

# Dennis Fong

416-602-6588 | dennis1.fong@gmail.com | [linkedin.com/in/dfong298/](https://www.linkedin.com/in/dfong298/) | [github.com/DFong298](https://github.com/DFong298)

## EDUCATION

---

### McMaster University

*Bachelor of Engineering (Software Engineering, 3.7 CGPA)*

Hamilton, ON

*Sep. 2020 – Present*

**Courses:** Data Structures (Java), Algorithms and Complexity, Intro to Machine Learning (Python, PyTorch), Software Design (Java, Python), Concurrent Programming (Go, Python), Databases (SQL)

## TECHNICAL SKILLS

---

**Languages:** Python, JavaScript, SQL, C/C++, Java, Go, MATLAB, HTML/CSS

**Libraries/Frameworks:** Git, AWS, TensorFlow, PyTorch, NumPy, Pandas, Node.js, Flask, React

## EXPERIENCE

---

### Cloud Engineer Intern

Jan. 2024 – Aug. 2024

*Purolator*

*Mississauga, ON*

- Orchestrated a real-time data pipeline to monitor **500+** AWS resources using **CloudWatch**, **Lambda**, and **S3**
- Automated deployment of **IAM** roles in **47** accounts with **CloudFormation** to warrant cross account monitoring
- Collaborated with solution architects to design applications adhering to best practices and architectural patterns
- Visualized data on **Amazon QuickSight**, leveraging **SQL** queries executed on **Athena** against a **Glue** database

### Teaching Assistant - Algorithms and Complexity

Jan. 2024 – Apr. 2024

*McMaster University*

*Hamilton, ON*

- Facilitated **200** students' learning of algorithms and computational tractability assisting them with understanding complex topics such as **dynamic programming**, **network flow**, **divide & conquer**, and **complexity classes**

### Quantitative Developer - DeGroot Finance and Investment Council

Oct. 2022 – Feb. 2024

*McMaster University*

*Hamilton, ON*

- Analyzed sentiment of financial headlines with **89%** accuracy by implementing natural language processing model
- Implemented mean reversion trading strategy in **Python** yielding a **278%** profit over 10 years on historical data
- Projected stock value within a **9%** margin from true market price by utilizing long-short term memory in PyTorch
- Created and verified validity of **11** algorithms through back-testing on **over 1,000,000** historical data points

### Undergraduate Research Assistant

May. 2023 – Aug. 2023

*McMaster University*

*Hamilton, ON*

- Solved software version dependencies and conflicts in polynomial time with linear programming in **Python** and **Z3**
- Explored methods of optimizing package management in supercomputers by formulating them as SAT problems
- Detected cycles in dependencies at **97%** accuracy by devising an algorithm that implements dynamic programming

## PROJECTS

---

### Dog Breed Classifier | *Python, TensorFlow, NumPy, Pandas*

- Deployed a deep learning model using the Xception architecture to classify **120** dog breeds with **94%** accuracy
- Preprocessed dataset of **10,000+** images by resizing, and rescaling resulting in a **17%** increase in model accuracy
- Evaluated metrics such as precision, recall, and confusion matrix to gain deeper insights into model performance

### Track.it | *JavaScript, SQL, MySQL, React, Express.js, Node.js*

- Built a full-stack web application that tracks the daily and total change of a portfolio created by the user
- Constructed front-end design with React to display user interface for **up to 10 pages** using **MaterialUI**
- Used Express.js to communicate **various relevant data** between the front-end and the database entities
- Created and maintained database schema using MySQL to store asset data leveraging Express.js functionality

### London Tube Pathfinding | *Python, PyTest, flake8*

- Experimented with pathfinding algorithms on **more than 300 train stations** in the London Tube transit system
- Applied **object oriented programming** and **SOLID principles** to create proficient and maintainable code
- Tested algorithm implementations using **PyTest** and **Flake8** for automated error checking and code organization
- Utilized **dynamic programming** in order to reduce algorithm execution time from **factorial to exponential**