~30%.
- Automated operational reporting workflows, reducing manual support overhead by ~15%.

PROJECTS

rag-verdict- pytest for RAG agents: a behavioral test framework for RAG/LLM systems
github.com/Shauryagulati/ragverdict
- Built an open-source framework that tests **RAG agent behavior**, whether tools fire, whether citations resolve to real documents, and whether the agent refuses out-of-corpus questions, instead of just averaging quality scores like RAGAs or DeepEval.
- Designed a **pluggable adapter** (Python or HTTP) that connects any RAG system in any language, plus an **LLM-as-judge** layer with structured output that degrades gracefully when no API key is present.

Suture- AI-native operations layer for a cardiology practice
github.com/Shauryagulati/suture-ai
- Built an AI platform handling the full referral to outreach loop (intake, review, prior-auth, outreach), with document extraction, human-in-the-loop review, and multi-channel patient outreach.
- Built an **LLM extraction pipeline with a real eval harness** (per-field precision/recall tracked across prompt changes) and a **3-stage hybrid-RAG** (structured + semantic) prior-auth engine over payer policy rules.
- Engineered **fail-closed tenant isolation** at the ORM layer (missing tenant context raises; cross-tenant reads return 404), field-level PHI encryption, and a PHI-safe audit log.

Multi-Agent Knowledge Retrieval System
- Designed a **multi-agent RAG pipeline** with LangChain: Planner (Deepseek), Retriever (BAAI/bge-m3 + cross-encoder), Synthesizer (Llama 3.1), and Reviewer for quality assurance and follow-ups.
- **Validated system performance** using the **RAGAs framework**, achieving **0.78** Faithfulness and **0.74** Answer Relevancy scores across complex domain-specific validation sets.