Vedaang Chopra

vedaangchopra@gatech.edu — +1 (404) 740-9905 — Atlanta, GA — LinkedIn — GitHub — Website

Agentic AI Engineer — Tool Use · Multi-Step Orchestration · Execution-Based Evaluation

Work Experience

Georgia Institute of Technology — CORE Robotics Lab @ Siemens Atlanta, GA

Graduate Research Assistant — Advisor: ***Dr. Matthew Gombolay*** *Jan 2026 – Present*

- Built an AI agent (LangGraph) that writes CAD code (the instructions that define a 3D part), runs it, compares the 3D shape with the correct one, and repairs its own mistakes in up to three tries. The language model is frozen; all gains come from the loop.
- Gave the agent evidence when a shape is wrong: which program step likely caused the error, and how the shapes differ. A separate checker found 60 of 60 injected faults on hand-built tests (not yet tested on real model errors).
- Measured it: shape-overlap (IoU; 1.0 is perfect) rose from 0.82 to 0.92 on 200 expert prompts and from 0.57 to 0.70 on 200 beginner prompts versus one-shot generation, and beat a self-debug baseline by about 0.05. Repair still uses the correct shape as guidance; a version without it is next. Extra feedback did not help on a 12-sample test (4 of 12 solved in every setup).

Fortinet Technologies Inc. (AIOps R&D) Bengaluru, India

Software Development Engineer I & II (ML/AI Track) *Feb 2021 – Jul 2025*

- Led an AI troubleshooting assistant from hackathon prototype (top-5 finalist, 2023) to production: the language model plans steps, calls tools on network data (structured function calling), checks its own guesses and reports a root cause. Mean time to resolve issues fell by ∼70% (self-reported).
- Rebuilt the pipeline that collects network-device logs (OpenSearch, async code in Go), raising its speed from under 50 to over 2,000 events per second (40×), and connected the data team's Python models to Go backend services.
- Built the anomaly-detection method behind patent PCT/IN2022/058026 (Filed); it cut manual troubleshooting by ∼75% (self-reported).

Education

Georgia Institute of Technology Atlanta, GA

M.S. in Computer Science *(Machine Learning),* ***GPA: 3.8/4.0*** *Aug 2024 – Dec 2026*

Maharaja Surajmal Institute of Technology New Delhi, India

B.Tech. in Information Technology, CGPA: 8.8/10.0 *Aug 2016 – Aug 2020*

Selected Projects & Publications

SHASTRA — Agent Trace Modeling & Planning (team research, GT) Jan 2026 – Present

- Built the tools that turn recorded agent runs into step-by-step graphs: each run is split into blocks, a language model labels what each block does, and an append-only label registry keeps names consistent across runs.
- Wrote the analysis code to study whether agents repeat the same sub-tasks. This is early-stage: the reuse measurement is not finished and retrieval is not built. Planner and search were written by teammates.

ATHENA — Idea-to-Storyboard Generation (team project, GT) 2025 – 2026

- Designed a 14-step LangGraph pipeline that turns an idea into a storyboard (scenes, shots, script, images). Each step hands a validated data record to the next, so failures are easy to trace; about 800 automated tests.
- Ran a 20-run stress test that exposed two bugs (scenes collapsing onto one beat, image failures hidden as "done"); the fix is committed and being re-verified.

CERBERUS & ARTEMIS — Vision-Language Research (GT) Aug 2025 – Present

- Fine-tuned a 7B Qwen-2.5 with LoRA to answer image questions (training loss 0.74 to 0.10; held-out test pending) and measured image-to-text matching (77.7% top-5 on PixMo validation). Built a leak-free routing benchmark; a learned router did not beat a per-task lookup (50.9% vs. 56.4%).

Publications & IP

- **IEEE ICAIA 2026** (accepted; presenting Nov 16, 2026) — "Integrating Machine Learning and Memory Forensics for Enhancing Cybersecurity in IoT-Enabled Energy Systems" (with Sonika Malik). **Patent (Filed)** PCT/IN2022/058026.

Technical Skills

Agents: LangGraph, LangChain, tool/function calling, agentic RAG, multi-step planning, structured outputs (Pydantic), LLM/VLM-as-a-judge

Evaluation: execution-based evaluation, benchmarking, paired bootstrap, IoU, Chamfer/Hausdorff, Recall@k

ML: PyTorch, Hugging Face Transformers, LoRA fine-tuning (PEFT), CLIP, ONNX Runtime, scikit-learn

Programming: Python, Go, SQL/PostgreSQL, OpenSearch, FastAPI, Docker, Linux, Git