Chao Xu

mgcclx@gmail.com · (217) 778-9067 · https://chaoxuprime.com · Github: chaoxu

Skills

Python· Java · Git · Algorithm Implementation/Research · Haskell · SQL
Machine Learning · Scala· Spark · Hadoop· Hive · Pig · LaTeX· APL· C++

Education

2018, PhD in Computer Science, University of Illinois at Urbana-Champaign
Advisors: Karthik Chandrasekaran and Chandra Chekuri. Specialization: Algorithms.

2013, BS in Mathematics and Applied Mathematics & Statistics, Stony Brook University
Minor in Computer Science.

Experience

Research Scientist, Yahoo! Research (Verizon Media), New York, NY, USA.
Scalable Machine Learning Group. June 2018-now

- Lead development of the household clustering algorithm and its data pipeline. Coordinate between multiple teams to obtain data, and measure resulting performance on ads targeting and attribution. (Scala, Spark, Pig, Hive).
- Maintain the transition platform on Search-to-Native Ads during the sale of Yahoo search business to Microsoft. Apply machine learning to improve the Search Ads click prediction model, and calibrates with the Native Ads click through rate. (Hadoop, Pig, Hive).
- Research, design and implement bipartite matching based ad placement and pricing system for Native Ads. Determine the effect of floor prices on ads revenue. (Hadoop, Hive).

Software Engineer, Google, Mountain View, CA, USA.
Google Analytics Backend. Feb.-Aug. 2013

- Maintain the critical custom filter component used by every single request to Google Analytics. Refactoring by introducing reflections. (C++).
- Introduce algorithmic improvements to a load partition problem in data centers. Substantial running time reduction from $O(n^2)$ to $O(n)$. (C++).
- Solve backward compatibility issues for customers by designing short regular expressions generator for integer ranges. (Haskell, Regex).

Visiting Researcher, National Institute of Informatics, Tokyo, Japan.
Hosted by Ken-ichi Kawarabayashi. Jun.-Aug. 2017

- Work on algorithms related to graph homomorphism and generalized matching with application to pair programming.

SageMath
Google Summer of Code. June-Aug. 2015

- Implement the unweighted and weighted matroid intersection algorithm. (Python, Cython).
- Improve the running time for 4-connectivity computation from $O(n^5)$ to $O(n^{4.5}\sqrt{\log n})$. (Python, Cython).
- Produce end of project report for publication in the matroid research community. Maintain detailed documentation for developers.