Ayush Ranjan

aranjan1@ucsc.edu | (831) 266-5973 | linkedin.com/in/ayuranjan
github.com/ayuranjan | ayuranjan.github.io

Summary

Engineer with 2.5 years of experience across **backend**, **full-stack**, and **AI systems**. At **Capgemini**, I led **data modeling** and built **Java-based diagnostic tools** for Mercedes-Benz. At **UCSC**, I developed **LLM agents**, **vector search**, and **multimodal AI applications** using **Python**, **LangChain**, and **PostgreSQL**. Skilled in **Java**, **Python**, **SQL**, and **LangGraph**, with a focus on scalable, production-grade systems.

Education

University of California, Santa Cruz **Sep 2023 – August 2025**
Master of Science (MS) in Computer Science *CGPA: 3.92/4*

- **Relevant Coursework:** Analysis of Algorithms, Design and Implementation of Database Systems, Deep Learning for Advanced Computer Vision, Artificial Intelligence (AI), Applied ML: Deep Learning(DL), Computer Networks

Manipal University, Jaipur **July 2017 – May 2021**
Bachelor of Technology (B.Tech.) in Information Technology

- **Relevant Coursework:** Operating Systems, Data Mining and Warehousing, Data Science, Cryptography and Network Security, Advanced Data Structures, Natural Language Processing

Work Experience

University of California, Santa Cruz ***Santa Cruz, CA***
Teaching Assistant *Jan 2024 – Present*

- **CSE-180 Database Systems I**: Designed assignments and facilitated projects/labs on **SQL**, **PostgreSQL internals**, transaction management, indexing strategies, **stored functions**, and **PL/pgSQL**.
- **CSE-182 Introduction to Database Management Systems**: Facilitated projects and labs on **relational data models**, SQL, **Python database access**, transactions, stored functions, and PL/pgSQL.
- **CSE-115A Introduction to Software Engineering**: Mentored student groups with an **Agile** focus on software projects.

AI Explainability and Accountability (AIEA) Lab, UCSC ***Santa Cruz, CA***
Graduate Researcher *Oct 2024 – Present*

- Conducting applied research on improving the **reliability and explainability** of LLM-based university chatbots, focused on campus-wide use cases such as enrollment, deadlines, housing, and course queries.
- Prototyping advanced **agentic RAG architectures** including **Classic RAG, Chain of Thought, Agentic RAG, Adaptive RAG, Corrective RAG, and RAT RAG** to evaluate retrieval efficiency and hallucination resistance.
- **Fine-tuning open-source LLMs** to align with university-specific tone, structure, and factual accuracy, enabling domain-adaptive generation for student and administrative queries.
- Implementing **LangGraph-based multi-agent workflows** with support for **loop guards**, **tool chaining**, and **hallucination checks** and custom doc relevance scoring.
- Evaluating retrieval strategies using Ragas with **A/B testing**, grounding-score analytics, and student usability feedback to iteratively improve answer quality and trustworthiness.

Information Retrieval and Knowledge Management Lab, UCSC ***Santa Cruz, CA***
AI Research Intern *July 2024 – Sep 2024*

- Partnered with a stealth hardware startup to **develop a 0-to-1 multimodal AI agent for smart wearable devices** (camera-integrated earphones), enabling on-device calorie estimation, object detection, and real-time video summarization.
- Built a modular **LangChain + Dialogflow** pipeline with a unified memory layer to manage stateful voice-video interactions; supported context switching, multi-intent execution, and follow-up reasoning.
- Implemented **RAG-based intelligent query routing** using vector similarity on real-time audio/video inputs, enabling fallback to external tools like **web search or OCR pipelines** based on input ambiguity.
- Developed a multi-threaded memory manager to asynchronously encode and cache historical observations (images, transcripts) into vector embeddings using **Hugging Face transformers**, with storage in **Pinecone**.
- Integrated the prototype with a **local edge pipeline** (**FFmpeg**, Whisper, and custom vision models), **achieving sub-500ms inference latency for key commands** and enabling real-time calorie detection via food segmentation and pose estimation.

Capgemini Technology Services India Limited ***Mumbai, India***
Software Engineer I & Software Engineer II *July 2021 – Aug 2023*

- **Software Engineer II** (Oct 2022 – Aug 2023)
  - **Headed the Data Modeling Team** for Mercedes-Benz's XDIS platform, focusing on backend schema evolution for **vehicle network topology change requests** (e.g., ECU reconfigurations, bus architecture edits).
  - Designed a lightweight **ETL pipeline in Java** to process large XML diagnostic files: extracted raw telemetry data, transformed it into updated entity structures, and loaded it into IBM **Db2** tables—supporting seamless data migration.
  - Wrote and tuned complex **SQL queries and views** in Db2 to support schema validation, relational consistency checks, and historical topology comparisons for Change Request(CR) automation workflows.

    - **3rd Place at Innocircle 2022, Mercedes-Benz Internal Innovation Forum**: Implemented micro frontend architecture to complement the existing process, enabling users to modify their vehicle network topology and review changes, **eliminating previous dependencies and saving more than 50% of the time**.
    - **Prototyped an AI-assisted validation system** for **2,500+** historical Change Requests by embedding symbolic vehicle network topologies using custom **Word2Vec** and **Sentence-BERT models**. Used **LightGBM**-based anomaly scoring to **flag rare configurations** (top 5% outliers), and applied KMeans clustering (K=8) with nearest-neighbor search to **recommend optimal topologies** based on similarity to **850+** successful prior deployments.
- **Software Engineer I** (July 2021 – Sep 2022)
    - **Role**: **Java Backend Developer**
    - Worked on **XDIS**, a **SOAP**-based diagnostic tool structured around a three-tier monolithic Java architecture used by **Mercedes-Benz** service teams for vehicle automation and diagnostics.
    - Implemented and maintained backend modules using **Core Java**, **JAXB**, and **JDBC**; performed **data modeling** for diagnostic entities; applied **design patterns** and contributed to performance-critical sections of the legacy system.
    - Dramatically optimized **XML** file migration time by an impressive **66.67%**. Additionally, concurrently implemented indexing strategies for associated IBM Db2 database tables, enhancing the tool's robustness.
    - Optimized export testing by creating a wrapper around the Autosar framework and implementing an **XML file** import strategy, **reducing overall testing time by 40%** and speeding up export time for **individual modules by 17% on average**.
- **Software Engineer Intern** (Jan 2021 – May 2021)
    - **Role**: **Java Full Stack Developer**
    - Built a full-stack **Medical Portal**: **Spring Boot REST MVC** backend exposing 17 JSON CRUD/search endpoints documented with **Swagger**, paired with a **React** single-page application using **Redux** and **Axios**, giving the team faster turnaround on new features.
    - Integrated **MySQL** via JPA/Hibernate and secured the APIs with **Spring Security** and JWT, implementing role-based access control (**RBAC**) for Admin, Doctor, and Patient roles; added automated tests with **JUnit**, **Jest**, and **React Testing Library** for reliable releases.
    - Containerized the stack with **Docker Compose**, set up a **GitHub Actions** CI pipeline, and deployed to a local **Minikube Kubernetes** cluster to provide quick, environment-consistent demos for QA and stakeholders.

Selected Projects

PgVector+ | ***C/C++, PL/pgSQL, Database Systems, Vector Similarity*** *Ppt* ***Jan 2024 – Mar 2024***

- Designed and built a **custom PostgreSQL extension** on top of pgvector to support hybrid similarity-dissimilarity search and low-level query composition, bridging gaps in vector DB functionality seen in systems like Pinecone and Qdrant.
- Prototyped a `compound_similarity()` operator in **PL/pgSQL** to support queries like "similar to X, unlike Y" using cosine and inner-product thresholds.
- Prototyped PL/pgSQL-based `search_similar_vectors()` function to simulate centroid-based multi-query composition and validate set-based similarity retrieval.
- Earned an **A grade** in CSE 215 for system-level innovation in vector search acceleration and database extensibility.

Unveiling Glitches in CLIP | ***Hugging Face, Python, pgVector, OpenAI-API*** *Arxiv* ***Jan 2024 – March 2024***

- Conducted in-depth analysis of the CLIP model's image comprehension capabilities. Identified and documented **14 systemic faults**, including **four novel faults**, impacting CLIP's interpretation of images using **two novel methodologies**.
- Implemented the Discrepancy Analysis Framework (**DAF**) to analyze discrepancies in image similarity rankings between CLIP and **DINOv2** and utilized **OpenAI's GPT API** to identify and analyze faults systematically. Utilized the Transformative Caption Analysis for CLIP (**TCAC**) approach to evaluate CLIP's response to transformations applied to images.
- Achieved **A+ grade** in CSE 290D Neural Computation at UCSC for this project.

Video to MP3 Converter Service | ***Python, Flask, Docker, Kubernetes, RabbitMQ, MongoDB*** *GitHub* ***Dec 2023 – Jan 2024***

- Designed a modular **microservices-based** system with four components: auth service, API gateway, uploader, and converter.
- Used **FastAPI** and **Flask** for building secure, high-performance REST endpoints, authenticated with **JWT tokens** and **role-based access control**; ensured protected video operations via the centralized API gateway.
- Enabled asynchronous video-to-MP3 conversion using **RabbitMQ** and **MoviePy**, supporting non-blocking task execution.
- Deployed services in **Docker** containers and orchestrated with **Kubernetes** via **Minikube** for scalable local development.

Technical Skills

Programming Languages: Python, Java, TypeScript, JavaScript, C/C++, Go, Rust, SQL
AI/ML Tooling: LangChain, LangGraph, Hugging Face Transformers, Sentence-BERT, Word2Vec, LightGBM, scikit-learn
Backend Development: Spring Boot, REST APIs, GraphQL, Microservices, Core Java, FastAPI, Flask, PL/pgSQL, JDBC
Frontend Development: React, Next.js, Redux, Axios, HTML/CSS, Swagger / OpenAPI, JWT, RBAC
Databases: PostgreSQL, MySQL, Oracle, IBM Db2, MongoDB, Redis, Vector Databases (pgvector, Pinecone)
Data Engineering: ETL (Java/XML), Liquibase, ffmpeg, Whisper, Pandas, NumPy
DevOps & Infrastructure: Docker, Docker Compose, Kubernetes, Minikube, Git, GitHub Actions, Jenkins, Maven, Gradle
Testing and Automation: JUnit, Mockito, Jest, React Testing Library, Playwright, TDD, BDD
Cloud and Distributed Systems: AWS (EC2, S3, RDS, Lambda, DynamoDB), GCP, Azure
Practices & Miscellaneous: Agile/SAFe, CI/CD, Debugging, Troubleshooting, Documentation, SDLC Lifecycle