

SANJAY MANIVASAGAM

San Francisco, CA | (530) 979-6675 | sanmanivas@ucdavis.edu
github.com/Sanjaavyv7 | linkedin.com/in/sanjaymanivasagam | sanjaymanivasagam.online

TECHNICAL SKILLS

Languages: Python, SQL, TypeScript, JavaScript, R, C/C++

Machine Learning & AI: PyTorch, TensorFlow, scikit-learn, XGBoost, Hugging Face Transformers, LangChain, LangGraph, RAG, Vector Embeddings, Cross-Encoder Reranking, Multi-Agent Orchestration, Agentic Systems, Model Context Protocol (MCP), LLM Observability, Prompt Engineering, LLM APIs (OpenAI, Anthropic)

Frameworks: FastAPI, Express, React, Next.js, Pandas, NumPy

Infrastructure & Backend: Docker, GitHub Actions, PostgreSQL, Redis, Apache Kafka, AWS (EC2, S3, Lambda, RDS), REST APIs, Drizzle ORM, BullMQ

Data Science & Analytics: Statistical Modeling, Hypothesis Testing, Experiment Design, Data Wrangling, Data Visualization, Tableau, Power BI

Methodologies: Production ML Deployment, Model Evaluation, RAG Pipelines, A/B Testing, Feature Engineering, Distributed Systems

EXPERIENCE

Founder & Agentic Systems Architect

Sep 2025 to Present

ClearAgent

San Francisco, CA (Remote)

- Architected a **LangGraph-based multi-agent orchestration system** with a planner-executor pattern coordinating verification, audit, and policy-enforcement subagents for autonomous AI agents operating under EU AI Act Articles 12, 14, and 19 (enforcement Aug 2026).
- Engineered a **semantic policy retrieval engine** combining vector embedding search with cross-encoder reranking over the EU regulatory corpus, surfacing context-relevant compliance clauses to enforcement agents in real time.
- Designed an **event-sourced audit ledger with cryptographic chain-of-custody**, capturing token usage, tool-call traces, and policy-violation flags per agent invocation across heterogeneous LLM providers including OpenAI and Anthropic.
- Built a **custom evaluation harness** measuring policy adherence, audit completeness, and decision provenance across agent invocations, replacing manual compliance review with deterministic, reproducible scoring.

AI/ML Engineering Intern

Jun 2025 to Sep 2025

DataCorp Traffic Private Limited

Remote, Berwick-upon-Tweed, UK

- Architected an **LLM-powered report automation system** using Python, OpenAI API, and Docker that processed 200+ monthly reports for a multi-tenant SaaS platform, cutting delivery time **85%** (3 to 5 days down to 4 hours) and enabling scale from 15 to 30 enterprise accounts without additional headcount.
- Engineered a **schema-aware, tenant-scoped SQL generation pipeline** with regex-based Excel validation and ETL workflows, eliminating cross-tenant data access across 50K+ queries and reducing runtime errors 80% and support tickets 40%.
- Built end-to-end as 1 of 2 developers on the core engineering team, owning the full stack from template validation to **JWT-authenticated API endpoints** with role-based access control and audit logging.

Software Engineering Intern, Agents & Frontend

Jun 2024 to Sep 2024

Headstarter

San Francisco, CA

- Extended a **code-generation AI agent** to understand multi-file GitHub repositories and propose cross-file refactors using Python and LLM orchestration frameworks, reducing average time from agent run to first useful suggestion by 20 to 30%.
- Shipped an **agent review dashboard** in React and TypeScript that consolidated PR review into a single workflow, cutting clicks-to-accept by **50%** and adopted by a majority of active users within 2 weeks of launch.
- Delivered 5+ production features in cross-functional collaboration with 2 to 4 senior engineers and a founder, deployed via Docker, Kubernetes, and Kafka on AWS infrastructure using Git-based CI/CD workflows.

TECHNICAL PROJECTS

Groundtruth, AI Evaluation & Reliability Platform | Python, Evaluation Infrastructure, Reproducible ML Systems, Evidence-Based Verification, CI/CD Automation [↗](#)

- Built an **evaluation infrastructure for AI systems** focused on **deterministic execution**, reproducible metrics, and evidence-backed validation.
- Designed a **meta-evaluation engine** that verifies evaluation artifacts using structured claims, threat models, and automated consistency checks.
- Implemented **170 automated tests**, CI pipelines, **negative-control validation**, and deterministic artifact generation for reproducible evaluation workflows.
- Developed modular engineering infrastructure emphasizing **auditability, traceability, and reliable benchmarking** for AI systems.

Interrogating Agents, LLM-vs-LLM Interrogation & Persuasion Benchmark | Python, Ollama, ChromaDB, RAG, LLM-as-a-Judge [↗](#)

- Built an **LLM-vs-LLM interrogation benchmark**: two-channel RAG (ChromaDB, MiniLM sentence embeddings, dialogue-phase metadata filters) over 18 interrogation technique cards, with a **blinded LLM judge** scoring matched control/treatment legs in one batched call, cancelling calibration bias.
- Owned **reproducibility engineering**: offline-hardened local inference (Ollama, Llama 3.1 8B) at zero API cost, automatic runtime discovery across macOS/Linux/Windows, one-command setup, and a **GitHub Actions CI matrix** across 3 OSes and 2 Python versions.

Chain-of-Thought Prompting vs. Fine-Tuning for Table Reasoning | Python, PyTorch, Hugging Face, FLAN-T5 [↗](#)

- Benchmarked **11 conditions** spanning zero-shot, plain and structured **CoT prompting**, and answer-only versus rationale-trace **fine-tuning** on FLAN-T5 (250M/780M) over WikiTableQuestions with dual seeds, isolating a defensible negative result: zero-shot (**0.241 EM**) beat every intervention.
- Ran the **TabFact cross-dataset generalization study** and **McNemar's significance testing**, showing fine-tuning overfit to task format and failed to transfer (below the 0.551 majority-class floor), with error attribution across lookup, aggregation, and multi-hop reasoning types.

L-Store, HTAP Database Engine Built from Scratch | Python, Columnar Storage, Bufferpool, Two-Phase Locking [↗](#)

- Built a **hybrid transactional/analytical (HTAP) database engine** from scratch in Python (5-person team): lineage-based columnar storage splitting immutable base pages from append-only tail pages, with indirection columns resolving each record's latest version at read time.
- Implemented an **LRU bufferpool** with dirty-page write-back, a **background merge** compacting tail updates into read-optimized base pages, and multithreaded transactions under **record-level two-phase locking**, validated by throughput autograding across 3 milestones.

EDUCATION

University of California, Davis

Graduated 2026

Bachelor of Science in Data Science | GPA: 3.60/4.0

Honors: 3x Dean's Honors List (Spring 2023, Spring 2025, Spring 2026)

Relevant Coursework: Artificial Intelligence, Algorithms for Data Science, Data Processing Pipelines, Databases for Data Science, Big Data & High Performance Computing