

Tianyi Zhang

404-422-2842 | tzhang708@gatech.edu | LinkedIn | Google Scholar

EDUCATION

Georgia Institute of Technology
Master of Science in Computer Science

Atlanta, GA
Aug 2025 – Dec 2026

Emory University
Bachelor of Science in Quantitative Sciences

Atlanta, GA
Aug 2021 – May 2025

PROFESSIONAL EXPERIENCE

LinkedIn Corporation

Mountain View, CA
May 2026 – Present

AI/ML Engineer Intern | LLM for RecSys, LLM Post-Training

- Contributing to LinkedIn's **out-of-network retrieval** system for homepage-feed candidate generation, applying **LLM-based generative retrieval** to surface relevant content beyond a member's immediate network.
- Led a research direction on **adaptive and compressed member representation**—modeling profile signals, temporal engagement history, long-history compression, and member-regime differences in retrieval.
- Built a **multi-axis retrieval experimentation framework** studying tradeoffs among **Recall@K**, diversity, efficiency, and evaluation fidelity across cold-start, sparse-, stale-, and active-user regimes.
- Developed a reusable **out-of-network experimentation pipeline** for standardized training, retrieval evaluation, configuration reuse, execution tracking, and experiment logging, improving team velocity and reproducibility for LLM retrieval research.

Georgia Tech Efficient & Intelligent Computing Lab · NVIDIA Collaboration

Atlanta, GA

Student Researcher | LLM Evaluation, Agentic AI

Jul 2025 – Dec 2025

- Co-authored **AgentSuite (ICML 2026)**, an automated pipeline auditing validity flaws in LLM-agent benchmarks; matched expert judgments at **0.79–0.87 F1** across **6 benchmarks** and showed flaws affect **~23% of tasks**, reshuffling **63% of model rankings** on a 30-LLM leaderboard.
- Built the **taxonomy-guided LLM-judge** stage and **rule-based structural detectors** powering the pipeline, decomposing agent tasks into user/environment/ground-truth/evaluation components; generalized to unseen benchmarks with no retuning and surfaced 100+ previously undocumented issues.

Emory Graph Mining Lab

Atlanta, GA

Machine Learning Researcher | AI for Science, Graph Representation Learning

Dec 2023 – May 2025

- First-authored a **causal brain-connectivity** method (**AIME 2025**) that recovered **directional** dependencies invisible to correlation-based graphs, improving downstream diagnostic prediction across **~8K patients** via Granger-directed graphs.
- Co-authored a **state-based Transformer (IEEE EMBC 2025, oral)** that modeled temporal transitions across latent functional-connectivity states, outperforming static-connectivity baselines on clinical prediction.

PUBLICATIONS

H. Suh*, S. Lee*, B. Ji*, R. Khare, B. Khan, H. Kim, **T. Zhang**, et al. “AgentSuite: Toward More Reliable Agent Evaluation with a Component-Based Benchmark Auditing Pipeline” *International Conference on Machine Learning (ICML)* 2026.
T. Zhang, K. Han, J. Nie, et al. “Causal Brain Connectivity: Integrating Granger Directed Graphs in fMRI Analysis.” *International Conference on Artificial Intelligence in Medicine (AIME)* 2025.
J. Nie, K. Han, **T. Zhang**, et al. “Brain Network State Transformer: Leveraging State Functional Connectivity.” *IEEE Engineering in Medicine and Biology Society (EMBC)* 2025. (Oral)

PROJECTS

loopcanary: Agent-Loop Observability Library 🐞

Apr 2026 – Present

- Building an **open-source Python library for agent-loop observability**—it watches an LLM agent's execution and diagnoses when the loop goes off-rails (repeated tool calls, silent context compression, un-recalled memory writes, cost run-away, or a safety monitor firing), emitting a per-run timeline to inspect or stream live.
- Grew from a **Python/Pydantic v2** stress-test harness for LLM safety monitors—typed trajectory model, abstention-aware verdicts, an adversarial transformation suite, and a **MALT** adapter over METR's labeled agent transcripts.

coact: Governance Substrate for Two Coding Agents 🤖

Jul 2026 – Present

- Building a **governance substrate** that lets Claude Code and Codex work in the same repo safely—capability policies, advisory locks and a shared task board, worktree isolation with merge gates, and an append-only journal for replayable oversight—organized around **security, controllability, and cost**. A single dependency-free **Go** binary.

TECHNICAL SKILLS

Programming: Python, Go, SQL, Java, R, C/C++ **ML/AI:** PyTorch, CUDA, Hugging Face, vLLM, SGLang, LangChain, MLflow **Infra:** Docker, AWS, GitHub Actions, FastAPI, Linux **Data:** PostgreSQL, MongoDB, Redis, Apache Spark