CMPT 733
Data Preparation

SLIDES BY:

JIANNAN WANG

https://www.cs.sfu.ca/~jnwang/

Data Preparation is the Bottleneck!

Doing data science is like cooking

Collection

Cleaning

Integration

Analysis

How much time will be spent on the preparation?

Outline

Data Collection

Data Cleaning

Data Integration

Outline

Data Collection

- Where to collect
- How to collect

Data Cleaning

Data Integration

Where to Collect?

Internal Data

- Application Database (Tabular Data)
- System Logs (Text Files)
- Documents (Word, Excel, PDF)
- Multimedia Data (Video, Audio, Image)

Where to Collect?

External Data

- Web Pages
- Web Service (https://www.programmableweb.com/)
- Open Data (data.vancouver.ca, www.data.gov)
- Knowledge Base (Wikidata, Freebase)

Data Classification

Data

Structured

Team		W	L	Pct	GB	Conf	Div	Home	Away	L10	Strk
1	Rockets	20	4	.833	-	11-2	3-2	8-3	12-1	9-1	W9
2	Warriors	21	6	.778	0.5	9-4	2-1	8-3	13-3	8-2	W6
3	Spurs	19	8	.704	2.5	9-4	4-1	13-2	6-6	8-2	W4
4	Timberwolves	16	11	.593	5.5	13-5	4-1	9-4	7-7	6-4	W2

Semi-structured

CLE - James Layup Shot: Missed — 06:48 CLE

CLE - James Rebound (Off:1 Def:0) — 06:46 CLE

CLE - James Reverse Layup Shot: Made (2 PTS) — 06:45 CLE 9-15

Stoppage: Out-of-Bounds — 06:29

Unstructured

Is LeBron breaking the aging curve?

Kevin Pelton
ESPN Staff Writer
5:10 AM PT

During his 15th NBA season, Cleveland Cavaliers star LeBron James is performing at a level that echoes the prime that saw him win four MVPs.

As James nears his 33rd birthday later this month, his performance at that age stands up to any of his predecessors, including Michael Jordan's 1995-96 season that produced an MVP, a then-record 72 wins and a championship. (Because James entered the NBA directly out of high school, NBA experience isn't the best way to compare how he's aging to his peers. After all, Jordan's 15th year was actually his final one in the NBA at age 40.)

Challenges

Data Discovery

- How to find related data?

> - Domain knowledge
> - Information retrieval skills

Data Privacy

- How to protect user privacy?

> - Data masking
> - Differential privacy

Security

- How to avoid a data breach?

> - Follow data access rules
> - Encrypt highly confidential data

Getting Data

From CSV Files

From JSON Files

From the Web

From HDFS

From Databases

From S3

From Web APIs

Load Data From CSV Files

CSV is a file format for storing tabular data

rankings.csv

```
Team,Win,Loss,Win%
Houston Rockets,20,4,0.833
Golden State Warriors,21,6,0.778
San Antonio Spurs,19,8,0.704
Minnesota Timberwolves,16,11,0.593
Denver Nuggets,14,12,0.538
Portland Trail Blazers,13,12,0.52
New Orleans Pelicans,14,13,0.519
Utah Jazz,13,14,0.481
```

Reading CSV File (pandas library)

```python
import pandas as pd

df = pd.read_csv('rankings.csv')
```

Load Data From JSON Files

JSON is a file format for storing nested data (array, dict)

players.json

```json
{
    "Kobe Bryant" :{
            "Born": "08/23/1978",
            "Number": ["8", "24"],
            "Team": ["Los Angeles Lakers"]
    },
    "Michael Jordan":{
            "Born": "02/17/1963",
            "Number": ["23"],
            "Team": ["Chicago Bulls", "Washington Wizards"]
    }
}
```

Reading JSON File (pandas Libaray)

```python
import pandas as pd
df=pd.read_json("players.json")
```

Web Scraping

Open web pages

- urllib2 (https://docs.python.org/2/library/urllib2.html)
- request (http://docs.python-requests.org/en/master/)

Parse web pages

- Beautiful Soup (https://www.crummy.com/software/BeautifulSoup/)
- lxml (http://lxml.de/)

Putting everything together

- Scrapy (https://scrapy.org/)

Before you scrape

Check to see if CSV, JSON, or XML version of an HTML page are available – better to use those

Check to see if there is a Python library that provides structured access (e.g., tweetPy)

Check that you have permission to scrape

From "Deb Nolan. Web Scraping & XML/Xpath"

If you do scrape

Be careful to not to overburden the site with your requests

Test code on small requests

Save the results of each request so you don't have to repeat the request unnecessarily

From "Deb Nolan. Web Scraping & XML/Xpath"

Outline

Data Collection

Data Cleaning

- Dirty Data Problems
- Data Cleaning Tools
- Example: Outlier Detection

Data Integration

Dirty Data Problems

From Stanford Course:

1) Parsing text into fields (separator issues)
2) Missing required field (e.g. key field)
3) Different representations (iphone 2 vs iphone 2nd generation)
4) Fields too long (get truncated)
5) Formatting issues – especially dates
6) Outliers (age = 120)

Data Cleaning Tools

Python

- Missing Data (Pandas)
- Deduplication (Dedup)

OpenRefine

- Open-source Software (http://openrefine.org)
- OpenRefine as a Service (RefinePro)

Data Wrangler

- The Stanford/Berkeley Wrangler research project
- Commercialized (Trifacta)

Outlier Detection

The ages of employees in a US company

1	20	21	21	22	26	33	35	36	37	39	42	45	47	54	57	61	62

Mean = $\frac{1}{n}\sum_{i=1}^{n} x_i = 37$

Stddev = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \text{mean})^2} = 16$

$[37 - 2 * 16, 37 + 2 * 16]$ = $[4, 70]$

Outlier Detection

The ages of employees in a US company

1	20	21	21	22	26	33	35	36	37	39	42	45	47	54	57	61	62	400

Mean = $\frac{1}{n}\sum_{i=1}^{n} x_i = 56$

$[56 - 2 * 83, \ \ 56 + 2 * 83]$ = [-109, 221]

Stddev = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \text{mean})^2} = 83$

Outlier Detection

The ages of employees in a US company

1	20	21	21	22	26	33	35	36	37	39	42	45	47	54	57	61	62	400

Median = $\text{median}(x_i) = 37$

$[37 - 2 * 15,\ \ 37 + 2 * 15]$ = $[7, 67]$

MAD =$\text{median}(|x_i - \text{median}(x_i)| = 15$

Outline

Data Collection

Data Cleaning

Data Integration

- Data Integration Problem
- Three Steps (Schema Matching, Entity Resolution, Data Fusion)
- Example: Entity Resolution

Data Integration Problem

Data Source 1 (from CourSys)

First Name	Last Name	Mark
Michael	Jordan	50
Kobe	Bryant	48

Data Source 2 (from survey)

Name	Background
Mike Jordan	C++, CS, 4 years
Kobe Bryant	Business, 2 years

Data Integration???

Integrated Data

Name	Mark	Background
Michael Jordan	50	C++, CS, 4 years
Kobe Bryant	48	Business, 2 years

Data Integration: Three Steps

Schema Mapping
- Creating a global schema
- Mapping local schemas to the global schema

Entity Resolution
- You will learn this in detail later

Data Fusion
- Resolving conflicts based on some confidence scores

Want to know more?
- Anhai Doan, Alon Y. Halevy, Zachary Ives. Principles of Data Integration. Morgan Kaufmann Publishers, 2012.

Entity Resolution

Apple iPad 2 MC775LL/A Tablet (64GB Wifi + AT&T 3G Black) NEWE
Apple iPad XX6LL/A Tablet (64GB, Wifi + AT&T 3G, Black) NEWEST MODEL

$660 and up
(3 stores)
Compare
(Share and Compare)

Apple iPad 2 MC775LL/A 9.7" LED 64 GB Tablet Computer - Wi-Fi - 3G ...
Brand Apple · **Weight** 1.40 lb · **Screen size** 9.70 in
There's more to it. And even less of it. Two cameras for FaceTime and HD video recording. The dual-core A5 chip. The same 10-hour battery life. All in a thinner, lighter design.... more...

$642 and up
(10 stores)
Compare
(Share and Compare)

Black iPad 8gb
The iPad 2 is the second and current generation of the iPad, a tablet computer designed, developed and marketed by Apple. It serves primarily as a platform for audio-visual media... more...

$599
eCRATER
Compare
(Share and Compare)

Output of Entity Resolution

ID	Product Name	Price
r_1	iPad Two 16GB WiFi White	$490
r_2	iPad 2nd generation 16GB WiFi White	$469
r_3	iPhone 4th generation White 16GB	$545
r_4	Apple iPhone 3rd generation Black 16GB	$375
r_5	Apple iPhone 4 16GB White	$520

(r_1, r_2) , (r_3, r_5)

Entity Resolution Techniques

Similarity-based

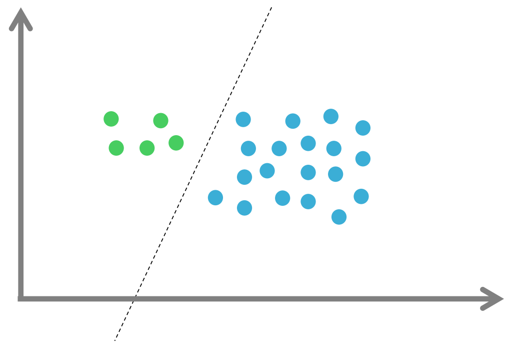
- Similarity Function (e.g., $\text{Jaccard}(r, s) = |\frac{r \cap s}{r \cup s}|$)
- Threshold (e.g., 0.8)

Jaccard(r1, r2) = 0.9 ≥ 0.8   Matching

Jaccard(r4, r8) = 0.1 < 0.8   Non-matching

Learning-based

- Represent a pair of records as a feature vector



Similarity-based

Suppose the similarity function is Jaccard.

Problem Definition

Given a table T and a threshold θ, the problem aims to find all record pairs $(r, s) \in T \times T$ such that $\text{Jaccard}(\mathrm{r}, \mathrm{s}) \geq \theta$

The naïve solution needs n^2 comparisons

Filtering-and-Verification

Step 1. Filtering

- Removing obviously dissimilar pairs

Step 2. Verification

- Computing Jaccard similarity only for the survived pairs

How Does Filtering Work?

What are "obviously dissimilar pairs"?

- Two records are obviously dissimilar if they do not share any word.
- In this case, their Jaccard similarity is zero, thus they will not be returned as a result and can be safely filtered.

How can we efficiently return the record pairs that share at least one word?

- To help you understand the solution, let's first consider a simplified version of the problem, which assumes that each record only contains one word

A simplified version

Suppose each record has only one word. Write an SQL query to do the filtering.

r_1	Apple
r_2	Apple
r_3	Banana
r_4	Orange
r_5	Banana

SELECT T1.id, T2.id

FROM Table T1, Table T2

WHERE T1.word = T2.word and T1.id < T2.id

Does it require n^2 **comparisons ?**

Output: (r1, r2), (r3, r5)

A general case

Suppose each record can have multiple words.

r_1	Apple, Orange
r_2	Apple
r_3	Banana
r_4	Orange, Apple
r_5	Banana

Flatten →

r_1	Apple
r_1	Orange
r_2	Apple
r_3	Banana
r_4	Orange
r_4	Apple
r_5	Banana

1. This new table can be thought of as the **inverted index** of the old table.
2. **Run the previous SQL on this new table** and remove redundant pairs.

Not satisfied with efficiency?

Exploring stronger filter conditions

- Filter the record pairs that share zero token
- Filter the record pairs that share one token
- ....
- Filter the record pairs that share k tokens

Challenges

- How to develop efficient filter algorithms for these stronger conditions?

Jiannan Wang, Guoliang Li, Jianhua Feng.
[Can We Beat The Prefix Filtering? An Adaptive Framework for Similarity Join and Search.]
SIGMOD 2012:85-96.

Not satisfied with result quality?

TF-IDF

- Use weighted Jaccard: $\mathbf{WJaccard}(r, s) = \frac{wt(r \cap s)}{wt(r \cup s)}$

Learning-based

- Model entity resolution as a classification problem
- How to generate feature vectors?

M. Bilenko and R. J. Mooney. [Adaptive duplicate detection using learnable string similarity measures](). In KDD, pages 39–48, 2003

Crowdsourcing

- Build a hybrid human-machine system (like Iron Man) for entity resolution

Summary

Data Collection
- Where to collect, How to Collect

Data Cleaning
- Dirty Data Problems, Data-cleaning tools

Data Integration
- Schema Mapping, Entity Resolution, Data Fusion

Plan for a 1-year Data Strategy

Team 1. SFU President Office

Team 2. BC Government

Team 3. Justin Trudeau Campaign Team

Team 4. Vancouver Hockey Team

Team 5. BC Children's Hospital

Team 6. Data Science Startup