

Akash Sampath

AI/ML Engineer

(312) 982-4542 | akashsampath16@gmail.com | San Francisco, CA | [linkedin.com/in/akash-sampath/](https://www.linkedin.com/in/akash-sampath/)

Summary

AI/ML Engineer with 5+ years of experience designing and deploying production-scale machine learning systems, Generative AI applications, and autonomous AI agents. Specialized in building LLM-powered multi-agent workflows, retrieval-augmented generation (RAG) systems, and scalable cloud-native ML platforms using Python, PyTorch, LangGraph, FastAPI, and AWS. Proven track record of improving model accuracy, reducing inference latency, and delivering enterprise-grade AI solutions through distributed systems, MLOps, and modern software engineering best practices.

Professional Experience

Anthropic

AI/ML Engineer

Apr 2024 – Present

San Francisco, CA

- Developed autonomous software engineering agents using Python, PyTorch, LangGraph, and MCP that performed repository analysis, multi-step planning, code generation, and debugging workflows for over 180,000 monthly users.
- Improved task completion accuracy by 34% through advanced prompt orchestration, retrieval-augmented reasoning, agent memory optimization, and structured evaluation pipelines built with Python, LangGraph, FAISS, and OpenTelemetry.
- Reduced end-to-end inference latency by 41% using semantic context retrieval, response caching, asynchronous tool execution, parallel agent orchestration, and optimized long-context processing with Python and Redis.
- Engineered multi-agent workflows using Python, LangGraph, MCP, FastAPI, PyTorch, and Tree-sitter, enabling repository understanding, autonomous planning, tool calling, multi-file code editing, and intelligent debugging across complex codebases.
- Built scalable retrieval pipelines with Python, FAISS, PostgreSQL, Redis, Voyage embeddings, and Pydantic, delivering semantic code search, dependency-aware context retrieval, artifact generation, and automated repository indexing.
- Designed evaluation frameworks using Python, PyTorch, HumanEval, SWE-bench, MLflow, and OpenTelemetry to benchmark reasoning quality, tool utilization, code correctness, regression detection, and continuous agent performance improvements.
- Architected cloud-native microservices with FastAPI, Docker, Kubernetes, Kafka, Redis, and AWS EKS, enabling resilient agent orchestration, asynchronous task execution, distributed processing, and enterprise-grade deployment scalability.
- Developed secure enterprise integrations connecting GitHub, Jira, Slack, CI/CD pipelines, and internal APIs through MCP, deploying containerized services on AWS using S3, RDS, EKS, and CloudWatch monitoring.
- Implemented distributed backend services using FastAPI, gRPC, PostgreSQL, Docker, Kubernetes, Prometheus, Grafana, and AWS infrastructure, supporting scalable inference, observability, fault tolerance, and high-availability production workloads.
- Automated software delivery through GitHub Actions, Jenkins, Docker, Kubernetes, Terraform, Prometheus, Grafana, MLflow, and OpenTelemetry, improving continuous deployment, monitoring, reproducibility, governance, and operational reliability.

Accenture

Machine Learning Engineer

May 2020 – Jul 2023

India

- Developed real-time customer intent prediction pipelines using Python, PyTorch, and Hugging Face Transformers, enabling accurate intent classification and personalization across multi-channel customer interactions for improved engagement.
- Improved recommendation accuracy by 32% using hybrid models combining collaborative filtering and deep learning, enhancing next-best-action predictions and increasing engagement across marketing and call center channels.
- Reduced inference latency by 41% by optimizing real-time scoring APIs using FastAPI, Redis caching, and asynchronous processing, ensuring sub-100ms response times for high-volume customer interaction systems.
- Designed and deployed scalable NLP pipelines using BERT-based models and spaCy for intent detection and sentiment analysis, supporting automated customer support workflows and improving response consistency across enterprise systems.
- Built and maintained machine learning models for churn prediction, conversion scoring, and upsell recommendations using Python, Scikit-learn, and XGBoost, driving data-driven decision-making across marketing and customer engagement teams.
- Engineered large-scale data pipelines using Apache Spark, Kafka, and Airflow to process real-time customer interaction data, enabling continuous feature updates and improving model performance in dynamic production environments.
- Developed microservices-based architecture using FastAPI, Docker, and Kubernetes, enabling modular deployment of recommendation and NLP services with high scalability, fault tolerance, and seamless integration across enterprise systems.
- Deployed and managed machine learning workloads on AWS using S3, EC2, and SageMaker, enabling scalable model training, versioning, and real-time inference for customer engagement use cases across distributed systems.
- Improved system throughput by 35% and reduced infrastructure cost by 28% by implementing distributed processing and auto-scaling on AWS, supporting over 3,000+ daily active users across customer interaction platforms.
- Integrated real-time decisioning systems with AWS Kinesis, DynamoDB, and Lambda, enabling event-driven architectures for instant personalization, dynamic recommendations, and seamless customer experience across digital and call center channels.

Technical Skills

Programming Languages: Python, SQL, Bash, Java

AI/ML & Generative AI: PyTorch, Scikit-learn, XGBoost, Hugging Face Transformers, LangGraph, MCP, RAG, AI Agents, Multi-Agent Systems, Prompt Engineering, Agent Memory, Tool Calling, Function Calling, LLM Evaluation, Reasoning Systems, Fine-tuning, Embeddings, Semantic Search, Vector Search, BERT, NLP, Intent Classification, Sentiment Analysis, Recommendation Systems, Churn Prediction, HumanEval, SWE-bench, Knowledge Graphs, Hybrid Search, Context Engineering, Prompt Optimization, LLMOps

Frameworks & Backend Development: FastAPI, gRPC, Pydantic, REST APIs, Microservices, Asynchronous Programming, WebSockets

Data Engineering & Streaming: Apache Spark, Apache Kafka, Apache Airflow, AWS Kinesis, ETL Pipelines, Data Processing, Feature Engineering, Data Modeling, Data Validation, Batch Processing, Stream Processing

Databases & Storage: PostgreSQL, Redis, FAISS, DynamoDB, Amazon S3, Amazon RDS, Vector Databases, Pinecone, ChromaDB

Cloud, DevOps & MLOps: AWS (EKS, EC2, SageMaker, S3, RDS, Lambda, CloudWatch), Docker, Kubernetes, Terraform, GitHub Actions, Jenkins, MLflow, OpenTelemetry, Prometheus, Grafana, CI/CD, Containerization, Model Deployment, Model Versioning, Distributed Systems, Auto Scaling, Infrastructure as Code (IaC), GPU Computing, Inference Scaling

Developer Tools & Enterprise Integrations: Git, GitHub, Jira, Slack, Tree-sitter, Repository Analysis, Code Generation, Automated Debugging, Code Review Automation, Dependency Analysis, GitOps

Soft Skills: Analytical Thinking, System Design, Software Architecture, Cross-functional Collaboration, Stakeholder Management, Project Ownership, Agile Development, Time Management, Adaptability, Continuous Learning, Attention to Detail

Education

Master of Science in Artificial Intelligence

Depaul University

Certifications

AWS Certified Machine Learning — Associate

AWS Certified AI Practitioner

Anthropic: Model Context Protocol

Anthropic: Claude with Amazon Bedrock

Projects

Explainability Agent — RAG & Copilot Integration

Tech Stack: Python, Microsoft Copilot Studio, LangChain, FAISS, LightGBM, Sentence Transformers, Optuna, DVC

- Developed a Retrieval-Augmented Generation (RAG) explainability agent integrated with Microsoft Copilot Studio to generate transparent, evidence-backed explanations for AI recommendations using semantic retrieval and LLMs.
- Built a hybrid retrieval and generation pipeline using FAISS, Sentence Transformers, and LightGBM ranking, improving retrieval relevance while reducing LLM token usage through adaptive cost-aware prompting and caching.
- Implemented an end-to-end evaluation and optimization workflow with DVC, Optuna, confidence calibration, and retrieval quality metrics, enabling reproducible experiments and efficient model tuning.

Encrypted Communication Framework for Autonomous Systems

Tech Stack: Python, PyTorch, Reinforcement Learning, Autoencoders, AES-GCM, PKI, Blockchain

- Designed and implemented a secure communication framework for autonomous systems using PKI-based authentication and AES-GCM encryption to ensure confidential and authenticated data exchange.
- Developed a deep learning-based anomaly detection module using PyTorch autoencoders and explored reinforcement learning for adaptive communication channel selection under simulated adversarial network conditions.
- Built an end-to-end proof of concept with encrypted audit logging and privacy-preserving communication concepts, evaluating system performance through simulation-based security and reliability experiments.