Bhaskar Gurram

AI Engineer · Research Software · Multimodal RAG · MLOps at Scale

Ashburn, VA | gurrambhaskar.ai@gmail.com | (513) 652-2523 | bhaskar.website
linkedin.com/in/bhaskar-gurram | github.com/bhaskargurram-ai
scholar.google.com/citations?user=0tSEbKIAAAAJ

Summary

AI engineer with 5+ years building production research software: large-scale retrieval and knowledge-graph systems (50M+ embeddings, sub-100ms P99), multi-agent orchestration platforms, and GPU-accelerated training/serving on Kubernetes. Sole author of Unwind, an Apache-2.0 reversibility layer for agentic tool use published to PyPI, npm, and ghcr.io with full open-source governance. Strong teaching and mentoring record (500+ graduate students, 20+ supervised capstones). M.S. thesis in multimodal neuroimaging; three 2026 arXiv preprints on evaluation rigor and selective risk control in AI systems.

Technical Skills

Languages: Python, SQL, JavaScript, TypeScript, R, C++, Bash

Open Source & Research Software: Apache-2.0 licensing, PyPI packaging, npm publishing, container publishing (ghcr.io), OSS governance (CODEOWNERS, GOVERNANCE, SECURITY policy, Code of Conduct, CITATION.cff), OpenSSF Scorecard, GitHub Actions CI, codecov, pre-commit, devcontainers

ML & GenAI: LangChain, LangGraph, LlamaIndex, RAPTOR, HyDE, Self-RAG, Corrective RAG, LoRA, QLoRA, PEFT, DPO, RLHF

Infrastructure: AWS SageMaker/S3/EC2/Lambda/EKS/Bedrock, GCP Vertex AI/GKE, Docker, Kubernetes, MLflow, Airflow, GitHub Actions, FastAPI, TorchServe, Streamlit

Data Systems: FAISS, Pinecone, ChromaDB, Qdrant, Weaviate, Milvus, Neo4j, Elasticsearch

Experience

AI Engineer, Zasti Inc – Ashburn, VA May 2024 – present

- Architected production RAPTOR RAG system with hierarchical summarization, Neo4j knowledge-graph integration, and dual-stage vector retrieval (FAISS + Pinecone): 45% faster queries, 38% better factual accuracy, 50M+ document embeddings at sub-100ms P99 latency.
- Designed multi-agent orchestration platform (LangGraph, LlamaIndex) with autonomous task planning, tool use, self-reflection loops, and short/long-term memory — automating 95% of clinical research workflows; manual review cycles cut from 3 days to under 4 hours.
- Engineered multimodal RAG pipeline (GPT-4V vision, custom OCR post-processors, FHIR-compliant parsers) with hallucination detection via chain-of-thought consistency scoring: 92% retrieval precision across 15+ heterogeneous lab-report formats.
- Cut LLM inference costs 75% (~$2K/month) via GPTQ 4-bit quantization, speculative decoding, dynamic batching with vLLM, and prompt-token caching strategies across agent tooling — 98% performance parity on clinical benchmarks (MIRAGE, MedQA).
- Built a shared agent harness (LangGraph through MCP) and four automation pipelines on it — sales, resource tracking, marketing, and internal software-engineering agents — adopted company-wide including leadership, absorbing work that would otherwise have required dedicated headcount in each vertical, with human approval gates before any consequential action.
- Diagnosed and fixed non-convergent agent loops by restructuring work as an explicit task-dependency map with single-task execution rather than whole-goal reasoning, and layered validation gates that caught hallucinated agent output before it reached a human reviewer.

Graduate Teaching Assistant, University of Cincinnati – Cincinnati, OH Aug 2022 – Apr 2024

- Designed and delivered hands-on workshops in Python, cloud computing, and ML for 500+ graduate students across a full year of continuous appointment with the same faculty member; 90% assignment completion, 95% positive feedback.
- Mentored 20+ graduate students on capstone research (NLP, CV, predictive analytics) using differentiated instruction — frequent check-ins and alternative explanations for students stuck on core concepts, peer-level method discussions with advanced students; 4 projects selected for departmental showcase, 2 led to published conference abstracts.

Deep Learning Developer, Technocolabs Softwares – Remote Feb 2021 – July 2022

- Led 4-member team building deep-learning models for a medical-diagnostics product (PyTorch CNNs, ResNet/ EfficientNet transfer learning): 88% diagnostic accuracy on held-out clinical test sets, 45% higher system efficiency.
- Designed pattern-recognition algorithm combining attention-based feature extraction with ensemble classification: +25% predictive accuracy over baseline for clinical decision support.

Selected Projects

Unwind — a reversibility layer for agentic tool use (Apache-2.0, sole author)

Python (90%), TypeScript (8%), MCP, Docker, uv, Typer, mkdocs, pytest
Available via PyPI (unwind-mcp), npm (unwind-mcp), ghcr.io container.

- Built a transparent MCP proxy that sits between any AI agent and any MCP server, infers which tool calls are reversible, maintains a durable cross-server undo log, synthesizes inverse operations, and interrupts the human only for genuinely irreversible actions.
- Contributed the R0–R4 reversibility taxonomy — nullipotent, self-reversible, compensable, mitigable-only, irreversible — treating reversibility as ordinal and environment-relative (the same write_file is R1 on a git-backed tree, R4 on a versionless one), with orthogonal dimensions for blast radius, externality, and a time-decaying reversibility half-life.
- Designed it to fail safe asymmetrically: misclassifying irreversible as reversible is catastrophic while the reverse is merely annoying, so unknown tools, failed classification, timeouts, and classifier crashes all escalate to a human. Never auto-allows under uncertainty.
- Shipped with full open-source governance: Apache-2.0, CODEOWNERS, GOVERNANCE.md, SECURITY.md, CONTRIBUTING.md, Code of Conduct, CITATION.cff, OpenSSF Scorecard, CI with coverage reporting, pre-commit hooks, devcontainer, Docker/compose, and a published mkdocs site.
- Building ReversiBench, a reversibility benchmark with a live sandbox; committed to publishing figures only with bootstrap confidence intervals from the live harness rather than quoting numbers early.

Autonomous Multi-Agent Research Platform with RLHF

Python, LangGraph, GPT-4o, PPO, FAISS, Neo4j, FastAPI, PostgreSQL

- Planner–Retriever–Reasoner–Validator agent loop with persistent memory and tool use across PubMed, ArXiv, Semantic Scholar APIs.
- PPO reward model on 12K researcher preference pairs — +41% synthesis quality, −68% hallucinated citations vs base GPT-4o on SciEval.
- GKE deployment, 5K+ concurrent sessions, sub-200ms median latency; LangSmith + Prometheus + Grafana observability.

EEG–fMRI Multimodal Fusion for Neural Decoding (M.S. Thesis)

Python, PyTorch, MNE, Nilearn, TCNs

- Fused 70-channel EEG with 3T fMRI (Wakeman–Henson multi-subject dataset) via temporal alignment and feature concatenation; preprocessing with SSS, ICA/wavelet-ICA, DWT (EEG) and motion/slice-timing correction with MNI normalization over fusiform and occipital ROIs (fMRI).
- Temporal convolutional networks with dilated causal convolutions classified famous/unfamiliar/scrambled faces at 84.8% within-subject and 81.1% cross-subject (LOSO) accuracy (ROC-AUC 0.93/0.90) — beating EEG-only (65.5%), fMRI-only (74.6%), and GRU/LSTM-CNN/SVM baselines.

Education

University of Cincinnati, MS in Computer Science — Aug 2022 – May 2024

- GPA: 3.95/4.0
- Thesis: A Multimodal Neuroimaging Method for the Prediction of Visual Stimuli (Advisor: Prof. Vikram Ravindra)

University of California San Diego, MicroMasters in Data Science — Apr 2021 – Jan 2022

SRM University, BTech in Computer Science — May 2018 – Apr 2022

- GPA: 3.98/4.0

Publications & Preprints

- Gurram, B. (2026). Valid Per-Field Selective Risk Control for Document Extraction: Three Failure Modes, a Validity Ladder, and When Conditioning Pays. arXiv:2608.14639 [cs.LG, cs.AI, cs.CL]. Code (Apache-2.0): github.com/bhaskargurram-ai/verifydoc
- Gurram, B. (2026). Auditing Automated Evaluation, Error Propagation, and Runtime Mitigation in Tool-Using Language Agents. arXiv:2604.16706 [cs.AI, cs.CL, cs.MA]. Code and data: github.com/bhaskargurram-ai/agenthallubench
- Gurram, B. (2026). VerifyDocBench: Measuring Per-Field Calibration, Selective Risk, and Grounding for Document Extraction — at Scale, Across Models and Languages. Under review at arXiv (companion benchmark paper to arXiv:2608.14639).