Algorithms for Big Data

Spring 2016

CUNY Graduate Center

Prof. Matthew P. Johnson

Prerequisites: Algorithms, and mathematical maturity; also, basic knowledge of discrete math, and linear algebra.

Texts: *Foundations of Data Science* (Hopcroft and Kannan) and *Data Stream Algorithms Lecture Notes* (Chakrabarti)

Assessment: Several bi-weekly problem sets and a course project. The project can take the form of either an expository paper, a nontrivial implementation project, or performing original research on a problem.

Catalog/course description: This course addresses algorithmic problems in a world of big data, i.e., problems in settings where the algorithm's input—the data—is too large to fit within a single computer's memory. Traditional analysis of algorithms generally assumes full storage of data and considers running times polynomial in input size to be efficient. Operating on massive-scale data sets such as those of tech companies such as Google, Facebook, etc., or on indefinitely large data streams, such as those generated by sensor networks and security applications, leads to fundamentally different algorithmic models. In previous decades, DBMS settings where the data fits on a machine's disk but not in memory motivated the external memory or I/O model (e.g. external mergesort and B-trees). More recently, models such as MapReduce/Hadoop have appeared for computing on data distributed across many machines (e.g. PageRank computation or matrix multiplication). Finally, streaming and sketching algorithms solve problems in linear or sublinear time, on sequences (e.g. finding missing, random, or frequent elements) and on graphs (e.g. finding matchings and counting triangles, deciding bipartiteness and connectivity). Other topics will include approximate matrix multiplication, the secretary problem, and compressed sensing.

Rationale: Traditional analysis of algorithms generally assumes full storage of data and considers running times polynomial in input size to be efficient. Operating on massive-scale data sets such as those of tech companies such as Google, Facebook, etc., or on indefinitely large data streams, such as those generated by sensor networks and security applications, leads to fundamentally different algorithmic models. MapReduce/Hadoop in particular has seen widespread adoption in industry.

Learning goals: The student will learn what the modern models for massive-scale data algorithms are, how to analyze algorithms in these models, and when to use them. In particular, the student will gain experience with MapRedude/Hadoop and with a number of streaming/sketching algorithms.