Muhammad Qasim

Applied ML Engineer — Retrieval Systems & Search Infrastructure

kas-sim.github.io linkedin.com/in/kas-sim github.com/Kas-sim amkassim444@gmail.com

Technical Summary

Languages: Python (4 yrs), Java (2 yrs), C++ (1.5 yrs), SQL
AI Infrastructure: LlamaIndex, LangChain, ChromaDB, Weaviate, HuggingFace, Streamlit
Retrieval Systems: RAG Pipelines, Hybrid Search (Sparse/Dense), Cross-Encoders, OCR Ingestion
Core Systems: Linux, Docker, CI/CD (GitHub Actions), File Locking/Concurrency, Memory Management
Algorithms: Inverted Indices, Tries, Vector Quantization, Graph Traversal, O(1) Lookups

Selected Projects

MQNotebook — Enterprise-Grade RAG System Python, LlamaIndex, OCR

Engineered a local RAG engine capable of parsing complex formats (scanned PDFs, PPTX speaker notes, XLSX) using a custom Tesseract + Poppler OCR pipeline.

Implemented a **Hybrid Search** architecture combining dense vector retrieval with a **Cross-Encoder Reranker**, improving context precision by 40% compared to naive cosine similarity.

Solved critical Windows file-locking (WinError 32) issues in persistent vector stores by architecting a dynamic, session-isolated storage handler.

Optimized context injection to reduce token usage by **60%**, deploying the solution to Streamlit Cloud with BYOK (Bring Your Own Key) security.

DevShelf — Distributed Vertical Search Engine Java, Systems Architecture

Architected a search engine from first principles (no Lucene/ElasticSearch), implementing a custom **Positional Inverted Index** for O(1) keyword retrieval.

Engineered a custom **Vector Space Model** ranking algorithm incorporating TF-IDF and user popularity signals.

Built a specialized ”Offline Indexer” to pre-process corpora, reducing runtime query latency to sub-millisecond levels.

Implemented an O(L) Trie-based autocomplete system and a Levenshtein Distance fuzzy matcher for typo tolerance.

BabyGPT — LLM Fundamentals Python, TensorFlow

Built a character-level LSTM language model from scratch to understand the mathematics of sequence modeling.

Implemented a custom tokenizer and temperature-controlled sampling loop for generative text synthesis.

Experience

Arch Technologies (Remote)

Machine Learning Intern *Present*

Fine-tuned BERT models for NLP classification tasks; optimized preprocessing pipelines for improved training throughput.

Collaborated with engineering teams to integrate PyTorch models into internal production prototypes.

Education

Sukkur IBA University

Bachelor of Computer Science — Distributed Computing & AI Systems *Expected 2028*