

# Tejas Narayanan

tejasn100@gmail.com | (408) 913-0595 | [linkedin.com/in/tejas-n](https://www.linkedin.com/in/tejas-n)  
[tejasnarayanan.com](https://tejasnarayanan.com) | [github.com/tnarayanan](https://github.com/tnarayanan)

## EDUCATION

---

### Stanford University

September 2020 - June 2025

BS/MS, Computer Science; Minor, Psychology; GPA: 3.99

- **Relevant Coursework:** Parallel Computing, Algorithms, Compilers, Computer Systems in C, Digital System Architecture, Databases, Deep Learning for Computer Vision, Operating Systems, Computer Networks & Security, Linear Algebra & Multivariable Calculus

## TECHNICAL SKILLS

---

**Languages:** C++, Python, PyTorch, Keras, TensorFlow, SQL, JavaScript, Node.js, Electron, HTML, CSS, Java, Android

**Areas:** Machine Learning, Deep Learning, Algorithm Design, Data Structures, Databases, App Development, Web Applications

**Tools:** Git, Unix, Terminal, LaTeX, Kafka, Cassandra, Docker, GSuite, Microsoft Office

## EXPERIENCES

---

### Course Assistant

April 2024 - present

Stanford University Computer Science Department

- Course assistant for CS 143: Compilers. Holding office hours, grading assignments, and designing assignments for 130+ students

### Software Engineering Intern

June 2023 - August 2023

Citadel

- Developed an end-to-end Python/C++ service to read raw trading position data, enrich the data, and publish it to a new trading view
- Parallelized column enrichment and data queries using batch Kafka consumption and multithreading to achieve 8x processing speedup
- Designed a generalizable system for defining and parsing custom column filters that can be used across all trading views
- Worked with traders to design the view and columns for their use cases. Deployed new service and view to production

### Software Engineering Intern

June 2022 - August 2022

Bloomberg

- Optimized the performance of a Kafka pipe that received real-time trade data. Prevented severe lag spikes when writing to Cassandra database during high trade volume
- Created new tools for Kafka pipe analysis in Python. Modified multiple large existing systems written in C++
- Achieved speed improvement of 2,200x under maximum load. Deployed changes to internal production

### Section Leader (TA)

January 2022 - December 2022

Stanford University Computer Science Department

- Section leader for introductory computer science classes: CS 106A (Python) and CS 106B (C++)
- Taught lessons to 10-student sections, graded assignments, and helped students debug assignments

### Research Engineering Intern (AI/Machine Learning)

June 2021 - September 2021

Ford Motor Company

- Implemented a reinforcement learning-based adaptive data selection pipeline for computer vision (CV) tasks
- Trained a Neural Data Filter (NDF) using OpenAI Gym, PyTorch, and Python to learn which data to keep at various stages of task model training, then used the NDF to dynamically filter training data during task model training
- Ran models on production datasets using GPU clusters. Improved task model accuracy by 27%

### Machine Learning Lab Director

September 2020 - June 2023

Stanford ACM

- Organized meetings, revamped presentations & onboarding project, and created training videos
- Mentored multiple teams participating in the project. Advised them on designing machine learning models and using PyTorch
- Built a convolutional neural network (CNN) using Python and PyTorch to predict average income based on satellite imagery

## PROJECTS

---

### Ghostwriter: Dynamic Programming and Deep Learning for Lyric Generation

Python, PyTorch

- Created deep learning language models to generate lyrics for inputted music data. Developed hybrid models that combined dynamic programming algorithms with language model outputs to constrain lyric generation to align with the beat of the input music
- Winning Stanford CS 224N project (Best Project, Best Poster). Link to paper: [tejasnarayanan.com/files/Ghostwriter\\_Paper.pdf](https://tejasnarayanan.com/files/Ghostwriter_Paper.pdf)

### Sixteen Pixels is (Almost) All You Need: Crafting Parameterized Image Uncrumpling Models

Python, PyTorch

- Created an algorithm that takes in a crumpled image and outputs its uncrumpled form. Utilized generative computer vision algorithms
- Procedurally generated crumpled images using Python and Blender (a 3D rendering engine) to efficiently procure large training dataset
- Winning Stanford CS 231N project (Best Poster). Link to paper: [tejasnarayanan.com/files/SixteenPixels\\_Paper.pdf](https://tejasnarayanan.com/files/SixteenPixels_Paper.pdf)

## CERTIFICATIONS

---

Deep Learning Specialization, deeplearning.ai (Coursera, 2020)    Algorithms Specialization, Stanford University (Coursera, 2017)