PART 0. INTRODUCTION: WHAT IS BIGDATA?

Instructor: Sangmi Lee Pallickara
Computer Science, Colorado State University
http://www.cs.colostate.edu/~cs435

Goal of this course
- Understanding fundamental concepts in Big Data
- Learn about the effective Big Data technologies and how to apply them
  - Computing paradigm
  - Data retrieval technology
  - Dataflow management for the large applications
  - Scalable algorithms

What is Big Data?
- Things one can do at a large scale that cannot be done at a smaller one
  - To extract new insights
  - Create new forms of values

- Big Data is about analytics of huge quantities of data in order to infer probabilities
  - Big Data is NOT about trying to “teach” a computer to “think” like humans
  - Providing a quantitative dimension it never had before

The three(or four) Vs in Big Data
- **Volume**
  - Voluminous
  - It does not have to be certain number of petabytes or quantity.

- **Velocity**
  - How fast the data is coming in?
  - How fast you need to be able to analyze and utilize it

- **Variety**
  - Number of sources or incoming vectors

- **Veracity**
  - Can you trust the data itself, source of the data, or the process?
  - User entry errors, redundancy, corruption of the values
  - Data cleaning
Use all of the data, or just a little? [1/5]

London, 17th century

- Estimate the population of London, which then used to estimate the number of eligible ‘fighting men’ available to the king
- Dispel the myth that the plague was more rife in years of monarchy changes
- Investigate how London was expanding

http://www.theactuary.com/archive/old-articles/part-3/who-was-captain-john-graunt-3F/

Use all of the data, or just a little? [2/5]

- Using a 1658 map he assumed that there were 54 families living in each space of 100 yards square
- Within the walls of London there were 220 such squares, making 11,880 families there
- The number of deaths in London as a whole, both within and outside the walls, was four times the number of deaths within the walls alone

http://www.theactuary.com/archive/old-articles/part-3/who-was-captain-john-graunt-3F/

Use all of the data, or just a little? [3/5]

- John Graunt’s population example

- Conducting censuses
  - Costly and time-consuming

- The US Constitution mandated one every decade
  - As the growing country measured itself in millions

Use all of the data, or just a little? [4/5]

- By the late nineteenth century, even that was proving problematic
  - Data outstripped the Census Bureau’s ability to keep up.

  - The 1880 census took 8 years to complete
  - The information was obsolete even before it became available!

  - The 1890 census was estimated to require a full 13 years

Use all of the data, or just a little? [5/5]

- The Census Bureau contracted with Herman Hollerith
  - An American inventor used his idea of punch cards and tabulation machines for the 1890 census
  - Reduced duration to less than one year

- Methods of acquiring and analyzing big data is very expensive

In 2009 a new flu virus was discovered.

Image of the newly identified H1N1 influenza virus
Source: CDC Influenza Laboratory
H1N1 virus

- Combines elements of the viruses that causes bird flu and swine flu
- Some projected outbreak to be on the scale of the 1918 Spanish flu that had:
  - Infected half a billion people
  - Killed tens of millions
- No vaccine was ready
- Public health authorities needed to know where it already was.

In the United States,

- The Centers for Disease Control and Prevention (CDC)
  - Requested doctors inform patients of new flu cases
  - Tabulated the numbers once a week
- Typically 1-2 week reporting lag
- What will be the problems with this system?

Detecting influenza epidemics using search engine query data -1/2

http://www.nature.com/nature/journal/v457/n7232/full/nature07634.html

Detecting influenza epidemics using search engine query data -2/2

- Predict the spread of the winter flu in the United States
  - Nationally
  - Specific regions and states
- Looking at what people were searching for on the Internet
  - Google
    - More than three billion search queries every day
    - Saves all the queries
    - 50 million most common search terms
    - Americans type and compared the list with CDC data
- On the spread of seasonal flu between 2003 and 2008
  - Identify areas infected by the flu virus by what people searched for on the Internet

How did they build a prediction model? (1/3)

- Aggregating historical logs of online web search queries
  - Between 2003 and 2008
  - Computed a time series of weekly counts for 50 million of the most common search queries in the US
  - Weekly counts were kept for every query in each state
- Simple model that estimates the probability that a random physician visit in a particular region is related to an influenza-like illness (ILI)

How did they build a prediction model? (2/3)

- The probability that a random search query submitted from the same region is ILI-related
  \[ \text{logit}(p(t)) = a \text{logit}(Q(t)) + \epsilon \]
  where:
  - \( p(t) \) is the percentage of ILI physician visits at time \( t \)
  - \( Q(t) \) is the ILI-related query fraction at time \( t \)
  - \( a \) is the multiplicative coefficient
  - \( \epsilon \) is the error term
  \[ \text{logit}(p) = \ln(p/(1 - p)) \]
How did they build a prediction model? (3/3)

- Each of the 50 million candidate queries was separately tested
- To identify the search queries that could most accurately model the CDC ILI visit percentage in each region
- Generated the highest scoring search queries

Evaluation of combining top-scoring queries

Maximal performance at estimating out-of-sample points during cross-validation was obtained by summing top 45 search queries

Most co-related queries

<table>
<thead>
<tr>
<th>Search query topic</th>
<th>Top 45 queries</th>
<th>Next 55 queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Influenza complication</td>
<td>11 18.15</td>
<td>5 3.40</td>
</tr>
<tr>
<td>Cold/flu remedy</td>
<td>8 5.05</td>
<td>6 5.03</td>
</tr>
<tr>
<td>General influenza symptoms</td>
<td>5 2.60</td>
<td>1 0.07</td>
</tr>
<tr>
<td>Term for influenza</td>
<td>4 3.74</td>
<td>6 0.30</td>
</tr>
<tr>
<td>Specific influenza symptom</td>
<td>4 2.54</td>
<td>6 3.74</td>
</tr>
<tr>
<td>Symptoms of an influenza complication</td>
<td>4 2.21</td>
<td>2 0.92</td>
</tr>
<tr>
<td>Antibiotic medication remedies</td>
<td>3 6.23</td>
<td>3 3.17</td>
</tr>
<tr>
<td>General influenza</td>
<td>2 0.18</td>
<td>1 0.32</td>
</tr>
<tr>
<td>Symptoms of a related disease</td>
<td>2 1.66</td>
<td>2 0.77</td>
</tr>
<tr>
<td>Antiviral medication</td>
<td>1 0.39</td>
<td>1 0.74</td>
</tr>
<tr>
<td>Related disease</td>
<td>1 6.66</td>
<td>3 3.77</td>
</tr>
<tr>
<td>Unrelated to influenza</td>
<td>0 0.00</td>
<td>19 28.37</td>
</tr>
<tr>
<td>Total</td>
<td>45 49.40</td>
<td>55 50.60</td>
</tr>
</tbody>
</table>

The top 45 queries were used in the final model; the next 55 queries are presented for comparison purposes. The number of queries in each topic is indicated, as well as query-volume-weighted counts, reflecting the relative frequency of queries in each topic.

Data sources

- Public dataset from the CDC’s Influenza Sentinel Provider Surveillance Network
- Average percentage of all outpatient visits to sentinel providers that were ILI-related
- Query submitted to the Google search engine
- 2003 – 2008

What makes this work innovative?

- There have been other approaches:
  - CDC’s weekly report
  - Call volume to telephone triage advice lines
  - Tracking over-the-counter drug sales
  - Online activity for influenza surveillance
    - Search queries submitted to a Swedish medical website
    - Tracking search keys (‘flu’, or ‘influenza’) in Yahoo search queries

- What is the innovation in Google’s approach?
Part 0. Introduction
Course Introduction

Related research areas

- Storage systems
  - How can we efficiently resolve queries on massive amounts of input data?
  - The input dataset may be presented in the form of a distributed data stream

- Machine learning
  - How can we efficiently solve large-scale machine learning problems?
  - The input data may be massive; stored in a distributed cluster of machines

- Distributed computing
  - How can we efficiently solve large-scale optimization problems in distributed computing environments?
  - For example, how can we efficiently solve large-scale combinatorial problems, e.g. processing of large scale graphs?

Related courses in CS at CSU

- Machine Learning/Statistics (CS480, CS545)
- Distributed Systems (CS455, CS555)
- Database Systems (Cs430, CS530)
- Computer Communications (CS457, CS557)
- Computer Security (Cs356, CS556)

About me

- Research area
  - Big Data
    - Storage, retrievals, analytics and visualization

- Projects
  - Glean
    - Predictive analysis at scale
  - Galileo
    - Distributed data storage system for large scale geospatial time-series datasets
  - Mendel
    - Genomic data storage system
  - GeoLens
    - Visual Analytics
  - Columbus
    - Exploratory visual analytics

- Undergraduate research
  - FleetFeet
    - Online sensor data acquisition and visual analytics

Communications (1/2)

- Course Website
  - http://www.cs.colostate.edu/~cs435
- Announcements: Check the course website at least twice a week.
- Schedule (course materials, readings, assignments)
- Policies

- Canvas
  - Assignment submission
  - Discussion board
  - Grades
Communications (2/2)
- Contact Instructor
  - sangmi@colostate.edu
  - Office hour
    - Tuesday 2:00 – 3:00PM
    - Friday 10:30 – 11:30AM and by appointment
  - Office: CSB456
  - URL: http://www.cs.colostate.edu/~sangmi
- Contact GTA
  - Saptashwa Mitra
  - Office hour: TBA

Course Structure
1. Lectures
2. Assignments
3. Quizzes and Exams
4. Term Projects
5. Help Sessions

Assignments (1/2)
- Providing hands-on experience
- Assignment 1: Set up your Hadoop cluster and Ngram (Unigram, bigram) generation and summarization of e-books using MapReduce
- Assignment 2: Ngram corpus analysis using Hadoop: Author detection using an NLP algorithm
- Assignment 3: Identifying the author/publishing year/genre using Ngram corpus analysis using AWS Elastic MapReduce Service (EMS) and HBase
  - Thank you for the grants, AWS!

Assignments (2/2)
- All assignments will be individual submissions
- Required programming language: Java
- Grading will be based on demo
- Late submission policy
  - Up to 2 days of late submission is accepted
  - 10% deduction per 24 hours will be applied
- Assignment submission
  - Canvas
Course Structure

1. Lectures
2. Assignments
3. Quizzes and Exams
4. Term Projects
5. Help Sessions

Quizzes and Exams (1/2)
- Pop quizzes
  - About 10 quizzes this semester
  - Simple questions
  - Each quiz is about \( \approx 1\% \) of the course grade
  - 2 lowest scores will be eliminated at the end of the semester
  - Open book, open note
  - No Internet, no collaboration

Quizzes and Exams (2/2)
- Exams
  - Midterm and Final
  - 40% of the course grade
  - Mixture of simple questions and comprehensive questions

Term Project
- Comprehensive Big Data software designing experiences
- Phase 0: Find your teammate!
- Phase 1: Term Project Proposal
- Phase 2: Final submission: Software and Report
- Phase 3: Presentation (Class presentation) and demonstration of your software
- Term project is a team effort 2-3 students per team
  - No single member team allowed
- http://www.cs.colostate.edu/~cs435/

How do I select my Term Project topic?

- Do you have your team mate?
  - Yes: Sit together!
  - No: Go through the example topics

- Did you find interesting topic?
  - Yes: Submit your team
  - No: Do you have team mate?
    - Yes: Go through the example topics
    - No: List your name on the Canvas discussion board with your strengths Topics that you are interested in
Example topics from previous courses

- Similarity and Clustering on The Million Song Dataset
- Methods Used to Analyze Eclipse and Mozilla Bug Data
- Movie Recommendation using Collaborative Filtering
- Climate Visualization and Predictive Analysis
- Wikipedia Page Traffic Statistics Analysis
- Trending Topics and Page Count Prediction on Wikipedia Traffic Log Data
- Supporting Emergency Response During Natural Disasters with Twitter Data

Help Sessions

- 4 lab sessions are planned
  - Installing and configuring Apache Hadoop
  - Regarding Assignment 1
  - Regarding Assignment 2
  - Regarding Assignment 3

- Additional lab sessions (up to 3 sessions) will be scheduled based on student requests
- Help sessions will be led by GTA and recorded

Course Structure

1. Lectures
2. Assignments
3. Quizzes and Exams
4. Term Projects
5. Help Sessions

Grading Policy

<table>
<thead>
<tr>
<th>Category</th>
<th>Points</th>
<th>Percentage of Final Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming Assignments</td>
<td>30</td>
<td>30%</td>
</tr>
<tr>
<td>(10% x 3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Term Project</td>
<td>20</td>
<td>20%</td>
</tr>
<tr>
<td>Quizzes and participation</td>
<td>10</td>
<td>10%</td>
</tr>
<tr>
<td>Midterm Exams (20% x 2)</td>
<td>40</td>
<td>40%</td>
</tr>
</tbody>
</table>

Term Project Grading Policy

- 20% of course grade

<table>
<thead>
<tr>
<th>Category</th>
<th>Points</th>
<th>Percentage of Final Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 0</td>
<td>1</td>
<td>1%</td>
</tr>
<tr>
<td>Phase 1</td>
<td>5</td>
<td>5%</td>
</tr>
<tr>
<td>Phase 2</td>
<td>8</td>
<td>8%</td>
</tr>
<tr>
<td>Phase 3</td>
<td>6</td>
<td>6%</td>
</tr>
<tr>
<td>Class presentation (3%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software demo (3%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Final Letter Grade
Letter grades will be based on the following standard breakpoints
• >=90 is an A,
• >=88 is an A-,
• >=86 is a B+,
• >=80 is a B,
• >=78 is a B-,
• >=76 is a C+,
• >=70 is a C,
• >=60 is a D,
• and <60 is an F

Course Timeline

Course Policy
• No make-up for missed exams
  - Except in extraordinary circumstances (e.g., major illness, family emergency)

• No make-up for missed quizzes
  - Except for the case where student provided an advance notice to the instructor based on an emergency

Plagiarism
• We use plagiarism detection software
• Please do not copy code or sentences from your references or your colleague's work
• Discuss with each other, but do not write your software together!

Course Policy
• No Cell-phones in the class.
• No Laptops during the exam, and quiz.
  - If you need to use a laptop, please sit in the back row.
  - I will ask you to turn off your laptop if needed.

• No text book.
  - I provide several on-line materials and research papers.
  - Rely on the lecture notes

More importantly,
• Attend the class, ask questions, and discuss
• Check the course web page and Canvas regularly
• Try new technologies and apply them
• Share your experiences with other students in class
Questions?