Andrew Wang
📞 647-916-9309 | 📧 a387wang@uwaterloo.ca | 🌐 GitHub: github.com/AndrewWang12345

Technical Skills

Languages: Python, C++, JavaScript/TypeScript, C, SQL, HTML, CSS,
Libraries & Frameworks: NumPy, Pandas, Docker, PyTorch, Detectron2, Cython, Flask, React
Concepts: Object Detection, Geospatial Processing, Parallel Computing, CI/CD, Rest API

Work Experience

Machine Learning Engineer – Western University Global Meteor Network Jan 2025 – Apr 2025

- Built and deployed full-stack geospatial applications using **FastAPI**, **React**, and **Mapbox** to visualize real-time satellite contrails and flight paths, with server-side acceleration using **Redis** caching and **Cython** optimization.
- Developed cloud-integrated backends for secure media delivery from **Google Cloud Storage** and remote **SSH** servers, parallelized KML parsing for dynamic station field-of-view rendering.
- Developed a scalable preprocessing pipeline for video annotation and trained a **Detectron2** Mask R-CNN for instance segmentation (85% precision) and used **HDBScan** for instance tracking

Python Programmer – H2O Geomatics Sept 2023 – Dec 2023

- Achieved **15× speedup** in satellite picture analytics pipeline using **Cython**, **NumPy**, and **parallelization**
- Built GUI tools for **uncertainty-aware raster merging** using confidence-weighted fusion
- Reduced memory and I/O overhead with **Numba**, **memory-mapping**, and low-level optimizations
- Collaborated in a fast-paced Agile team using **modular Python OOP** and **Git version control**

Data Analyst – Meteorological Service of Canada Jan 2023 – Apr 2023

- Conducted **anomaly detection** on multi-year **weather station time series** across North America
- Built models using **logistic regression** and **KNN classifiers** to identify sensor malfunctions
- Filtered Canadian stations using custom **geolocation scripts** in **Python** and **AWK**
- Delivered reproducible workflows with **automated test harnesses** and technical documentation

Projects

AI Chatbot – PyTorch + MERN stack

- Built a **transformer-based chatbot** from scratch with **self-attention**, **BPE tokenization**, and **contextual memory**
- Implemented **CUDA acceleration** and exported **checkpointed model weights** for future reuse

Racket String Tension App – React Native + Python

- Built a **React-Native** application to calculate string tension based on thickness and frequency
- Trained a regression model using collected data for accurate tension prediction
- Containerized backend with **Docker**, set up **CI/CD workflows** in **GitHub Actions** for deployment.
- Deployed app the Google Play store and backend to Azure

Regional Event Aggregator (In Progress) – Flask, React, FullCalendar.js

- Developing full-stack tool to extract and organize **campus events** from **Instagram posts** using **NLP** and web scraping
- Backend powered by **Python Flask**, with dynamic UI powered by **React** and **FullCalendar**
- **Frontend** calendar UI built with FullCalendar.js and dynamic event syncing.

Education

University of Waterloo — B.Eng. Computer Engineering, Co-op Sept 2022 – Present

Relevant Courses

ECE 250 – Algorithms & Data Structures (completed on Linux environment)

- Implemented the data structures and algorithms in C++ including binary search trees, hash tables, graphs, Stacks, Queues, Maps, Sets, sorting, graph search, minimal spanning tree
- Learned algorithms and design techniques like greedy, divide and conquer, dynamic programming

ECE 252 – Systems Programming & Concurrency (completed on Linux environment)

- Implemented in C language: processes and pthreads, system calls, concurrency (semaphore, mutex, monitors, and barrier synchronization), user-level memory management, performance and correctness of concurrent systems, deadlock detection and recovery, and file systems.