

EDUCATION

- > **Northeastern University** | Khoury College of Computer Sciences | Honors Boston, MA
Candidate, B.S. Computer Science & Mathematics \ \ Physics & Chemistry Minor \ \ 3.87 GPA 2021 – May 2025

RESEARCH AND WORK | * := Part-Time, † := Under Non-Disclosure Agreement

- > **Computational Protein Design Co-op @ Tessera Therapeutics**[†]: Gene Therapy Jan. 2024 – June 2024
 ◦ Analyzed biological data for protein engineering of genetic medicines.
 ◦ Developed tool to visualize and analyze protein structure and mutational data using Dash and plotly.
- > **Software Engineering Co-op @ Nth Cycle**: Metal Refining Startup Jan. 2023 – June 2023
 ◦ Identified problems with data capture and analysis such as manually transcribing pdfs, siloed data analysis procedures, nonstandardized datasheets, and exclusively using Excel/paper. Designed and implemented a webapp (see LUCAS project) which used drag-and-drop libraries and MongoDB for a no-code solution.
 ◦ Fixed problems with internet, printers, laptops, AzureVPN and Remote Desktop in lieu of an IT department.
- > **Peer Tutoring***: Discrete, Fundies 1 & 2, OOD, Algo, Phys 2, Calc 3, LinAlg, Prob & Stats Sep. 2022 – June 2023
 ◦ Demonstrate patience and professionalism with tutees having a variety of skill levels. 164 hours total ≈ 4 hrs/wk
- > **Sijia Dong Lab***: Computational chemistry research group Dec. 2021 – Jan. 2023
 ◦ Used automation scripts and ML to investigate ligand-protein binding.
- > **Northeastern Undergrad Computer Architecture Research Group***: Nov. 2021 – May 2022
 ◦ Learned C and beginner CUDA through programming challenges given by Prof. Kaeli.
- > **Internship with Town of Holliston Director of Technology***: Nov. 2020 – Feb. 2021, Apr. 2021 – May 2021
 ◦ Setup and resolve problems with phones, printers, desktops, software, and file storage for municipal departments.

COURSES AND TECH STACK | * := In progress/learning, † := Included lab portion

- > **Tech**: Typescript, React, Python, MongoDB, Java, Lean4, Bash, Linux, C, Mathematica, Rust*
- > **CS**: AI*, Software Development*, Networks & Distributed Systems, Object Oriented Design, Logic & Computation
- > **Math**: Statistics & Stochastic Processes, Linear Algebra, Number Theory, Group Theory, Calculus 3, Diff. Eq.
- > **Chemistry**: Organic Chemistry 1[†] & 2[†], Physical Chemistry[†], Analytical Chemistry*[†], Quantum Chemistry*[†]
- > **Physics**: Quantum Computation and Information, Electronics*[†], Modern Physics

PROJECTS

- > **LUCAS** | ReactTS, MongoDB, Azure, ⚡RJSF, ⚡Blockly 📄/lucas-demo-screenshots Feb. 2023 – Jun. 2023
 Webapp which enabled drag-and-drop design of forms and data analysis pipelines and graphics. Enabled parsing of files output from a specific machine to upload multiple documents with same metadata, dimensional analysis in pipelines, linking a sample to the other steps in its history, which could have both branching and merging, and the capability to make graphs and tables, all without (them) coding. *Access to live demo available on request. Written for Nth Cycle.*
- > **ML Excitation Model** | Python, Maestro, MATLAB, Bash, TensorFlow Aug. 2022 – Jan. 2023
 Identify what aspects of chloramide, flavins, and styrene influence their excitation wavelengths when they are in proximity to each other. Have written a MATLAB/Python/bash script that transforms the raw coordinates of each atom into more usable and interpretable features, which then can be used as input for a neural network or rounded for use in a neural network to predict excitation wavelengths and strengths. *Completed as part of Sijia Dong Lab.*
- > **Automated Docker** | Python, Schrodinger, Slurm, QtPy 📄/automated_docking_script July 2022
 Designed a GUI add-on and backend script for Schrodinger Maestro that automated a time-consuming and error-prone process of generating docking poses for three molecules in all permutations. *Completed as part of Sijia Dong Lab.*
- > **N-Bullets in Racket** | Intermediate Student Language (Racket subset) 📄/nbulletsrkt Apr. 2022
 Rewrote a project for a Java-based course in Racket to compare the two languages. Racket was terser, easier to properly test, and (subjectively) more readable. Additional findings and opinions are in the README.
- > **Grave Finder** | ReactJS, PHP, MySQL, Cloudflare, cPanel, SSL 📄/findagravemiddleborough.ml Dec. 2020
 Website to view grave data for a local cemetery preservation nonprofit. Makes use of join tables, searching, filtering, React+MaterialUI forms, and an administrator login with a cooldown period after too many login attempts.