Jonathan Stokes

stokesj84@gmail.com • 215-207-1143 • Oakland, CA • https://sophist0.github.io/ • https://github.com/sophist0

Profile

I have a PhD. in Electrical Engineering from Drexel University. My research was in graph sampling which gave me experience in software engineering, applied math, and the fundamentals of machine learning. Most of my programming experience is in Python, however I have also worked in Java. I thrive off the ability to work in a collaborative environment where I can get feedback on my work. I believe I do my best work in the processes of learning new skills and look forward to new challenges.

Education

PhD. of Electrical Engineering — **Drexel University**; Philadelphia, PA
Advisor: Steven Weber — Graduation: December 2018 | GPA: 3.7/4.0

Masters of Electrical Engineering — **Drexel University**; Philadelphia, PA
Advisor: Steven Weber & Mark Hempstead — Graduation: December 2015 | GPA: 3.7/4.0

Bachelors of Science in Electrical Engineering — **University of Rhode Island**; Kingston, RI
Graduation: May 2012 | GPA: 3.8/4.0

Experience

Federal Software Engineer: Intel - Pheonix AZ — January 2022 - Present

Project: Testing video object tracking.

- Created an object tracking ground truth dataset, generating video and location information in the Unity engine.
- Wrote a framework in Python to calculate object tracking metrics relative to the ground truth dataset.
- Used the results from the testing framework to make recommendations regarding improving the object detection and tracking algorithms.

Project: Optimized an object tracker.

- Wrote a framework to tune a Kalman filter based tracker using the Bayesian Expected Improvement algorithm.
- Modified the Expected Improvement algorithm to avoid resampling points in the parameter space.
- Showed that the resulting optimization algorithm could achieve better results than random sampling given the same sampling budget.

Senior Software Engineer: General Dynamics - Groton CT — January 2021 - December 2021

Project: Estimating construction times.

- Collected and cleaned the data stored on various databases.
- Fit Gaussian process regression models to the data using Pytorch and using Bayesian optimization.
- Experimented with using Reinforcement Learning to select a construction process using Stable-Baselines implementations of DQN, and A2C.

Project: Classifying documents.

- Collected and cleaned the data stored on various databases.
- Used a bag of words model combined with a gradient boosting model solve a binary classification problem with 95% accuracy.
- Used Pytorch implementation of an LSTM module to solve the same binary document classification problem with over 95% accuracy.

Software Engineer (Contract): Centrly - San Diego CA — January 2020 - April 2020

Project: Designing and testing web scrapers.

- Created several web scrapers using Python, Selenium, Azure, and AWS to collect business intelligence data.
- Required correctly scoping various projects, designing web scrapers, implemented and tested scrapers with regard to initial objectives.

Kaggle Competition: Kuzushiji Recognition Competition — August 2019 - October 2019

Project: Segment and classify Japanese handwritten script characters.

- Combined U-Net for character segmentation and MobileNet for character recognition.
- Required setting up the hardware and software environment for a deep learning project, managing large data sets, training deep learning models, and working with TensorFlow.

Student and Research Assistant: Drexel University - Philadelphia PA September 2013 - September 2018

Project: Estimate the expected cost of star sampling to find a target node in a large graph.

Procedure: Identified a tractable problem. Read related research. Formulated solution. Iterated. Tested solution.

Reason: Given a graph and computational constraints, one may want to know if star sampling will find a target node.

- Estimated the expected unit and linear cost to find a target node in Erdős Rényi (ER) graphs under three types of star sampling: Star sampling with replacement (SS-R), Star sampling with center removal (SS-C), Star sampling with star removal (SS-S).
- Proved asymptotically in the size of the graph the probability of finding a target node on a given sample under the three variants of star sampling is approximately equivalent.
- Coded the simulations showing the estimates of the expected unit and linear cost of using star sampling to find a target node are accurate on ER graphs and can be accurate on real-world graphs.

Project: Estimate the expected steps for a biased random walk in a large graph to find a maximum degree node.

Procedure: Identified a tractable problem. Read related research. Formulated solution. Iterated. Tested solution.

Reason: Given a graph and computational constraints, determine if a biased random walk will find a target node.

- Developed a Self Avoiding Walk Jump (SAWJ) algorithm to search large graphs for maximum degree nodes.
- Modeled a rough upper bound on the expected number of steps required by SAWJ to find a maximum degree node using a discrete time Markov chain model which is shown to be accurate if the joint degree distribution of the graph is known.
- Coded the simulations showing that SAWJ outperformed competing algorithms in the literature on degree assortative ER graphs and some degree assortative real-world graphs.

Key Papers

Journal of Internet Mathematics: "*Graph search via sampling with and without replacement*" (2020)
IPL: "*Common greedy wiring and rewiring heuristics do no guarantee maximum assortative of given degree*" (2018)
KDD Workshop MLG: "*Star Sampling with and without Replacement*" (2017)
IEEE BigData: "*The Self-Avoiding Walk-Jump (SAWJ) Algorithm for Finding Maximum Degree Nodes in Large Graphs*" (2016)

Experience Using

- Artificial Neural Networks
- Deep Reinforcement Learning
- Gaussian Processes
- Graph Theory
- K-Nearest Neighbor
- Linear Regression
- Markov Chains Models
- Naive Bayes
- Principle Component Analysis
- Reinforcement Learning
- Long Short-Term Memory Models
- Support Vector Machines

Academic Background - Interests

- Artificial Intelligence
- Algorithm Design
- Detection & Estimation
- Graph Theory
- Information Theory
- Machine Learning
- Optimization
- Applied Probability
- Stochastic Processes