New York University Tandon School of Engineering
Computer Science and Engineering
Course Outline CS6513 Big Data

Professors: Raman Kannan rk1750@nyu.edu

Office Hours: email and weekly virtual meetings and weekly lab with TA (TBD)

Statement of Academic Integrity

Students are expected to follow standards of excellence set forth by New York University. Such standards include respect, honesty, and responsibility. This class does not tolerate violations to academic integrity including:

- Plagiarism
- Cheating in an exam
- Submitting your own work toward requirements in more than one course without prior approval from the instructor
- Collaborating with other students for work expected to be completed individually
- Giving your work to another student to submit as his/her own
- Purchasing or using papers or work online or from a commercial firm and presenting it as your own work

Please refer students to the Tandon code-of-conduct for addition information at:http://engineering.nyu.edu/life/student-affairs/code-of-conduct

Instructor allows students to source knowledge from any source including friends, colleagues, internet, library, papers and books.

All evaluations are open book and open notes and your problem solving abilities and your ability to work with other students are assessed.

Students who violate will be turned over to Deans Office and in the past Instructor has given F.

Course Pre-requisites

This offering of the course is for students who wish to prepare for a career in processing very large amounts of data. As prerequisite, students must have significant experience in programming, mathematical background, and some knowledge of algorithms. Of benefit for this course, but not required, is some basic knowledge in databases.

Course Description

Big Data requires the storage, organization, and processing of data at a scale and efficiency that go well beyond the capabilities of conventional information technologies. The course reviews the state of the art in Big Data analytics and in addition to covering the specifics of different platforms, models, and languages, students will look at real applications that perform massive data analysis and how they can be implemented on Big Data platforms.

Topics discussed include:

1. DataStores: SQL and NoSQL stores,
2. Map reduce over Mongo,
3. Apache Spark,
4. large-scale data mining using R or python or C# or java/scala and
5. visualization.

The curriculum will primarily consist of technical readings and discussions and will also include programming projects where participants will prototype data-intensive applications using existing Big Data tools and platforms, namely R, Relational, non-Relational, and Spark. Students may choose to use R, python and/or Spark over java or Scala.

Course Objectives

1. To learn about basic concepts, technical challenges, and opportunities in big data management and big data analysis technologies.
2. To learn and get hands-on experience analyzing large data sets using a combination of R,MySQL and mongo or any other non-relational database.
3. To learn and get hands-on experience analyzing large data sets using Apache Spark.
4. To learn about different types of scenarios and applications in big data analysis, including for structured, semi structured, and unstructured data.

Course Structure

Materials posted on classes plus intensive interaction via the e-learning platform. There will also be a reading list of research papers, and students are expected to complete three projects: 1) dataStore using SQL or mongo;2) Perform basic data science using R involving Shiny or R-Serve ; 3) Perform basic data science using Apache-Spark.

Readings

The required text for the course is: **Mining of Massive Datasets**. Rajaraman and Ullman, Cambridge University Press, 2011. Available online at http://infolab.stanford.edu/~ullman/mmds/book.pdf

Additional reading: **Data-Intensive Text Processing with MapReduce**. J. Lin and Chris Dyer, Morgan Claypool , 2010. Available online at http://lintool.github.io/MapReduceAlgorithms/

A list of journal and conference papers, available on the internet or via the Dibner electronic library, challenges from real-world, additional notes and presentations will be provided.

Software Requirements

The course requires the following software packages, all freely available:

1. The R Project for Statistical Computing, http://www.r-project.org/
   Optional R Studio, http://www.rstudio.com/
2. MySQL for relational, mongo for document oriented data.
3. Spark over java or scala will also be provided.
4. Any python – recommend version 3 or above.

All class related work must be done IBM cloud so the work can be centrally evaluated. A login will be provided free of charge. There is no installation required. Students are encouraged to use Xming (on Windows) and Quartz (on Mac) if there is aversion to command line interactivity.

.

Other Technical Requirements

We will be performing all our work on IBM Cloud. All the (functional) projects and tests required for this course have to be delivered on the IBM Cloud.

Access to IBM Cloud will be provided by the instructor free of cost.

Course requirements

Students are expected to do, and will be graded on: (a) 3 significant homework projects giving them hands-on experience in high volume data processing and graded as shown below:

1) P1, project 1 is to demonstrate proficiency in data storage technologies (20%)
2) P2, project 2 is to demonstrate proficiency in data processing (20%)
3) CP/P3, the Class Project is a much larger project and carries (40%) – the CP/P3
4) An Expectation Report to be submitted within 2 weeks carries(4%)
5) A reflection report to be submitted 13th or the 14th week carries (4%)
6) A weekly progress report, class discussions for 12 weeks (12%)

Course Topics by Week: Subject to adjustment/revision

Week 1: Course Overview. This is course is project driven and processing large number of files or

Records held in a persistent store is of particular interest. Some of the practical applications I am interested in are:

1. Broad applications of text analytics and social media programming
2. Broad applications of genetic information ;
3. Building and enhancing R with a framework for very large scale distributed computing in R using RServ and other distributed computing primitives available in R.
4. Students are expected to learn Sql/mongo/R and Spark
5. and engineer a non-trivial application.

Week 2: Databases and Big Data: Persistence, Transactions, Querying, Indexing and SQL

Week 3: Introduction to R Programming Language from Data Analytics Perspective I

Week 4: Introduction to R Programming Language from Data Analytics Perspective II

Week 5: Basic Data Mining and Statistics in SQL and R – Project 1 Due.

Week 6: Distributed Problem Solving and Flynn's Taxonomy

Week 7: Text Processing in R and Spark – basics TF/IDF, Word2Vec, LDA, Entity Extraction.

Week 8: Learning for Scalable Text Analysis

Week 9: Parallelism in Text Processing

Week 10: Algorithms for Big Data: Finding Similar Items – Project 2 Due

Week 11: parallelizing Cross Validation, LOOCV

Week 12: parallelizing Stochastic Gradient Descent, Boosting, Locally Sensitive Hashing

Week 13: SparkML – certain fundamental algorithms used in Machine Learning

Week 14: SparkKnife, Occams Razor, Classifiers as instructions and MISD – Project 3 due

Grade distribution will be as follows:

Top 30% of students will get A and A-,

Next 25% B+,

Next 25% B,

Next 20% other grades

Hadoop is not included in the topics and any work done on Hadoop is acceptable for this course. Pure Machine Learning or AI or Deep Learning is not the focus of this course. We are seeking to study,understand and engineer systems and PoCs highlighting parallel and distributed computing.

This is not a course on analytics but many programming challenges are drawn from many disciplines including machine learning, and data mining.

Schedule:

For calendar visit here
https://www.nyu.edu/registrar/calendars/university-academic-calendar.html

Our virtual class will be held over the webex or zoom (or the tool NYU provides) on classes from 9 to 10 PM on TUESDAY, each week. Attendance and active participation is required and submitting a weekly progress report is also required to receive class participation grade of 1% point toward your grade except for the first and the last week (12%). First week students are asked to submit an expectation report (4%) and a reflection report on the final week (4%). Submitting a weekly progress report is also required to receive class participation grade. Based on student feedback, the deadlines are rigid and I will not grant any extension.

This course requires a lot of work, allowing students to define their own projects.

Central learning objective are

1. distributed problem solving, leveraging large volume and variety of data. This course does not address velocity, the third V of Big Data.
2. Thinking and devising distributed problem solving architecture is an essential learning objective.
3. Implementing/engineering solutions using R,java,scala and Spark is the third learning objective.

To achieve these objectives we will use data from financial services, text analytics. Our focus will not be analytics or statistics or the mathematics. But given some analytical function how to compute solve problems using that function in a distributed environment. Given some statistics or mathematics how to setup parallel solution so that problems that cannot be solved in a single computer is solved using a cluster of computers.

Students are encouraged to study Ken Birman (Cornell, ISIS), Yale (Linda Gelertner) and Condor (Processor hunter) and PVM …from the 90s. Think about strategies to incorporate missing features into R and Spark. Students are also encouraged to review beta language (Aarhus University)– object mobility in a heterogeneous distributed environment, like the internet – or Gul Agha's Actor Model, parallel Deep Neural Networks [1][2][3], Data Classification[1][2][3], Analyzing Genomic Data [1][2][3][4][5], text mining [1][2][3], some algorithms in data mining and others.