THE VALUE OF DATA

Soho, London, 1854

The mystery of cholera

The Hunt for the invisible cholera

Cholera: Galen’s miasma theory
John Snow: 1813–1858

The Survey of Soho

Data collection

What the data showed ...

What the data showed ...

616 deaths, 8 days later ...
Cholera notice ca. 1866

Thirty years before discovery of V. cholerae

John Snow: Father of Epidemiology

Epidemiology’s Success Stories

Value of data: Not just epidemiology

(Paper) notebooks no longer good enough
THE GROWTH OF DATA

Wikipedia
≈ 5.9 TB of data
(Jan. 2010 Dump)
1 Wiki = 1 Wikipedia

US Library of Congress
≈ 235 TB archived
≈ 40 Wiki

Sloan Digital Sky Survey
≈ 200 GB/day
≈ 73 TB/year
≈ 12 Wiki/year

NASA Center for Climate Simulation
≈ 32 PB archived
≈ 5,614 Wiki

Facebook
≈ 100 TB/day added
≈ 17 Wiki/day
≈ 6,186 Wiki/year
(as of Mar. 2010)
Large Hadron Collider
≈ 15 PB/year
≈ 2,542 Wikipedias/year

Google
≈ 20 PB/day processed
≈ 3,389 Wiki/day
≈ 7,300,000 Wiki/year
(Jan. 2010)

Internet (2016)
≈ 1.3 ZB/year
≈ 220,338,983 Wiki/year
(2016 IP traffic, Cisco est.)

“There were 5 exabytes of data online in 2002, which had risen to 281 exabytes in 2009. That’s a growth rate of 56 times over seven years.”
-- Google VP Marissa Mayer

Data: A Modern-day Bottleneck?
← Rate at which data are produced

← Rate at which data can be understood

• A buzz-word: no precise definition?
• Data that are too big to process by “conventional means”
• A call for Computer Scientists to produce new techniques to crunch even more data
• Storage, processing, querying, analytics, data mining, applications, visualisations …
How many V’s in “Big Data”?

• Three V’s:
  — Volume (large amounts of data)
  — Velocity (rapidly changing data)
  — Variety (different data sources and formats)

• Maybe more (Value, Veracity)

“BIG DATA” IN ACTION …

Social Media

What’s happening here? (Trendsmap)

“What are the hot topics of discussion in an area”

• Analyse tags of geographical tweets

What’s the fastest route? (Waze)

“What’s the fastest route to get home right now?”

• Processes real journeys to build background knowledge
• “Participatory Sensing”

Christmas Predictions for Stores

“What will be the hot items to stock up on this Christmas? We don’t want to sell out!”

• Analyse product hype on Twitter, Search Engines and Social Networks
• Analyse transaction histories
Get Elected President (Narwhal)

“Who are the undecided voters and how can I convince them to vote for me?”

- User profiles built and integrated from online sources
- Targeted emails sent to voters based on profile

Predicting Pre-crime (PredPol)

“What areas of the city are most need of police patrol at 13:55 on Mondays?”

- PredPol system used by Santa Cruz (US) police to target patrols
- Predictions based on analysis of 8 years of historical crime data
- Minority Report!

IBM Watson: Jeopardy Winner

“William Wilkinson’s “An Account of the Principalities of Wallachia and Moldavia” inspired this author’s most famous novel.”

- Indexed 200 million pages of structured and unstructured content
- An ensemble of 100 techniques simulating AI-like behaviour

“BIG DATA” NEEDS “MASSIVE DATA PROCESSING” ...

Every Application is Different ...

- **Data** can be
  - Structured data (*JSON, XML, CSV, Relational Databases, HTML form data*)
  - Unstructured data (*text document, comments, tweets*)
  - And everything in-between!
  - Often a mix!

Every Application is Different ...

- **Processing** can involve:
  - Natural Language Processing (*sentiment analysis, topic extraction, entity recognition, etc.*)
  - Machine Learning and Statistics (*pattern recognition, classification, event detection, regression analysis, etc.*)
  - Even inference! (*Datalog, constraint checking, etc.*)
  - And everything in-between!
  - Often a mix!
Scale is a Common Factor ...

- Cannot run expensive algorithms

I have an algorithm.
I have a machine that can process 1,000 input items in an hour.

If I buy a machine that is $n$ times as powerful, how many input items can I process then?

Depends on algorithm complexity of course!

Note: Not the same machine!

Scale is a Common Factor ...

- One machine that’s $n$ times as powerful?  
  vs.

- $n$ machines that are equally as powerful?

Scale is a Common Factor ...

- Data-intensive (our focus!)
  - Inexpensive algorithms / Large inputs
    - e.g., Google, Facebook, Twitter

- Compute-intensive (not our focus!)
  - More expensive algorithms / Smaller inputs
    - e.g., climate simulations, chess games, combinatorials

- No black and white!

“MASSIVE DATA PROCESSING” NEEDS “DISTRIBUTED COMPUTING” ...

Distributed Computing

- Need more than one machine!

  - Google ca. 1998:

  - Google ca. 2014:
**Data Transport Costs**

- Need to divide tasks over many machines
  - Machines need to communicate
    - ... but not too much!
  - Data transport costs (simplified):

![Diagram showing Main Memory, Solid-state Disk, Hard-disk, and Network with need to minimise network costs]

**Data Placement**

- Need to think carefully about where to put what data!

```
I have four machines to run my website. I have 10 million users.
Each user has personal profile data, photos, friends and games.
How should I split the data up over the machines?
Depends on application of course!
(But good design principles apply universally!)
```

**Network/Node Failures**

- Need to think about failures!

```
Lot of machines: likely one will break!
```

**Network/Node Failures**

- Need to think (even more!) carefully about where to put what data!

```
I have four machines to run my website. I have 10 million users.
Each user has a personal profile, photos, friends and apps.
How should I split the data up over the machines?
Depends on application of course!
(But good design principles apply universally!)
```

**Human Distributed Computation**

```
Similar Principles!
```

**“DISTRIBUTED COMPUTING” LIMITS & CHALLENGES …**
Distribution Not Always Applicable!

Distributed Development Difficult

- Distributed systems can be complex
- Tasks take a long time!
  - Bugs may not become apparent for hours
  - Lots of data = lots of counter-examples
  - Need to balance load!
- Multiple machines to take care of
  - Data in different locations
  - Logs and messages in different places
  - Need to handle failures!

Frameworks/Abstractions can Help

- For Distrib. Processing
- For Distrib. Storage

Based on 2013 slides by Twitter lead architect: Raffi Krikorian

"Twitter Timelines at Scale"

Big Data at Twitter

- 150 million active worldwide users
- 400 million tweets per day
  - 4,600 tweets per second
  - max: 143,199 tweets per second
- 300 thousand queries/sec for user timelines
- 6 thousand queries/sec for custom search

What should be the priority for optimisation?
Supporting timelines: write
- 300 thousand queries per second

Supporting timelines: read
- 300 thousand queries per second

High-fanout
- @ladygaga • 31 million followers
- @katyperry • 28 million followers
- @justinbieber • 28 million followers
- @barackobama • 23 million followers

Supporting text search
- Information retrieval
  - Earlybird: Lucene clone
  - Write once
  - Query many

Timeline vs. Search
- 4,600 requests/sec -> O(n) write
- 300,000 requests/sec -> O(1) read
- 6,000 requests/sec - 4,600 requests/sec
- 4,600 requests/sec -> O(1) write
- 6,000 requests/sec - 4,600 requests/sec

Twitter: Full Architecture
Big Data at Twitter

- 150 million active worldwide users
- 400 million tweets per day
  - 4,600 tweets per second
  - max: 143,199 tweets per second
- 300 thousand queries/sec for user timelines
- 6 thousand queries/sec for custom search

What the Course Is/Is Not

- Data-intensive not Compute-intensive
- Distributed tasks not networking
- Commodity hardware not big supercomputers
- General methods not specific algorithms
- Practical methods with a little theory

What the Course Is!

- Principles of Distributed Computing [3 weeks]
- Distributed Processing Frameworks [4 weeks]
- Principles of Distributed Databases [3 weeks]
- Information Retrieval [3 weeks]

Course Structure

- ~1.5 hours of lectures per week [Monday]
- 1.5 hours of labs per week [Wednesday]
  - To be turned in by Friday evening
  - Mostly Java
  - In Laboratorio 1 (Cuarto Piso, DCC)

http://aidanhogan.com/teaching/cc5212-1/

Course Marking

- 45% for Weekly Labs (~3% a lab!)
- 35% for Final Exam
- 20% for Small Class Project
Outcomes!

Outcomes!

Outcomes!

Outcomes!

Outcomes!

Outcomes!
Questions?