

Angad Kalra

Toronto, ON

angadkalra94@gmail.com

angadkalra.com

github.com/angadkalra

Education

MSc in Applied Computing, Data Science Concentration

September 2018 -December 2019

University of Toronto, Department of Computer Science

Courses: Data Science Methods, Communication and Collaboration, Quantum Computing, Machine Learning in Healthcare, Algorithms for Collective Decision Making

BSc Computer Science and Mathematics

September 2012 – May 2018

University of British Columbia

Technical Experience

Galiano Medical Solutions – Vancouver, BC

May 2018 – September 2018

Full Stack/Machine Learning Engineer

- Implemented a web application allowing doctors to search through patient database and find similar X-rays to their patient's.
- Built search functionality as a combination of deep learning and Elasticsearch document search.
- Developed using ReactJS, Django, TensorFlow, Docker, and Elasticsearch.

Center for Molecular Medicine and Therapeutics – Vancouver, BC

May 2017 – August 2017

Research Assistant

- Implemented a deep-CNN in TensorFlow to predict protein binding sites for the FOXP3 protein.
- Transformed a dataset of 80,000 DNA sequences to one-hot encodings and trained the network for twelve hours. Achieved 80% prediction accuracy on first attempt.

Vision Critical – Vancouver, BC

January 2016 - August 2016

Software Developer

- Responsibilities were fixing bugs, writing integration tests, improving test coverage in deployment pipeline, and learning new technologies simultaneously.
- Replaced legacy code by rewriting with HTML5 and various JavaScript frameworks (Grunt, jQuery, Jasmine).

Recent Projects

Co-Director of Code the Change Foundation

- Created a nonprofit in Vancouver, BC, with a group of engineers from UBC.
- Purpose is to help nonprofits and charities around the world with their technical needs. All projects are open-source and available at: <https://codethechangeubc.org/projects.html>

Portfolio Selection Strategies using Quantum Computing

- Using D-Wave Leap (quantum annealer), ran graph algorithms for NP-complete problems on financial datasets in order to compare solution quality against classical algorithms. Implemented in Python 3 in Jupyter Notebook.

Technical Skills

Programming Languages: Python 3, JavaScript.

Libraries: Tensorflow, NumPy, Pandas, scikit-Learn, PySpark, Matplotlib.

Frameworks: Django, Flask, ReactJS, Apache Spark, Docker.

Databases: PostgreSQL, MySQL, Elasticsearch.