CS167
Introduction to Big-data
Instructor: Ahmed Eldawy
Welcome to UCR! (Virtually)
Class information

• Classes: Tuesday, Thursday 2:00 – 3:20 PM via Zoom
• Instructor: Ahmed Eldawy
• Office hours: Monday, Thursday 11:00-11:50 (Conflicts?)
• TA: Payas Rajan and Xin Zhang
• Website: http://www.cs.ucr.edu/~eldawy/21SCS167/
• Email: eldawy@ucr.edu Subject: “[CS167] …”
• Slack workspace https://join.slack.com/t/cs167s21/shared_invite/zt-odmd6bxu-x4rLDpFmDIXoRRuRv3iuIa
Class Logistics

• All classes will be recorded and published after class
• Ask questions in the chat window
• Raise your hand (in Zoom) if you have a question that you would like to ask verbally
Lab Logistics

- All labs will be on Zoom
- Attend the session that you are enrolled in
- The TA will share their screen
- Students will follow the instructions on their machines
- Ask questions in the chat
- If you have a question, you can share your screen (privately) with the TA to get help!!
Course work

• Active participation (10%)
• Assignments (15%) (3% x 5)
• Labs (30%) (3% x 10)
• Mid-terms (15% x 3)

• All exams will be open slides, notes, and book.
Textbook

- No required textbook
- Recommended textbooks


   ISBN-10: 1491912219
Presurvey Results

Assignment submission

- iLearn: 38%
- Gradescope: 60%
- eLearn: 2%

Online conversation system

- Slack: 42%
- Discord: 31%
- Piazza: 21%
- Other: 6%

OS

- Windows: 57%
- Mac: 31%
- Linux: 12%

- Windows
- Mac
- Linux
- Slack
- Discord
- Piazza
- Other
Background

Java

Scala

Javascript

SQL

Linux command-line tools

JSON data format

Java

52 responses

Scala

52 responses

Javascript

52 responses

SQL

52 responses

Linux command-line

52 responses

JSON

52 responses
Excitements/Concerns

• Excited
  ▪ Play with large amounts of data
  ▪ Not sure/Exploring
  ▪ Distributed frameworks
  ▪ Learn about machine learning
  ▪ Search/Sort big-data
  ▪ Internals of big-data systems
  ▪ Move from DBMS to big-data
  ▪ Learn new technology

• Concerns
  ▪ Steep learning curve
  ▪ Extreme workload
  ▪ Not having the necessary background
  ▪ Not being the right course for me
  ▪ Online teaching
Course goals

• What are your goals?

• Understand what big data means
• Identify the internal components of big data platforms
• Recognize the differences between different big data platforms
• Explain how a distributed query runs on big data
Superhero
Ant-Man/Wasp

Get smaller to understand how ants work and what they are capable of.

Use this knowledge to control thousands of ants and do amazing things!
Big-data Expert

• Understand how the big-data platforms really work
• Control those thousands of processors efficiently to carry out your task
Syllabus

- Overview of big data
- Big-data storage
- Big-data processing
- Structured data processing
- Column-based storage and retrieval
- Big-spatial data
- Document databases
- Machine learning on big data
- Big-data visualization
Introduction
Big Data
Straight Ahead
All of the information

Information you need!
Interest in Big Data in the US

March 2012: Obama administration unveils BIG DATA initiative: $200 Million in R&D investment

June 2013: Washington Post is calling Obama “The Big Data President”
Interest in Big Data in Europe

- **March 2014:** David Cameron and Angela Merkel talking about Big Data in a Computer Expo in Hannover, Germany.
The creation and consumption of data continues to grow by leaps and bounds and with it the investment in big data technology and analytics. For 2017, Forbes compiled a list of predictions for the $203 billion big data analytics market.
Job Market

Big Data Market Worldwide Segment Revenue Forecast 2011-2026

Big Data Market Forecast Worldwide from 2011 to 2026, by segment (in billion U.S. dollars)

Three Four V’s of Big Data

Volume

- Scale of Data
- 40 zettabytes (40 trillion gigabytes) of data will be created by 2020, an increase of 300 times from 2005
- 6 billion people have cell phones
- World population: 7 billion

The New York Stock Exchange captures 1 to 2 billion trades during each trading session

Velocity

- Analysis of streaming data
- Modern cars have close to 100 sensors that monitor items such as fuel level and tire pressure
- By 2016, it is projected there will be 18.9 billion network connections, almost 2.5 connections per person on earth

Variety

- Different forms of data
- 4 billion+ hours of video are watched on YouTube each month
- 30 billion pieces of content are shared on Facebook every month
- 400 million tweets are sent per day by about 200 million monthly active users

Veracity

- Uncertainty of data
- 27% of respondents in one survey were unsure of how much of their data was inaccurate

The FOUR V’s of Big Data

- From traffic patterns and music downloads to web histories and medical records, data is recorded, stored, and analyzed to enable the technology and services that the world relies on every day. But what exactly is big data, and how can these massive amounts of data be used?

- As a leader in the sector, IBM data scientists break big data into four dimensions: Volume, Velocity, Variety and Veracity

- Depending on the industry and organization, big data encompasses information from multiple internal and external sources such as transactions, social media, enterprise content, sensors and mobile devices. Companies can leverage data to adapt their products and services to better meet customer needs, optimize operations and infrastructure, and find new sources of revenue.

- By 2015, 4.4 million IT jobs will be created globally to support big data, with 1.5 million in the United States

- As of 2011, the global size of costs in healthcare was estimated to be 150 exabytes (1.6 trillion gigabytes)

- By 2014, it’s anticipated there will be 420 million wearable, wireless health monitors

Sources: McKinsey Global Institute, Twitter, Cisco, Gartner, IDC, SAS, IBM, NCPC, GAO
Big Data Vs Big Computation

• Full scans (e.g., log processing)
• Range scans
• Point lookups
• Iterations
• Joins (self, binary, or multiway)
• Proximity queries
• Closures and graph traversals
Big Data Applications

- Web search
- Marketing and advertising
- Data cleaning
- Knowledge base
- Information retrieval
- Internet of Things (IoT)
- Visualization
- Behavioral studies
Publicly Available Datasets

• Data.gov
• Data.gov.uk
• UCR STAR [https://star.cs.ucr.edu]
• Twitter Streaming API
• GDELT [http://www.gdeltproject.org/]
• Kaggle.com
Big Data Landscape 2014

Big Data Landscape 2016

Big Data Landscape 2016 (Version 3.0)

Infrastructure
- Hadoop
- Hadoop in the Cloud
- Spark
- Data Storage
- Data Integration

Analytics
- BI Platforms
- BI/Analytics
- Log Analytics
- Machine Learning

Applications
- Customer Service
- Security
- Open Source

Last Updated 3/23/2016
© Matt Turck (@mattturck), Jim Hao (@jimhao), & FirstMark Capital (@firstmarkcap)

http://mattturck.com/2016/02/01/big-data-landscape/
Big Data Landscape 2018
Components of Big Data
Components of Big Data

**Big-data Libraries**
MLlib (Machine Learning), GraphX, Visualization

**High-level Languages**
SparkSQL, Pig, SQL++, HiveQL

**Distributed Computing**
MapReduce (Hadoop and Google), Resilient Distributed Dataset (Spark), Hyracks (AsterixDB)

**Big Data Distributed Storage**
Hadoop Distributed File System, Cloud storage systems (Amazon S3 and Google File System), Key-value stores

**Cloud Services**
Amazon Web Services, Microsoft Azure, and Google Cloud Platform

**Coordination/Cluster Management**
Oozie, Yarn, Kubernetes
Storage of Big Data

• Data is growing faster than Moore’s Law
• Too much data to fit on a single machine
• Partitioning
• Replication
• Fault-tolerance
Hadoop Distributed File System (HDFS)

- The most widely used distributed file system
- Fixed-sized partitioning
- 3-way replication
- Write-once read-many
- See also: GFS, Amazon S3, Azure Blob Store
Indexing

- Data-aware organization
- Global Index **partitions** the records into blocks
- Local Indexes organize the records in a partition
- Challenges:
  - Big volume
  - HDFS limitation
  - New programming paradigms
  - Ad-hoc indexes
Fault Tolerance

• Replication

• Redundancy

• Multiple masters
# Key-value Stores

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Email</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jack</td>
<td><a href="mailto:jack@example.com">jack@example.com</a></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Jill</td>
<td><a href="mailto:jill@example.net">jill@example.net</a></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Alex</td>
<td><a href="mailto:alex@example.org">alex@example.org</a></td>
<td></td>
</tr>
</tbody>
</table>
Streaming

• Sub-second latency for queries
• One scan over the data
• (Partial) preprocessing
• Continuous queries
• Eviction strategies
• In-memory indexes
## Structured/Semi-structured

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Email</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jack</td>
<td><a href="mailto:jack@example.com">jack@example.com</a></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Jill</td>
<td><a href="mailto:jill@example.net">jill@example.net</a></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Alex</td>
<td><a href="mailto:alex@example.org">alex@example.org</a></td>
<td></td>
</tr>
</tbody>
</table>

**Document 1**

```json
{  "id": 1,  "name":"Jack",  "email": "jack@example.com",  "address": {"street": "900 university ave",  "city": "Riverside",  "state`: "CA"},  "friend_ids": [3, 55, 123]}
```

**Document 2**

```json
{  "id": 2,  "name": "Jill",  "email": "jill@example.net",  "hobbies": ["hiking", "cooking"]}
```
Traditional Distributed Computing

Centralized Big Data

Coordinator

Workers

Ship data to computation paradigm
e.g., High performance computing (HPC)
Big-data Computing

Send program and task information to where the data is

Coordinator

Storage/Compute Nodes

Ship compute to data paradigm
Task Execution

• MapReduce
  ▪ Map-Shuffle-Reduce
  ▪ Resiliency through materialization

• Resilient Distributed Datasets (RDD)
  ▪ Directed-Acyclic-Graph (DAG)
  ▪ In-memory processing
  ▪ Resiliency through lineages

• Hyracks
• Stragglers
• Load balance
Query Optimization

• Finding the most efficient query plan
• e.g., grouped aggregation

• Cost model (CPU – Disk – Network)
Provenance

- Debugging in distributed systems is painful
- We need to keep track of transformations on each record
Big Graphs

• Motivated by social networks
• Billions of nodes and trillions of edges
• Tens of thousands of insertions per second
• Complex queries with graph traversals
Structured Data Processing

• There is a need for processing structured and semi-structured data
• Let the big-data system know about the structure of the data and processing
• Allow the system to optimize query processing
• Examples: Algebricks, SparkSQL, and Pig
Hadoop Ecosystem

- Pig
- Apache Ambari
- Yet Another Resource Negotiator (YARN)
- MapReduce Query Engine
- Hadoop Distributed File System (HDFS)
Spark Ecosystem

- Spark SQL
- MLlib
- GraphX
- SparkR
- Spark Streaming

Data Frames

Resilient Distributed Dataset (RDD) a.k.a Spark Core

Yet Another Resource Negotiator (YARN)

Hadoop Distributed File System (HDFS)

Kubernetes
Hyracks Data-parallel Platform

AsteixDB
HiveSterix
Other compilers

Algebricks Algebra Layer
Hadoop MapReduce Compatibility
Pregelix

HiveQL
PigLatin

MapReduce Jobs
Pregel Jobs

Hyracks jobs
Impala

- Query Parser
- Query Planner
- Query Executor
- Yet Another Resource Negotiator (YARN)
- Hadoop Distributed File System (HDFS)
SpatialHadoop

Pig Latin + Pigeon

Spatial Visualization

MapReduce Processing + Spatial Query Processing

Yet Another Resource Negotiator (YARN)

Hadoop Distributed File System (HDFS) + Spatial Indexing