Information Extraction, the Semantic Web, and all that

3 hours in the course “Technologies du Web”
at the École nationale supérieure des Télécommunications
in Paris/France in fall 2010

by Fabian M. Suchanek

This document is available under a Creative Commons Attribution Non-Commercial License
Organisation

• 2h class on Information extraction

• Short break

• 1h class on the Semantic Web

• Web-site: http://suchanek.name/ → Teaching
Motivation

Elvis Presley
1935 - 1977

Will there ever be someone like him again?
Motivation

Elvis Presley: The Early Years
Elvis spent more weeks at the top of the charts than any other artist.
www.fiftiesweb.com/elvis.htm
Motivation

Another singer called Elvis, young

Personal relationships of Elvis Presley – Wikipedia
...when Elvis was a **young** teen.... another girl whom the singer's mother hoped Presley would .... The writer called Elvis "a hillbilly cat"

en.wikipedia.org/.../Personal_relationships_of_Elvis_Presley
Motivation

SELECT * FROM person
WHERE gName='Elvis'
AND occupation='singer'

<table>
<thead>
<tr>
<th>GName</th>
<th>FName</th>
<th>Occupation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis</td>
<td>Presley</td>
<td>singer</td>
</tr>
<tr>
<td>Elvis</td>
<td>Hunter</td>
<td>painter</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

1: Elvis Presley
2: Elvis ...
3. Elvis ...

Information Extraction

Another Elvis

Google
Motivation: Definition

**Information Extraction** (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

Elvis Presley was a famous rock singer. ...
Mary once remarked that the only attractive thing about the painter Elvis Hunter was his first name.
Motivation: Examples

<table>
<thead>
<tr>
<th>Title</th>
<th>Type</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business strategy Associate</td>
<td>Part time</td>
<td>Palo Alto, CA</td>
</tr>
<tr>
<td>Registered Nurse</td>
<td>Full time</td>
<td>Los Angeles</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Motivation: Examples

Information Extraction: Techniques and Challenges

Ralph Grishman

1 Introduction

This volume takes a broad view of information extraction and filtering information from large volumes of text.

<table>
<thead>
<tr>
<th>Author</th>
<th>Publication</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grishman</td>
<td>Information Extraction...</td>
<td>2006</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Information Integration Papers**

- **Answering Queries Using Templates With Binding Patterns.** In PODS 1995, specify binding patterns.
- **Querying Semistructured, Heterogeneous Information** (with Dallan Quass, A semantics. Also, a **A shorter Version** that appeared in DOOD '95.)
Motivation: Examples

<table>
<thead>
<tr>
<th>Product</th>
<th>Type</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynex 32&quot; LCD TV</td>
<td>$1000</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>
**Information Extraction** (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

<table>
<thead>
<tr>
<th>Person</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis Presley</td>
<td>American</td>
</tr>
</tbody>
</table>

**Instance Extraction**

- Elvis Presley: singer
- Angela Merkel: politician

**Fact Extraction**

**Ontological Information Extraction**

And beyond
**Instance Extraction** is the process of extracting entities with their **class** (i.e., concept, set of similar entities)

Elvis was a great artist, but while all of Elvis’ colleagues loved the song “Oh yeah, honey”, Elvis did not perform that song at his concert in Hintertuepflingen.

<table>
<thead>
<tr>
<th>Entity</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis</td>
<td>artist</td>
</tr>
<tr>
<td>Oh yeah, honey</td>
<td>song</td>
</tr>
<tr>
<td>Hintertuepflingen</td>
<td>location</td>
</tr>
</tbody>
</table>

...some of the class assignment might already be done by the Named Entity Recognition.
Instance Extraction: Hearst Patterns

**Instance Extraction** is the process of extracting entities with their class (i.e., concept, set of similar entities)

<table>
<thead>
<tr>
<th>Entity</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis</td>
<td>artist</td>
</tr>
</tbody>
</table>

**Elvis was a great artist, but while all of Elvis’ colleagues loved the song “Oh yeah, honey”, Elvis did not perform that song at his concert in Hintertüpflingen.**

**Idea (by Hearst):**

Sentences express class membership in very predictable patterns. Use these patterns for instance extraction.

**Hearst patterns:**

- X was a great Y
Elvis was a great artist.

Many scientists, including Einstein, started to believe that matter and energy could be equated.

He adored Madonna, Celine Dion and other singers, but never got an autograph from any of them.

Many US citizens have never heard of countries such as Guinea, Belize or Germany.

Idea (by Hearst):

Sentences express class membership in very predictable patterns. Use these patterns for instance extraction.

Hearst patterns:

- X was a Y
- Ys, such as X1, X2, ...
- X1, X2, ... and other Y
- many Ys, including X,
Instance Extraction: Hearst Patterns

Hearst Patterns on Google

"cities such as"
About 5,300,000 results (0.43 seconds)

* News for "cities such as"
  Unknown Cities Are Getting Richer - 23 hours ago
  Cities such as Aurangabad, Curitiba in Brazil, Xiaochang in China, and lumped together, BCG found, with the mostly poor, ...
  BusinessWeek - 3 related articles

Cities That Could Steal Your Job: New Outsourcing Hot Sp
From overlooked American cities such as Boise, Idaho and Winnipeg t like Cluj-Napoca, Romania, or the Philippines' Iloilo City, ...
images.businessweek.com/ss/09/05/0504_outsourcing.../1.htm - Cache

Wildcards on Google

"many *, including **"
About 1,670,000,000 results (0.19 seconds)

* Putco 401127 Chrome Trim Mirror Covers, Fits many Fords including ...
  Fits many Fords including the F-150, F-250 Super Duty, and many more from 1999 to 200
  Brand: Putco, Mfr Part#: 401127. Lowest Price $72.89 ...
  www.streetperformance.com/part/.../869788-401127.html - Cached - Similar

Skyfire Mobile Browser closed down in many countries including ...
  1 Jul 2010 ... Skyfire Mobile Browser closed down in many countries including Pakistan.
  3rd. Share/Bookmark. No comments. Skyfire, the web browser with ...
pakistannewsblog.com/skyfire-mobile-browser-closed-down-in-many-countries-including-
pakistan/ - Pakistan - Cached

Idea (by Hearst):
Sentences express class membership in very predictable patterns. Use these patterns for instance extraction.

Hearst patterns:
- X was a Y
- Ys, such as X1, X2, ...
- X1, X2, ... and other Y
- many Ys, including X,
Instance Extraction: Hearst Patterns

Hearst Patterns can extract instances from natural language documents.

Input:
• Hearst patterns for the language (easily available for English)

Condition:
• Text documents contain class + entity explicitly in defining phrases

Idea (by Hearst):

Sentences express class membership in very predictable patterns. Use these patterns for instance extraction.

Hearst patterns:
• X was a Y
• Ys, such as X1, X2, ...
• X1, X2, ... and other Y
• many Ys, including X,
Instance Extraction: Set Expansion

Seed set: {Russia, USA, Australia}

Result set: {Russia, Canada, China, USA, Brazil, Australia, India, Argentina, Kazakhstan, Sudan}
Instance Extraction: Set Expansion

**Most corrupt countries**

<table>
<thead>
<tr>
<th>Rank</th>
<th>Country</th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>174</td>
<td>Uzbekistan</td>
<td>1.7</td>
<td>1.8</td>
<td>1.7</td>
</tr>
<tr>
<td>175</td>
<td>Chad</td>
<td>1.6</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>176</td>
<td>Iraq</td>
<td>1.5</td>
<td>1.3</td>
<td>1.5</td>
</tr>
<tr>
<td>176</td>
<td>Sudan</td>
<td>1.5</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>178</td>
<td>Myanmar</td>
<td>1.4</td>
<td>1.3</td>
<td>1.4</td>
</tr>
<tr>
<td>179</td>
<td>Afghanistan</td>
<td>1.3</td>
<td>1.5</td>
<td>1.8</td>
</tr>
<tr>
<td>180</td>
<td>Somalia</td>
<td>1.1</td>
<td>1.0</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Result set: {Russia, Canada, China, USA, Brazil, Australia, India, Argentina, Kazakhstan, Sudan}
Instance Extraction: Set Expansion

Seed set: {Russia, Canada, China, USA, Brazil, Australia, India, Argentina, Kazakhstan, Sudan}

Most corrupt countries

<table>
<thead>
<tr>
<th>Rank</th>
<th>Country</th>
<th>Score 1</th>
<th>Score 2</th>
<th>Score 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>174</td>
<td>Uzbekistan</td>
<td>1.7</td>
<td>1.8</td>
<td>1.7</td>
</tr>
<tr>
<td>175</td>
<td>Chad</td>
<td>1.6</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>176</td>
<td>Iraq</td>
<td>1.5</td>
<td>1.3</td>
<td>1.5</td>
</tr>
<tr>
<td>176</td>
<td>Sudan</td>
<td>1.5</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>178</td>
<td>Myanmar</td>
<td>1.4</td>
<td>1.3</td>
<td>1.4</td>
</tr>
<tr>
<td>179</td>
<td>Afghanistan</td>
<td>1.3</td>
<td>1.5</td>
<td>1.8</td>
</tr>
<tr>
<td>180</td>
<td>Somalia</td>
<td>1.1</td>
<td>1.0</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Result set: {Uzbekistan, Chad, Iraq,...}

Try, e.g., Google sets: [http://labs.google.com/sets](http://labs.google.com/sets)
Instance Extraction: Set Expansion

Set Expansion can extract instances from tables or lists.

Input:
- seed pairs

Condition:
- a corpus full of tables

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>1.7</th>
<th>1.8</th>
<th>1.7</th>
</tr>
</thead>
<tbody>
<tr>
<td>174</td>
<td>Uzbekistan</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>Chad</td>
<td>1.6</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>176</td>
<td>Iraq</td>
<td>1.5</td>
<td>1.3</td>
<td>1.5</td>
</tr>
<tr>
<td>176</td>
<td>Sudan</td>
<td>1.5</td>
<td>1.6</td>
<td>1.8</td>
</tr>
<tr>
<td>178</td>
<td>Myanmar</td>
<td>1.4</td>
<td>1.3</td>
<td>1.4</td>
</tr>
<tr>
<td>179</td>
<td>Afghanistan</td>
<td>1.3</td>
<td>1.5</td>
<td>1.8</td>
</tr>
<tr>
<td>180</td>
<td>Somalia</td>
<td>1.1</td>
<td>1.0</td>
<td>1.4</td>
</tr>
</tbody>
</table>
Instance Extraction: Evaluation

In science, every system, algorithm or theory should be evaluated, i.e. its output should be compared to the gold standard (i.e. the ideal output)

Algorithm output:
\[ O = \{Einstein, Bohr, Planck, Clinton, Obama\} \]

Gold standard:
\[ G = \{Einstein, Bohr, Planck, Heisenberg\} \]

Precision:
What proportion of the output is correct?

\[
\begin{array}{c|c|c|c|c|c}
| O \land G | \\
| O | & | G |
\end{array}
\]

Recall:
What proportion of the gold standard did we get?

\[
\begin{array}{c|c|c|c|c|c}
| O \land G | \\
| G |
\end{array}
\]
Instance Extraction: Evaluation

**Explorative** algorithms extract everything they find. (very low threshold)

Algorithm output:
O = \{Einstein, Bohr, Planck, Clinton, Obama, Elvis, Heisenberg, \ldots\}

Gold standard:
G = \{Einstein, Bohr, Planck, Heisenberg\}

**Precision:**
What proportion of the output is correct?

BAD

**Recall:**
What proportion of the gold standard did we get?

GREAT
Instance Extraction: Evaluation

Conservative algorithms extract only things about which they are very certain (very high threshold)

Algorithm output:
O = \{Einstein\}

Gold standard:
G = \{Einstein, Bohr, Planck, Heisenberg\}

Precision:
What proportion of the output is correct?

GREAT

Recall:
What proportion of the gold standard did we get?

BAD
Instance Extraction: Evaluation

You can’t get it all...

The F1-measure combines precision and recall as the harmonic mean:

\[ F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \]
Instance Extraction

**Instance Extraction** is the process of extracting entities with their **class** (i.e., concept, set of similar entities)

Approaches:
- Hearst Patterns (work on natural language corpora)
- Set Expansion (for tables and lists)
- ...many others...

On top of that:
- Iteration
- Cleaning

And finally:
- Evaluation
Information Extraction (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

<table>
<thead>
<tr>
<th>Person</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis Presley</td>
<td>American</td>
</tr>
</tbody>
</table>

Elvis Presley: singer
Angela Merkel: politician
Fact Extraction

**Fact Extraction** is the process of extracting pairs (triples,...) of entities together with the relationship of the entities.

<table>
<thead>
<tr>
<th>Event</th>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costello sings</td>
<td>2010-10-01, 23:00</td>
<td>Great American Music Hall, San Francisco, CA</td>
</tr>
</tbody>
</table>

10/1/2010 Friday 11:00p

Costello Sings Lowe/Nick Sings Elvis (late show)
THE BAND: Paul Revelli, Ruth Davies, Bill Kirchen, Bob Andrews, Derek Huston, Austin...
Fact Extraction: Wrapper Induction

Observation: On Web pages of a certain domain, the information is often in the same spot.
Fact Extraction: Wrapper Induction

Observation: On Web pages of a certain domain, the information is often in the same spot.

Idea: Describe this spot in a general manner.
A description of one spot or multiple spots on a page is called a wrapper.

A wrapper can be similar to an XPath expression:

```
```

It can also be a search text/regex

```
>.*</b>(TV
```
Fact Extraction: Wrapper Induction

We manually label the fields to be extracted, and produce the corresponding wrappers (usually with a GUI tool).
Fact Extraction: Wrapper Induction

We manually label the fields to be extracted, and produce the corresponding wrappers.

Then we **apply** the wrappers to all pages in the domain (i.e., we determine the spots of the pages that the wrappers point to).

<table>
<thead>
<tr>
<th>Title</th>
<th>Rating</th>
<th>ReleaseDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Titanic</td>
<td>7.4</td>
<td>1998-01-07</td>
</tr>
</tbody>
</table>
Fact Extraction: Wrapper Induction

Wrapper induction can extract entities and relations from a set of similarly structured pages.

Input:
• Choice of the domain
• (Human) labeling of some pages
• Wrapper design choices

Condition:
• All pages are of the same structure

Can the wrapper say things like
  “The last child element of this element”
  “The second element, if the first element contains XYZ”

If so, how do we generalize the wrapper?
Fact Extraction: Pattern Matching

Known facts *(seed pairs)*

<table>
<thead>
<tr>
<th>Person</th>
<th>Discovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Einstein</td>
<td>K68</td>
</tr>
</tbody>
</table>

The patterns can either
- be specified by hand
- or come from annotated text
- or come from seed pairs + text

X ha scoperto il Y

Bohr ha scoperto il K69 nel anno 1960.
Fact Extraction: Pattern Matching

Einstein ha scoperto il K68, quando aveva 4 anni.

Bohr ha scoperto il K69 nel anno 1960.

The patterns can be more complex, e.g.
- regular expressions
  X discovered the .{0,20} Y
- POS patterns
  X discovered the ADJ? Y
- Parse trees

<table>
<thead>
<tr>
<th>Person</th>
<th>Discovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Einstein</td>
<td>K68</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Person</th>
<th>Discovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bohr</td>
<td>K69</td>
</tr>
</tbody>
</table>

Try
Fact Extraction: Pattern Matching

Einstein ha scoperto il K68, quando aveva 4 anni.

Bohr ha scoperto il K69 nel anno 1960.

<table>
<thead>
<tr>
<th>Person</th>
<th>Discovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Einstein</td>
<td>K68</td>
</tr>
<tr>
<td>Bohr</td>
<td>K69</td>
</tr>
</tbody>
</table>

First system to use iteration: Snowball

Watch out for semantic drift: Einstein liked the K68
Fact Extraction: Pattern Matching

Einstein ha scoperto il K68, quando aveva 4 anni.

Pattern matching can extract facts from natural language text corpora.

Input:
• a known relation
• seed pairs or labeled documents or patterns

Condition:
• The texts are homogenous (express facts in a similar way)
• Entities that stand in the relation do not stand in another relation as well
Fact Extraction: Pattern Matching

With 97% of the votes counted, it is now certain that Brazil’s presidential race will go to a second round. Dilma, made an unexpectedly poor showing, at just over 46% of all votes counted so far. That will rise a smidgen, but her expected gains there will not be enough to secure an absolute n

Try this out:
http://viewer.opencalais.com/
Fact Extraction: Cleaning

Fact Extraction commonly produces huge amounts of garbage.

- Web page contains bogus information
- Deviation in iteration
- Formatting problems (bad HTML, character encoding mess)
- Regularity in the training set that does not appear in the real world
- Web page contains misleading items (advertisements, error messages)
- Different thematic domains or Internet domains behave in a completely different way
- Something has changed over time (facts or page formatting)

⇒ Cleaning is usually necessary, e.g., through thresholding or heuristics
Fact Extraction is the process of extracting pairs (triples,...) of entities together with the relationship of the entities.

Approaches:
• Fact extraction from tables (if the corpus contains lots of tables)
• Wrapper induction (for extraction from one Internet domain)
• Pattern matching (for extraction from natural language documents)
• ... and many others...
Information Extraction (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

<table>
<thead>
<tr>
<th>Person</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis Presley</td>
<td>American</td>
</tr>
<tr>
<td>Angela Merkel</td>
<td>politician</td>
</tr>
</tbody>
</table>
Ontological Information Extraction (IE) tries to create or extend an ontology through information extraction.

Angela Merkel is the German chancellor.
...Merkel was born in Germany.
...A. Merkel has French nationality.

<table>
<thead>
<tr>
<th>Person</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angela Merkel</td>
<td>German</td>
</tr>
<tr>
<td>Merkel</td>
<td>Germany</td>
</tr>
<tr>
<td>A. Merkel</td>
<td>French</td>
</tr>
</tbody>
</table>
Ontological Information Extraction (IE) tries to create or extend an ontology through information extraction.

Challenges:
1. Map entity names to ontological entities
2. Disambiguate entity names
3. Use the relationships from the ontology
4. Make the ontology consistent

Wikipedia is a free online encyclopedia
- 3.4 million articles in English
- 16 million articles in dozens of languages

Why is Wikipedia good for information extraction?
- It is a huge, but homogenous resource (more homogenous than the Web)
- It is considered authoritative and covers many different aspects (more authoritative than a random Web page)
- It is well-structured with infoboxes and categories
- It provides a wealth of meta information (inter article links, inter language links, user discussion,...)

Elvis Presley

~Infobox~
Born: 1935
...

Categories: Rock singers

Exploit Infoboxes

Elvis Presley

Categories: Rock singers

~Infobox~
Born: 1935
...

Rock Singer

type

born

1935

Exploit Infobloxes
Exploit conceptual categories
Exploit conceptual categories

Elvis Presley

Born: 1935

Exploit Infoboxes

categories: Rock singers

WordNet

Person

subclassOf

Singer

~Infobox~

Born: 1935

...
Elvis Aaron Presley (January 8, 1935 – August 16, 1977) was one of the most popular American singers of the 20th century....

Example: Elvis in YAGO

YAGO
- 3m entities, 28m facts
- focus on precision 95%
  (automatic checking of facts, manual relations, link with WordNet)
  http://mpii.de/yago

DBpedia
- 3.4m entities
- 1b facts (also from non-English Wikipedia)
- large community
  http://dbpedia.org

Freebase
Community project on top of Wikipedia data
(bought by Google, but still open)
http://freebase.com

KYLIN/KOG
Part of the Intelligence in Wikipedia project
Ontological IE: Reasoning

Goal:
Extract ontological information from natural language documents

"Elvis was born in 1935"

Main Challenges:
• deliver canonic relations
• deliver canonic entities
• deliver consistent facts

died in, perished in, was killed in
Elvis, Elvis Presley, The King
born (Elvis, 1970)
born (Elvis, 1935)

Idea: These problems are interleaved, solve all of them together.
Ontological IE: Reasoning

Ontology

First Order Logic Formulae

type(Elvis_Presley,singer)
subclassof(singer,person)
...
appears("Elvis","was born in","1935")
...
means("Elvis",Elvis_Presley,0.8)
means("Elvis",Elvis_Costello,0.2)
...
born(X,Y) & died(X,Z) => Y<Z
appears(A,P,B) & R(A,B) => expresses(P,R)
appears(A,P,B) & expresses(P,R) => R(A,B)
...

New facts with
1. canonic relations
2. canonic entities
3. consistency with the ontology

SOFIE system
Ontological IE: Reasoning

Reasoning-based approaches use logical rules to extract knowledge from natural language documents.

Current approaches use either
- Weighted MAX SAT
- or Datalog
- or Markov Logic

Input:
- often an ontology
- manually designed rules

Condition:
- homogeneous corpus helps
Ontological Information Extraction (IE) tries to create or extend an ontology through information extraction.

Current hot approaches:
• extraction from Wikipedia
• reasoning-based approaches
**Information Extraction** (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).
Open Information Extraction

**Information Extraction** (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

**Open Information Extraction/Machine Reading/Macro Reading** aims at information extraction from the entire Web.

Vision of Open Information Extraction:
- the system runs perpetually, constantly gathering new information
- the system creates meaning on its own from the gathered data
- the system learns and becomes more intelligent, i.e. better at gathering information

Rationale for Open Information Extraction:
- We do not need to care for every single sentence, but just for the ones we understand
- The size of the Web generates redundancy
- The size of the Web can generate synergies
Open IE: KnowItAll &Co

KnowItAll, KnowItNow and TextRunner are projects at the University of Washington (in Seattle, WA).

Subject | Verb | Object | Count
--- | --- | --- | ---
Egyptians | built | pyramids | 400
Americans | built | pyramids | 20
... | ... | ... | ...

Valuable common sense knowledge (if filtered)

http://www.cs.washington.edu/research/textrunner/
Open IE: KnowItAll &Co

KnowItAll, KnowItNow and TextRunner are projects at the University of Washington (in Seattle, WA).

TextRunner took .80 seconds.

Retrieved 391 results for Predicate containing "built" and Argument 2 containing "pyramids"

Grouping results by predicate. Group by: argument 2 | argument 1

built - 159 results

Egyptians (297), aliens (71), Pharaohs (40), built the pyramids
Egyptians (26), Khufu (18), Maya (9), 30 more... built the Great Pyramid
Imhotep (8), Pharaoh Zoser (4), Egyptians (2), King Djoser (2) built the Step Pyramid
two symbols of life (4), 6th dynasty kings (3), King Seneferu (3), Snefru (3) built two large Pyramids
Egyptians (8) built the Great Pyramids
ancient Egyptians (6) built more than 90 royal pyramids
colonial silver city of Taxco (3), Explore (2) built the gigantic pyramids of the Sun
Central America (2), part of Mexico (2) built great cities, temples and pyramids

http://www.cs.washington.edu/research/textrunner/
Open IE: Read the Web

“Read the Web” is a project at the Carnegie Mellon University in Pittsburgh, PA.

Initial Ontology

Natural Language Pattern Extractor

Krzewski coaches the Blue Devils.

Table Extractor

Krzewski  Blue Devils
Miller    Red Angels

Mutual exclusion Learner
sports coach != scientist ?

Rule Learner
coaches => is paid by?

Type Check Learner
If I coach, am I a coach?

http://rtw.ml.cmu.edu/rtw/
Open IE: Read the Web

http://rtw.ml.cmu.edu/rtw/
Open Information Extraction

Open Information Extraction/Machine Reading/Macro Reading aims at information extraction from the entire Web.

Main hot projects
• TextRunner
• Read the Web

Input:
• The Web
• Read the Web: Manual rules
• Read the Web: initial ontology

Conditions
• none
Information Extraction (IE) is the process of extracting structured information (e.g., database tables) from unstructured machine-readable documents (e.g., Web documents).

<table>
<thead>
<tr>
<th>Person</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis Presley</td>
<td>American</td>
</tr>
<tr>
<td>Angela Merkel</td>
<td>politician</td>
</tr>
</tbody>
</table>

✓ Elvis Presley singer
✓ Angela Merkel politician

and beyond

Ontological Information Extraction

nationality

Germany flag
Semantic Web: Motivation

We just saw how to move from unstructured data to structured data

But even between structured data, interaction is difficult...

<table>
<thead>
<tr>
<th>Person</th>
<th>Occupation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elvis P.</td>
<td>singer</td>
</tr>
</tbody>
</table>
Semantic Web: Motivation

We just saw how to move from unstructured data to structured data.

But even between structured data, interaction is difficult, in particular if the data is in different formats.

We just saw how to move from unstructured data to structured data.
Semantic Web: Motivation

We just saw how to move from unstructured data to structured data.

But even between structured data, interaction is difficult, in particular if the data is in different formats, on different machines or devices.
Semantic Web: Motivation

We just saw how to move from unstructured data to structured data. But even between structured data, interaction is difficult, in particular if the data is in different formats, on different machines or devices or in different companies.
Motivation: Use cases

Examples:

• Booking a flight
  Interaction between office computer, flight company, travel agency, shuttle services, hotel, my calendar

• Finding a restaurant
  Interaction between mobile device, map service, recommendation service, restaurant reservation service

• Web search
  Interaction between client, search service, Web page content provider

• Web service composition
  Interaction between client and Web services and Web services themselves

• Intelligent home
  Fridge knows my calendar, orders food if I am planning a dinner

• Intelligent cars
  Car knows my schedule, where and when to get gas, how not to hit other cars, what are the legal regulations
The Semantic Web

Goals:
• make computers „understand“ the data they store
• allow them to answer „semantic“ queries
• allow them to share information across different systems

Techniques: (= this class)
• defining semantics in a machine-readable way (RDFS)
• identifying entities in a globally unique way (URIs)
• defining logical consistency in a uniform way (OWL)
• linking together existing resources (Linked Data)

The Semantic Web is an evolving extension of the World Wide Web, in which data is made available in one standardized semantic format.
RDF (the Resource Description format) is a format of knowledge representation that is similar to the Entity-Relationship-Model.

“Elvis was born in Tupelo”

RDF is used as the only knowledge representation language. => All information is represented in a simple, homogeneous, computer-processable way.

http://www.w3.org/TR/rdf-primer/
URIs

A **URI** (uniform resource identifier) is similar to a URL, but it is not necessarily downloadable. It identifies a concept uniquely.

![Diagram](image.png)

```
Elvis:  http://elvis.org/Myself
bornIn: http://mpii.de/yago/resource/bornIn
Tupelo: http://tupelo.com
```

URIs are used as globally unique identifiers for resources.

**=>** Knowledge can be interlinked. A knowledge base on one server can refer to concepts from another knowledge base on another server.

http://www.ietf.org/rfc/rfc3986.txt
Name Spaces

A **namespace** is a shorthand notation for the first part of a URI.

Our statement is a triple of 3 URIs -- quite verbose


Namespaces make our statement much less verbose

Namespace  e :=  http://elvis.org/
Namespace  yago :=  http://mpii.de/yago/

e:Myself  yago:bornIn  http://tupelo.com

Namespaces are used to abbreviate URIs

=> Namespaces with useful concepts can become popular.
   This facilitates a common vocabulary across different knowledge bases.
Name Spaces: Sharing

If two RDF graphs share one node, they are actually one RDF graph.

A machine can follow the links and retrieve more information in the neighboring ontology.
A number of standard vocabularies have evolved

- **rdf:** The basic RDF vocabulary
  
  [http://www.w3.org/1999/02/22-rdf-syntax-ns#](http://www.w3.org/1999/02/22-rdf-syntax-ns#)

- **rdfs:** RDF Schema vocabulary
  
  [http://www.w3.org/1999/02/22-rdf-syntax-ns#](http://www.w3.org/1999/02/22-rdf-syntax-ns#)

- **dc:** Dublin Core (predicates for describing documents)
  
  [http://purl.org/dc/elements/1.1/](http://purl.org/dc/elements/1.1/)

- **foaf:** Friend Of A Friend (predicates for relationships between people)
  
  [http://xmlns.com/foaf/0.1/](http://xmlns.com/foaf/0.1/)

- **cc:** Creative Commons (types of licences)
  
  [http://creativecommons.org/ns#](http://creativecommons.org/ns#)
Sharing: Creative Commons

A number of standard vocabularies have evolved

cc: Creative Commons (types of licences)
http://creativecommons.org/ns#

Creative Commons is a non-profit organization, which defines very popular licenses, notably
- CC-BY: Free for reuse, just give credit to the author
- CC-BY-NC: Free for reuse, give credit, non-commercial use only
- CC-BY-ND: Free for reuse, give credit, do not create derivative works

Try this
A **class** (also called concept) can be understood as a set of similar entities.

A **super-class** of a class is a class that is more general than the first class (a super-set in the set-theoretic interpretation).
A class (also called concept) can be understood as a set of similar entities.

The fact that an entity belongs to a class is expressed by the type predicate from the standard namespace rdf (http://w3c.org/...).

The fact that a class is a sub-class of another class is expressed by the subclassOf predicate from the standard namespace rdfs (http://w3c.org/...).
**Meta Data**

**Meta-Data** is data about classes and properties

Properties themselves are resources in RDF

RDFS can be used to talk about classes and properties, too

=> There is no concept of „meta-data“ in RDFS

http://www.w3.org/TR/rdf-schema/
Reasoning

„A person can only be born in one place“

owl:FunctionalProperty
[rdf:type]
:bornIn

:bornIn example:Meat
owl:disjointWith example:Fruit

„Meat is not Fruit“

owl:Class
[rdf:type]
:Meat
owl:disjointWith :Fruit

The owl namespace defines vocabulary for set operations on classes, restrictions on properties and equivalence of classes.

OWL vocabulary can be used to express properties of classes and predicates

=> We can express logical consistency

http://www.w3.org/TR/owl-guide/
OWL: Undecideability

The consistency of an OWL ontology is **undecideable**.

What do you mean, we are undecideable?

That means: We may encounter an ontology for which we cannot decide whether it contains a contradiction or not.

OWL comes with the following decideable sub-sets (**profiles**):
- OWL-EL
- OWL-RL
- OWL-QL
- OWL-DL
SPARQL: Matching

SPARQL (SPARQL Protocol and RDF Query Language) is the query language of the Semantic Web.

PREFIX e: <http://elvis.org/>

SELECT ?loc
WHERE {
  e:elvis e:livesIn ?loc.
}

SPARQL queries can be seen as sub-graph matching.
SPARQL: Endpoints

**SPARQL** (SPARQL Protocol and RDF Query Language) is the query language of the Semantic Web.

Many ontologies provide a “SPARQL endpoint”, i.e. a service than can
- receive SPARQL queries sent by a machine
- receive SPARQL queries typed by a human in a Web interface

http://esw.w3.org/SparqlEndpoints

### Currently Alive SPARQL Endpoints

(Alphabetical. Let's avoid PoorMansHypertext and in-your-face URLs, please)

<table>
<thead>
<tr>
<th>Project</th>
<th>status</th>
<th>SPARQL endpoint</th>
<th>Webform</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBC Programmes and Music</td>
<td>(2010-06-29)</td>
<td>endpoint</td>
<td>Ajax based Visual Query Builder</td>
</tr>
<tr>
<td>Bio2RDF</td>
<td>(2010-01-07)</td>
<td>List of 40</td>
<td>n/a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPARQL endpoints</td>
<td></td>
</tr>
<tr>
<td>BioGateway</td>
<td>(2010-01-07)</td>
<td>endpoint</td>
<td>webform</td>
</tr>
</tbody>
</table>
SPARQL: Example

Example at http://dbpedia-live.openlinksw.com/sparql/:

```
select distinct ?x {
    <http://dbpedia.org/resource/Elvis_Presley>
    <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?x
}
limit 100
```
Existing Ontologies

The **Linking Open Data Project** aims to interlink all open RDF data sources into one gigantic RF graph ([link](http://www4.wiwiss.fu-berlin.de/lodcloud/)).

Hundreds of data sets are nowadays available in RDF
([http://www4.wiwiss.fu-berlin.de/lodcloud/](http://www4.wiwiss.fu-berlin.de/lodcloud/))
- US census data
- BBC music database
- Gene ontologies
- DBpedia general knowledge (and hub vocabulary), + YAGO, + Cyc etc.
- UK government data
- geographical data in abundance
- national library catalogs (Hungary, USA, Germany etc.)
- publications (DBLP)
- commercial products
- all Pokemons
- ...and many more
Currently (2010)
• 200 ontologies
• 25 billion triples
• 400m links

http://richard.cyganiak.de/2007/10/lod/imagemap.html
But back to the original question...

Will there ever be a famous singer called Elvis again?

SELECT ?x WHERE {
  ?x :hasGivenName "Elvis" .
  ?x rdf:type singer .
}
But back to the original question...

http://mpii.de/yago

We found him!

Can we find out more about this guy?

?x = Elvis_Costello
?singer = wordnet_singer_110599806
?d = 1954-08-25
But back to the original question...

http://mpii.de/yago

Alright, and even more?
Querying Semantic Data

**Sindice** is an index for the Semantic Web developed at the DERI in Galway/Ireland.

Sindice exploits
- RDF dumps available on the Web
- RDF information embedded into HTML pages
- RDF data available by cool URIs
- inter-ontology links

[Tummarello ISWC 2007]
Querying Semantic Data

... far from perfect... but far from useless...
Conclusion

• We have seen that there is a need to move from keyword-based treatment of information to semantic treatment

• We have seen approaches for instance and fact extraction (Hearst patterns, Set Expansion, Pattern-based and Wrappers)

• We have seen more ambitious techniques for information extraction (from Wikipedia, Reasoning-based, KnowItAll, ReadTheWeb)

• We have seen the knowledge representation model of ontologies, RDF
  In a nutshell, RDF is a kind of distributed entity-relationship model

• We have seen what knowledge bases exist and how they are linked

• We have seen that there is indeed a promising singer called Elvis