

Ishan Trivedi

Bengaluru, Karnataka | +91 9116770965 | ishtrivedi18@gmail.com

[Portfolio](#) | [GitHub](#) | [Hugging Face](#) | [X](#)

EDUCATION

Sir M. Visvesvaraya Institute of Technology, Bengaluru

2024 – Present

B.E. in Computer Science & Engineering, Visvesvaraya Technological University — CGPA: 8.5/10

- Relevant Coursework: Data Structures, Linear Algebra, Discrete Mathematics, Object-Oriented Programming

PROJECTS

Agent Canary | *Python, LangGraph, FastAPI, AWS Bedrock, React*

[GitHub](#)

- Built a 5-agent LangGraph pipeline (Strategist, Attacker, Evaluator, Defender, Reporter) on AWS Bedrock that autonomously attacks any HTTP-based AI agent, scores findings, and generates guardrail patches.
- Implemented a two-layer evaluation system — deterministic detectors plus an LLM judge — producing a 4-case consensus verdict per attempt, with a patch-and-retest loop (up to 3x) until findings are verified fixed.
- Designed 12 attack strategies mapped to the MITRE ATLAS/ASI taxonomy, exposed through a FastAPI backend, a SQLite audit trail with semantic deduplication, and a React dashboard with live streaming.

IncidentOps | *Python, FastAPI, PostgreSQL, pgvector, Azure*

[GitHub](#)

- Building an Azure-first FastAPI backend that ingests and indexes engineering evidence (PostgreSQL/pgvector) to power citation-backed, grounded answers rather than free-form generation.
- Built a deterministic collection pipeline (separate service) that discovers, filters, redacts, and normalizes source evidence before it reaches retrieval — reducing the risk of sensitive data leaking into answers.
- Added workflow run persistence, approval gates, eval endpoints, prompt-injection checks, and metrics for auditability.

LogSage | *Python, PyTorch, QLoRA, Hugging Face, AWS EC2*

[GitHub](#) | [HF](#)

- Fine-tuned a 4-bit quantized Qwen2.5-7B-Instruct adapter via QLoRA (r=16) for structured log triage, producing schema-validated JSON (issue, root cause, severity, fix, confidence) instead of raw text.
- Curated a 1,116-row instruction dataset (1,004 train / 112 eval) across low/medium/high severity incidents; trained 3 epochs on an AWS EC2 g5.2xlarge (A10G), reaching 0.789 best eval loss in ~34 minutes.
- Built a full observability pipeline (TensorBoard, JSONL metrics, training curves) and published the adapter to Hugging Face with a reproducible Colab inference notebook and schema-validated post-processing.

audioforge | *Python, PyTorch, Hugging Face, AWS EC2*

[GitHub](#) | [HF](#) | [Write-up](#)

- Trained and compared two architectures on FSD50K (36,796 training / 4,170 held-out clips, 200 multi-label sound categories): a from-scratch CNN and a LoRA-adapted Audio Spectrogram Transformer (AST).
- LoRA fine-tuning of only 0.52% of AST's 86.8M parameters beat the fully-trained CNN's mAP by 84% (0.302 → 0.557), converging in roughly 3x the wall-clock time rather than 200x.
- Owned AWS GPU infrastructure end-to-end — VRAM sizing, quota increases, spot-capacity fallback, cost-gated instance start/stop — and caught and fixed six production bugs, including a silently-broken metric function and a mismatched LoRA target-module name, via smoke tests before committing GPU hours to full runs.

TECHNICAL SKILLS

LLM / RAG / Agents: LangGraph multi-agent orchestration, hybrid retrieval (pgvector + full-text), prompt design, LLM-as-judge evaluation, JSON schema validation, tool-calling

AI / ML: PyTorch, Hugging Face Transformers, PEFT/LoRA/QLoRA fine-tuning, CNNs, supervised fine-tuning, model evaluation

Backend / Infra: FastAPI, OAuth 2.0, PostgreSQL, pgvector, Docker/Compose, AWS EC2 & Bedrock, Git/GitHub

Programming: Python, Java, TypeScript basics

Data: NumPy, pandas, scikit-learn, PostgreSQL, SQL, JSON/JSONL, Jupyter/Colab

ACHIEVEMENTS & CERTIFICATIONS

- Hugging Face LLM Course** — Transformers, Datasets, Tokenizers, Accelerate, Hub workflows, fine-tuning, model sharing
- DeepStation Community** — participant in AI systems-focused learning community