Tyler Vergho

Cupertino, CA • (408) 499-0354 • tvergho@gmail.com • linkedin.com/in/tyler-vergho • tvergho.me • github.com/tvergho

2 years of professional software engineering experience. A highly motivated self-starter. Pursuing a M.S. to focus on generative AI.

WORK EXPERIENCE

CivicBell, Software Engineer *Mountain View, CA* November 2022 – Present

Developing features for a civic engagement platform built using React/React Native, Express, Node, and Postgres, deployed on AWS.

- Lead a team of 4 developers. Coordinated major UI revamps and redesigned APIs for user registration, mobile push notifications, and local news, increasing customer conversion by ~30%. Trained and onboarded one new developer to the team.
- Identified and optimized SQL query inefficiencies that reduced initial load time by ~80%.
- Spearheaded backend integration testing using Jest to increase coverage on critical user-facing features.

Amazon Web Services, Software Engineering Intern *Seattle, WA* June 2022 – September 2022

Intern on the AWS DynamoDB Streams team.

- Developed a system in Python to automate load balancer management for network application traffic. This reduced cumulative team developer time expenditure by ~5/days each month.
- Collaborated with senior engineers to deploy changes to production infrastructure handling >10 million requests/second.

Canopy, Software Engineering Intern *San Francisco, CA* April 2022 – June 2022

Worked alongside senior engineers to build and ship features for a Series A startup developing a SPV investment management platform.

- Revamped integration with Plaid and Modern Treasury APIs to simplify user experience on >$1 million in payment flows/month.
- Engineered a new frontend customer onboarding form in collaboration with the legal team for the launch of a VC funds product.
- Streamlined package dependencies to cut bundle size and reduce First Contentful Paint load time by ~30%.

Dartmouth DALI Lab, Software Engineer *Hanover, NH* August 2020 – June 2023

Collaborated in teams of developers and designers to develop full-stack applications within an on-campus development lab at Dartmouth.

- Oversaw growth of the lab from 25 to 40+ developers. I facilitated weekly workshops and meetings, organized onboarding and mentorship programs, and expanded documentation and technical resources as software engineering lead (one-year term).
- Developed a Ruby on Rails course management platform over 9 months. Launched to 4,000+ students in fall 2022.
- Architected an anti-phishing contest platform (GoPhish) managing 1,800+ participants and $10,000 in prizes for Dartmouth. Mentored and led development of a full-stack React Native live game show application integrating Firebase, PayPal, and AWS.

EDUCATION

Dartmouth College, Hanover, NH

Master of Science – Computer Science (AI concentration) 2024 (expected)

Bachelor of Arts – Computer Science 2023, GPA: 3.83 (cum laude)

Relevant Coursework: Full-Stack Web Development, Smartphone Programming, Discrete Mathematics, Software Design & Implementation, Algorithms, Computer Architecture, Machine Learning & Statistical Analysis, Artificial Intelligence, Music & AI

RESEARCH & PROJECTS

Ongoing

Personality Detection in Recommendation Models: Integration of personality models into neural recommendation systems, with the Minds, Machines, and Society Group at Dartmouth.

GPT-4 and Misinformation Assessment: Assessing GPT-4 and other LLMs' capacity to detect misinformation (collaboration with McGill).

AI Safety and Alignment Whitepaper: Crafting a report on AI alignment strategies, both technical & policy, with Berkeley's SPAR.

Art to Music: Leading a human-computer interaction project translating user drawings into music using generative music models.

Completed

Sentence Transformers Rust: A port of the Sentence Transformers library for semantic text embeddings, developed in Rust using the Burn machine learning framework. Code available here.

Logos: Search engine for intercollegiate policy debate evidence. Built using React and vector-based semantic search. Code available here.

Deepfake Detection: Evaluating CNNs' capacities to serve as universal deepfake detectors. Results available here.

AI Evidence Production: Automating debate evidence production with GPT-4. Built with Python and OpenAI APIs. Code available here.

Generative AI Audio: Experimenting with audio generation using deep neural networks (diffusion models and GANs). Code available here.

SKILLS & INTERESTS

Languages: Python, TypeScript/JavaScript, HTML/CSS, SQL, Java, Rust

Frameworks: React/React Native, Express, Node.js, Flask, Postgres, Ruby on Rails, Jest, PyTorch, Tensorflow, GGML

Tools: AWS, Google Cloud, Docker, PostgreSQL, MongoDB, Git, CUDA

Awards: Winner of the 2021 & 2022 National Debate Tournaments and Rex Copeland Award for intercollegiate debate