

Amos Jaimes

Boston, MA | (617) 934-9015 | jaimes.a@northeastern.edu | [LinkedIn](#) | [GitHub](#)

EDUCATION

Northeastern University

MS in Applied Machine Intelligence; GPA – 3.8/4.0

Indian Institute of Information Technology

Advanced Data Science Diploma; GPA – 3.6/4.0

Sep 2024 – Mar 2026

Boston, MA

Dec 2023 – Jul 2024

Bangalore, India

SKILLS

Languages: Python, TypeScript, SQL, C++, C, Java

Frameworks & APIs: FastAPI, Flask, NestJS, Next.js, React Native, REST APIs, Docker

AI/ML & CV: LangChain, LangGraph, RAG, Prompt Engineering, OpenCV, MediaPipe, Open3D, scikit-learn, XGBoost, PyTorch, HuggingFace, Pandas, NumPy

Databases & Cloud: PostgreSQL, Firebase, Supabase, Vector Databases (pgvector, FAISS), Azure, AWS S3, Redis, MongoDB, GitHub Actions, CI/CD

Data & Orchestration: Apache Airflow, Azure Data Factory, DuckDB, PySpark, MLflow, Evidently AI, Power BI

PROFESSIONAL EXPERIENCE

Slay

Aug 2025 – Dec 2025

AI/ML Engineer Intern

Boston, USA

- Developed a personalized beauty product recommendation system using ML models and data-driven algorithms, directly improving customer engagement metrics across the platform.
- Built a real-time 3D face processing application in Python (Flask) and Three.js, streaming RGBD depth data from iPhone via Record3D; implemented a 468-point MediaPipe Face Mesh pipeline with OpenCV for head pose estimation and face shape classification using Open3D mesh generation.
- Maintained and optimized Azure SQL and SQL Server databases supporting 4 production applications — identified and resolved 8 slow-running queries, reducing report latency by 40%, and improved 12 Azure Data Factory ETL pipeline reliability from 85% to 99.5% uptime.

Northeastern University

Mar 2025 – Jun 2025

Data Engineer (Experiential Learning)

Boston, USA

- Designed end-to-end data workflows for a production-grade booking platform using FastAPI, PostgreSQL, and Azure, implementing CRUD business logic, relational schema design, and asynchronous background jobs to optimize API response times.
- Collaborated directly with stakeholders to gather requirements, conducted gap analysis, and delivered RESTful backend services following agile sprint cycles with regular code reviews.
- Tested web application with integrated chatbot, filed and tracked bugs, and coordinated cross-team resolutions to accelerate production deployment.

National Instruments (NI)

Aug 2022 – Jul 2024

Software Engineer (Partner Support)

Bangalore, India

- Led Scrum ceremonies for two teams and acted as Tech Lead triaging critical production bugs; engineered C and LabVIEW instrument drivers integrating hardware into the NI Testing Environment.
- Developed and maintained 1,200+ migration pipelines converting MSI/EXE packages to NI Package Manager format, supporting reliable software distribution across the engineering organization.
- Mentored interns, co-authored 5+ published software solutions, and collaborated with partners to resolve customer queries across engineering and software purchase workflows.

PROJECTS

GYM Coaching App *Full-Stack AI Platform* | [Code](#)

- Architected a multimodal AI coaching platform across 3 service planes (Core Services in NestJS, AI Orchestration, Media Pipeline) supporting Powerlifting, Bodybuilding, Rehab, and Diet programs; implemented a human-in-the-loop system where AI drafts plans and coaches validate before delivery.
- Designed a hybrid video analysis pipeline combining MediaPipe 33-point skeleton extraction with Gemini 2.5 Pro VLM for rep counting via signal processing; integrated a multi-provider LLM router (Claude Sonnet, Haiku, Gemini Flash) with Zod-validated structured JSON outputs and RAG via pgvector (1536-dim) across 50+ sessions.
- Built a React Native mobile app (Expo SDK 52 + WatermelonDB) with offline execution and video upload queue; deployed 9 concurrent BullMQ async pipelines (video analysis, meal-photo, plan generation, weekly adaptation, notifications) with idempotent job keys and cost tracking on every LLM call.

Subscription Health & Experimentation Engine *Product Data Science & Analytics* | [Code](#)

- Architected an end-to-end DuckDB ETL pipeline processing 30GB+ of transactional and behavioral streaming logs via out-of-core SQL, engineering core metrics including MRR, Customer LTV, Content Completion Rate, and 30-day cohort churn flags.
- Designed and simulated a randomized A/B test ($n \approx 149,860$) using Python and SciPy two-sample T-tests to validate a statistically significant 4.12% lift in LTV ($p < 0.00001$) for an Auto-Renew Nudge feature.

- Developed an interactive Streamlit dashboard for portfolio health visualization and a Power BI executive report with complex DAX measures and conditional formatting for self-service experiment reporting.

Dynamic Visual Ranker *End-to-End MLOps & Recommendation System* | [Code](#)

- Architected an end-to-end recommendation engine using a two-tower retrieval pipeline (PyTorch + OpenAI CLIP) to generate semantic embeddings for e-commerce product images and text metadata.
- Trained an XGBoost Learning-to-Rank model (XGBRanker) optimizing NDCG, with a PySpark distributed feature pipeline storing vectors in a Redis feature store for sub-100ms retrieval.
- Automated daily ML retraining with Apache Airflow, tracked experiments via MLflow, and deployed the ranking engine as a FastAPI/Docker microservice with Evidently AI monitoring for feature drift.

High-Throughput Audio Processing Pipeline *Asynchronous ML Pipeline* | [Code](#)

- Architected an asynchronous microservices pipeline using FastAPI and Redis to decouple high-latency inference from user-facing APIs, achieving sub-100ms response times under concurrent audio streams.
- Integrated OpenAI Whisper for speech recognition and RoBERTa-based models for real-time sentiment analysis; engineered a Singleton-pattern model warm-up strategy reducing cold-start latency by 90%.
- Designed a MongoDB job lifecycle state-management system with structured logging, health-check endpoints, and containerized services via Docker enabling 5x traffic spike horizontal scaling.

Distributed Job Scheduler *Distributed Systems & Infrastructure* | [Code](#)

- Designed a distributed job scheduling system using FastAPI, PostgreSQL, Redis Streams, and Docker with leader election, reconciliation logic, and resource-aware CPU/memory scheduling across scalable worker nodes.
- Implemented worker heartbeat monitoring, crash detection, and auto-rescheduling with exponential backoff retry; supported Python, shell, ETL, and custom Docker workload execution via isolated container runtimes.
- Integrated Prometheus and Grafana for real-time observability of queue depth, worker health, and job latency; engineered ACID-backed PostgreSQL state transitions for correctness under concurrent restarts.

Customer Feedback Classifier *Production ML System* | [Code](#)

- Fine-tuned DistilBERT using PyTorch and HuggingFace Transformers with a custom training loop, learning rate scheduling, and gradient clipping; built an async FastAPI inference backend supporting batch prediction of up to 100 texts per request.
- Built a LangGraph-based intelligent agent with conditional routing for query complexity analysis, GPT-4o-mini integration, and automated SageMaker deployment pipeline with S3 artifact upload and auto-scaling.
- Integrated CloudWatch for custom metrics (inference latency, prediction confidence, error rates) with SNS alarm integration; containerized with Docker using graceful model fallback and modular layered architecture.

Medical Data ETL Pipeline *Cloud-Native Data Engineering* | [Code](#)

- Built a cloud-native ETL pipeline extracting FDA drug approvals and clinical trial data from OpenFDA and ClinicalTrials.gov APIs; orchestrated workflows via Apache Airflow DAGs with KubernetesPodOperator for isolated containerized task execution.
- Stored processed data in an AWS S3 Data Lake with date-based partitioning and automated CI/CD via GitHub Actions for DAG validation, test execution, and triggered deployments.
- Implemented data governance and quality checks on API-sourced data; built a custom frontend dashboard for real-time visibility into DAG runs, task statuses, and error logs.

Research Agent System *Multi-Agent AI Research Platform* | [Code](#)

- Built a multi-agent collaboration system with specialized AI agents (Researcher, Critic, Synthesizer, Insight Generator) orchestrated via LangGraph for state management and multi-session research project organization.
- Integrated arXiv API for automated paper discovery and PDF ingestion; implemented semantic search using SentenceTransformers (all-MiniLM-L6-v2) and FAISS indexing for fast persistent vector search with evidence retrieval.
- Developed a Streamlit web interface with agent collaboration visualization, interactive dashboard, and CLI with session management and research commands (ingest, query, research) with export capabilities.

CPS AI Assistant *RAG-based Conversational Agent* | [Code](#)

- Built an enterprise-grade RAG web application to automate academic advisor workflows, integrating LangChain and Supabase vector databases for real-time query processing across multiple enterprise systems and APIs.
- Designed automated data collection workflows for structured and unstructured web data with ingestion pipelines featuring validation, profiling, and schema definitions ensuring consistency across sources.
- Delivered measurable reduction in student query response time through automated service delivery, with comprehensive documentation enabling seamless knowledge base updates.