Akhil Shekkari

Portfolio | GitHub | LinkedIn | shekkari.akhil@gmail.com | +1 (425) 426-8292

Summary

AI Engineer with 3+ years of industry experience building and deploying **production LLM systems**. Specialized in Agents, post-training, and inference optimization, with a focus on evaluation and scalable system design.

Experience

AI Engineer | Atrium | Silver Spring, USA June 2025 - August 2025

- Automated Statistical Analysis Plan generation from clinical trial documents for **Pfizer's team**, reducing drafting time by up to **60%**.
- Evaluated knowledge graphs, naive chunking, and semantic chunking strategies; benchmarked **embedding models** (OpenAI) and **hybrid retrieval** (dense + BM25 sparse), and selected the best-performing RAG configuration on internal test sets.
- Built a custom **LLM-as-a-Judge** evaluation system with structured rubrics scoring completeness, guideline adherence, and hallucination detection against ground truth SAPs, achieving **82% precision** and **78% recall**.
- Conducted multiple prompt experiments improving generation consistency by **20%** and reducing manual revision cycles by **30%**.

Machine Learning Engineer | Tezo | India July 2021 - July 2024

- Built a **RAG** powered chatbot over SharePoint repositories, giving **200+** employees a unified search interface across **1,000+** internal documents with sub-second retrieval.
- Engineered a **semantic search** pipeline (embeddings + vector DB) reducing document lookup time by **60%**, and added LLM-based summarization cutting manager review time by **35%**.
- Optimized **chunking** and **retrieval** by benchmarking chunk sizes, overlap strategies, and hybrid search, improving answer relevance and reducing hallucination rates across document types.
- Built an **evaluation** pipeline to measure RAG output quality tracking faithfulness, retrieval relevance, and hallucination rates across queries, enabling data-driven iteration on the retrieval and generation stages.
- **Trained ML models** for insurance policy scoring and fraud detection, improving fraud recall by **15%** and enabling earlier intervention across the claims pipeline.

Projects

RL Post-Training for LLM Reasoning at Scale GitHub 2025

- Post-trained **Qwen-2.5** into a reasoning model in raw PyTorch, reproducing a DeepSeek-R1-style pipeline: supervised fine-tuning on chain-of-thought traces followed by **GRPO** reinforcement learning to elicit long-horizon reasoning without a learned critic.
- Scaled **RL** training across multiple GPUs using **FSDP** with mixed precision and gradient accumulation; overlapped rollout generation and policy updates via **NCCL** to maximize hardware utilization.
- **Profiled** distributed training with **Nsight Systems** and **Nsight Compute**, diagnosing all-reduce communication stalls and CUDA kernel launch overhead to reduce idle time and improve GPU efficiency.
- **Evaluated performance** on GSM8K and MATH using a SymPy-based symbolic equivalence harness, observing consistent gains over the base model after SFT and GRPO.

LLM Inference Engine from Scratch GitHub 2025

- Built a modular **inference engine** with interchangeable attention backends: naive $O(T^2)$, streaming $O(T)$ with online softmax, and a **Triton**-fused FlashAttention kernel with tiled memory access.
- Implemented **paged KV**-cache with on-demand allocation, reducing peak memory from $O(T^2)$ to $O(T)$ compared to contiguous caching.
- Benchmarked **prefill** and **decode** throughput across **4K–16K** context lengths, quantifying Triton kernel-fusion speedups over Python-level implementations.
- Profiled GPU kernels to characterize memory-bound vs compute-bound regimes and measured global memory throughput relative to **roofline** limits.

AI Agent Framework GitHub 2025

- Built a **multi-step reasoning agent from scratch** with function calling, **MCP** integration for external tool servers, Pydantic-validated structured outputs, sliding-window memory with compaction, and evaluated on the **GAIA** benchmark.

Technical Skills

Languages & Frameworks: Python, PyTorch, Triton, SQL, Pydantic, FastAPI, Gradio, LangChain, LlamaIndex
ML & Training: LoRA/QLoRA, FSDP, Hugging Face Transformers, Unsloth, W&B, ClearML, Scikit-learn
Infrastructure: Docker, Kubernetes, AWS SageMaker, Azure ML, Snowflake, GitHub Actions, CI/CD

Education

University of Maryland, College Park Expected May 2026
Master of Science in Applied Machine Learning
Coursework: Deep Learning, NLP, Advanced ML, Reinforcement Learning, Optimization, Probability & Statistics