CS 5433: Big Data Management

***Time*:** 4.30pm – 7.10pm Wednesday

Instructor*:*
Dr J. P. Thomas
email: jpt@cs.okstate.edu.
Office Hours: Wednesday: 11.00am-12.15pm
Office: 201 MSCS

Teaching Assistant:
Kun Chen
email:kuchen@ostatemail.okstate.edu
Office Hours: Thursday 2:00pm - 3.30pm.
Office: MSCS 313
Alternate times to meet on an individual basis can be arranged. Contact the instructor or TA by email.

***Prerequisite*:**
CS 5423, CS 4283 or CS 5283 or equivalents
Knowledge of Programming

***Course Description*:**
Data storage and processing for big data, Map-Reduce model for big data processing within the Hadoop software framework, big data warehouse for summarization, query and analysis using Hive, Data munging and transformation using Pig; Streaming data using Flume; Transferring structured data using Sqoop; Loading, Querying and transforming data using HBase; Setting up distributed services using ZooKeeper; Data Processing using Apache Spark; Writing Data Pipeline jobs for Hadoop; Machine learning in Spark and R.

Course Objectives
This course provides an introduction to Distributed data processing using the Hadoop Framework and Apache Spark. In the Hadoop framework, the Hadoop Distributed File System (HDFS) for storage and the MapReduce programming model for processing data will be explored in detail. Writing MapReduce programs, Streaming/Pipes API will be covered. In Apache Spark RDD transformations, real-time stream processing will be explored in detail. The second area covers a wide range of tools that are a part of Hadoop Ecosystem. Some of the tools like Pig, Sqoop, Flume, ZooKeeper, HBase, and Kafka. Introduction to Data analytics. Data analytics implementation in R and Spark will be studied. The final topic to be covered will be distributed analysis and patterns for big data processing

Course Outline

Topics to be covered

A. Parallel and Distributed Data Processing Frameworks
    a. Hadoop
        i. Hadoop File Systems (HDFS)
        ii. MapReduce Programming Model
        iii. Joins in MapReduce Jobs
        v. Streaming/Pipes API
        vi. MapReduce Features
    b. Spark
        i. Salient Features of Spark
        ii. RDD
        iii. Anatomy of Spark Jobs
        iv. Spark on YARN
        v. Stream-Processing
        vi. Real-time Querying

B. Big Data EcoSystem
    a. Hive for data summarization, query, and analysis
    b. Analyzing large data sets using Apache Pig
    c. Streaming data using Apache Flume
    d. Importing/Exporting Structural data using Apache Sqoop
    e. Writing distributed services using Apache ZooKeeper
    f. ETL operations using HBase
    g. Apache Hive™ data warehouse software facilitates and quering using SQL syntax.
    h. Distributed real-time streaming with Kafka

C. Big Data Analytics using MLlib and R
    a. Basic Data Analytic Methods
    b. Advanced Analytical theory and methods:
        i. Clustering
        ii. Association Rules
        iii. Regression
        iv. Classification
        v. Naïve Bayes

D. Distributed Analysis and Patterns for Big Data Processing
    a. Computing with Keys
    b. Keyspace Patterns
    c. Design Patterns
    d. Summarization
    e. Indexing
    f. Filtering
    g. Toward Last-Mile Analytics
    h. Fitting a Model and Validating Models

Textbooks

There is no textbook for this class. The following serve as reference texts:

1. Tom White, *Hadoop: The Definitive Guide,* 4th Edition, O'Reilly, 2014
2. Edward Capriolo, Dean Wampler, Jason Rutherglen, *Programming Hive*: *Data Warehouse and Query Language for Hadoop,* O'Reilly, 2012
3. Alan Gates, *Programming Pig,* O'Reilly, 2011
4. Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharia, *Learning Spark,* O'Reilly, 2014
5. EMC Education Services, *Data Science and Big Data Analytics,* Wiley, 2015
6. Benjamin Bengfort & Jenny Kim*, Data Analytics with Hadoop - An Introduction for Data Scientists*, O'Reilly, 2016
7. Mark Grover, Ted Malaska, Jonathan Seidman & Gwen Shapira, *Hadoop Application Architectures: Designing real-world big data applications,* O'Reilly, 2015

Communication medium

All notes and class announcements will be on Desire2Learn (D2L)/Brightspace

***Grading*:**

- 2 Quizzes = 50
- Individual Programming Assignments = 150
- Group Project = 100
- Finals = 75

Total – 375 marks

***Letter Grades*:**

Grade A: 90 - 100 %
Grade B: 80 – 89 %
Grade C: 70 –79 %
Grade D: 60 - 69 %
Fail (Grade F): 0-59 %

Attendance Policy:

Attendance is strongly encouraged, but not required. Students are responsible for any material covered in class. Some of the material covered in class will not be in the required textbook. Announcements about tests etc. will be made in class and/or by email. Students are to check their emails regularly (using their class accounts).

Late submission penalty:
1 calendar day late: 10% penalty - date based on submission
2 calendar days late: 20% penalty - date based on submission
3 calendar days late: 30% penalty - date based on submission
4 calendar days late: 40% penalty - date based on submission
5 calendar days late: 50% penalty - date based on submission
6 or more calendar days late: 100% penalty - date based on submission

***Examinations/Tests*:** No discussion of any kind (except with the instructor) is allowed. No access to any type of written material is allowed. Students who **do not** comply with the described collaboration policy will receive a grade of F in the course. Furthermore, the case will be reported to the University Officials.

Drop and Add Policy: Students will be allowed to drop as long as the University permits them to do so. A grade of W or F will be determined on the basis of the points earned until that time.

Academic Dishonesty/misconduct: The Computer Science departmental policy for academic dishonesty and misconduct applies to this class. In addition, a student attempting to gain unfair advantage by keeping an examination paper longer than the time permitted is guilty of academic misconduct.

Computer Usage: The Computer Science departmental policy for computer usage applies to this class. Exceptions will be made for students whose companies permit use of company machines for academic work. Students taking advantage of the exception must have two-way email access.

Americans with disabilities act: The Computer Science departmental policy for students with disabilities applies to this class. Anyone who has a need for examinations by special arrangements should see the instructor as the earliest possible opportunity during scheduled office hours.