Krrish Choudhary

Jaipur, Rajasthan 302031, India | +91 7877624097 | krrishchoudhary109@gmail.com
linkedin.com/in/krrish-choudhary | github.com/Krrish2004 | Google Scholar | Codeforces (Expert) | Kaggle

Summary

Software & AI Engineer (B.Tech CSE, LNMIIT '27; **Codeforces Expert**) who builds AI assistants and the backends that run them, in **TypeScript**, **Python**, **Rust**, and **C++**. At **Incredible**, shipped a production **AI agent** platform connecting **300+** apps and handling **∼10M** events a month; raised task completion **∼45%**. At **GoQuant**, cut **∼120 µs** from a sub-millisecond trading data path. Founder of **Verace**: a new **LLM** architecture and open-source memory tools for AI coding agents. **3** published papers.

Education

LNMIIT, Jaipur — *Aug 2023 – Expected May 2027*
Bachelor of Technology (B.Tech), Computer Science Engineering — *Jaipur, India*

- **Coursework**: Data Structures & Algorithms, Operating Systems, Computer Networks, DBMS, Deep Learning, LLMs.
- **Capstone (BTP)**: GNN-based deepfake detection using spatio-temporal relationship modeling.

Eden International, Bhilwara — *Apr 2021 – Mar 2022*
12th Grade (CBSE) – 80% — *Bhilwara, India*

Experience

Founder & Engineer — *Apr 2026 – Current*
Verace Pvt Ltd (verace.in) — *Remote / India*

- Founded **Verace** to build a new **LLM** architecture (Verace V1) and open-source memory tools for AI coding assistants; as sole engineer, completed **1,579** tasks across **14** sprints and launched its web platform (verace.in) in **TypeScript** / **Next.js 15**.
- Owned the **system design** of the cloud backend that serves the model to **∼1,000** early users at p95 **∼250 ms**: **gRPC** microservices behind a **GraphQL** gateway on **Kubernetes** (**AWS** EKS / **GCP** GKE, **Terraform**), **Kafka** event streaming, **PostgreSQL** + pgvector as its memory store, and **vLLM** / **Hugging Face** serving with Prometheus / Grafana monitoring.

Software Developer — *Aug 2025 – Mar 2026*
Incredible — *Remote*

- Shipped **Incredible Chat**, a conversational **AI agent** that works across **100+** connected apps (Slack, LinkedIn, Google Workspace), and raised the relevance of the context it retrieves by **∼30%** in production using **RAG** (embedding-based semantic search) and custom **LLM** fine-tuning, confirmed by offline evals and online A/B tests.
- Built the trigger system that lets Incredible's agents act on their own, on a schedule or when a connected app fires an event, across **300+** third-party services and **∼10M** events/month, using **LangChain**, **MCP**, and **OpenAI** / **Anthropic** function calling with webhooks, timezone-aware cron, **OAuth2** account management, and at-least-once delivery.
- Raised task-completion rate **∼45%** over the single-agent baseline by designing a dual-agent **LLM** architecture that turns a user's plain-language request into an executable **Python** workflow across **100+** integrated services and custom agent tools.
- Built the core of the **Incredible Voice Assistant** in **Rust**, so users can call it by voice from any application in **8** locales, answering in under **300 ms** on under **80 MB** of idle memory, with on-device NLP and real-time screen capture.
- Cut end-to-end deploy time **∼70%** across **12+** services by rebuilding the **CI/CD** pipelines on **GitHub Actions** with layered build caching, parallel test stages, and **Helm**-managed releases on **Kubernetes**.
- Delivered the Incredible **API** platform, used by **300+** developers for **∼1M** calls/month (agentic reasoning, deep research, web search, image / video generation, OCR), plus a developer portal with API keys, usage analytics, and SDK generation.

Software Developer — *May 2025 – Jul 2025*
Kanchan — *Bhilwara, India*

- Cut median page-load time from **∼3.2 s** to **∼1.4 s** (**∼56%** better Largest Contentful Paint) on mobile and desktop by building a CDN-cached asset pipeline with image optimization, fingerprinted bundles, and edge caching.
- Lifted user engagement **∼40%** by leading the full website revamp across **3** partner teams (content, design, ops) and shipping a redesigned **React** front end with a modular component architecture.

Software Developer Intern — *May 2025 – Jun 2025*
Vaidsys Technologies — *Remote*

- Shipped a full-stack task-management app that stayed responsive under load (**100+** concurrent users at p95 **∼180 ms**), with an **Angular** (**TypeScript**) front end using **RxJS** state management and a **Node.js** / **Express** REST backend on **MongoDB**.
- Restricted each user to the data their role allows with **JWT** auth and **role-based access control (RBAC)**, and added **TDD** CI checks on **GitHub Actions** plus a **Playwright** end-to-end suite so partner ops could regression-test releases.

Back-End Engineer Intern — *Feb 2025 – Apr 2025*
GoQuant Technologies Inc. — *Miami, FL (Remote)*

- Raised throughput **15%** and cut p99 tick-to-trade latency **∼120 µs** on the firm's sub-millisecond order-book path (Binance / Deribit **WebSocket** feeds) by profiling the hot path and moving it to allocation-free structures in **low-latency C++**.
- Unified Binance and Deribit market data into one feed for the firm's OEM trading engine by building adapters that normalize L2 order-book and trade-tick formats, with **WebSocket** reconnection, heartbeat handling, and gap detection.

Projects

Agentic Memory: Persistent Memory MCP Server for AI Agents GitHub
Python, PostgreSQL, pgvector, Model Context Protocol (MCP)

- Built and open-sourced a memory server that lets AI coding assistants (Claude Code, Cursor, Antigravity, Codex CLI) retain memory across sessions, exposed over **MCP** (Model Context Protocol): six memory layers behind one ranked `unified_query`, prompt-injection-safe storage, and bitemporal belief revision that keeps a full audit history and never overwrites past beliefs.
- Isolated each tenant's data with a multi-tenant **PostgreSQL** + pgvector backend using Row-Level Security; shipped it as a Claude Code marketplace plugin with **9** autonomous hooks, validated by **98** automated tests.

Verace V1: Manifold-Based Multimodal LLM Architecture GitHub
Python, PyTorch, Triton

- Designed a from-scratch text-and-vision **LLM** architecture where memory per token stays constant and each token uses only the layers it needs (**2**–**128**), replacing softmax attention, **KV**-cache, and routed mixture-of-experts (**MoE**) with continuous manifold-based mechanisms and a latent energy critic that picks the lowest-energy branch in tree-search generation; ∼**40%** lower peak memory than a same-size transformer on 8K-token inputs.
- Implemented the reference version in **PyTorch** with custom **Triton** GPU kernels and a **7**-module unit / integration test suite.

DriftCall: OpenEnv RL Evals Gym for Voice-First Indic Concierge Agents under API Drift GitHub
Python, FastAPI, PyTorch, Unsloth, TRL, MLflow, Weights & Biases

- Trained a voice-first Indic concierge agent (flights, cabs, food, hotels, payments) to notice when the APIs it depends on change: drift-detection recall rose **+65 pp** with calibrated confidence after **500** GRPO steps on one V100, fine-tuning **Gemma 4** with **LoRA** (r=16) and **GRPO** via Unsloth + TRL in a custom **OpenEnv**-compliant **reinforcement learning (RL)** gym.
- Generated **200,000+** training episodes with a drift engine that mutates **5** versioned mock APIs in **20** patterns (schema, policy, T&C, pricing, auth), scored by **5** deterministic rewards with Brier calibration; added speech in and out with **faster-whisper-small ASR** (int8, 5 Indic languages) and **Kokoro-82M TTS** behind a **FastAPI** loop.

Research Papers & Publications

Hybrid GraphRAG for Cross-Lingual Legal Citation Retrieval: A Multi-Signal Fusion Approach for Swiss Legal Information Systems Paper · GitHub
IJECS Vol. 15, Issue 04, April 2026 · co-authored with T. Kandoi, S. Patel

- Built an English-to-German search pipeline that finds the right Swiss legal citations across **2.47M** court considerations, scoring Macro F1 **0.691** on the validation set (**+111%** over a BM25-only baseline) while returning only citations in the corpus, so it cannot invent one; hybrid **GraphRAG** retrieval (BM25 + BGE-M3 + Personalized PageRank + Leiden) with 3-stage verification (RRF, BGE-reranker-v2-m3, agentic Qwen2.5-7B) on FAISS + 2×T4 GPUs.

StreamRAG v2: Native Parser Daemons and Three-Tier Extraction for Real-Time Code Graphs Paper · Live Demo
*IJTRP Vol. 2, Issue 2, February 2026 · shipped as an **MCP** (Model Context Protocol) server*

- Cut AI code editors' token use **47%** and turns **19%** by keeping a live code graph that updates in under **0.05 ms** per change (**20K+** lines, 8 languages incl. **TypeScript**); built as an **MCP server** on a 10-stage Delta GraphBridge pipeline.

The Artificial Brain: A Neuroscience-Inspired Architecture for Multimodal AI Systems Paper
IJTRP Vol. 2, Issue 2, February 2026 · co-authored with T. Kandoi

- Proposed a brain-inspired design for multimodal **AI**: specialized Small Language Models process inputs in parallel like sensory cortices, a larger reasoning model is activated only when needed, and sleep-cycle fine-tuning consolidates new knowledge.

Accomplishments

College Rank #1 – AlgoUniversity Graphs Camp LinkedIn Post

- Ranked **#1** in college at AlgoUniversity Graphs Camp by solving **30+** problems in **3** days with **17+** advanced graph algorithms (Dinic's max-flow, min-cost max-flow, centroid decomposition, heavy-light decomposition, 0-1 BFS).
- Selected in the top **50** of **40,000+** applicants and named to the AlgoUniversity Hall of Fame; **Codeforces Expert** (**1600+**).

LUSIP Research Intern (LNMIIT, 2024) – Singing Deepfake Voice Detection

- Detected AI-generated (deepfake) singing voices with ∼**85%** validation accuracy on a ∼**12K**-sample dataset by training CNN + LSTM classifiers on Mel-spectrogram features in **PyTorch**, during a competitive research internship.

Technical Skills

Languages: TypeScript, JavaScript, Python, Rust, C++, Go, Java, SQL, Bash
AI Agents & LLM: Function Calling, RAG (Retrieval-Augmented Generation), Fine-tuning, LoRA, GRPO, Prompt Engineering, LangChain, MCP, PyTorch, TensorFlow, Hugging Face Transformers, vLLM, OpenAI API, Anthropic API, ASR, TTS, pgvector
Full-Stack & Backend: Next.js, React, Angular, Node.js, Express, REST, gRPC, GraphQL, WebSockets, Microservices, System Design, System Architecture, Kafka, OAuth2, JWT, RBAC, FastAPI, Flask
Data & Cloud: PostgreSQL, MongoDB, AWS, GCP, Azure, Docker, Kubernetes, Terraform, CI/CD (GitHub Actions), Linux
Practices: Agile, Scrum, TDD, Unit / Integration / End-to-End Testing, A/B Testing, Code Review