

Evan Li

I like (very!) contemporary jazz piano, video games and photography

 hc3li@edu.uwaterloo.ca

 in/hanchen-li/

 evanli.me

 [@Evanli1999](https://github.com/Evanli1999)

Skills

Languages: Python, SQL, C++, C, Scala, Java, JavaScript, MATLAB,
Tools: Pandas, Scikit-Learn, Scipy, StatsModels, Flask, Redis/RQ
AWS (S3, EC2), Google Cloud (GCS, DataStore), Git, Linux

Experience

Data (Science/Engineering) Intern / Clearbanc / Toronto, ON May-Aug 2019

- Developed new revenue forecasting models (**XGBoost**, **RF**, **novel spectral analysis**)
- Designed a novel time series classification algorithm detecting erratic revenue patterns
- Collaborated on a risk classification framework used to suggest repayment period in offer adjudication based on revenue **forecast uncertainty** and a **delinquency classifier**
- Implemented a new multithreaded model testing framework using **Redis** and **RQ** that improved backtesting time by 5x

Core Software Developer / IBM Canada Varicent / Toronto, ON May-Aug 2018

- Implemented API Keys for **REST API (C#)** and **Web App (JavaScript)**
- Developed Federated Access controls to automatically configure new user accounts
- Designed and implemented individual model access controls (e.g. certain user roles, or users connected through SSH, IP whitelisting)
- Improved **SAML** protocol to enforce audience and lifespan assertions of IDP responses

Behavioural Team Member / WATonomous / Waterloo, ON Sep-Dec 2018

- Designed and implemented localized obstacle map to avoid object collisions using C++
- Developed waypoint processor to plan vehicle path to find waypoints on a map

Projects

Somewhat-Blue / Chess AI / Python (Tensorflow), C++, EC2 Aug 2018

- Implemented AlphaZero's Self-Play learning algorithm
- Trained a **Convolutional Neural Net** to evaluate board positions in order to suggest the moves that should be evaluated using **Monte-Carlo Tree Search**
- Reduced tree search time by **54%** using **Alpha-Beta pruning**

VM / Vim Clone / C++ Nov 2018

- Supports most of Vim's core features (e.g. motions, search, macros, repeat commands)
- Implemented **MVC pattern** to control input/output, state logic used **visitor pattern**

Scala!? / Scala Subset Compiler / Scala Sep-Dec 2018

- Compiler for a typed subset of **Scala** to **MIPS assembly**
- Implemented **Lexer/Parser (CYK)**, **type safety** and **garbage collector (Cheney)**
- Features include **lexical scoping**, **closures** and **tail call optimization**

Education

University of Waterloo

Bachelors of Computer Science, Honours, Co-op, 2022