KRISNA PRANAV

Software Engineer | AI/ML & Agentic Systems | Python · Rust · Go · TypeScript · C++/C

Coimbatore, India | krisna.pranav@gmail.com | linkedin.com/in/krishpranav | github.com/krishpranav | krishpranav.github.io

SUMMARY

Software engineer who has been coding since the 8th grade, with 5+ years of professional experience across backend, systems, and full-stack engineering, including **3+ years focused specifically on AI/ML**: agentic systems, LLM orchestration, and retrieval-augmented generation. I build RAG and tool-using LLM pipelines, on-device vector search engines, and the high-performance infrastructure that supports them, on a foundation of systems programming in Rust, C++, and Go. Comfortable owning a problem end-to-end: model integration, infra, and the product surface around it.

TECHNICAL SKILLS

AI / ML:	Agentic AI systems, LLM orchestration (Anthropic Claude, OpenAI APIs), tool-use & function calling, multi-step reasoning pipelines, prompt engineering & evaluation, Retrieval-Augmented Generation (RAG)
ML Infra:	Embeddings, vector indexing (HNSW, IVF), vector databases, semantic search, inference optimization, ONNX runtime, model quantization basics, GPU/CPU pipelining
MLOps:	Model deployment, inference serving, observability & tracing for LLM workflows, prompt/tool-use evaluation, model & pipeline versioning, CI/CD for model integration
Systems:	Rust (tokio async I/O, lock-free concurrency, zero-copy serialization), C/C++, Go, memory profiling, Linux systems programming
Backend & Infra:	gRPC, REST, GraphQL, PostgreSQL, Redis, Docker, Kubernetes, AWS, distributed systems
Languages:	Rust, Go, C/C++, Python, TypeScript

EXPERIENCE

AI/ML Engineer, Turing — Present

Remote, agentic systems & LLM infrastructure on Anthropic models

- Build agentic AI systems and LLM workflows on Anthropic models: the orchestration layer, tool-use plumbing, and the reasoning pipelines behind multi-step automation.
- Write the performance-critical infrastructure in Rust and C++, with Python at the ML boundary for model integration.
- Own the backend for AI orchestration: request routing, concurrent task execution, and state management for long-running agent runs.
- Focus on inference optimization, distributed systems for AI workloads, and the MLOps side of shipping frontier models into production.

AI Engineer, BlockStars — Apr to Jul

Freelance / Remote, computer-vision + LLM reasoning pipeline

- Built a video-based LLM parser: live video streams feed into YOLO for object detection, and the structured detections get passed to an LLM that reasons about the scene.
- Wrote the pipeline in C++, Rust, and Python, optimizing for low-latency frame processing and throughput from incoming stream to LLM output.
- Pick up additional AI projects across the company, covering model integration and deployment.

Senior Rust Engineer, Aether (AI Desktop Agent) — Oct to Dec

Contract, on-device retrieval & vector search

- Built a privacy-first AI desktop assistant in Rust: a local semantic engine indexes files, generates embeddings on-device, and answers context-aware queries in under 100ms.
- Designed the on-device vector store with HNSW retrieval, tuning graph parameters (M, ef_construction) to reach recall@10 above 0.95 on roughly 100K-document corpora, no GPU required.
- Owned the full stack: indexer, embedding pipeline, vector store, file watcher, IPC layer, and the Tauri UI. Shipped macOS and Linux binaries that idle under 50MB.

AI/ML Engineer, Legal RAG Platform (Contract) — Apr to Jun

Remote, retrieval-augmented legal information system

- Built a production retrieval-augmented system that answers procedural, compliance, and legal questions grounded in a curated knowledge base of statutes, government notifications, circulars, and judgments, with every answer returning verifiable source citations.

- Designed the ingestion pipeline end-to-end: PDF parsing, semantic chunking, metadata tagging (section, clause, date, jurisdiction), and embedding upsert into a hybrid vector + keyword store with reranking.
- Implemented multimodal document understanding so forwarded notices and scanned pages were parsed via vision-LLM extraction and fed into the same RAG + citation pipeline.
- Built an automated knowledge-base monitoring agent that scrapes designated sources on a schedule, detects new resolutions and court orders, and re-indexes the live knowledge base without manual intervention.

Senior Software Engineer, PayPal (External Contract) **Dec to May**

High-throughput backend migration

- Migrated critical C++ services to Go on a tier handling 15 to 20 million requests/day, keeping p99 latency on par with the legacy system while simplifying deploys.
- Built transaction-processing concurrency with goroutines and bounded channels for backpressure; ran the cutover with zero downtime via gradual traffic shifting.

Software Engineer, Trading Platform (Confidential) **Mar to Oct**

On-chain trading infrastructure

- Contributed to a high-frequency on-chain trading platform processing 20 to 25M+ in monthly trading volume, with feature depth comparable to leading on-chain trading terminals: real-time price feeds, fast swap execution, wallet/whale tracking, and live charting.
- Built low-latency trade execution paths in Rust, optimizing for sub-second order placement and minimal slippage during periods of high network congestion.
- Developed the trading interface in Next.js/TypeScript, including real-time order books, position tracking, and charting driven by WebSocket data streams.

Senior Full-Stack Engineer, PrivateFirm **Feb to Sep**

Full-stack web across multiple product surfaces

- Built and maintained full-stack web applications using Next.js, React, Golang, Python, and Node.js across several product surfaces.
- Cut p95 API latency by ~40% by restructuring queries, eliminating N+1 patterns, and adding a Redis cache for hot reads.
- Owned backend architecture decisions and led code review across the team.

AI/ML Engineer, Crossing Hurdles **Present**

Remote, improving production AI systems for clients including Auth0 and Handshake AI

- Improve client-facing AI systems for Auth0 and Handshake AI, building the real-time inference and orchestration paths in Python, Rust, C++, and TypeScript.
- Built a streaming inference gateway that fans out concurrent LLM calls and streams tokens back to clients over WebSockets/SSE, cutting time-to-first-token to under 300ms under load.
- Cut real-time retrieval latency with an in-memory embedding cache and incremental re-indexing, keeping vector search under 50ms p99 on live traffic.
- Hardened the serving layer in Rust with backpressure, request batching, and timeout/retry handling so streaming pipelines stay stable under bursty concurrent load.

Earlier roles: ~4 years across roughly ten contracts building production mobile apps (Flutter, React Native), full-stack web (Next.js, React, Node.js), and blockchain systems (Solidity, Solana/Anchor).

SELECTED OPEN-SOURCE WORK

Tauri (github.com/tauri-apps/tauri) - Core contributor to the Rust framework for building cross-platform desktop applications. Fix memory-leak bugs in the core runtime and contribute to core architecture.

- Fixed a memory leak in **tauri-runtime-wry** where `WithWebview` leaked Objective-C retains on macOS/iOS: replaced `Retained::into_raw()` with scoped `Retained` bindings via `Retained::as_ptr()`, keeping ObjC retain/release balanced so WebContent processes are no longer kept alive for the whole app lifetime (stabilized RSS: 116.5 to 116.9 MB over a 3-min stress loop). PR #15224, merged upstream.
- Contribute to core runtime architecture and concurrency internals across the Tauri codebase.

EDUCATION

BE, Computer Science Engineering, MCET