Shihan Ai

g3aishih@gmail.com

LinkedIn : aishihan
GitHub: g3aishih

Education

University of Toronto — Toronto, CA
Honours Bachelor of Science in Computer Science — *Sept 2012 - June 2017*
Major GPA: 3.58/4.00; CGPA: 3.40/4.00

Programming Skills

Languages: Python, Java, JavaScript, Go, SQL, HTML, CSS, MATLAB

Technologies: Node, Express, React, Redux, Electron, PostgreSQL, MongoDB, OpenCV, NumPy, NLTK

Professional Experience

Yaar Inc — Toronto, CA
Software Engineer — *Feb 2018 - Present*

- **Yaar Core Product**: Worked as a team of two and used Electron, JavaScript, React, Redux, Go and PostgreSQL to create Yaar, an A.I powered team collaboration tool
- **Yaar Front-end Components**: Developed modular React components using JavaScript for the control panel, chat box, message bubble, hover container, link container and thumbnail previewer of Yaar's front-end
- **Yaar Front-end Communication**: Designed the structure of the Redux Store and created Redux Actions and Reducers to communicate between the front-end React components
- **Yaar Back-end**: Designed the database schema and communication protocols between the front-end and the back-end of Yaar using Go
- **Yaar Database Queries**: Wrote PSQL queries to retrieve and update records such as user information, user tasks, channels members, team messages, links, and link cookies
- **Training Data Collection**: Created a script using JavaScript and collected training data consisting of 56,000 audio samples from workers on Amazon Mechanical Turk saying phrases such as "Hey Yaar" and "Sleep Yaar"
- **Natural Language Model Integration**: Worked as a team of two with a Machine Learning Engineer to integrate a speaker recognition model and a keyword detection model into Yaar

IBM — Markham, CA
Software Engineer Intern — *May 2015 - Aug 2016*

- **DB2 Analytics Platform**: Designed the back-end of an analytics platform using JavaScript and DB2; wrote SQL queries to pull customer data from pre-existing databases and generated infographics using D3.js to provide product managers with real-time insight from massive amounts of live customer data
- **YouTube Analytics**: Created a script using Python, SQL, and the YouTube Analytics API to capture the audience retention rates of DB2 promotional videos and generated infographics with the data to gauge the effectiveness of the promotional videos
- **Twitter Bot**: Developed a bot with NodeJS, JavaScript, DB2, and the Twitter API to automate the promotion of DB2 on Twitter and saved up to an hour of work per day for six other interns

University of Toronto — Toronto, CA
Research Assistant — *May 2014 - Aug 2014*

- **Lexical Analysis**: Performed lexical analysis on five hundred tweets to understand the structure of natural language on Twitter and created a script with Python and NLTK that extracted the important words of one hundred new tweets; manually verified the important words of the new tweets and achieved an accuracy of 87%
- **Sentiment Analysis**: Developed a script using Python, SQL, NLTK, and the Twitter Search API which collected movie related tweets from Twitter, extracted their sentiment, and stored the result in a MySQL database
- **Twitter Bot**: Built a Twitter bot with Python and NLTK that responded to movie related tweets with show-times, theater locations and movie recommendations

Projects

Optimized Interactive Foreground Extraction June 2017 - July 2017

- **Foreground Extraction**: Engineered a Computer Vision algorithm using Python, NumPy, and OpenCV that can classify pixels in an image as either the foreground or the background of the image
- **Optimization**: Independently researched and implemented methods to decrease processing time when processing high resolution photos taken by modern day cameras
- **Benchmark Testing**: Performed benchmark tests against traditional foreground extraction methods such as GrabCut and improved the processing time of 12-Megapixel images from an average of 74 seconds to 3.6 seconds

Exemplar-Based Image Inpainting May 2017 - June 2017

- **Image Inpainting**: Researched and implemented a Computer Vision algorithm in Python, NumPy, and OpenCV that can crop out user defined areas in an image and replace the cropped-out areas with visually plausible textures
- **Hyper-Parameter optimization**: Analyzed and identified the optimal hyper parameters to product results with little or no optical artifacts
- **Practical Applications**: Identified real world applications for the algorithm such as digital acne removal