Krishna Chaitanya Bodepudi

St. Louis, MO (Open to Relocate) | 6468211711 | kcbodepudi@gmail.com | LinkedIn | Portfolio

MACHINE LEARNING ENGINEER

Machine Learning Engineer with 3+ years of experience building and deploying production-grade ML inference systems, LLM-powered backends, and embedding-based retrieval pipelines. Strong focus on FastAPI-based services, NLP workflows, and scalable retrieval architectures, with emphasis on reliability, failure handling, and performance optimization. Delivered applied ML systems that reduced inference latency by up to 25% and decreased malformed or partial responses by approximately 30% through robust validation, batching, and system-level optimizations.

TECHNICAL SKILLS

- **Programming & Data:** Python, SQL, Pandas, NumPy, PostgreSQL, SQLite, MongoDB, SQLAlchemy
- **Machine Learning & Applied AI:** Supervised and unsupervised learning, feature engineering, model evaluation, NLP pipelines, semantic similarity, Retrieval-Augmented Generation (RAG), embedding-based retrieval
- **LLM & NLP Systems:** Hugging Face Transformers, embeddings, vector similarity search, prompt refinement, context window optimization, hallucination mitigation, response validation
- **Inference & Backend Engineering:** FastAPI, REST APIs, async request handling, batching strategies, latency optimization, failure-safe inference design
- **MLOps, Cloud & Systems:** Docker, CI/CD (GitHub Actions), AWS (EC2, S3), Linux, Git, experiment tracking (basic), inference logging and monitoring
- **Vector Search & Supporting:** FAISS (local vector indexing, semantic retrieval, top-k optimization), PyTorch, Scikit-learn, Streamlit, basic Spark concepts, Tableau, Power BI

PROFESSIONAL EXPERIENCE

Machine Learning Engineer — Optum (UnitedHealth Group) USA (Remote)

January 2024 – Present

- Designed and deployed Retrieval-Augmented Generation (RAG) pipelines over structured and unstructured healthcare data, improving answer relevance and response accuracy by 35–40% in internal evaluations.
- Built FAISS-based semantic retrieval systems indexing 50K+ records, enabling sub-second similarity search and reducing noisy or irrelevant context injection by approximately 30%.
- Implemented strict input validation, response normalization, and fallback logic across inference APIs, reducing malformed or partial LLM outputs by 30% under concurrent workloads.
- Optimized inference throughput using async request handling, batching strategies, and response-size controls, achieving 20–25% reductions in end-to-end latency at peak concurrency.
- Analyzed model outputs across hundreds of real-world samples to detect semantic drift, low-confidence retrieval, and hallucination risks, driving iterative retrieval and prompt tuning cycles.
- Productionized FastAPI-based ML inference services with structured logging and monitoring, supporting stable usage across multiple internal teams and eliminating ad-hoc model execution paths.

Software Engineer — New Mek Solutions Hyderabad, India

January 2022 – December 2023

- Developed and deployed ML-backed inference services using FastAPI and Docker, standardizing model access across 3+ internal NLP and analytics workflows and reducing manual execution effort by 40%.
- Built and maintained REST APIs supporting concurrent internal requests, improving service reliability and reducing response variance by approximately 25%.
- Designed Python and SQL data pipelines to support model training and evaluation on datasets ranging from 100K to 2M+ records, improving data preparation consistency and experiment reproducibility.
- Implemented NLP pipelines using Hugging Face Transformers for classification, summarization, and information extraction, automating processing of thousands of unstructured documents per run.
- Improved average API response times by 25–35% through async request handling, query optimization, and backend performance tuning.

PROJECTS

Persistent Memory Layer for LLM Applications GitHub

Python | FastAPI | FAISS | SQLite | SQLAlchemy | LM Studio

- Designed and implemented a persistent memory service to overcome LLM context window limits while preventing cross-task and cross-user memory leakage across 10K+ stored memory entries.
- Built FAISS-based semantic retrieval pipelines enabling top-k context selection, reducing irrelevant prompt injection by approximately 35% during downstream inference.
- Implemented namespace-based isolation and request scoping, validating correctness under concurrent multi-session access with zero observed cross-context leakage.
- Developed an async FastAPI backend with durable SQLite persistence, ensuring consistent memory retrieval behavior across service restarts, crashes, and failure scenarios.
- Reduced average prompt size by 30–40% by decoupling conversational history from long-term memory retrieval, improving inference efficiency and response consistency.

EDUCATION

Saint Louis University St. Louis, MO

Master of Science in Information Systems GPA: 3.90