Jeremy Hsu

jeremyhsu.me | jeremyhsu@college.harvard.edu | github.com/hsujeremy

Education

Harvard University | B.A. in Computer Science — May '24

Selected Coursework Data Structures and Algorithms · Operating Systems · Database Systems · Computational Complexity Probability · Functional Programming · Design of Useful Interactive Systems · Linear Algebra and Differential Equations

Experience

Facebook | Software Engineer Intern — Menlo Park, CA · Starting May '22

Lime | Software Engineer Intern – Partnerships Platform — San Francisco, CA · May '21 – Aug '21

- Developed a server-side API to enable seamless MaaS integrations between Lime vehicles and public transit providers which launched in August 2021 starting with riders in Berlin.
- Architected and implemented the end-to-end refund flow along with the lifecycle APIs for users and for fetching real-time vehicle data through QR code scanning or license plate entry.
- Extended the functionality of a company-wide internal administrative portal by creating UI components that enable assigning permission access-control roles and viewing mobility data feed instructions for regions and external partners.

Cisco | Software Engineer Intern – Webex Media Engine — San Jose, CA · May '20 – May '21

- Overhauled and extended the team's suite of internal debugging and triage tools by writing new and updating existing automation, analysis, and visualization scripts, deployed using custom-built Jenkins jobs and CLI commands, saving engineers multiple hours per day.
- Collaborated with overseas teams to develop and deploy a data aggregation module leveraging the Elastic API that resulted in Kibana dashboards loading up to 5x faster.
- Created and monitored interactive dashboards visualizing critical media quality and network performance metrics, enabling engineers to identify problematic trends and issues more rapidly.

Activities

Datamatch | Algorithm Lead — Cambridge, MA · Sep '19 – Present

- Leading a team of 25+ student developers to improve Datamatch's core matching and scoring algorithms.
- Wrote the mutual and pseudorandom matching algorithms for Datamatch's matching service for graduating seniors.
- Improved Datamatch's user-to-user compatibility score function by implementing semantically-nuanced sentence embeddings with Sentence-BERT and by inversely weighting questions to account for distribution polarity in survey responses.
- February 2021 launch matched 42,000+ students across 34+ universities, including Harvard, MIT, and Berkeley.

Projects

ShuttleDB [C]

A column-store database that can efficiently handle columns with up to 10M tuples with support for selecting, fetching, joining, loading, column arithmetic, and multiple aggregation types. ShuttleDB contains common modules such as a parser, optimizer, and persistence on disk, as well as advanced optimizations to minimize IO costs such as primary and secondary indexing with B-trees, custom TCP streaming protocols for large payloads, zone maps, hash joins, and multithreaded shared scans.

YouTube Party [Node.js, React, Socket.io, YouTube iFrame API]

A full-stack web app that allows any number of users to stream YouTube videos synchronously using web sockets. The site synchronizes pause and play operations, keeps track of the number of online users, and includes an in-app chatroom.

ML-Enabled Spotify Curator [Flask, React, Celery, Redis, Scikit-Learn, Spotify Web API]

A full-stack web app that predicts whether a user will like a particular song based on musical characteristics of songs in their Liked Songs library. Song characteristics are fetched from the Spotify Web API while training and prediction tasks are handled asynchronously on the backend using Celery as a distributed task queue and Redis as both a transport store and message broker.

Roommate Hub [Swift, UIKit, MessageUI, Firebase]

An iOS app organizing the shared living experience for roommates. Features include secure user authentication, shared task lists, interactive roommate profiles, anonymous message forums, and embedded iMessaging.

MiniML [OCaml]

An interpreter for a subset of the OCaml functional programming language. MiniML supports unary operators, binary operators, conditional branches, recursive functions, higher-order functions, and exception handling.

Technical Skills

Programming Languages C · C++ · Java · Python · JavaScript · Swift

Frameworks & Technologies React · Ruby on Rails · Node.js · SwiftUI · UIKit · Firebase · Elasticsearch · Figma