

BS Mathematics at University of Washington '2021

<https://sm5art.github.io> | arturk@uw.edu | <https://github.com/sm5art> | Seattle/SF Bay Area

Summary

Self-taught programmer and currently a mathematics student/researcher at UW. I've had previous engineering internship experience in software engineering and in quantitative research. Whenever I'm not playing with algorithms, I like to rock climb and ski (season permitting).

Experience

Quantitative Finance Research Intern

Nipun Capital

Jun 2017 – Sept 2017

Foster City, California

At Nipun Capital, I was an independent research intern that got assigned research ideas throughout the internship. During my stay, I helped develop a new deployment system for dependency management because the company was using an ancient python 2.7 installation with terrible dependency management (ran into a lot of errors). The new deployment system featured auto-deployment scripts and auto-dependency management and moved the python version to 3.6 which was much needed for some machine learning libraries. The company had me do a lot of web-scraping work to pull signal data from financial sites based in China and test if these had any signal. One of these jobs required me to break a captcha which I successfully did using convolutional neural networks. One other project that I worked on included sentiment analysis on conference calls to see if those introduced any signal on those companies.

Technologies Used: Python data science stack including pandas, Theano, Keras, and numpy

Software Engineer Intern

Minted

Jun 2016 – Aug 2016

San Francisco, California

Minted is a e-commerce company selling crowd-voted and crowd-supplied art. At Minted, I was put on the fulfilment team which dealt with order processing (most stressful work at the company). All throughout, I worked on a stack of a large Flask monolith with business logic and queries written in Python with SQLAlchemy. One project that I worked on was address validation on the client side to help reduce costs to Minted. I also worked on an analytics dashboard (frontend/backend) to measure productivity of company-hired design "approvers" to incentivize faster work. I also worked to implement A/B testing to help the marketing sector of the company make better marketing decisions and test them as well. I gained familiarity with how order processing works in the case of a state machine. Any front-end work was done in Javascript with react-redux methodology and all backend queries and logic was done in python. We used MySQL to store user/product information.

Languages/ technologies: Python, MySQL, React-Redux JS,
deployment tools: vagrant, puppet, AWS, Rackspace

Projects

uw-rso-directory (<https://github.com/sm5art/uw-rso-directory>):

- For this project, I was inspired to write a better service to discover new clubs at my university because the service that existed was a 200 page online manual of clubs which was unusable.
- I web-scraped all the data from the manual and placed it in a UI where you can search using text query or sort them by types of club (academic/sports etc), which used infinite scroll methodology instead of page by page search.
- My choice of data delivery was much smarter than their templating approach, which made for a much quicker service.
- Demo at <https://uwrsodirectory.herokuapp.com/>
- Technologies used: webpack, Javascript, react, redux, express (web server), MongoDB

genetic pong (<https://github.com/sm5art/genetic-pong>):

- This project was my first tinkering into reinforcement learning. It's an algorithm that finds the best "brain" for a pong player given the state of the game and what actions the pong player can take at a given moment. It's inspired by Darwin's theory of evolution where after some number of generations the pong player converges to the best possible player given those training circumstances.
- Training demo at <https://www.youtube.com/watch?v=mFOkdGye7vY>
- Note: this is a collaboration project: my friend Sarthak from UW wrote the pygame portion of the project, I wrote the AI on top.
- Technologies used: Python (numpy + pandas + pygame)

kernel (<https://github.com/sm5art/kernel>):

- This project is a basic i386 kernel built using many operating systems resources that I've found on Wikipedia and operating system developer forums. It features basic graphical capabilities, a GDT code descriptor table, an interrupt table with handlers, basic CPU clock, and some system methods shared by these components.
- This was a fun project to learn more about lower-level system calls and how all languages boil down to machine instructions at some level.
- Technologies used: C, Makefile, gcc, nasm, ld, qemu-system-i386 for emulation

Education

Approaching BS Mathematics at University of Washington, Seattle, WA '2021, GPA: 3.3

Minor: Applied Mathematics

Notable Coursework: Linear/Matrix Algebra, Numerical Analysis, Mechanics, Programming I,II, Algorithms and Data Structures, Differential Equations, Vector Calculus, Intro to Complex Analysis, Statistics, Probability I, Artificial Intelligence
