Big Data Technologies: Course Information

Instructor:	Joseph Rosen
Email:	jrosen@iit.edu (preferred contact method)
Lecture:	Thursday, 6:45– 9:36 PM, online
Telephone:	312-860-0860 (m) (by appointment or for emergencies only)
Office Hours:	Tuesday, 5:15 - 6:15 PM by appointment. Other times available by appointment.

Course Description

Big data is the area of informatics focusing on datasets whose size is beyond the ability of typical database and other software tools to capture, store, analyze and manage. This course provides a rapid immersion into the area of big data and the technologies which have recently emerged to manage it. We start with an introduction to the characteristics of big data and an overview of the associated technology landscape and continue with an in depth exploration of Hadoop, the leading open source framework for big data processing. Here the focus is on the most important Hadoop components such as Hive, Pig, stream processing and Spark as well as architectural patterns for applying these components. We continue with an exploration of the range of specialized (NoSQL) database systems architected to address the challenges of managing large volumes of data. Overall the objective is to develop a sense of how to make sound decisions in the adoption and use of these technologies. Prerequisites: CS 425 or equivalent.

Required Texts

Tom White. 2015. Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale
(4th ed.). O'Reilly Media, Inc (TW)

Pramod J. Sadalage and Martin Fowler. 2012. *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison-Wesley.(PS)

Free Online

Jimmy Lin and Chris Dyer. 2010. *Data-Intensive Text Processing with Mapreduce*. Morgan and Claypool Publishers. https://vgc.poly.edu/~juliana/courses/BigData2014/Textbooks/MapReduce-algorithms-Jan2013-draft.pdf

Anand Rajaraman and Jeffrey David Ullman. 2011. *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA. http://infolab.stanford.edu/~ullman/mmds/book0n.pdf

Course Notes

Copies of each week's course lecture notes will be posted on the Blackboard system before class starts each week. Students should note that the lecture notes are meant to frame the lecture and class discussion, and alone will not provide the depth of knowledge required to successfully achieve the course objectives. Students are advised to take their own notes as necessary.

Other Readings

Readings will be from published research or industry online materials and available via Blackboard "Articles" or "Free Books and Chapters."

Equipment / Material Requirements

Students are required to have a 64-bit laptop/PC running Windows, MacOS, or Linux.

Students will also be required to set up an account on the Amazon Web Services cloud to install and configure a Hadoop environment. I will provide detailed instructions about this.

Students can try, on their own, to install the Hadoop sandbox on their laptop/PC. Often this will not work well, especially on Windows PCs due to lack of RAM. For ease of support and consistency, all assignments will be described and tested against the Azure environment.

Grading Policy

Short Quizzes	15%
Assignments	15%
Project / Paper Proposal	5%
Project / Paper Draft	5%
Project / Paper	25%
Half Term Exam	35%

Each short quiz will be open notes and books (but closed Google Search) and consist of multiple choice and short answer questions and should take no more than 10-15 minutes to complete. The lowest short quiz score will be dropped when calculating the overall quiz score. Since you can miss a quiz with no penalty there are no make ups or exceptions (except as explicitly provided for by IIT policy or if negotiated ahead of time with me).

The half term exam will be open notes and books (again closed Google Search) and consist of multiple choice and short answer questions, some longer essay questions and take 90-120 minutes. Contact me well ahead of time for possible accommodations if you will be unable to attend on the day of the exam.

Assignments, project/paper proposals and project/paper drafts must be submitted by 6:45pm on their announced due dates (or whatever makes sense for your time zone). You have up to 3 grace days to apply across all of these without penalty. For example, one assignment can be up to 3 days late or three assignments can each be up to one day late. Beyond this grace period 5% will be deducted from your score for each day the assignment is late. But contact me well ahead of time for possible accommodations.

Without prior negotiated accommodations the final project or paper must be submitted by 11:59pm on its due date. Beyond this 10% will be deducted from your score for each day the assignment is late.

Assignments may be reviews of research papers or projects (with write ups) applying big data technology.

By the end of term you will have completed a paper exploring some topic in big data technology more deeply (ex. security, architecture, specific tools, etc.) or conducted an investigational project where you will have applied big data technology to a problem of interest to you, your community or an organization with which you are affiliated.

Grade Distribution

A = 100 – 90, B = 89.9999 – 75, C = 74 .9999 – 60 (Undergraduate only)

Students with Disabilities

Reasonable accommodations will be made for students with documented disabilities. In order to receive accommodations, students must obtain a letter of accommodation from the Center for Disability Resources. The Center for Disability Resources (CDR) is located in 3424 S. State St., room 1C3-2 (on the first floor), telephone 312 567.5744 or disabilities@iit.edu.

Course Syllabus

For guidance, some order of the following may change a bit.

Module	Description	Details
1a	• Course introduction • What is big data?	• What is Big Data? • Characteristics of Big Data • Understanding Big Data with Examples • Big Data Processing Pipeline • Big Data Trends • The Risks of Big Data
1b	• Distributed systems concepts • Fault tolerance in distributed systems	
2a	• Hadoop Basics	• Describe the case for Hadoop • Identify the Hadoop Ecosystem architecture

		• Data Management - HDFS, YARN • Data Processing: MapReduce, Spark • Data Access - Pig, Hive, HBase, Storm, Solr • Data Governance & Integration - Falcon, Flume, Sqoop, Kafka, Atlas • Security - Kerberos, Knox • Operations - Ambari, Zookeeper, Oozie
2b	Hadoop Distributed File System	• Concepts • Operations • Command Line Interface • Java Interface • Fault tolerance
3	YARN MapReduce I	• YARN Concepts • YARN Architecture • Introduction to Developing Hadoop Applications • Illustrate the MapReduce model conceptually • Discuss how MapReduce works at a high level • Define how data flows in MapReduce • Job Execution Framework • Describe how jobs execute in YARN • Describe how to manage jobs in YARN • Write a MapReduce Program • Design and implement the Mapper class, Reducer class and driver • Build and execute the code then examine the output • Describe data set for programming problem • Use the MapReduce API • API overview • Mapper input processing and Reducer output processing data flow • Explore the Mapper, Reducer and Job class API • Managing, monitoring, and testing MapReduce jobs • Work with counters • Display job history and logs • Write unit tests for MapReduce programs • Characterizing and improving MapReduce job performance • Enhance performance in your MapReduce jobs • Working with different data sources in MapReduce • Fault tolerance
4	MapReduce II	
5a	Hive I	• Hive Basics • Hive Architecture • Hive Query Language • Loading and exporting data • Use cases of Hive • Steps in the data pipeline
5b	Hive II	

		• Create and Load Data • Create databases, internal tables, external tables, and partitioned tables • Learn about data types • Load data into tables and databases • Query and Manipulate Data • Query, sort, and filter data • Hive Operators and Functions • Hive Storage Formats
6a	Pig	• Pig Basics • Pig Latin • Data Processing Operators • Scripting with Pig
Project or Paper Proposal Due		
6b	Spark I (Spark Basics)	• Spark basics • Using the Spark shell for interactive data analysis • The features of Spark's Resilient Distributed Datasets • Writing and Deploying Spark Applications • Common Patterns in Spark Data Processing • Fault tolerance
7	Spark II (Spark Advanced)	
8	Spark III (Spark SQL)	• Spark SQL and the SQL Context • SchemaRDD basics • DataFrames basics • Transforming and Querying DataFrames • Loading and Saving Data • Use with Hive
9	**Mid Term Exam**	•
10a	Big Data Reference Architecture	• Data Warehouse and Data Mart Models • Lambda Architecture • Critiques and Alternatives
10b	Kafka	• Basics • Producers—Writing Messages • Consumers—Reading Messages • Architecture • Fault Tolerance
11	Spark IV (Spark Streaming)	• Basics • Transformations • Inputs • Outputs • Fault Tolerance
12a	NoSQL Database Landscape	• Motivations for NoSQL Databases • ACID and BASE ○ ACID: Atomicity, Consistency, Isolation, and Durability

		○ BASE: Basically Available, Soft State, Eventually Consistent ○ Types of Eventual Consistency • Four Types of NoSQL Databases ○ Key-Value Pair Databases ○ Document Databases ○ Column Family Databases ○ Graph Databases
12b	CAP Theorem	• Basics • Critique • Other ways to characterize NoSQL databases
13a	Key Value Databases	• Essential Features of Key-Value Databases • Key-Value Database Terminology • Key-Value Architecture Terms • Keys: More Than Meaningless Identifiers • Values: Storing Just About Any Data You Want • Designing for Key-Value Databases • Limitations of Key-Value Databases • Design Patterns for Key-Value Databases
13b	Wide Column Databases	• Introduction to Wide Column Databases • Wide Column Database Terminology • Designing for Wide Column Databases • Usage Patterns • MapReduce Integration
Project or Paper Draft Due		
14	Document Databases (MongoDB)	• JSON • What Is a Document • Managing Multiple Documents in Collections • Avoid Explicit Schema Definitions • Basic Operations on Document Databases • MongoDB overview • Aggregation Framework • Designing for Document Databases • One-to-Many Relations • Many-to-Many Relations • The Need for Joins • Modeling Common Relations • One-to-Many Relations in Document Databases • Many-to-Many Relations in Document Databases • Modeling Hierarchies in Document Databases
15	Document Databases III (MongoDB)	
Final Project or Paper Due		