Henry Steinitz Resumé

Contact

Phone
(631) 759-6093

Address
279 E. 10th Street
Apartment 3
New York, NY 10009

Email
hjsteinitz@gmail.com

Websites
henrysteinitz.com
github.com/henrysteinitz

Coursework

Machine Learning
Computer Systems
Complexity Theory
Algorithms
Statistics
Quantum Computing
Computability Theory
Real & Complex Analysis
Abstract Algebra
Point-set Topology
Algebraic Topology
Differential Geometry
Mathematical Logic
General Relativity
Neuroscience

Skills

C / C++
Python
Java
Javascript
Haskell
TensorFlow
Gensim
React
Redux / Alt
Node.js
Django
AWS
Ansible
Elasticsearch
Matlab

Work Experience

Software Engineer, Google — July 2018 - Present

- I work on Cloud Load Balancing infrastructure.

Software Engineer, Factr — December 2016 - July 2018

- I was one of two full-time engineers developing the system which involves working across a frontend (React), backend (Django, Node.js, Elasticsearch), and machine learning (Gensim) codebase.

Private Tutor — September 2015 - June 2018

- I teach math, physics, and computer science to college and advanced high school students.
- Earned an average 5 star rating on the tutoring platform WyzAnt based on 31 ratings. (wyzant.com/match/tutor/86214023)

Fundraiser, Oxfam — February 2016 - March 2016

- Canvassed in New York City to raise money for Syrian refugees.

Education

NYU Courant — M. S. in Mathematics (2018 -)
University of Chicago — B. S. in Mathematics with Honors (2011 - 2015)

- First-generation college student. Received the Odyssey scholarship for students from lower-income families.
- Dean's List all four years.
- Wrote a survey of results in algorithmic randomness based on *Algorithmic Randomness and Complexity* by Downey and Hirschfeldt. (http://www.henrysteinitz.com/steinitz-2013.pdf)

Programming Projects

Cajal — github.com/henrysteinitz/cajal

- Computational graph library written in Python with NumPy.
- Inspired by similar libraries like TensorFlow and Theano, current features include a simple implementation of general backpropagation, gradient descent, and L2 regularization.

Neural Turing Machine — github.com/henrysteinitz/neural-turing-machines

- Implementation of Graves et al. 2014 in TensorFlow.

Multilayer Perceptron Visualizer — henrysteinitz.com/neura

- A simple javascript application that randomly generates data based on a provided boolean relation, which is then used to train and visualize an MLP.
- The magnitude of the strength of each weight determines the thickness of the edges in the architectural graph. The user can then provide test inputs and watch information flow through the network. Training networks multiple hidden layers nicely visualizes the problem of exploding / vanishing gradients.