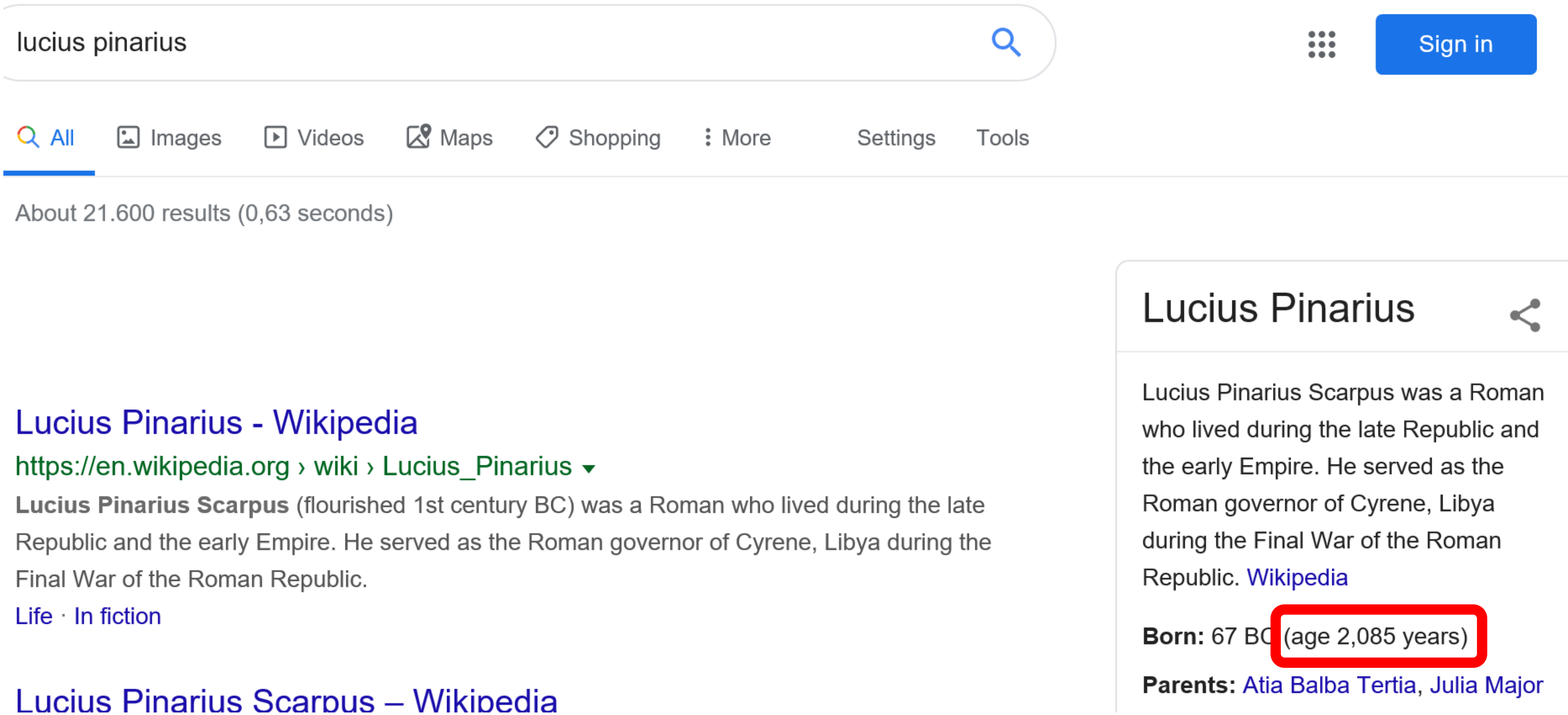
Information extraction 3. Design considerations, crawling and scraping

Simon Razniewski
Winter semester 2019/20

Announcements

- Assignments
  - Do not plagiarize
  - Submit outputs where asked
- No lecture nor tutorial next week
- Automating extraction?
  - Stay tuned…
- Visualizing KGs
  - https://www.wikidata.org/wiki/Wikidata:Tools/Visualize_data
  - https://angryloki.github.io/wikidata-graph-builder/?property=P40&item=Q3044&iterations=100&limit=100
  - https://angryloki.github.io/wikidata-graph-builder/?property=P737&item=Q937&iterations=100&limit=100
  - https://gate.d5.mpi-inf.mpg.de/webyago3spotlxComp/SvgBrowser/
  - https://developers.google.com/knowledge-graph



About 21.600 results (0,63 seconds)

Lucius Pinarius - Wikipedia
https://en.wikipedia.org › wiki › Lucius_Pinarius

Lucius Pinarius Scarpus (flourished 1st century BC) was a Roman who lived during the late Republic and the early Empire. He served as the Roman governor of Cyrene, Libya during the Final War of the Roman Republic.

Life · In fiction

Lucius Pinarius Scarpus – Wikipedia

Lucius Pinarius

Lucius Pinarius Scarpus was a Roman who lived during the late Republic and the early Empire. He served as the Roman governor of Cyrene, Libya during the Final War of the Roman Republic. Wikipedia

Born: 67 BC (age 2,085 years)

Parents: Atia Balba Tertia, Julia Major

- https://www.reddit.com/r/wikipedia/comments/dg6pnl/the_death_date_of_lucius_pinarius_wasnt_added_so/
- https://www.wikidata.org/wiki/Wikidata:Project_chat#unknown_values_for_people_who_have_long-since_died

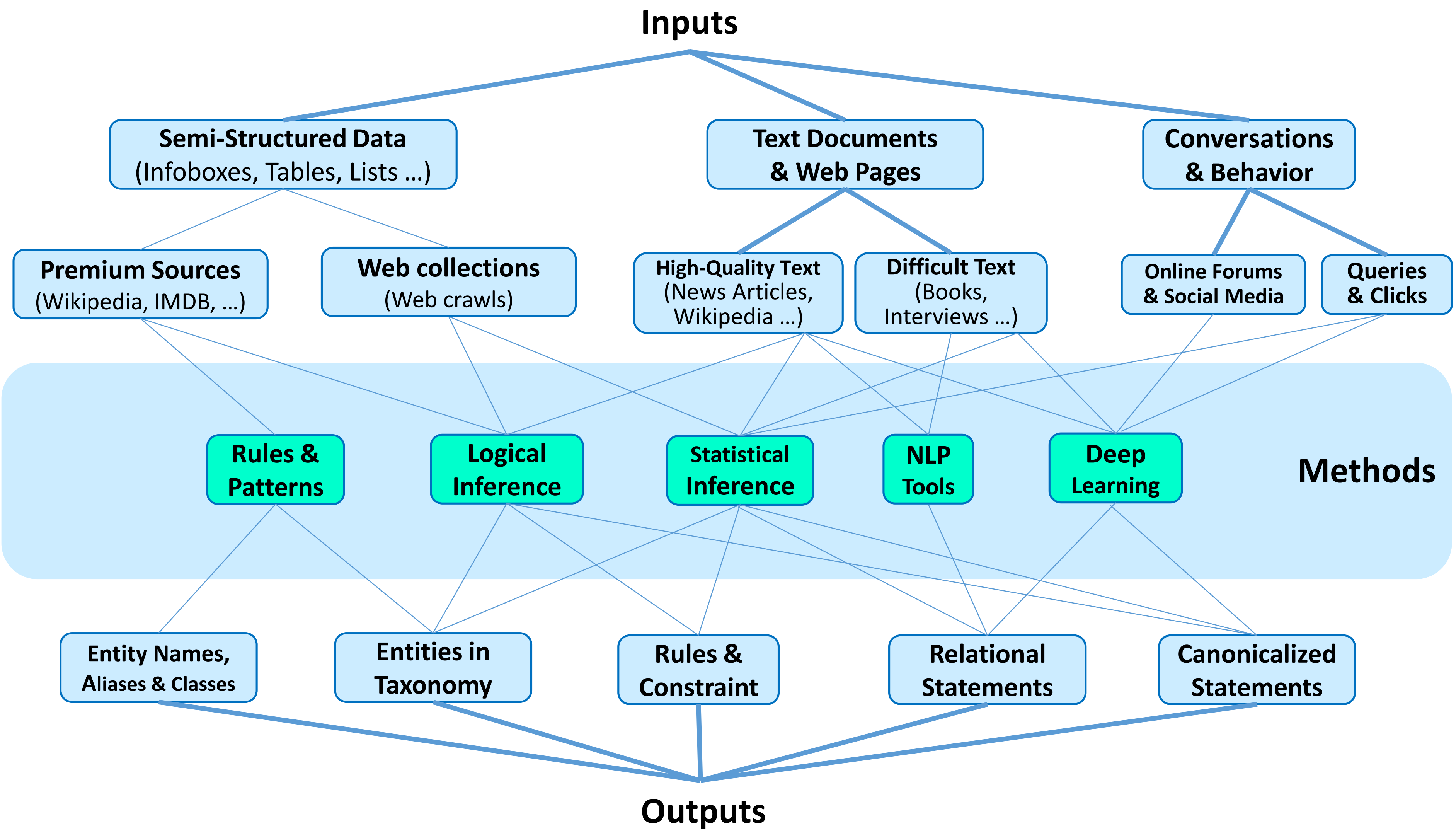
Outline

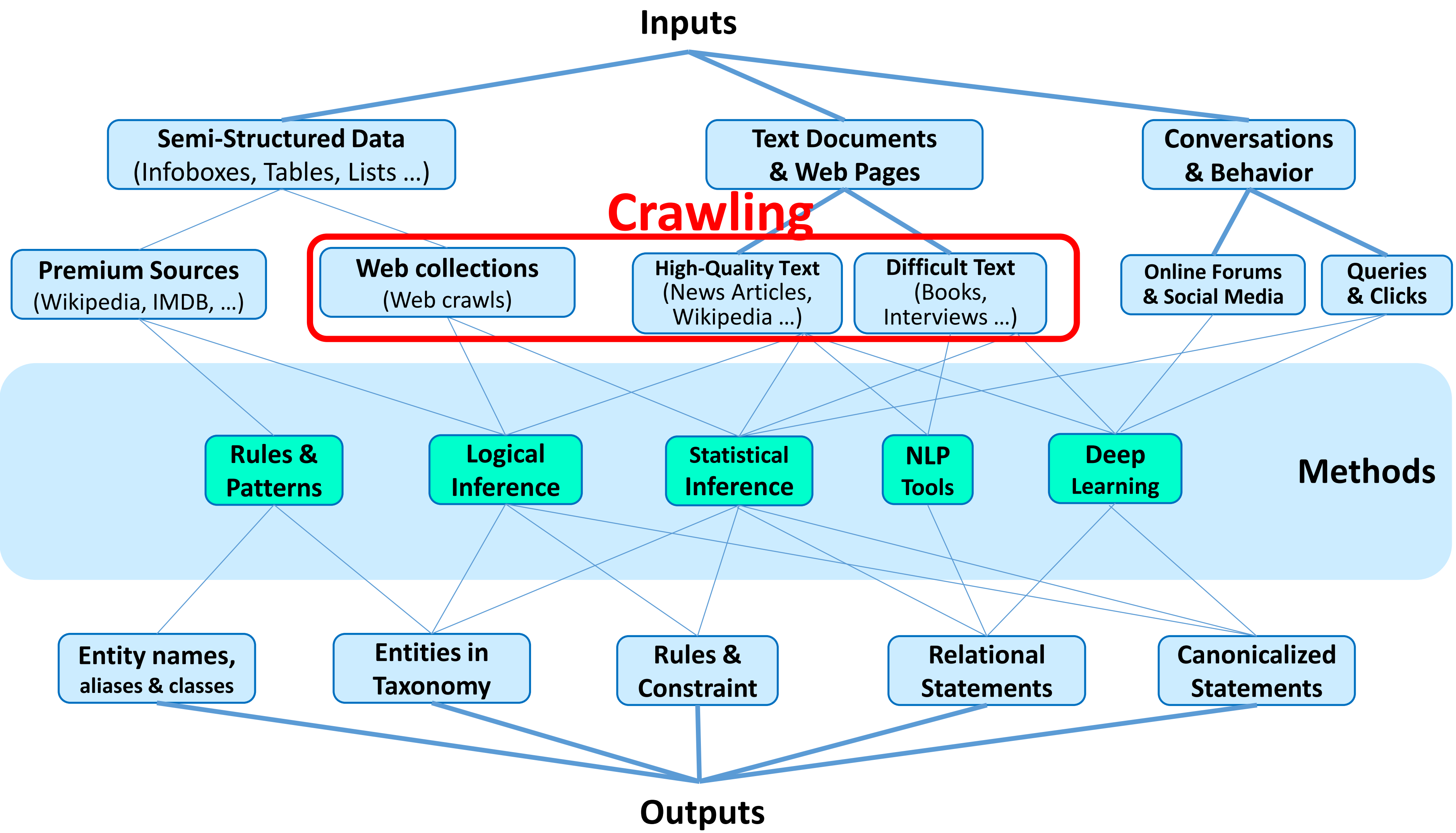
1. Design considerations
2. Crawling
3. Scraping

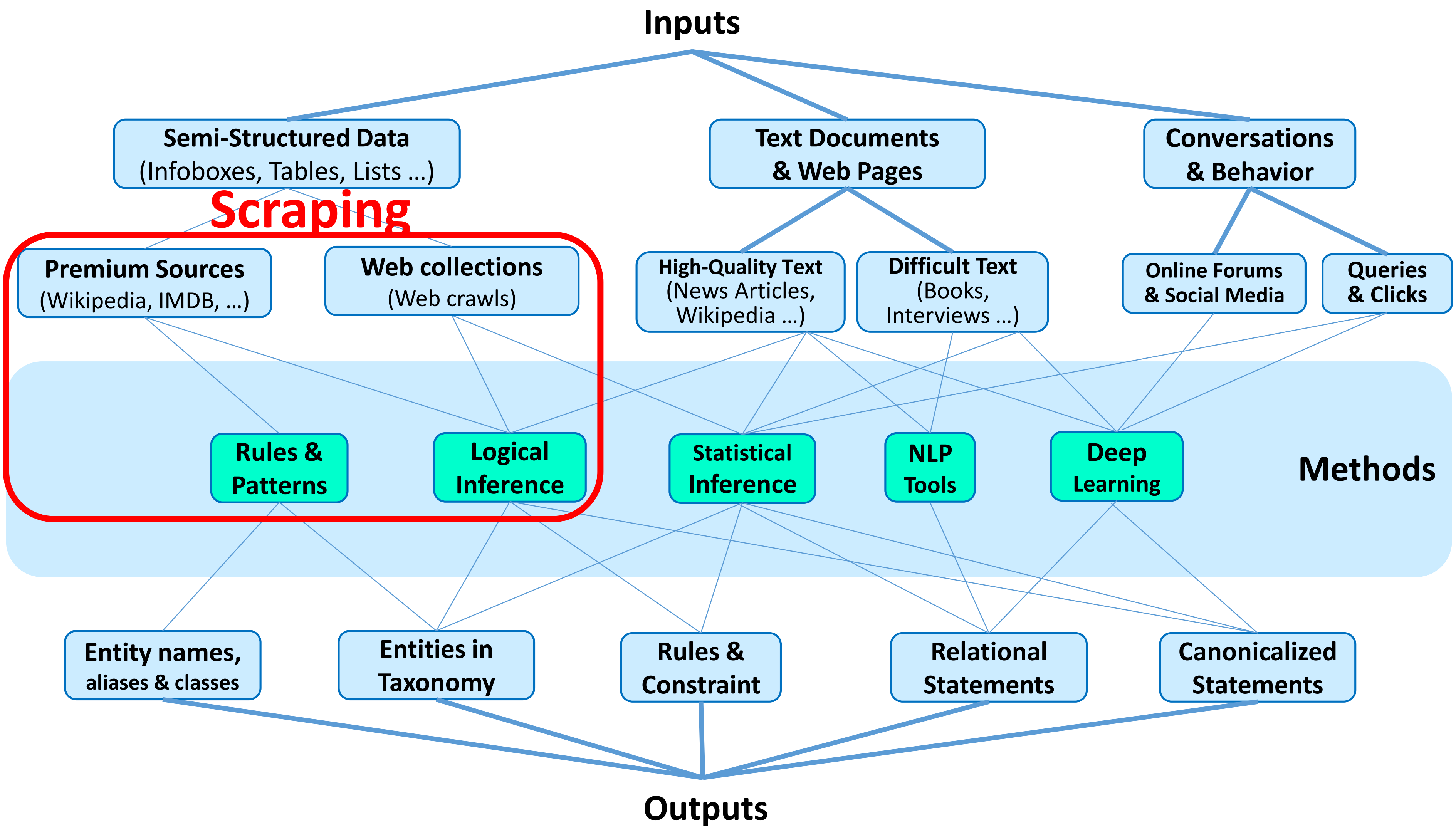
IE design considerations

1. What should be the output?
   - Type of information
   - Quality requirements

2. What is the best suited input?
3. Which method to get from input to output?


Inputs
Semi-Structured Data (Infoboxes, Tables, Lists …)
Text Documents & Web Pages
Conversations & Behavior
Premium Sources (Wikipedia, IMDB, …)
Web collections (Web crawls)
High-Quality Text (News Articles, Wikipedia …)
Difficult Text (Books, Interviews …)
Online Forums & Social Media
Queries & Clicks
Rules & Patterns
Logical Inference
Statistical Inference
NLP Tools
Deep Learning
Methods
Entity Names, Aliases & Classes
Entities in Taxonomy
Rules & Constraint
Relational Statements
Canonicalized Statements
Outputs


Inputs
Semi-Structured Data
(Infoboxes, Tables, Lists …)
Text Documents
& Web Pages
Conversations
& Behavior
Crawling
Premium Sources
(Wikipedia, IMDB, …)
Web collections
(Web crawls)
High-Quality Text
(News Articles,
Wikipedia …)
Difficult Text
(Books,
Interviews …)
Online Forums
& Social Media
Queries
& Clicks
Rules &
Patterns
Logical
Inference
Statistical
Inference
NLP
Tools
Deep
Learning
Methods
Entity names,
aliases & classes
Entities in
Taxonomy
Rules &
Constraint
Relational
Statements
Canonicalized
Statements
Outputs



Inputs

Semi-Structured Data
(Infoboxes, Tables, Lists …)

Text Documents
& Web Pages

Conversations
& Behavior

Scraping

Premium Sources
(Wikipedia, IMDB, …)

Web collections
(Web crawls)

High-Quality Text
(News Articles,
Wikipedia …)

Difficult Text
(Books,
Interviews …)

Online Forums
& Social Media

Queries
& Clicks

Rules &
Patterns

Logical
Inference

Statistical
Inference

NLP
Tools

Deep
Learning

Methods

Entity names,
aliases & classes

Entities in
Taxonomy

Rules &
Constraint

Relational
Statements

Canonicalized
Statements

Outputs

Outline

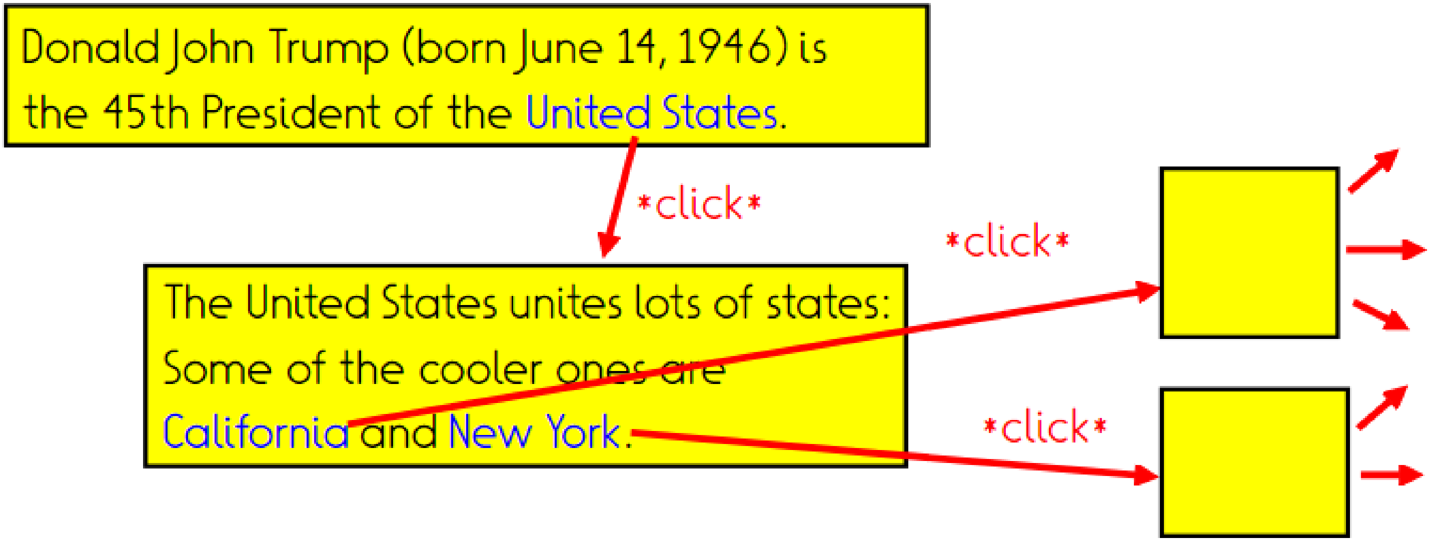
1. Design considerations
2. Crawling
3. Scraping

Acknowledgment

- Material adapted from Fabian Suchanek and Antoine Amarilli

Web Crawler

A Web crawler is a system that follows hyperlinks, collecting all pages on the way.

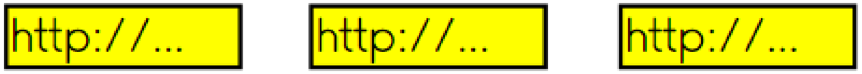


Donald John Trump (born June 14, 1946) is the 45th President of the United States.

The United States unites lots of states: Some of the cooler ones are California and New York.

A crawler does BFS on URLs

1. Start with queue of important URLs

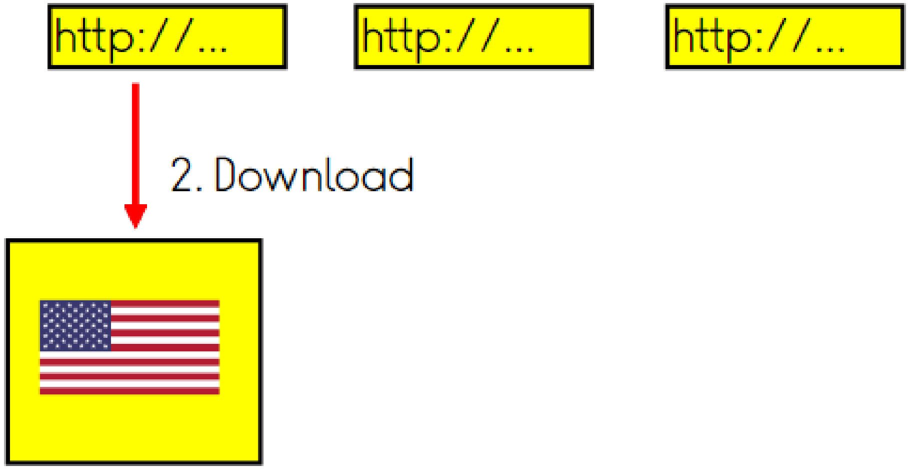


http://...

http://...

http://...

A crawler does BFS on URLs



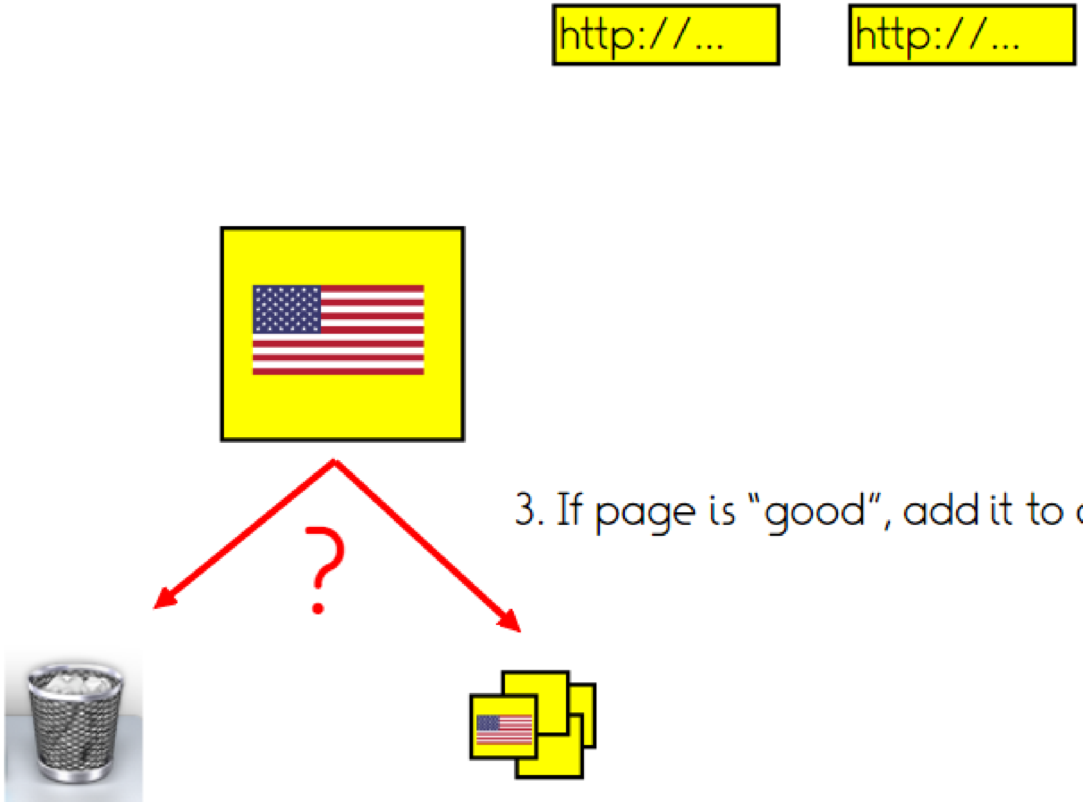
http://...

http://...

http://...

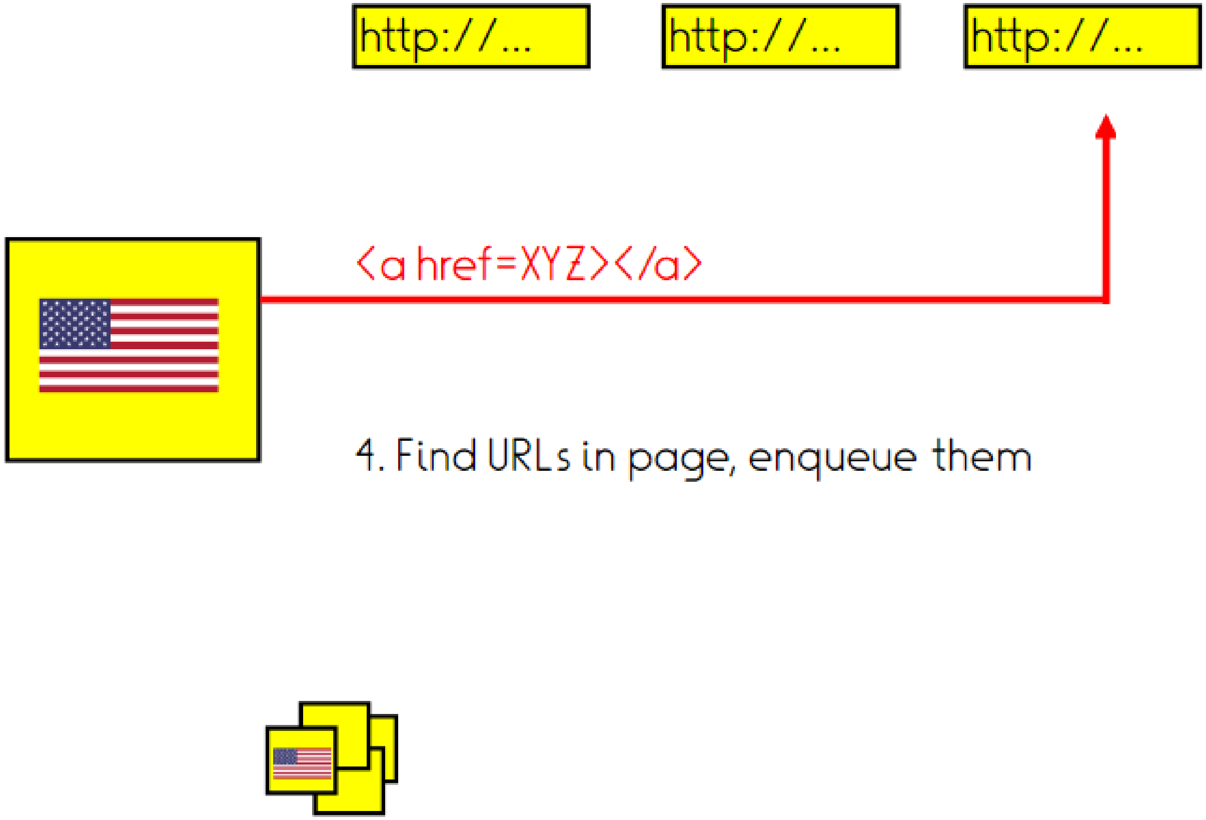
2. Download

A crawler does BFS on URLs



3. If page is "good", add it to corpus

A crawler does BFS on URLs



http://...

http://...

http://...

<a href=XYZ></a>

4. Find URLs in page, enqueue them

A crawler does BFS on URLs

http://... http://... http://...

5. repeat the process until you covered all pages

- within a certain depth
- in a certain domain
- with certain topics
- .

Finding new URLs

- In an HTML page
  - Hyperlinks <a href="...">
  - Media <img src="...">, <audio src="...">, <video src="...">, <source src="...">
  - Frames <iframe src="...">
  - JavaScript window.open("...") – undecidable in general
  - Page text by regular expressions.
- In other kinds of files (PDFs...).
- In sitemaps provided specifically to crawlers.

Freshness Problem

- Content on the Web changes
- Different change rates:

  online newspaper main page: every hour or so

  published article: virtually no change
- Continuous crawling, and identification of change rates for adaptive crawling:

  If-Last-Modified HTTP feature (not reliable)

  Identification of duplicates in successive request

Freshness problem (2)

- Prediction problem: Estimate page change frequency
  - From previous change behavior
  - Or from page content

- Optimization problem: Decide crawl frequency
  - Fixed budget → How to distribute them
  - Flexible budget → Cost-benefit framework needed

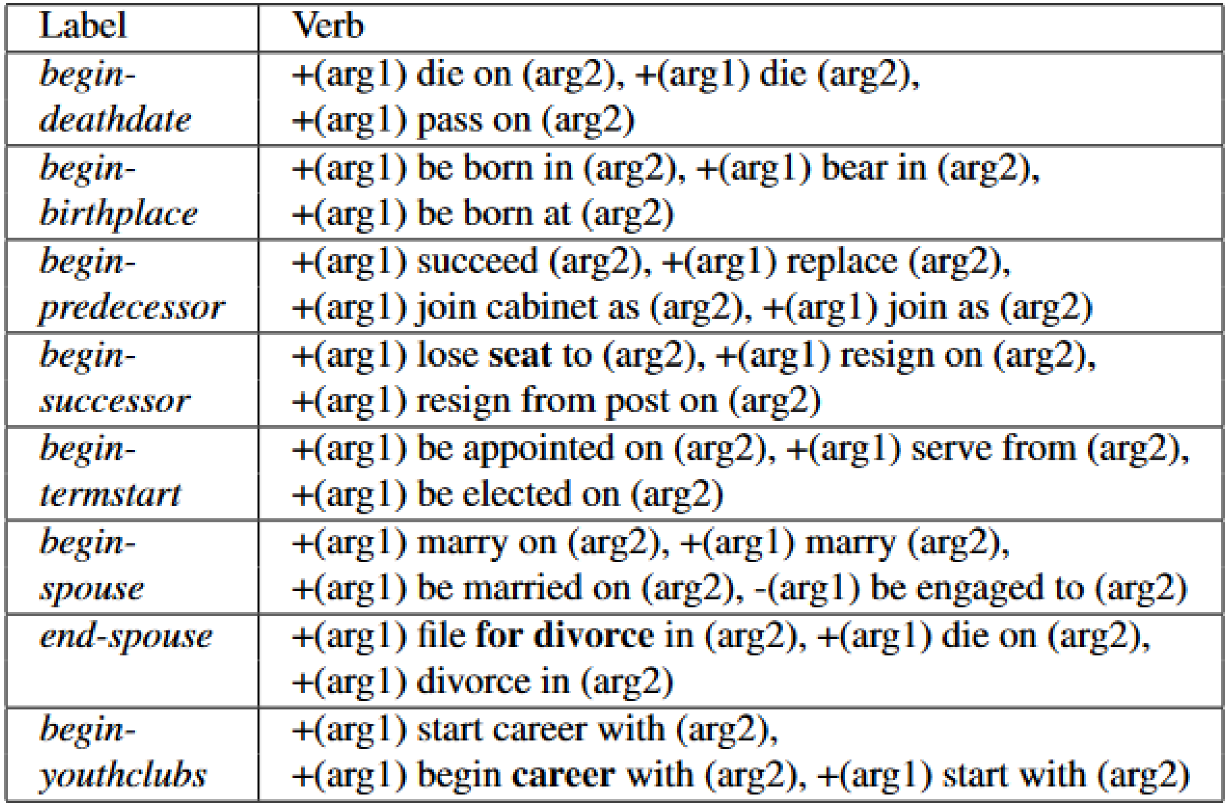
Estimating change frequencies

- Cho and Molina, TOIT 2003
  - Model changes as Poisson processes (i.e., memoryless/ statistically independent)
  - Extrapolate change frequency from previous visits

    → Daily visit for 10 days, 6 changes detected

    → Change frequency: 0.6 changes/day?
  - Extrapolation underestimates change frequency due to multiple change possibility
- Liang et al., IJCAI 2017
  - Monitor news websites
  - Build supervised prediction models based on page features
- Wijaya et al., EMNLP 2015
  - Wikipedia-specific
  - Learn state-change-indicating terms
  - E.g., engage, divorce

Wijaya et al., EMNLP 2015



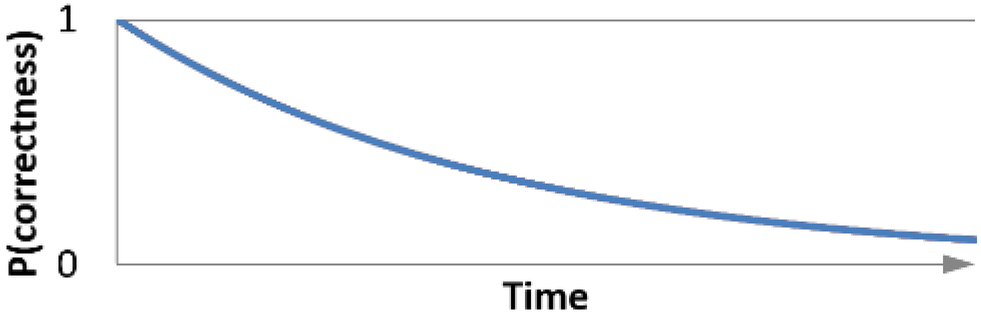
Label	Verb
begin-deathdate	+(arg1) die on (arg2), +(arg1) die (arg2), +(arg1) pass on (arg2)
begin-birthplace	+(arg1) be born in (arg2), +(arg1) bear in (arg2), +(arg1) be born at (arg2)
begin-predecessor	+(arg1) succeed (arg2), +(arg1) replace (arg2), +(arg1) join cabinet as (arg2), +(arg1) join as (arg2)
begin-successor	+(arg1) lose seat to (arg2), +(arg1) resign on (arg2), +(arg1) resign from post on (arg2)
begin-termstart	+(arg1) be appointed on (arg2), +(arg1) serve from (arg2), +(arg1) be elected on (arg2)
begin-spouse	+(arg1) marry on (arg2), +(arg1) marry (arg2), +(arg1) be married on (arg2), -(arg1) be engaged to (arg2)
end-spouse	+(arg1) file for divorce in (arg2), +(arg1) die on (arg2), +(arg1) divorce in (arg2)
begin-youthclubs	+(arg1) start career with (arg2), +(arg1) begin career with (arg2), +(arg1) start with (arg2)

Distributing crawl resources
[Razniewski, CIKM 2016]

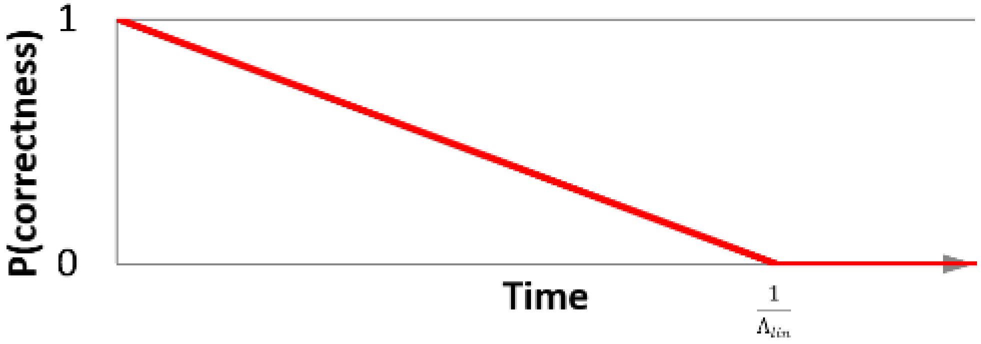
- Ingredients:
  - Benefit of an up-to-date website
    - Synonymous: cost of outdated website
  - Cost of a crawl action
  - Decay behavior

→ Page-specific recrawl frequency that maximizes benefit minus cost

Decay behaviour



$$z_{exp}(t) = e^{-\lambda_{exp} t}.$$



$$z_{lin}(t) = max(1 - \lambda_{lin} t, 0).$$

Observed decay behaviour

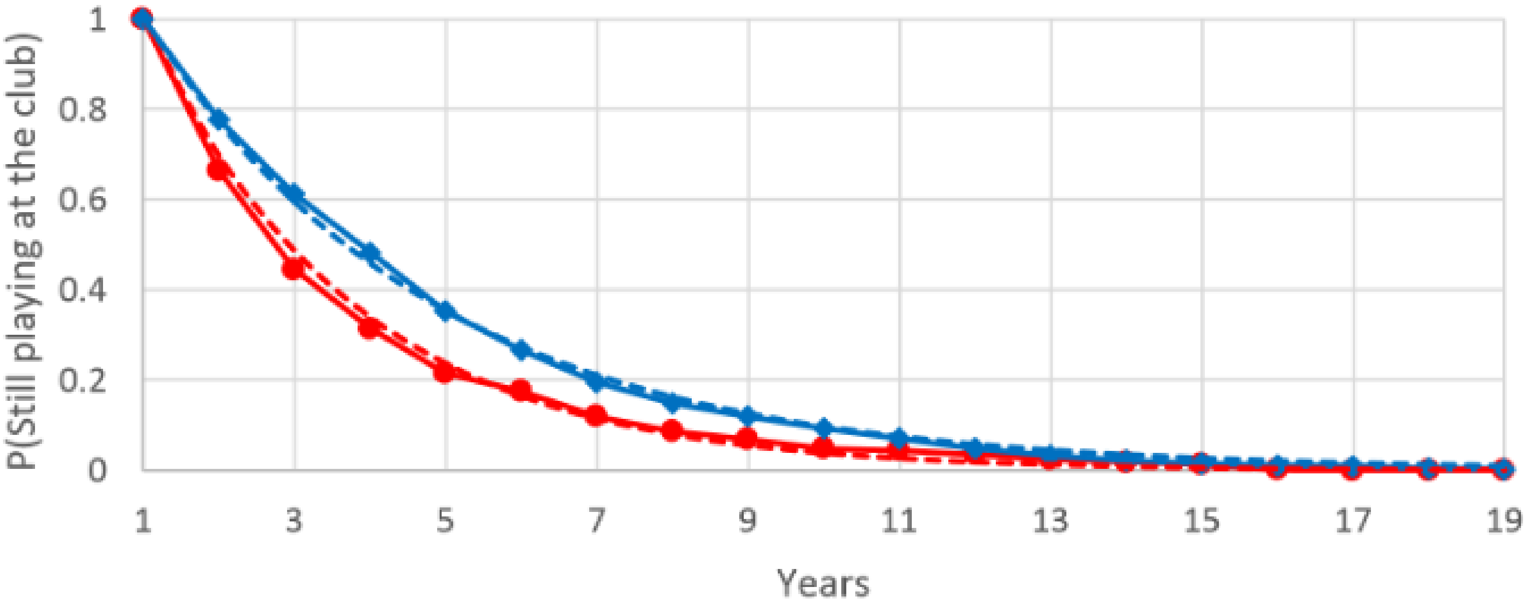
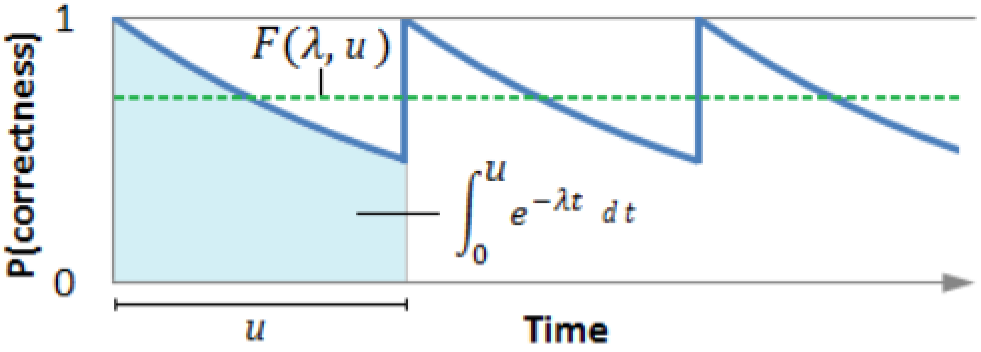


Figure 7: Decay behaviour of soccer players at Manchester United (blue) and Bayern München (red), observed (solid lines), and approximated by exponential decay curves with $\lambda = 0.26$ and 0.36, respectively (dashed lines).

Average freshness F



$$F(\lambda,u) = \frac{\int_0^u z(t)dt}{u}.$$

$$F_{lin}(\lambda,u) = 1 - \frac{\lambda \cdot u}{2}.$$

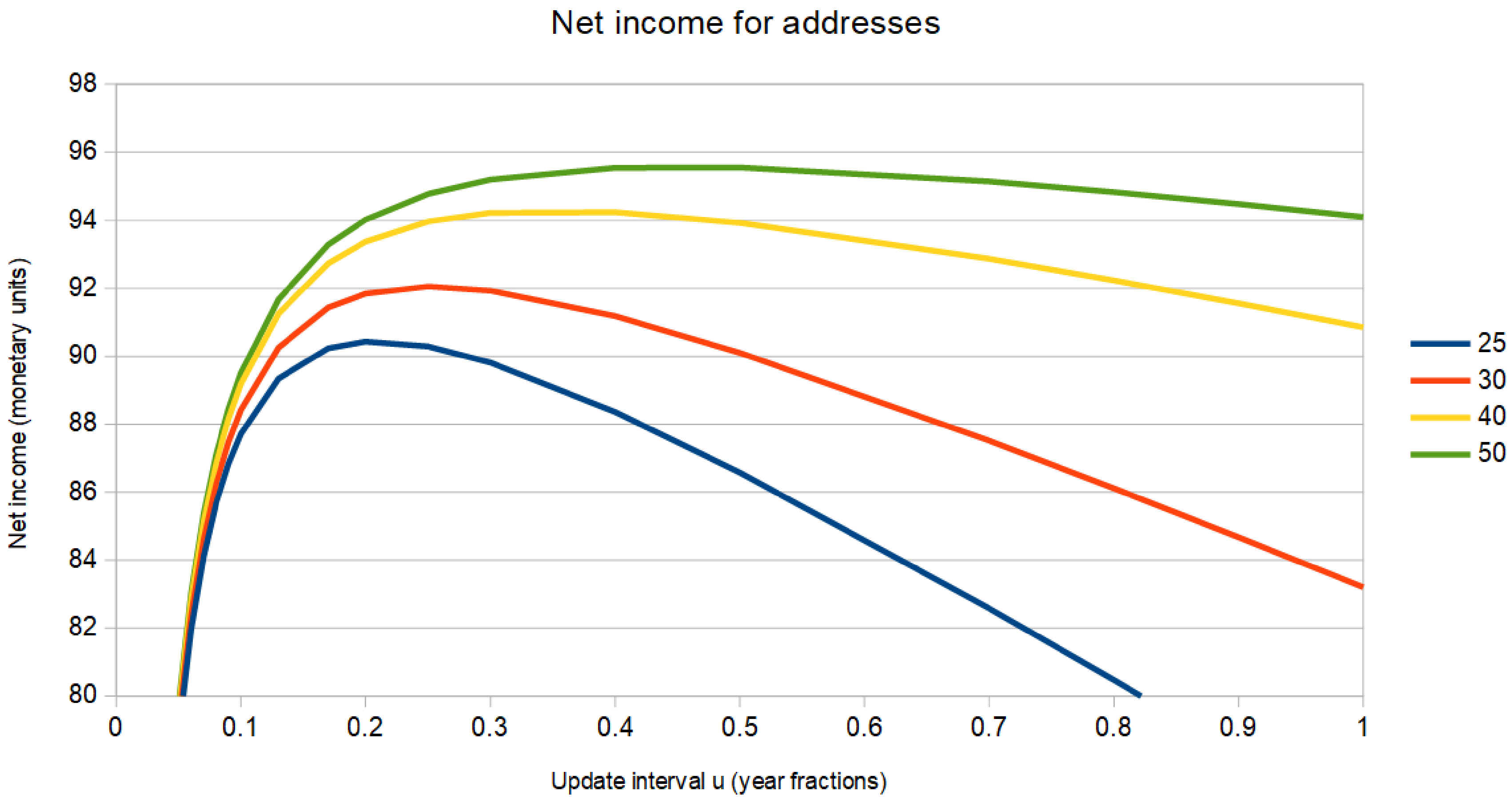
$$F_{exp}(\lambda,u) = \frac{1 - e^{-\lambda u}}{\lambda \cdot u}.$$

Net income NI

B…Benefit/time unit
F…Average freshness
Λ… decay coefficient
u…update interval length
C…cost of an update

$$\begin{tikzcd}
 & NI_{lin}(u) = B - \frac{B \cdot \lambda \cdot u}{2} - \frac{C}{u}. \arrow[dr] & \\
NI(u) = B \cdot F(\lambda, u) - \frac{C}{u}. \arrow[ur] \arrow[dr] & & \text{Optimum via common algebra} \\
 & NI_{exp}(u) = B \frac{1 - e^{-\lambda u}}{\lambda \cdot u} - \frac{C}{u}. \arrow[ur] &
\end{tikzcd}$$

Examples for address updates NI over u



Assumption: benefit over one year = 100 x cost of single crawl
Actual ratio magnitudes lower, e.g., 0.003 Cents/crawl
[http://www.michaelnielsen.org/ddi/how-to-crawl-a-quarter-billion-webpages-in-40-hours/]
(and for 580 $ on Amazon EC2)

Duplicate pages

- Prevent multiple indexing and penalize content farms.
- Prevent duplicate URLs by canonicalization.

  http://example.com:80/foo

  = http://example.com/bar/../foo

  = http://www.example.com/foo
- Detect duplicate pages by using a hash function.
- Detect near-duplicates (dates, etc.) by using a similarity function.

  (e.g., Broder's MinHash from 1997, used in AltaVista and later Google)

Crawl scheduling

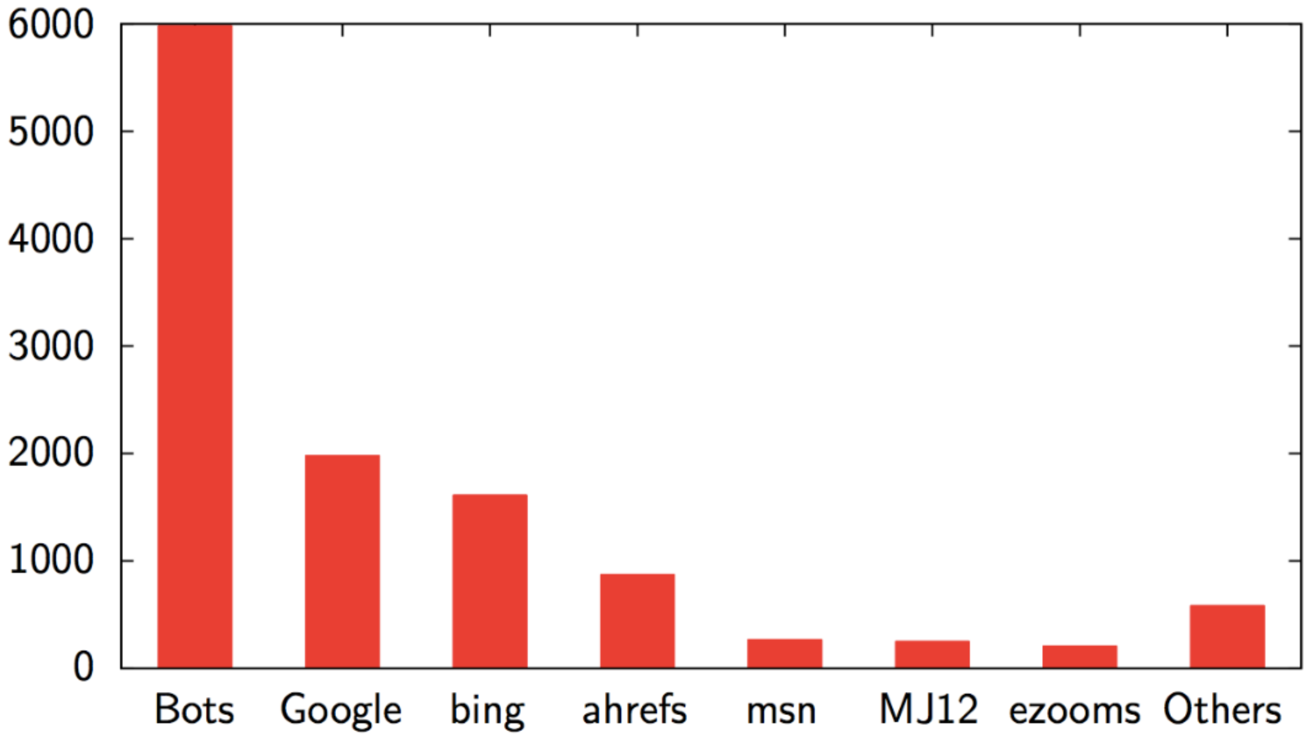
- Wait a minimal delay between requests to the same server.

  => Depends on the server (wikipedia.org vs your laptop).

  => Depends on the resource (large files...).

  => Generally, waiting at least one second is preferable.
- Requests to different servers can be parallelized.
- Requests should be run asynchronously.
- The HTTP connection should remain open.
- Requests can be distributed across multiple machines.
- Crawlers represent about 20% of Web traffic.

Crawler traffic



Traffic on a3nm.net as of September 2013 (out of 36593 requests).

Robot control (honor-based)

- Robot Exclusion Standard: http://example.com/robots.txt
  => Only at root level (not available for subfolders).
  => Filtering by User-agent.
  => Disallow directive to forbid certain pages.
  => Also: Allow, Crawl-delay, Host, Sitemap.
- HTTP header: X-Robots-Tag (less support):
  => X-Robots-Tag: noindex
- Meta tag: <meta name="robots" content="noindex">
  => Also nofollow, nosnipped, noarchive...
- Links: <a href="secret/" rel="nofollow">
- Engine-specific interfaces (e.g., Google Webmaster Tools).

=> No guarantees!

https://www.mpi-inf.mpg.de/robots.txt
https://www.google.de/robots.txt

Robot control with CAPTCHAs

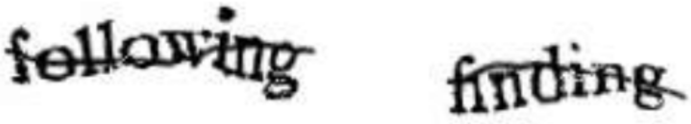
How can we discriminate against robots?

- Completely Automated Public Turing test to tell Computers and Humans Apart (trademarked by CMU, but patented by AltaVista).
- Making a computer able to recognize humans.
- Can be any AI problem: add two numbers, listen to a word, recognize an animal in an image, etc.

ReCAPTCHAs

CAPTCHAs can be used to

- digitize books
  Show one word that we know (to validate the user),
  and one word that we want to digitize (to digitize the book)


following
finding

- Show ads
  Ask the user to type a slogan

- Do recognition of street numbers in
  Google street view images

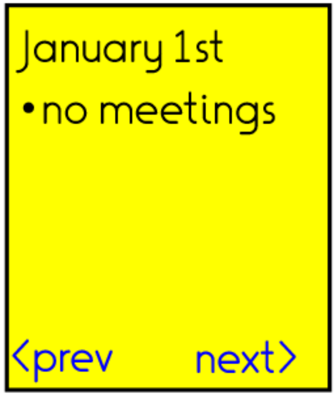
Breaking CAPTCHAs

- Employ humans to remotely solve CAPTCHAs (“sweatshops”, hundreds per hour)
- Sometimes there may be no ground truth → Try often enough
- Optical character recognition has improved and can solve some CAPTCHAs

"Robot Control" by Spider Traps

A spider trap (also: crawler trap, robot trap) is a set of web pages that cause a web crawler to make an infinite number of requests or cause a poorly constructed crawler to crash.
[Wikipedia/Spider trap]

Example:



January 1st
- no meetings

<prev next>

Spider traps can be intentional or unintentional.
Can be used to trap spiders that do not follow robots.txt :-)

http://foo.com/bar/foo/bar/foo/bar/foo/bar/.....

Deep web / dark web

- Pages that have no links to them.
- For instance, result pages from a search.
- 2001 estimate: the deep Web is hundreds of times larger than the reachable Web.
- Web form probing:

  => Need to figure out form constraints.

  => Need to come up with keywords.

  => Idea: feed back words from the website into the form.

Bergman, Michael K (August 2001). "The Deep Web: Surfacing Hidden Value". The Journal of Electronic Publishing, 7 (1)

We can use an existing Web crawl

	pages	size
ClueWeb	1b	25 TB
CommonCrawl	6b	100TB
Internet Archive	2b	80TB
enWikipedia	5m	30 GB
Dresden web table corpus	125m	
Twitter dumps 2016 US election	280m	
Reddit dumps	…	
Wikia dumps	…	

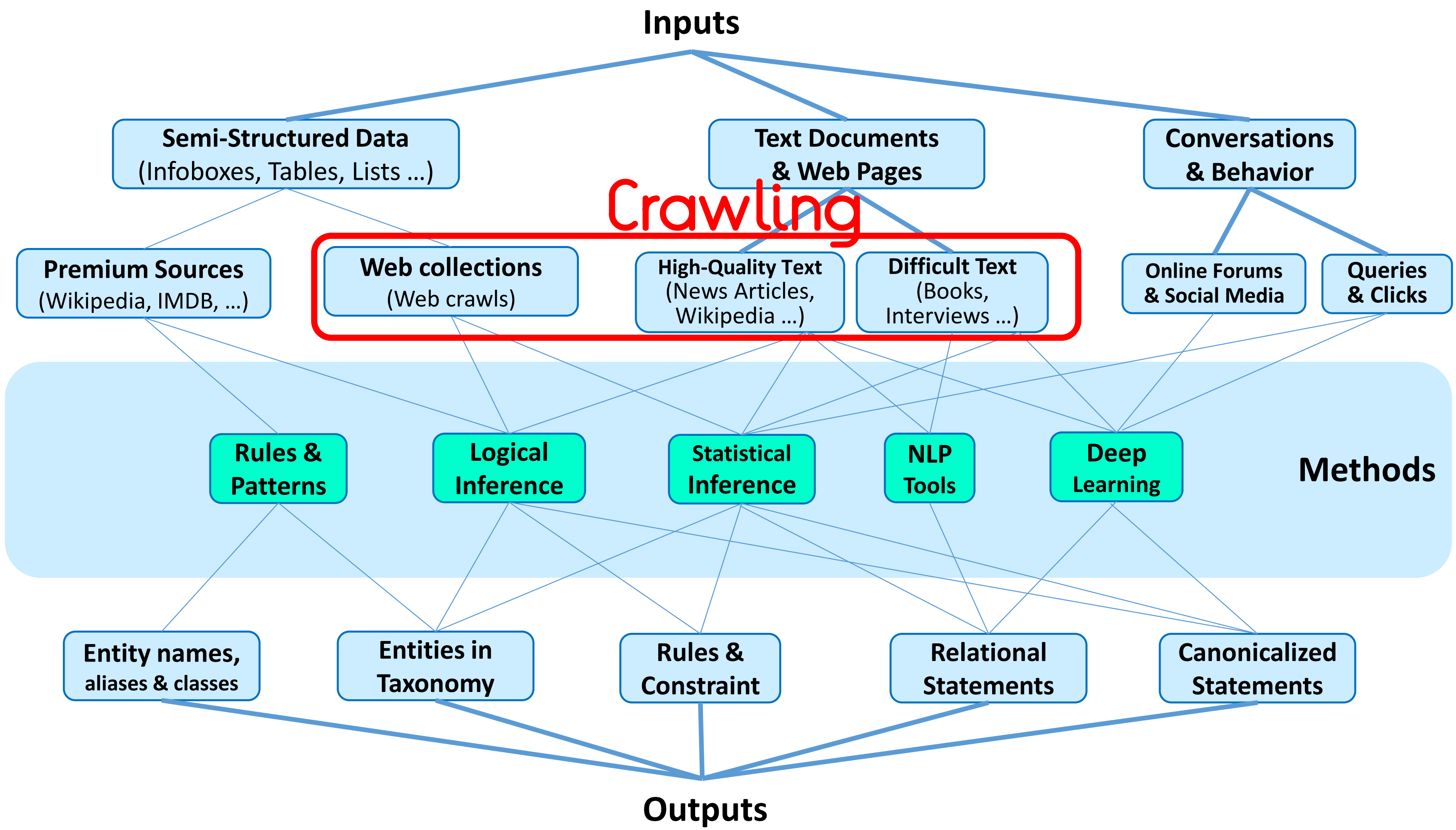
…

Insights from crawling mpi-inf.mpg.de

- URL ending inclusion/exclusion criteria need thought
- Long (machine-generated URLs) need exclusion
- Beyond that no issues
- 35 lines in Python
- Sequential runtime for 2000 pages: ~10 minutes
- Completeness?

Outline

1. Design considerations
2. Crawling
3. Scraping



Inputs

Semi-Structured Data
(Infoboxes, Tables, Lists …)

Text Documents
& Web Pages

Conversations
& Behavior

Crawling

Premium Sources
(Wikipedia, IMDB, …)

Web collections
(Web crawls)

High-Quality Text
(News Articles,
Wikipedia …)

Difficult Text
(Books,
Interviews …)

Online Forums
& Social Media

Queries
& Clicks

Rules &
Patterns

Logical
Inference

Statistical
Inference

NLP
Tools

Deep
Learning

Methods

Entity names,
aliases & classes

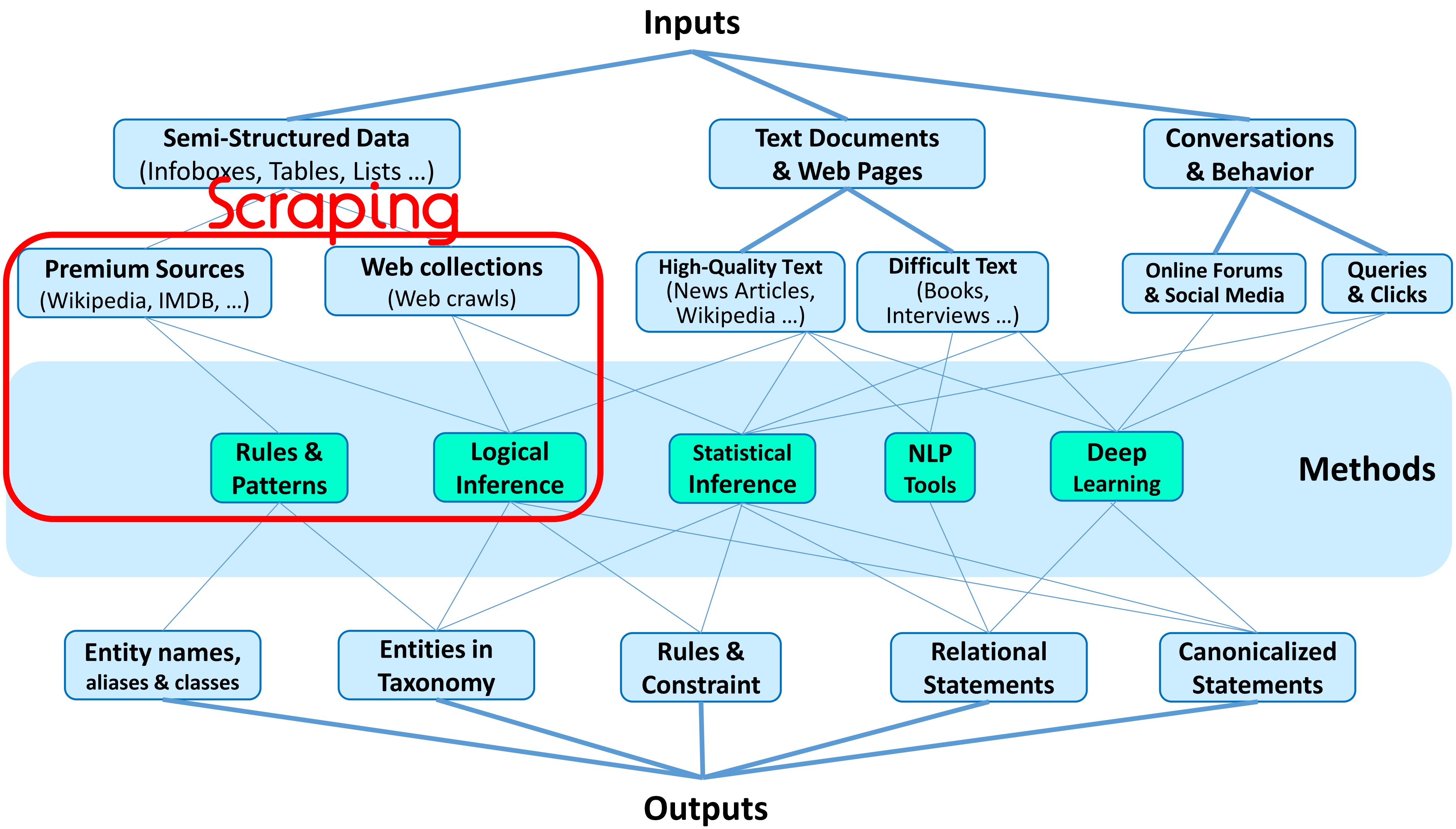
Entities in
Taxonomy

Rules &
Constraint

Relational
Statements

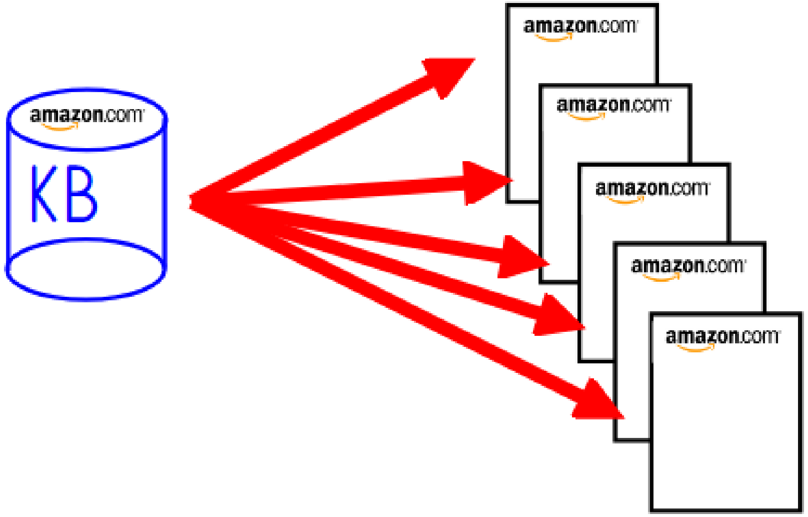
Canonicalized
Statements

Outputs

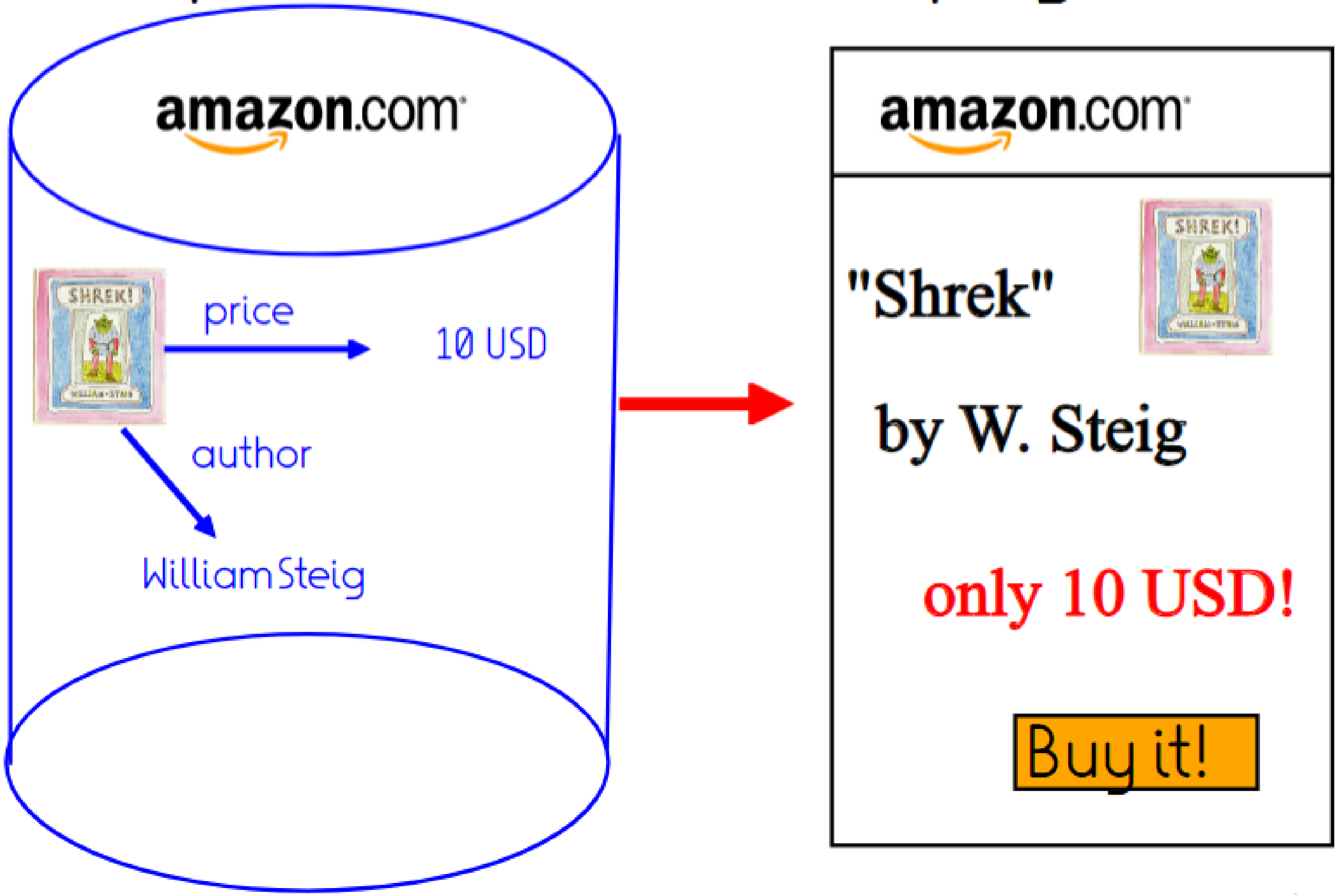

Inputs
Semi-Structured Data
(Infoboxes, Tables, Lists …)
Text Documents
& Web Pages
Conversations
& Behavior
Scraping
Premium Sources
(Wikipedia, IMDB, …)
Web collections
(Web crawls)
High-Quality Text
(News Articles,
Wikipedia …)
Difficult Text
(Books,
Interviews …)
Online Forums
& Social Media
Queries
& Clicks
Rules &
Patterns
Logical
Inference
Statistical
Inference
NLP
Tools
Deep
Learning
Methods
Entity names,
aliases & classes
Entities in
Taxonomy
Rules &
Constraint
Relational
Statements
Canonicalized
Statements
Outputs

Generated Web pages

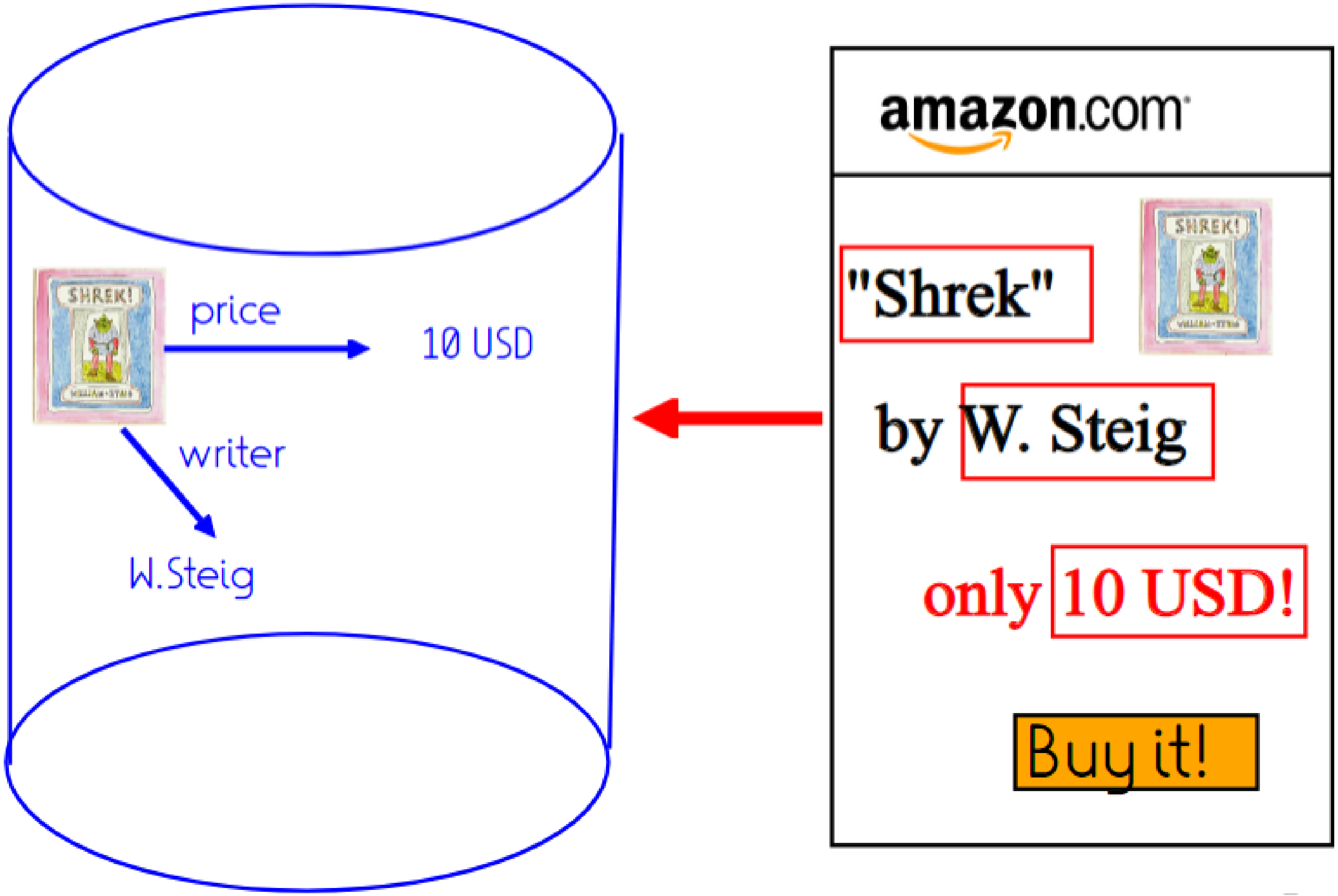
Web page generation is the process of producing several similar Web pages from a KB.



Example: Generated Web pages



Scraping aims to reconstruct the KB



price

10 USD

writer

W.Steig

amazon.com

"Shrek"

by W. Steig

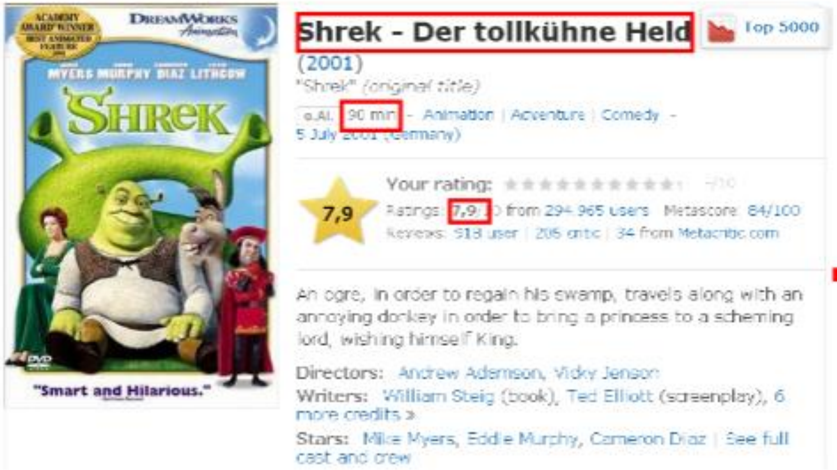
only 10 USD!

Buy it!

Def: Wrapper

A wrapper for a set of pages generated from the same KB is a function that extracts strings from such a page.

(Technically, it is the inverse function of the function that generated the page.
The strings still have to be disambiguated and put in relation to yield facts.
Different applications have different more specific definitions of the "strings".)
Kushmerick: Wrapper Induction



ACADEMY AWARD WINNER BEST ANIMATED FEATURE
DREAMWORKS Animation
MYERS MURPHY DIAZ LITHGOW
SHREK
DVD
"Smart and Hilarious."

Shrek - Der tollkühne Held Top 5000
(2001)
"Shrek" *(original title)*
o.Al. 90 min - Animation | Adventure | Comedy -
5 July 2001 (Germany)

7,9 Your rating: ★★★★★★★★★★ -/10
Ratings: 7,9/10 from 294.965 users Metascore: 84/100
Reviews: 918 user | 205 critic | 34 from Metacritic.com

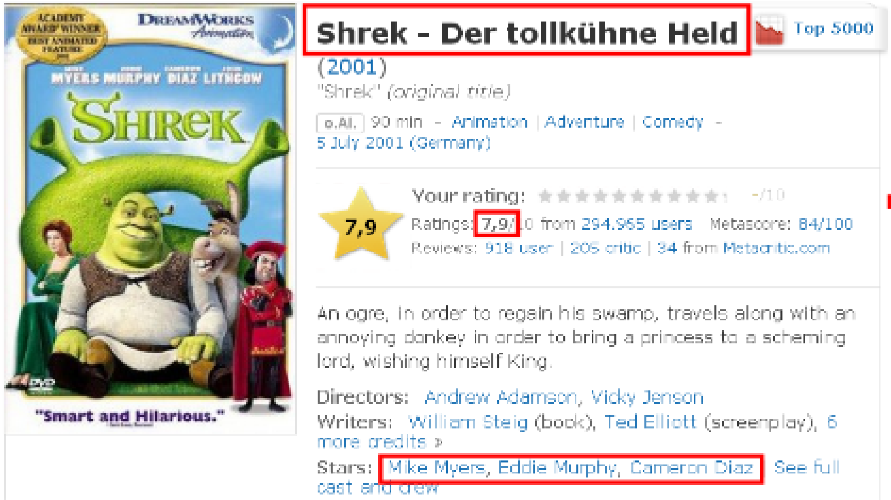
An ogre, in order to regain his swamp, travels along with an annoying donkey in order to bring a princess to a scheming lord, wishing himself King.

Directors: Andrew Adamson, Vicky Jenson
Writers: William Steig (book), Ted Elliott (screenplay), 6 more credits »
Stars: Mike Myers, Eddie Murphy, Cameron Diaz | See full cast and crew

→

"Shrek...",
"90 min",
"7.9"

Information is always in same place


ACADEMY AWARD WINNER BEST ANIMATED FEATURE
DreamWorks Animation
MYERS MURPHY DIAZ LITHGOW
SHREK
DVD
"Smart and Hilarious."

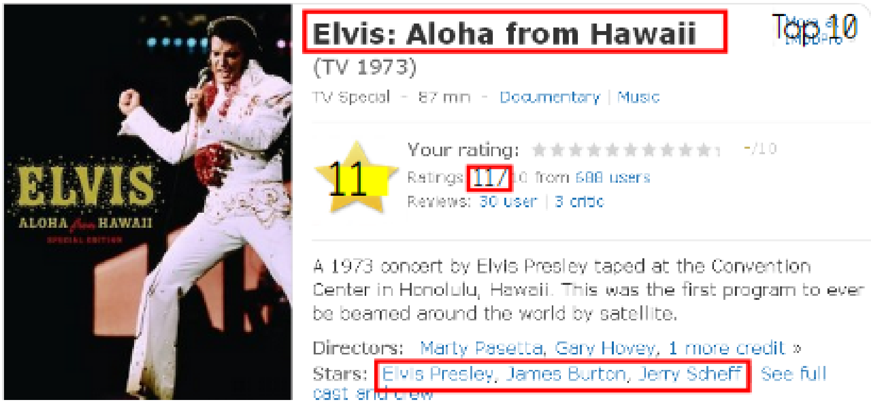
Shrek - Der tollkühne Held Top 5000
(2001)
"Shrek" *(original title)*
o.Al. 90 min - Animation | Adventure | Comedy -
5 July 2001 (Germany)

Your rating: -/10
Ratings: 7,9/10 from 294.965 users Metascore: 84/100
Reviews: 918 user | 205 critic | 34 from Metacritic.com

An ogre, in order to regain his swamp, travels along with an annoying donkey in order to bring a princess to a scheming lord, wishing himself King.

Directors: Andrew Adamson, Vicky Jenson
Writers: William Steig (book), Ted Elliott (screenplay), 6 more credits »
Stars: Mike Myers, Eddie Murphy, Cameron Diaz See full cast and crew

If we understand this...



Elvis: Aloha from Hawaii Top 10
(TV 1973)
TV Special - 87 min - Documentary | Music

Your rating: -/10
Ratings: 11/10 from 688 users
Reviews: 30 user | 3 critic

A 1973 concert by Elvis Presley taped at the Convention Center in Honolulu, Hawaii. This was the first program to ever be beamed around the world by satellite.

Directors: Marty Pasetta, Gary Hovey, 1 more credit »
Stars: Elvis Presley, James Burton, Jerry Scheff See full cast and crew

then we understand this.

Def: XPath

XPath is a formal language for selecting nodes in an XML document.

/ identifies the root node

K/T[i] identifies the i-th child with
tag T of the node identified by K

K/T is K/T[1] if K has one T child

```
<html>
 <body>
   <h1>Aloha from Hawaii</h1>
   <p>This is a really great movie</p>
   <p>Stars:<i>Elvis Presley</i></p>
 </body>
</html>
```

/html/body/p[2]/i

[https://www.w3schools.com/xml/xml_xpath.asp]
[https://devhints.io/xpath]

Task: XPath

Write XPath expressions that identify nodes whose text is "Shrek", "W. Steig", and "84 min".

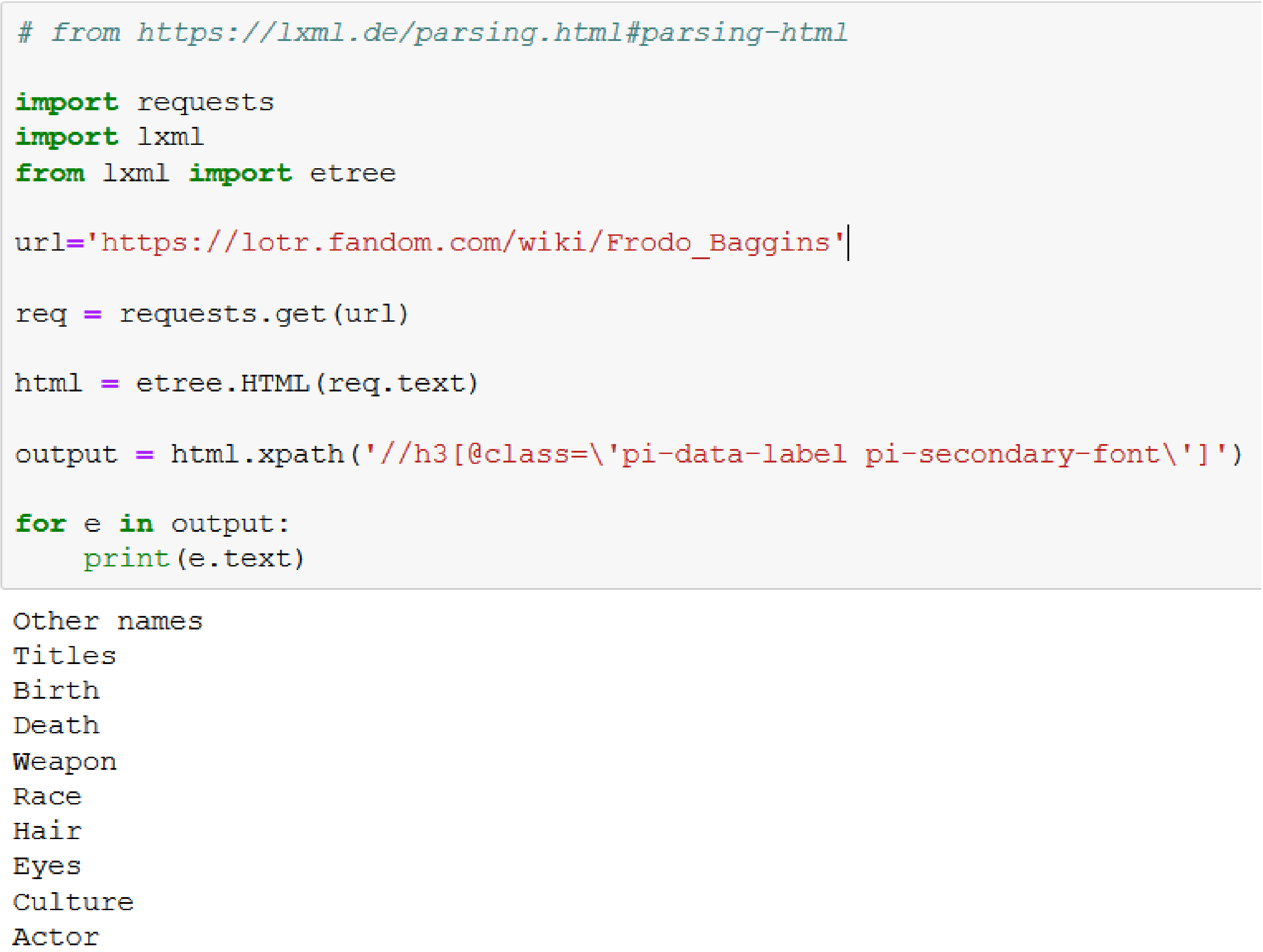
```
<html>
 <body>
  <b>Shrek</b>
  <ul>
   <li>Creator: <b>W. Steig</b></li>
   <li>Duration: <i>84m</i></li>
   </ul>
</body>
</html>
```

https://www.freeformatter.com/xpath-tester.html

Scraping: Browser

- “Try XPath” Firefox addin
- //h3[@class='pi-data-label pi-secondary-font']
- Firefox console
  - $x('//h3[@class=\'pi-data-label pi-secondary-font\']')
- //h3[@class='pi-data-label pi-secondary-font'] | //div[@class='pi-data-value pi-font']

Scraping in Python - XPath

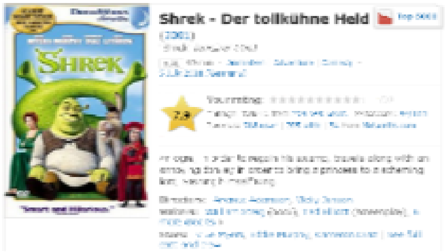


```
# from https://lxml.de/parsing.html#parsing-html

import requests
import lxml
from lxml import etree

url='https://lotr.fandom.com/wiki/Frodo_Baggins'

req = requests.get(url)

html = etree.HTML(req.text)

output = html.xpath('//h3[@class=\'pi-data-label pi-secondary-font\']')

for e in output:
    print(e.text)
```

```
Other names
Titles
Birth
Death
Weapon
Race
Hair
Eyes
Culture
Actor
```

Def: Wrapper induction

Wrapper induction is the process of generating a wrapper from a set of Web pages with strings to be extracted.



Shrek - Der tollkühne Held Top 5000

(2001)

[illegible]

7.9

[illegible]

+

"Shrek", "7.9"

=

/html/body/h1

/html/body/p[2]/i

Web page

+

Strings to be extracted

=

Wrapper

Wrapper induction

Wrapper Induction requires as input Web pages with strings to be extracted. These can come, e.g.,

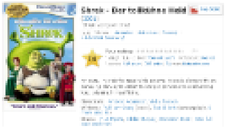
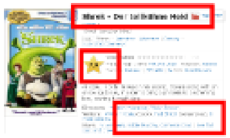
- from a KB

hasTitle(ShrekMovie, "Shrek")

- from manual extraction

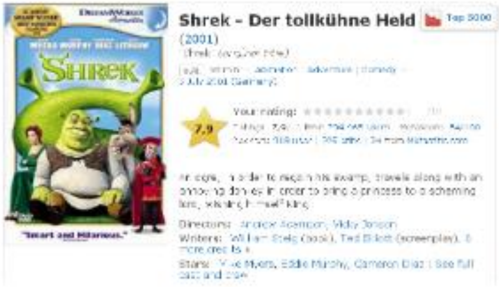
+ = "Shrek"

- from manual annotation in a GUI

 + = 

Detail Pages & List Pages

Wrappers can be learned across several detail pages:



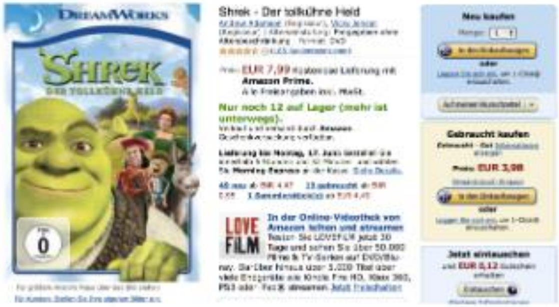
Shrek - Der tollkühne Held Top 5000
(2001)

Your rating:

7.9

An ogre, in order to regain his swamp, travels along with an annoying donkey in order to bring a princess to a scheming lord, wishing himself King.

Directors: Andrew Adamson, Vicky Jenson
Writers: William Steig (book), Ted Elliott (screenplay), 5 more credits »
Stars: Mike Myers, Eddie Murphy, Cameron Diaz | See full cast and crew



Shrek - Der tollkühne Held

Preis: EUR 7,99 Kostenlose Lieferung mit **Amazon Prime.**
Alle Preisangaben inkl. MwSt.

Nur noch 12 auf Lager (mehr ist unterwegs).

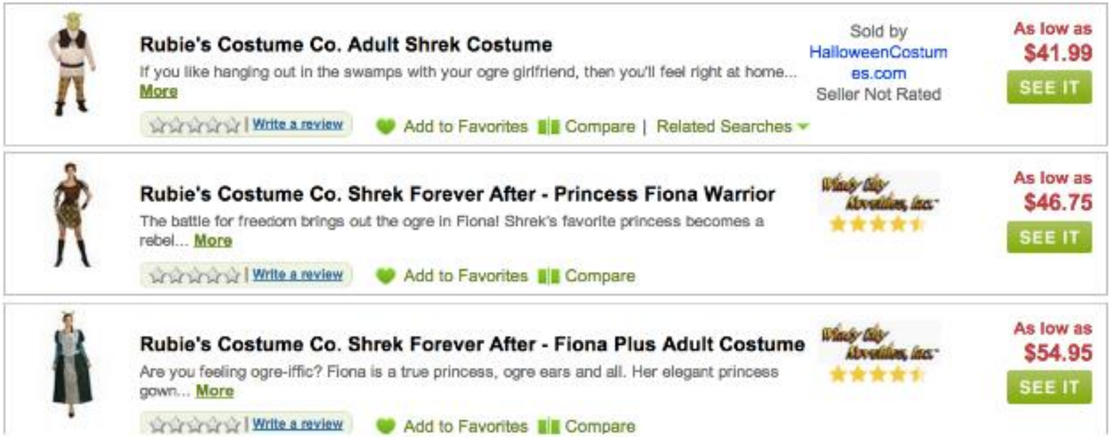
In der Online-Videothek von Amazon leihen und streamen

Neu kaufen
oder

Gebraucht kaufen
Preis: EUR 3,98
oder

Jetzt eintauschen
und EUR 0,12 Gutschein erhalten
Eintauschen

Wrappers can also be learned across items in a list:



Rubie's Costume Co. Adult Shrek Costume
If you like hanging out in the swamps with your ogre girlfriend, then you'll feel right at home... More
Sold by HalloweenCostumes.com
Seller Not Rated
As low as $41.99
SEE IT
☆☆☆☆☆ | Write a review   Add to Favorites   Compare | Related Searches

Rubie's Costume Co. Shrek Forever After - Princess Fiona Warrior
The battle for freedom brings out the ogre in Fiona! Shrek's favorite princess becomes a rebel... More
Windy City Novelties, Inc.
As low as $46.75
SEE IT
☆☆☆☆☆ | Write a review   Add to Favorites   Compare

Rubie's Costume Co. Shrek Forever After - Fiona Plus Adult Costume
Are you feeling ogre-iffic? Fiona is a true princess, ogre ears and all. Her elegant princess gown... More
Windy City Novelties, Inc.
As low as $54.95
SEE IT
☆☆☆☆☆ | Write a review   Add to Favorites   Compare

Data may exhibit structure

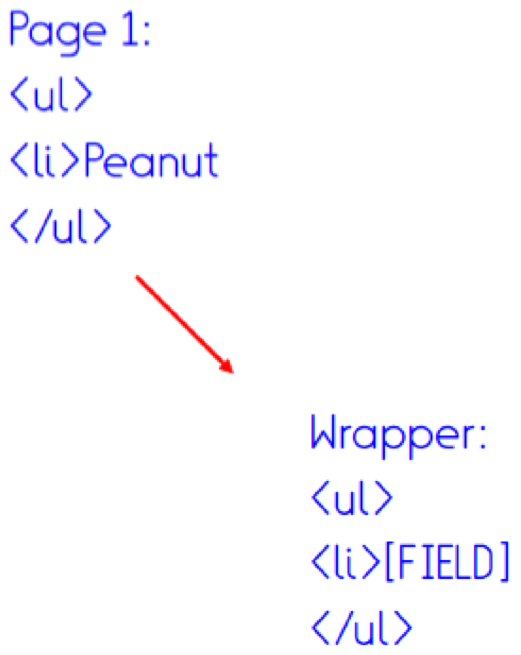
```
Dronkeys:
 <ul>
 <li>Eclair: female
 <li>Bananas: flexible
 </ul>
Shrek's kids:
 <ul>
 <li>Farkle: male
 <li>Fergus: male
```

```
family: tuple (
 name: string
 children: set (
   child: tuple (name: string,
                 gender: string)))
```

ROADRUNNER: Learn types

ROADRUNNER is a system that can learn the Web page structure.

Finds least upper bounds in regex lattice



Page 1:
<ul>
<li>Peanut
</ul>

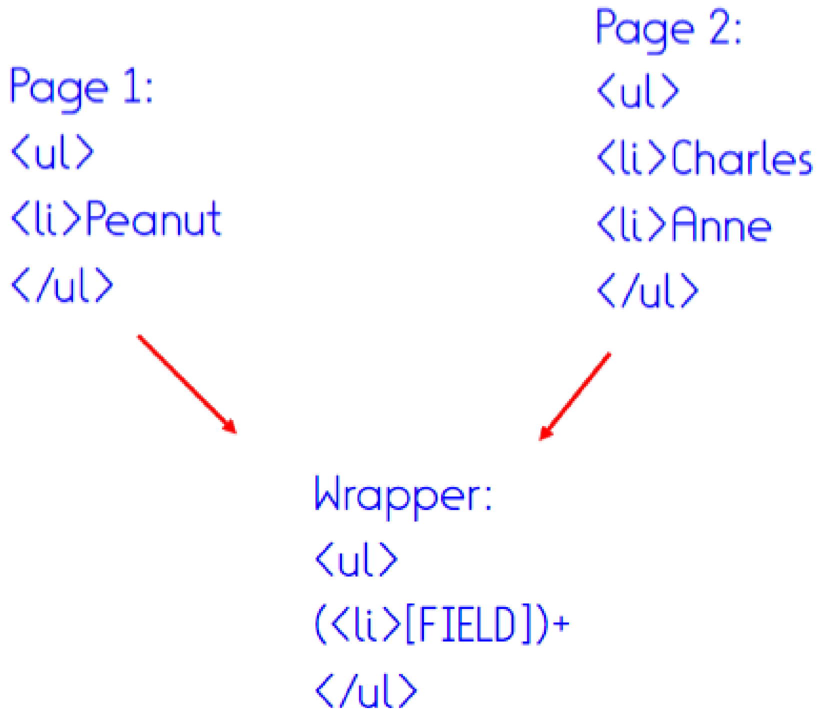
Page 2:
<ul>
<li>Charles
</ul>

Wrapper:
<ul>
<li>[FIELD]
</ul>

Crescenzi et al., VDLB 2001
http://www.vldb.org/conf/2001/P109.pdf

ROADRUNNER: Learn types

ROADRUNNER is a system that can learn the Web page structure.



Page 1:

```
<ul>
<li>Peanut
</ul>
```

Page 2:

```
<ul>
<li>Charles
<li>Anne
</ul>
```

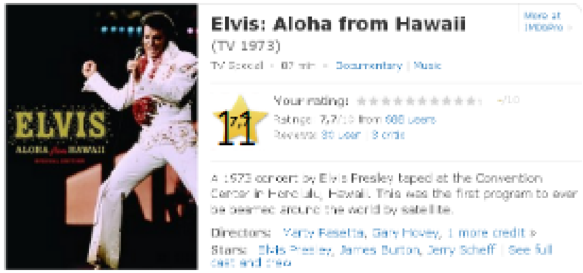
Wrapper:

```
<ul>
(<li>[FIELD])+
</ul>
```

Crescenzi et al., VDLB 2001
http://www.vldb.org/conf/2001/P109.pdf

Def: Wrapper Application

Wrapper application is the process of extracting its strings from a Web page.



Elvis: Aloha from Hawaii

More at IMDbPro »

(TV 1973)

TV Special - 87 min - Documentary | Music

Your rating: -/10

7,7

Ratings: 7,7/10 from 608 users

Reviews: 30 user | 8 critic

A 1973 concert by Elvis Presley taped at the Convention Center in Honolulu, Hawaii. This was the first program to ever be beamed around the world by satellite.

Directors: Marty Pasetta, Gary Hovey, 1 more credit »

Stars: Elvis Presley, James Burton, Jerry Scheff | See full cast and crew

+

/html/body/h1

/html/body/p[2]/i

=

"Elvis", "11"

Web page

+

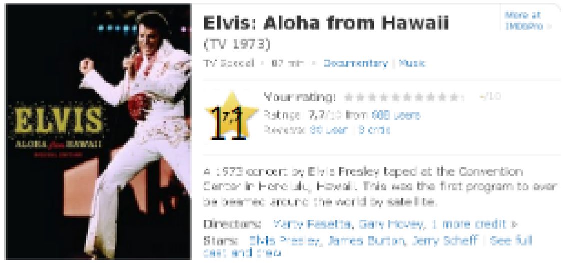
Wrapper

=

Strings

Def: Wrapper Application

Wrapper application is the process of extracting its strings from a Web page.



Elvis: Aloha from Hawaii

(TV 1973)

TV Special · 87 min · Documentary | Music

More at IMDbPro »

Your rating: -/10

Ratings: 7,7/10 from 698 users

Reviews: 30 user | 9 critic

A 1973 concert by Elvis Presley taped at the Convention Center in Honolulu, Hawaii. This was the first program to ever be beamed around the world by satellite.

Directors: Marty Pasetta, Gary Hovey, 1 more credit »

Stars: Elvis Presley, James Burton, Jerry Scheff | See full cast and crew

Web page

+

/html/body/h1

/html/body/p[2]/i

+

Wrapper

=

"Elvis", "11"

=

Strings

↓ Disambiguation, + relation ↓

hasActor(e42, ElvisPresley)

hasRating(e42, "11.0")

Facts

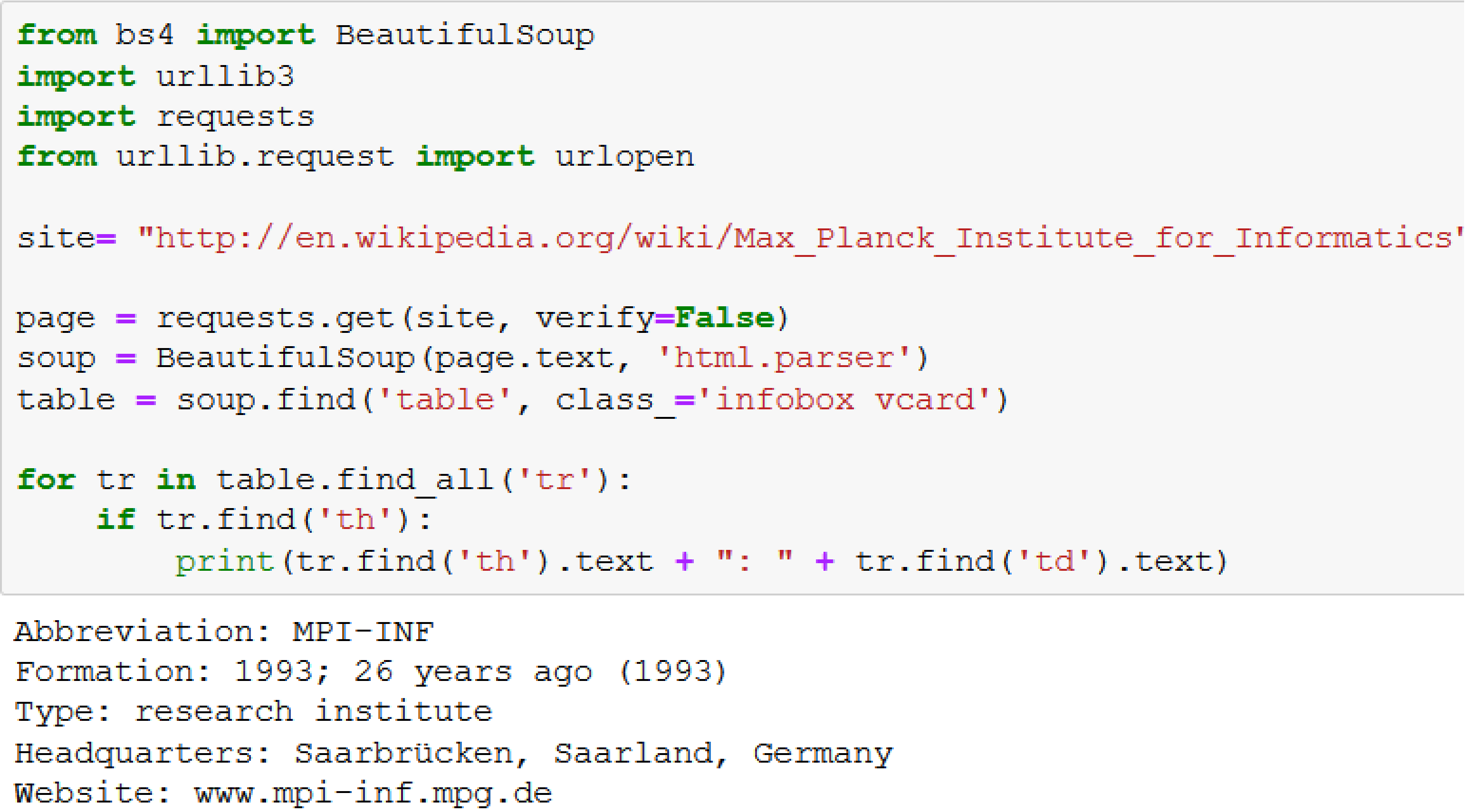
Scraping in Python – BeautifulSoup (1)

- Python library for pulling data out of HTML and XML files.

```
<html>
 <head>
  <title>
   The Dormouse's story
  </title>
 </head>
 <body>
   Once upon a time there were
three little sisters; and their names
were  <a class="sister"
href="http://example.com/elsie"
id="link1">Elsie</a> , <a
class="sister"
href="http://example.com/lacie"
id="link2"> Lacie</a> and …
```

```
soup.title
# <title>The Dormouse's story</title>

soup.title.string
# u'The Dormouse's story'

soup.title.parent.name
# u‘head'

soup.a
# <a class="sister" href="http://ex.com/elsie"
id="link1">Elsie</a>

soup.find_all('a')
# [<a class="sister" href="http://ex.com/elsie"
id="link1">Elsie</a>,
#  <a class="sister" href="http://ex.com/lacie"
id="link2">Lacie</a>,
#  <a class="sister" href="http://ex.com/tillie"
id="link3">Tillie</a>]
```

Scraping in Python – BeautifulSoup (2)



```
from bs4 import BeautifulSoup
import urllib3
import requests
from urllib.request import urlopen

site= "http://en.wikipedia.org/wiki/Max_Planck_Institute_for_Informatics'

page = requests.get(site, verify=False)
soup = BeautifulSoup(page.text, 'html.parser')
table = soup.find('table', class_='infobox vcard')

for tr in table.find_all('tr'):
    if tr.find('th'):
        print(tr.find('th').text + ": " + tr.find('td').text)
```

```
Abbreviation: MPI-INF
Formation: 1993; 26 years ago (1993)
Type: research institute
Headquarters: Saarbrücken, Saarland, Germany
Website: www.mpi-inf.mpg.de
```

XPath vs. BeautifulSoup vs …

- XPath: Generic query language to select nodes in XML (HTML) documents
  - Queries can be issued from Python, Java, C, …

- BeautifulSoup
  - Python library to manipulate websites as Python objects

- Scrapy
  - Python library to crawl websites

- Selenium
  - Actual scripted browser interaction
  → To get around Javascript etc.

https://www.udemy.com/tutorial/scrapy-tutorial-web-scraping-with-python/scrapy-vs-beautiful-soup-vs-selenium/

Assignment 3

- No crawling (ethics…)
- 1x Extraction from dump – infobox treasure
  - Remember design considerations
  - XML format, but essential content not structured by XML tags → pattern matching/regex
- 2x Scraping
  - BeautifulSoup recommended, but XPath fine as well

- Reading on large-scale WP extraction: DBpedia extraction framework

Take home

1. Think about goal, sources, methods
2. Crawling
    - BFS to achieve coverage
    - Challenges with traps and deep web
3. Scraping
    - Reverse-engineering of template-based websites