SCOTT WERWATH

1930 Channing Way, Apt. 3E ◇ Berkeley, California 94704
(804) 380-1188 ◇ sw@swerwath.com ◇ swerwath.github.io

TECHNICAL SKILLS

Languages	Python, Javascript, C, C++, Java, SQL
Frameworks	Node, .NET, Rails, MPI, Redis, RabbitMQ
Misc.	Git, NLP, ML, WebSockets, Relational Databases

EDUCATION

University of California, Berkeley September 2015—Expected May 2019
B.S. Electrical Engineering & Computer Sciences *GPA (major): 3.9, GPA (overall): 3.6*
Minor in Linguistics

EXPERIENCE

Google January 2017—Present
Software Engineering Intern *Mountain View, Calif.*

Developing novel Machine Learning techniques to disambiguate entity mentions based on linguistic context
Writing large-scale data processing pipelines for example generation, model training, and model evaluation

CITRIS Foundry September 2016—December 2016
Engineering Fellow *Berkeley, Calif.*

Prototyped embedded systems for Numericcal, a DSP and controls startup in the Foundry accelerator
Implemented and optimized linear algebra algorithms for high-performance embedded control systems

SolarCity (division of Tesla Motors) June 2016—August 2016
Software Engineering Intern *San Francisco, Calif.*

Designed and built Node.js WebSocket microservice to enable real time interaction and data streaming between customers and sales representatives
Refactored .NET routes and database schemas, reducing average customer-facing API response time by 75%

RESEARCH

UC Berkeley, Energy & Resources Group September 2016—December 2016

Developing integrated assessment modeling library for use by the White House, EPA, and other federal bodies to estimate the economic and environmental effects of policy decisions
Augmenting Julia library to allow users to run Monte Carlo simulations across computing clusters

UC Berkeley, Computer Sciences Division January 2016—June 2016

Identified main challenges in performing game analysis with distributed computing systems
Developed novel algorithm for solving arbitrary abstract strategy games on distributed systems with MPI
Deployed algorithm to the Savio High Performance Computing Cluster for testing and analysis