COSC6339: Big Data Analytics
Future: Systems for Big Data
Instructor: Carlos Ordonez

1 Course Contents

This is a "systems" + "data science" graduate course, which covers theory and programming to analyze big data, covering all available platforms and technologies today: database systems, data science languages, NoSQL and Hadoop. Past data mining research has been expanded to analyze big data, mixing structured and semi-structured content, with more complex statistical and machine learning models and exploiting parallel processing.

1. Big Data overview: Evolution from 3 Vs to 5 Vs (3+2), Data Lakes vs Data Warehouses vs Data Swamps, Data integration, pre-processing and cleaning, Parallel and Distributed Computing, Big Data Analytics problems: machine learning, graphs, streams.

2. Data Science language definition and philosophy, exploring data, data pre-processing and data cleaning, functional constructs, runtime in the OS, parallel processing in multicore CPUs and GPUs, main data structures (data frames, matrices).

3. Parallel DBMSs: parallel processing in multicore CPUs, parallel processing in a cluster with distributed storage, OLTP versus cubes, Data warehousing and denormalization, advanced SQL queries (pivoting, horizontal aggregation, keyword search, recursive queries), advanced programming (embedded SQL, UDFs, internal C), row/column/tile storage, indexing versus row ordering, ETL and pre-processing, machine learning and graph algorithms.

4. Hadoop ecosystem: file systems (Google, HDFS, Linux Posix), subsystems (Yarn, Storm, Zookeeper), storage (Hive, SPARQL, Parquet), containers vs virtual machines, text versus numeric processing, NoSQL and non-DBMS (MapReduce, Spark, MongoDB), Graph analytic systems, Search Engines (IR models, architecture, keywords, page rank, web spider).

The course will require reading CS research papers, mainly from Big Data and Database systems, available on DBLP, IEEE and ACM digital libraries. There is no textbook, but [2] and [1] are recommended.

2 Academic Background

Pre-requisites: For CS students it is encouraged, but not required, that the equivalent of COSC2430 (basic data structures and algorithms), COSC3380 (Database Systems), COSC3360 (Operating Systems), Machine Learning, were taken before. For students without an undergrad CS background it is encouraged you took before COSC6320 (Algorithms), COSC6345 (Programming Languages), COSC6373 (Parallel Computing). This is a course combining "systems", "machine learning" and "data mining". It is not an introductory course to machine learning or data science.
3 Grading

This is a tentative plan, which may change, depending on students CS background and interests. Keep in mind that this course will require a significant programming effort, which may be harder and will take more time than a test-based course.

- 90%: 3 programming homeworks (team-based):
  Programming language: Python, R and C++. System: your own computer with minimal hardware (multicore CPU, at least 4 GB RAM, 256 GBs storage); using a more powerful computer is acceptable, but will not produce any significant change in programming effort. OS: Windows, Linux or MacOS are fine.
  HW2 (30%): parallel DBMS (parallel query processing in the Vertica DBMS). Potential projects include: graphs, cubes, summarization, time series. Programming language: SQL and Python. System: a parallel DBMS using columnar storage; this Linux system will be provided by UH. OS: Linux.
  HW3 (30%): Hadoop Big Data Systems. Potential projects: document search, querying JSON data, gradient descent. Programming language: Java, Python. System: a Big Data system with data stored on HDFS; MongoDB or Spark; this Linux system will be provided by UH. OS: Linux.

- 10%: midterm exam around 10th week of classes. The exam will be in the classroom during official lecture time and will last 80 minutes. This exam will have 10 questions, which require a short answer or writing a few lines of source code (Python, C++, SQL). There is no final exam. The exam will be closed-everything (no computer or textbook allowed).

- 5%: extra credit, with in-class participation: answering instructor theory or programming questions; extra credit will also be given for good technical questions.

Homework details: There will be a total of 3-4 different analytic problems for each approach. In general, each analytic problem will be solved in the typical language or system used today. That is, there will be about 10-12 programming homeworks, where each team will solve 3. The homeworks will be proposed by the instructor, including analytic goal, programming language and target system. The instructor will randomly assign each homework to each team. However, a team may request a different homework with a written justification.

HW grading: homeworks will be graded in 2 ways. TAs will run each program to verify results and each team will prepare a 5-minute video.

Exam topics: parallel efficiency, external algorithms, distributed architectures, memory management, I/O optimization, efficient linear algebra, accelerating numerical methods and graph algorithms.

Data: This course edition will emphasize biomedical data, in text, tabular, binary or web form. However, there may be some traditional data sets coming from the UCI machine learning repository or the web.

Teams: Programs for each HW will be developed in teams of 4 students, randomly assigned by the professor; unfortunately, team partners cannot be chosen by students. It is expected a team stays together until the semester ends, unless some team members are lazy or do not work well in the team.

Final grade: all programs must be submitted and work in order to get at least C+. The scale to assign letter grades will be standard.

References