

Steve Li

steveli@college.harvard.edu

github.com/lithafnium

<http://steve-li.com>

978-735-3386

Education:

Harvard University

Cambridge MA

Relevant Coursework: Systems Programming and Machine Organization, Data Structures and Algorithms, Machine Learning, Probability, Linear Algebra and Differential Equations, Theory of Computation, Operating Systems, Abstraction and Design in Computation, Statistical Inference

05/2024

Technical Skills:

Java, C/C++ , Javascript (React.js, React-Native, Vue.js, Node.js, Electron.js, Jest, Typescript), PyTorch, Python, Bash, Flask, MySQL, PostgreSQL, Firebase, GraphQL, GCP, CI/CD, AWS (DynamoDB, S3, Lambda, EC2, ECS, EBS), Docker

Relevant Experience:

The Berkeley NLP Group | Research Intern

Remote, Jun 2022 - present

Instagram (Meta) | Software Engineer Intern

New York NY, May 2022 - Aug 2022

- Developed internal **logging infrastructure** using **C++** for performance metrics, logging filtering, and crash reports, providing support for multiple Instagram entities (accounts, comments, etc.) on **~100k** requests per day.
- Designed and built internal tools using **React** and **PHP** for testing Instagram filtering/downranking results from ML-generated classifications, reducing testing from **hours** to **minutes** and preventing faulty configurations from being pushed to production.
- Created soft action enforcement infrastructure to support filtering/downranking for multiple different types of Instagram entities using **C++**, operating on millions of requests per day.

Gamalon | Machine Learning Engineer Intern

Cambridge MA, Sept 2021 - Dec 2021

- Developed novel **neural network architecture** for **NLP topic hierarchies** based on probabilistic factor graphs using **PyTorch**,
- Designed a suite of **CLI** tools using **Bash**, **Python**, and **GPT-3** for model building and question/answer generation, reducing manual business analyst labor from **~4 weeks** to **~4 days**
- Created question answering and response interface with **Retrieval-Augmented Generation**-inspired algorithms based around google search results through Bayesian decision trees. Competitive with **GPT-3** question/answer generation

Amazon | Software Development Engineer Intern

Seattle WA, June 2021 - Aug 2021

- Developed public **REST API** for resource tagging for Amazon Connect, AWS's cloud call center, designed to be shipped to **hundreds of thousands** of customers by September.
- Spearheaded redesign of resource cleanup **lambda** workflows and architecture, reducing user throttling rates by over **70%**
- Implemented access control support using tagging through IAM roles, allowing for resource allocation and user restriction.
- Debugged **production** code and wrote **unit/integration tests** required for **CI/CD pipelines** to maintain AWS cloud resources.

Fractal | Software Engineering Intern

Cambridge MA, Jan 2021 - May 2021

- Created testing infrastructure with **~10x** fewer bugs to ensure quality software using **Jest**, **React-Testing-Library**, and **Enzyme**, complete with **Github Actions workflows**.
- Implemented AWS resource tracking with **logz.io docker** integrations and slack notifications through **Github Actions workflows** to monitor **EC2 instances** and **ECS clusters**, saving **thousands of dollars** in sunk costs.
- Developed **Electron.js app** and **Flask Python webserver** functionality with **e2e** and documentation.

Software Projects:

Wikipedia Topic Modeling (Top2Vec, Typescript, Flask)

Designed a topic visualizer for Wikipedia articles using **Top2Vec** and **transformer-based NLP models** trained on multilingual data. Created **flask** backend and **typescript graph editor** tool.

ChickadeeOS (C++)

A multicore x86-64 operating system that supports **syscalls**, **multithreading**, and virtual memory. Includes **caching**, synchronization objects such as **spinlocks** and **futexes**, a virtual file system, and an on-disk file system with directory trees. Created for CS161: Operating Systems

Spotify Curator (Python, Flask, Pandas, Scikit-learn, React.js)

Devised a song preference predictor by analyzing playlists and liked songs on Spotify. Utilized a random forest classifier to examine empirical song data from a user to predict song preferences.

Datamatch (C++, Python), <https://datamatch.me/>

Designed sentiment analysis scoring functions using **Sentence-BERT**. Refactored Gale-Shapley-inspired matchmaking algorithm to account for question similarities in answer distributions, creating matches for over 50,000 people across 30+ universities.

Clubs and Activities:

Harvard Radcliffe Orchestra | Oboist

Sept 2019 - Present

Wave Learning Festival | Technical Co-Director

California, April 2020 -

Built frontend **React.js** app with **e2e** testing used by over **10000** students **worldwide** in more than **60** countries. Redesigned technical workflow by creating **CI/CD** infrastructure on **AWS Amplify** servers with **DynamoDB** Backend, **S3** storage, **Lambda** function deployments, and **GraphQL**.

May 2021