CAMERON RYAN

GitHub

(+1) 585-794-6360 ◇ ryan.came@northeastern.edu

Personal Website: `perceptrons.org`

EDUCATION

Northeastern University *September 2023 - June 2027*
Candidate for Bachelors in Mathematics and Physics
GPA: 3.89, deans list

EXPERIENCE

Undergraduate Research Program University of Rochester Laboratory for Laser Energetics *Summer 2023*
I worked with other software engineers at the laboratory to develop an authentication system for the laboratory's control system.

High School Research Program University of Rochester Laboratory for Laser Energetics *Summer 2022*
I was an intern at the Laboratory for Laser Energetics at the University of Rochester. My work in involved individual research on the development of a containerized application management tool for scientific computing

Molecular Dynamics Simulation Analysis Software Development Volunteer Bioinformatics Laboratory, RIT *2022*
I worked with Dr. Gregory Babbitt in the RIT Bioinformatics department to recreate parts of a software called `cpptraj` that is used for statistical analysis of molecular dynamics simulations, to be used in a web demonstration of his protein analysis software.

INVOLVEMENT

Quant Team Member Disrupt (Northeastern Fintech club) *Spring 2025 - present*
I am currently working on developing market analysis, and trading algorithms with the Quant team.

PROJECTS

Neural Nets experiments in Jax *ongoing*
I performed several experiments training and analyzing neural networks using python `jax` library. Information about this is available on my website, and GitHub.

Vector Search Engine *Summer 2024*
My search engine uses embeddings from neural networks for images and text to search content on Wikipedia. The project involves many parts, including a multi threaded web crawler, a large scale vector database, and a web app.

Naturify *Spring 2023*
Naturify is a phone app that uses artificial intelligence to classify images of plants, animals, and fungi. Naturify utilizes two neural networks that I trained to make its predictions. Both of the neural networks were implemented in the python `torch` library. The user interface for Naturify was implemented in Swift, using both the SwiftUI and UIKit libraries.

1. **Image classification neural network** A 92 million parameter visual transformer was trained on a Nvidia A100 GPU, to map images from the iNaturalist dataset to a label set, containing 10,000 labels. The model has a 0.69 accuracy on the validation dataset.

2. **Location metadata neural network** A neural network was trained to map latitude and longitude coordinates to the label space of the iNaturalist data set. This approximates the probability density of a species being observed at a given coordinate, so the user can identify the most likely species to be in their particular area.

Personal Website *ongoing*
My website is a project that I have been working on

Image Garden *Spring 2023*
Image Garden is a phone app that utilizes neural networks to allow users to search text and images from their photo library. Neural networks map image vectors, and text queries to a high dimensional feature embedding, in which embeddings images or text sequences that are semantically similar will have lower euclidean distance. Note that I did not train the neural networks.

CameronOS *ongoing*
CameronOS is an unfinished project that I periodically work on. So far, I have created a basic kernel using C and assembly.

DNA file encoder *Spring 2023*
DNA file encoder is a simple C program that I wrote to encode the binary data from files into strings representing DNA sequences. In the future, if it becomes possible to synthesize large, specific DNA sequences from scratch, this could be used to encode and decode data, which could be stored in living cells.

Floral *Spring 2023*
Floral is a python neural network library that I implemented using the automatic differentiation features of Jax.

Earth Explore *Summer 2022*
Earth Explore is a phone app that uses artificial intelligence to identify bird songs. The neural network was trained by the Cornell Ornithology Lab.

Sign Translator *Fall 2021*
Sign Translator is a phone app that uses artificial intelligence to identify letter signs from the ASL alphabet. The neural network was trained on synthetic data, using Apple's CoreML framework.

WordLab *Summer 2021*
WordLab is a note taking app that uses artificial intelligence to semantically interpret people's notes, and answer questions from their notes. For example, a user can paste an excerpt from a textbook into the app, and then ask a question about the content. The AI can find the part of the text that contains the answer to their question. WordLab uses a pre trained BERT model.

Skin Lesion Identifier *Summer 2021*
I trained a convolutional neural network to classify images of skin lesions using the python `tensorflow` library. I then created an iOS app with Swift that used the neural network to identify abnormal skin growths. This app has not been published in the App Store because I was unable to obtain FDA approval. However, I have been contacted by physicians with interest in developing my work.

SKILLS

Machine learning frameworks
I am experienced with training neural networks using these machine learning frameworks:

1. **PyTorch** I am experienced with PyTorch through using it to train neural networks that I used in some of the phone apps that I created.

2. **Tensorflow** I used Tensorflow to train neural networks that I used in some of the phone apps that I created. Tensorflow was the first machine learning framework I used.
3. **Jax** I have done several informal projects using Jax, including coding a python neural network library, and independently experimenting with creating a new learning algorithm for neural networks.

Programming Language Experience
This is a list of all the programming languages that I have used, as well as a brief summary of my experience with them. It is approximately ordered by how experienced I am with each one.

1. **Python** Most of my experience involves using Jupyter notebooks, however I am also familiar with using Python in the command line, and in writing scripts.
2. **Swift** I have gained a large amount of experience with Swift from my experience coding phone apps.
3. **Rust** I used Rust to write the crawler for my image search engine. I have also used it for some other small in-progess projects that can be found on my GitHub.
4. **C** I have written code in C code for some phone apps, and I worked on several small projects with C, such as creating a shell, or making an Ising model simulation.
5. **C++** I have written code in C++ for some of my phone apps.
6. **Objective C** Objective C is an older programming language used primarily for developing iOS applications. I have written code in Objective C for some of my phone apps.
7. **Java** I learned to write code in Java for my AP Computer Science A class in high school. I got a 5 on the AP Computer Science A exam.
8. **JavaScript** JavaScript is used for coding web applications. I gained JavaScript experience initially from my work with Dr. Babbitt. My internship with Richard Kidder involved a large amount of coding in JavaScript. I also used JavaScript to code my personal website.
9. **HTML** HTML is used to render web pages. Much of my work under Richard Kidder involved using HTML.
10. **Perl** I am experienced with Perl from creating an authentication library in Perl during the summer of 2023 for Richard Kidder.
11. **PHP** I am experienced with PHP from creating an authentication library in PHP during the summer of 2023 for Richard Kidder.
12. **CSS** CSS is used for adding styles to web pages. From my work with Richard Kidder, I gained brief experience with CSS.

RESEARCH

[1] Gregory A. Babbitt, Madhusudan Rajendran, Miranda L. Lynch, Richmond Asare-Bediako, Leora T. Mouli, **Cameron J. Ryan**, Harsh Srivastava, Patrick Rynkiewicz, Kavya Phadke, Makayla L. Reed, Nadia Moore, Maureen C. Ferran, and Ernest P. Fokoue. “ATOMDANCE: Kernel-based denoising and choreographic analysis for protein dynamic comparison”. In: *Biophysical Journal* (Mar. 2024). issn: 0006-3495. doi: 10.1016/j.bpj.2024.03.024. url: http://dx.doi.org/10.1016/j.bpj.2024.03.024.

[2] **Cameron Ryan**. “CONTAINERIZED APPLICATION MANAGEMENT FOR CLOUD BASED SCIENTIFIC ANALYSIS”. In: (2023). url: https://www.lle.rochester.edu/media/publications/high_school_reports/documents/hs_reports/2022/Ryan_Cameron.pdf.