Nitin Jayavarapu

Fremont, CA | nitinjayavarapu12@gmail.com | +1-448-219-7141
GitHub | LinkedIn | Porfolio

Professional Summary

A Data Science graduate student and an AI Engineer with 2+ years of hands-on experience building and operating production ML systems in real-world environments. I focused on reliability problems—messy camera feeds, latency bottlenecks, and drift in live data—and iterate with a data-first approach to improve accuracy and stability. Experienced deploying containerized inference services with FastAPI and Docker on AWS, balancing performance, quality, and infrastructure cost under real usage.

Technical Skills

Programming & Data: Python, SQL, Pandas, NumPy, MySQL, SQL Server
Machine Learning & Deep Learning: PyTorch, TensorFlow, Scikit-Learn
Computer Vision: Object Detection (YOLOv5, YOLOv8), OpenCV
LLM & NLP Systems: Hugging Face, Embeddings, RAG, FAISS, Prompt Engineering, Groq API, LLaMA 3, Structured Output Extraction, Confidence Scoring
ML Systems & Deployment: FastAPI, Docker, MLflow, CI/CD, AWS (EC2, S3, Lambda), Render, Supabase (PostgreSQL)
Evaluation & Monitoring: Precision@k, Grounding & Response Quality Evaluation, Data Drift Detection, Latency Analysis
Integrations & Tooling: Twilio SMS API, PyMuPDF, WeasyPrint, REST API Design

Experience

AI Engineer January 2022 – November 2023
Black Box *Bangalore, India*

- Built and productionized a face recognition system used by 10,000+ daily users, where access workflows depended on accurate identification despite inconsistent lighting, partial occlusions, and noisy camera feeds across deployments.
- Diagnosed inference latency and GPU memory constraints that caused slow responses and accuracy regressions; redesigned batching and applied quantization, reducing end-to-end latency by ˜40% while preserving recognition quality under live traffic.
- Used production failure logs to find recurring misclassifications, curated hard-example datasets, and retrained models in successive releases, reducing repeat recognition errors by 20–30% in real environments.
- Deployed and operated containerized FastAPI inference APIs on AWS EC2, closing gaps in logging/health checks and handling traffic spikes and restarts to maintain ˜99.9% uptime while balancing latency, accuracy, and cost.

Education

University of West Florida Pensacola, FL
Master of Science in Data Science, GPA: 3.51/4.0 *January 2024 – December 2025*

- Relevant Coursework: Statistical Modeling, Machine Learning, Deep Learning, Cloud Computing

Technical Projects

Location-Aware Semantic Recommendation System | *Python, FastAPI, Embeddings, SQLite, Docker*

- Built a recommendation service for location-aware users (e.g., remote workers looking for "quiet cafés to work"), where keyword/rule-based filtering produced irrelevant results and missed user intent.
- Implemented embedding-based semantic ranking with cosine similarity and normalization, improving contextual matching versus keyword baselines and reducing "nearby but wrong" recommendations.
- Added embedding-level caching and TTL-based API caching to reduce repeated inference work, improving response latency by ˜40–60% during iterative user queries.
- Developed a feedback-driven personalization module (like/dislike/click) that re-weights category preferences within a session, improving ranking relevance as users interact.

MediBridge – AI Patient Discharge Summarizer & Follow-Up Coach | *Python, FastAPI, Groq (LLaMA 3), Supabase, Twil*

- Built for discharged patients who misunderstand clinical notes (40–50% nationally), causing medication errors and preventable readmissions; designed a three-stage LLM pipeline over 4,999 real medical transcriptions (MTSamples) to translate, extract, and safety-check discharge instructions.
- Added a confidence scoring layer that flags fields the model inferred rather than explicitly extracted, preventing silent hallucinations in safety-critical medical output.
- Deployed a Twilio SMS check-in bot with secondary LLM red-flag detection and a live caregiver dashboard; full system hosted on Render with FastAPI and Supabase.