Jack Wang

Software Engineering

github.com/jj22wang | jj22wang@uwaterloo.ca

Professional Experience

August 2016 - Present

Yelp

Data Quality Engineer — Python, Sklearn, Pandas

- Sourced new training and gold set data to significantly improve the performance of location accuracy models
- Fine tuned data model by building new features, adding new signals and adjusting decision thresholds
- Constructed a feedback loop using in-app survey questions to adjust for inaccurate model predictions
- Improved automatic business attribute detection to handle multiple active subattributes

January 2016 - April 2016

Capital One

Data Scientist — Python, PySpark, Hive, JavaScript, Sklearn

- Adapted and productionalized existing data models to service data visualization applications
- Designed a parallelized batching pipeline using pySpark and Hive to batch data prepping
- Migrated web services to use Docker to improve deployability with AWS

May 2015 - August 2015

Loyalty One

R&D Engineer — Scala, Spark, Kafka, Elasticsearch

- Developed a distributed computing API using Spark, to eliminate boilerplate code for ML feature generation
- Parallelized existing data analytics processes improving run times by 100 times
- Executed large scale performance testing to optimize data aggregation runtimes
- Provided an end to end streaming data pipeline to do real-time analysis of campaign statistics
- Integrated Jenkins into services built

Education

2014 – 2019 Candidate for B.SE. 3.7 Gpa
The University of Waterloo

Computer Skills

Languages	Python, Scala, Javascript
Libraries	Spark, MLlib, Jenkins, Sklearn, Pandas, Matplotlib,
Tools	Hive, MySQL, Elasticsearch, Kafka, Git, Docker

Side Projects

Tagger — Python

A python script to predict and tag song attributes using existing tags and filename

- Researched previous work on filename analysis done in Sony CSL's A Naturalist Approach to Music File Name Analysis
- Designed an algorithm to chunk and weight the different segments of a filename
- Reconciled chunked filename information with existing tag information and online databases

MeetMe — JavaScript

A Google Maps webapp to find the closest meeting point between a set of locations

- Interprets distance information to estimate where optimal search locations
- Aggregates distance matrixes to find an optimal meeting place

FunctionSolver — JavaScript

A web application to interpret, reduce and graph mathematical equations

- Wrote a recursive chunking algorithm to parse and syntax check user input