Vikram Chandar

chandarv@mcmaster.ca |+1 905-781-5826 | LinkedIn| Developer Portfolio| GitHub

EDUCATION

McMaster University Sept 2022 - Present
Bachelor of Engineering - Software Engineering Co-op
Cumulative GPA: 3.9/4.0; Dean's Honour List 2022-2024
Relevant Coursework: Object Oriented Programming, Data Structures and Algorithms, Digital Logic and Systems

WORK EXPERIENCE

Alluri Technologies @ BMO | Jr. Full-Stack Developer May 2023 – Sept 2023

- Implemented a **maven** framework to automate **selenium** testing on BMO **Investorline UI**. New **framework** and **selenium** implementation reduced code **runtime** by **65%**
- Used **SQL** to validate trader orders by sending **query requests,** extracting relevant information for **quantitative verification**
- Helped coordinate mobile **UI notifications** for limit and stop trades using **Springtools**

SKILLS

Programming Langugages: Python, Java, C, JavaScript, HTML, CSS, SQL, Bash, Verilog
Frameworks/Libraries: Maven, Selenium, TensorFlow, Scikit-learn, React, Node.js

PROJECTS

Stock Closing Price Forecaster
Utilized: Python, TensorFlow, pandas, matplotlib, seaborn, numpy

- Used a **Recurrent Neural Network** comprised of **LSTM** cells to predict future adjusted closing prices of tech stock with a **root mean squared error** of less than **0.05**.
- Analyzed stock trends using **pairgrids, histplots, pyplots and heatmaps** to determine the optimal **data shape** the **RNN model** based predictions on
- Graphed the comparison between actual and **RNN forecasted** closing prices visually using a **matplotlib** and **numpy**

GTRacer
Utilized: Python, Pygame, Pygbag, Asyncio

- Designed and implemented an interactive car racing game using **pygame** with software supported **N-key rollover** that generated over **2200 impressions** on itch.io during the first two weeks of launch
- Designed **Sprite Collisions** and obstacle generation algorithms to implement increased difficult scales based on performance and game customization
- Ported the python game to **WebAssembly** format to enable deployment on the internet with **near-native** performance

Quanser Robot Recycling Simulation
Utilized: Python, Quanser

- Designed a **Python-based** program that conceptualized a line-following Q-Bot that accurately gathers, transports and dumps recyclables into appropriate shipment containers
- Utlilized **IR sensors** to engineer algorithms that accurately navigated predefined paths, optimizing speed and efficiency
- Integrated **Colour sensors** to accurately determine off-loading location of recyclables coupled with **Ultrasonic sensors** that determined Q-Bot stopping position for precise dumping of waste
- Controlled **Linear Actuators** on the Q-Bot to enable safe disposal of bottles while ensuring efficiency well within time constraints

Email Spam Filter
Utilized: Python, Sklearn, Matplotlib, numpy, pandas, Flask, Tkinter

- Curated a diverse **dataset** of email samples and **preprocessed data** to extract relevant features such as **metadata**, text content and sender familiarity
- Employed state-of-the-art **Linear Regression**, **SVM** and **MultiNominalNB** models to create a **stacked classification model** with over **97% accuracy** in determining spam emails
- Designed a custom **GUI** with **Tkinter** paired with a **RESTful API** backend using the **Flask framework** to enable seamless communication between the **GUI frontend** and the **classification model**