CHRISTOPHER DINH

Irvine, California 92612 | (240) 535-3571 | cqdinh153@gmail.com | linkedin.com/in/cqdinh | github.com/cqdinh | cqdinh.github.io

EDUCATION

UNIVERSITY OF CALIFORNIA, IRVINE, Irvine, California
Master of Computer Science, Expected December 2020, GPA 3.97 / 4.0

UNIVERSITY OF MARYLAND – BALTIMORE COUNTY, Catonsville, Maryland
B.S., Computer Science - Data Science Track, 2015-2019, Major GPA: 4.0 / 4.0
B.S., Mathematics, 2015-2019, Major GPA: 3.9 / 4.0
magna cum laude, Cumulative GPA 3.8 / 4.0

TECHNICAL SKILLS

Languages:	Python (Expert), C++ (Proficient), Java (Proficient), C# (Prior Experience)
Web Development:	Javascript, HTML / CSS, node.js
Libraries:	Anaconda, PyTorch, Tensorflow, D3.js, OpenCV, Scikit-learn, Pandas, NumPy, Plotly
Other:	MySQL, Jupyter, Git, Visual Studio Code, GDB, Linux / Unix, Google Colab
Deployment:	Google Cloud Compute, Streamlit.io

EXPERIENCE

SWAPPIT CO, Irvine, California
ML Volunteer Project, 6/2020 – Present

- Designed and implemented a deep learning model to predict the aftermarket prices of sneakers based on real data
- Initial model prototyping with Scikit-learn and PyTorch and final implementation in Tensorflow / Keras
- Given 60k training instances, the model's predictions average within $56 of the true sale price
- Currently designing a recommendation system to suggest products to users

IBM, Rochester, Minnesota
Software Engineering Intern, Cloud Managed Application Systems, 6/2018 – 8/2018

- Learned and practiced Agile software development practices

MANTAROBOT CORP, Germantown, Maryland
Summer Intern, 5/2014 – 8/2014, 5/2015 – 8/2015, 5/2016 – 8/2016, 5/2017 – 8/2017

- Designed and implemented an OpenCV-based system in C# that uses a camera to automatically dock a robot with its charging station with its dock from up to 7 feet away with an angular error of less than 10 degrees and a linear error of less than 3 inches.
- Decreased control latency for a telepresence robot by an average of 50% by implementing WebRTC as a control method both in the browser and in an Android app.

PROJECTS

Toxic Comment Classification ● *Deep Learning Course Project* ● Code ● Kaggle
Deep Learning model built in PyTorch that uses BERT to determine if online comments are toxic
- Built a custom head on top of BERT that uses an attention mechanism to process comments longer than BERT's 512-token limit. After fine-tuning, the model achieved an average AUC of 0.959 across 6 types of toxicity.
- Trained and deployed the model on a Google Cloud Compute instance using a custom model server that is accessed by a Streamlit.io app.

Google CodeU ● *Personal Project* ● Code
Invite-only program working on a team with 2 other students supervised by a Google engineer
- Implemented features for a chat application in Java including multithreaded update polling and storage of server state in a SQLite database through JDBC.
- Participated in regular code reviews with a Google engineer and learned industry practices including unit testing, trunk-based development, collaboration using Github, and uniform code style.