ECI 2013 – Day 2

Labeled graph homomorphism and first order logic inference

Madalina Croitoru
University of Montpellier 2, France

croitoru@lirmm.fr
What is Knowledge Representation?
Semantic Web Motivation
What is the Web?

What is the Web 2.0?

What is the Web 3.0?
What is the Web?

• The WWW: World Wide Web

• Web is different from the Internet! (how?)

• Tim Berners Lee – CERN 1989
Anything can link to anything
Web Architecture

Resources identified by their addresses: URI
Uniform Resource Identifier
Web Architecture

Resources are accessed using the HTTP protocol

HyperText Transfer Protocol
What about the resources?

• Web content should be suitable for **human consumption**
  – Even Web content that is generated automatically from databases is usually presented without the original structural information found in databases

• Web uses
  – seeking and making use of information, searching for and getting in touch with other people, reviewing catalogs of online stores and ordering products by filling out forms
Keyword-Based Search Engines

• The Web would not have been the huge success it was, were it not for search engines

• Initial Web activities not particularly well supported by software tools
  – Except for keyword-based search engines (e.g. Google, AltaVista, Yahoo)
Problems of Keyword-Based Search Engines

• High recall, low precision.
• Low or no recall
• Results are highly sensitive to vocabulary
• Results are single Web pages
• Human involvement is necessary to interpret and combine results
• Results of Web searches are not readily accessible by other software tools
The Key Problem of Today’s Web

• The meaning of Web content is not machine-accessible: lack of semantics

• It is difficult to distinguish meaning
The Semantic Web Approach

• Represent Web content in a form that is more easily machine-processable.
• Use intelligent techniques to take advantage of these representations.
• The Semantic Web will gradually evolve out of the existing Web
Semantic Web

Today’s Web

Lots of unchartered land (data), and a guide (Google)

Tomorrow’s Web

Every worthy site labeled (i.e. data described in some standard way so both people and machines can make sense of it)

Copyright Greg Boutin (you can share it with attribution to this site)
Semantic Web will allow for...

- Knowledge will be organized in **conceptual spaces** according to its meaning.
- **Automated tools** for maintenance and knowledge discovery
- **Semantic** query answering
- Query answering over **several documents**
- Defining **who may view** certain parts of information (even parts of documents) will be possible.
Why is this an important topic?
Data, data everywhere

• The world contains an unimaginably vast amount of digital information which is getting ever vaster ever more rapidly
  – Facebook: 40 billion pictures etc.
New term: « BIG DATA »

• Joe Hallerstein (Berkley, California): « the industrial revolution of data »

• Results felt everywhere:
  – Business
  – Science
  – Government
  – Arts
How did we get here?

- Technology: sensors and gadgets are digitising lots of information that was previously unavailable
- Mobile phones (6.8 billion people, 4.6 mobile subscriptions)
- Internet: 1-2 billion users
What can we do with all this data?

• Social sciences: researchers are able to understand human behaviours at the population level rather than the individual

• Economics: Microsoft’s Bing can advise customers whether to buy an airline ticket now or wait for the price to come down
What current technologies are fueling this trend?

- Cloud computing: Internet is used as a platform to collect, store and process data. Amazon, Google, Microsoft major investors.
- Open source software
Knowledge Representation?!

- As data becomes more abundant, the main problem is no longer finding information but finding the relevant bits
- We need information about information... how is the semantic web fitting in the picture?
How can we structure things?

- Statements
- Describe some relationship between one thing and another
- Standardised by W3C as RDF
How things work?

• For the **Web**:
  - URI; HTTP; (X)HTML

• For the **Semantic Web**:
  - URI; HTTP; RDF
RDF

• Resource Description Format
• Machine – processable
• Statements: TRIPLES
RDF triplets:

- `<Subject> <Predicate> <Value>`
- **Example:**
  - `<Cat> <hasName> “Gizzy” .`
  - `<Madalina> <knows> <Cathy> .`
  - `<Madalina> <hasEmployer> <LIRMM>`
  - `<LIRMM> <site> <http://www.lirmm.fr>`
  - `<primaryKey> <column_name> “value” .`
The Web vs Semantic Web

• Web
  • (Madalina Croitoru, www.lirmm.fr/~croitoru)

• Semantic Web
  • (Madalina Croitoru, has_homepage, www.lirmm.fr/~croitoru)
RDF is a graph:

- `<Madalina> <has_homepage> <www.lirmm.fr/~croitoru> .`
- `<Madalina> <hasEmployer> <LIRMM>`
- `<LIRMM> <site> <http://www.lirmm.fr>丰满
What difference with XML?

• XML is about documents and not data
• Many ways to say the same thing in XML
• RDF is easy to put in relational databases
• RDF is easy to merge!
Ambiguity in XML

<page><hasAuthor><Madalina>

<author>
  <uri>page</uri>
  <name>Madalina</name>
</author>

<person name="Madalina"> 
  <work>page</work> 
</person>

<document href="http://www.lirmm.fr/~croitoru" author="Madalina"/>
RDF - syntax

• RDF was designed to provide a common way to describe information so it can be read (and understood) by computer applications.
• RDF descriptions are not designed to be displayed on the web.
• RDF descriptions are logic descriptions
RDF in its logic form:

- `<Madalina> <has_homepage> <www.lirmm.fr/~croitoru> .
  has_homepage("Madalina", "www.lirmm.fr/~croitoru")`
- `<Madalina> <hasEmployer> <LIRMM>
  hasEmployer("Madalina", "LIRMM")`
- `<LIRMM> <site> <http://www.lirmm.fr>
  site("LIRMM", "www.lirmm.fr")`
Logic and Knowledge Representation
Ontological Query Answering

Knowledge Base: RDF

Data / Facts
Ontological Query Answering

Knowledge Base

Data / Facts

Query

Answers ?
Ontological Query Answering

Challenge in KR: **query large fact bases**
Ontological Query Answering

Challenge in KR: \textit{query large fact bases}
Ontological Query Answering

Challenge in databases: take the ontology into account

Challenge in KR: **query large fact bases**
Ontological Query Answering

Knowledge Base

Data / Facts
Conj Positive
Atoms

Ontology

Query

Answers ?
Ontological Query Answering

Knowledge Base

Data / Facts

 Conj Positive

Atoms

Universal Knowledge

All mothers are women

Questions?

Query
Ontological Query Answering

Knowledge Base

Data / Facts
Conj Positive Atoms

Universal Knowledge
All mothers are women

Existential Rules
All humans have a mother

Rule Application: Forward Chaining or Backwards Chaining
Ontological Query Answering

Knowledge Base
- Data / Facts
- Conj Positive Atoms

Universal Knowledge
- All mothers are women

Existential Rules
- All humans have a mother

Answers?

Query

Rule Application: Forward Chaining or Backwards Chaining

Challenge in KR: query large fact bases
Ontological Query Answering

Knowledge Base

Data / Facts

*Conj Positive Atoms*

Universal Knowledge

*All mothers are women*

Existential Rules

*All humans have a mother*

Answers?

Query

Rule Application: **Forward Chaining or Backwards Chaining**

Challenge in KR: query **large fact bases**
Data / Facts

Relational Database

<table>
<thead>
<tr>
<th>parentOf</th>
<th>Male</th>
<th>Fem.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td>x</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Abstraction in first-order logic

\[ \exists x \ (\text{parentOf}(A,B) \land \text{parentOf}(A,C) \land \text{parentOf}(C,x) \land F(A) \land M(B) \land M(x) \) \]

RDF (Semantic Web)

Etc.

Or in graphs / hypergraphs

Etc.
Ontological Conjunctive Query Answering

Facts: Conjunction of positive atoms

Ontology

Query: Conjunction of positive atoms
Ontological Conjuctive Query Answering

Facts $\models$ Ontology $\models$ Conjunctive Query
Facts $\models$ Conjunctive Query

Simple RDF entailment

Database Conjunctive Query Answering

Deduction in the conjunctive, positive, existential fragment of FOL
Ontological Conjunctive Query Answering

Facts

Every cat is a vertebrate
If two people are brothers then they are relatives

Type Hierarchy

╞

 Conjunctive Query
Facts + Type Hierarchy \[\vdash\]
Conjunctive Query

RDF(S) entailment

Conjunctive Positive Fragment of FOL

Simple Conceptual Graphs Entailment
Ontological Conjunctive Query Answering

For every cat there exists a cat who is its mother
Facts + Rules \implies Conjunctive Query

Simple Conceptual Graphs Rules

DATALOG +
Facts: Conjunction of positive atoms

\{p(x, y), q(y, z, t), p(z, A), q(z, t, B)\}
BASIC GRAPHS (FACTS)
Relational Model

<table>
<thead>
<tr>
<th>BoyId</th>
<th>GirlId</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BoyId</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GirlId</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Eva</td>
</tr>
</tbody>
</table>
Conjunctive Query Answering

Fact \models Query

Table F \xrightarrow{\text{Conjunctive Query Q}} Query Answering

Graph F \xrightarrow{\text{Graph Q}} Homomorphism
Conceptual Graph: Facts

- Boy: John
- Girl: Eva
- TeddyBear: *
- Color: Red

Relations:
- siblingOf: Boy: John - Girl: Eva, 2
- give: Boy: John - TeddyBear: *, 3
- hold: Girl: Eva - TeddyBear: *, 1
- possess: TeddyBear: * - Color: Red, 2
Conceptual Graphs: Query
Conceptual Graphs Querying:

Labelled Graph Homomorphism (Projection)
Homomorphism

• Structure preserving mapping between two algebraic structure:
  – “Homos”: same
  – “Morphe”: shape

• Example of homomorphisms:
  – Isomorphism (bijective)
  – Automorphism (with itself) etc.
Projection

mapping from the nodes of the query
to the nodes of the fact

that

1. ‘respects’ the labels of mapped nodes
2. preserves the neighbourhood structure
Sound and complete algorithm wrt given semantics

• Soundness: preserves the “truth” (algorithm does not give any results which are untrue in the given semantics)

• Completeness: all truth can be obtained by the algorithm (algorithm addresses all possible inputs without any miss).
Projection

SOUND AND COMPLETE

w.r.t.

DEDUCTION IN THE POSITIVE EXISTENTIAL, CONJUNCTIVE FRAGMENT OF FIRST ORDER LOGIC
PROJECTION

Given a query graph* Q and a fact graph* F find if there is a projection P from Q to F

... as long as there are not projected nodes in the query find their pertinent matches

The operations we need to be able to do at a large scale
Projection algorithm

**Extend** \((ProjectedNodes, \text{Query}, \text{Fact})\)

Find the **extensions** \((\text{Node}, \text{Image})\) to current projection

\[
\text{\textbf{\textit{V Extend}}} (ProjectedNodes \cup \{(N, I)\}, \text{Query}, \text{Fact})
\]

\((N, I)\)
Extend \((\text{ProjectedNodes}, \text{Query}, \text{Fact})\)

Extend \((\Phi)\)
Choose node 1
Return node A and node B

Extend \((\{1, A\})\)
Extend \((\{1, B\})\)
Projection algorithm

Extend \((\text{ProjectedNodes}, \text{Query}, \text{Fact})\)

\[
\text{if no of } \text{ProjectedNodes} = \text{no of Query nodes} \\
\text{return true} \\
\text{else} \\
\text{select a node } n \text{ not yet projected} \\
\text{return list pertinent matches } i \text{ of } n \text{ according to} \\
\text{ProjectedNodes, Query, Fact}
\]

\[
\text{V Extend } (\text{ProjectedNodes} \cup \{(n,i)\}, \text{Query, Fact})
\]
Projection algorithm

Extend \((ProjectedNodes, Query, Fact)\)

if no of \(ProjectedNodes\) = no of \(Query\) nodes
return true
else
    select a node \(n\) not yet projected
    return list pertinent matches \(i\) of \(n\) according to \(ProjectedNodes, Query, Fact\)

\(\forall\) Extend \((ProjectedNodes \cup \{(n,i)\}, Query, Fact)\)
Possible images i of node 1

1. A good index: Constants
Possible images of node 3

2. A suitable data structure to efficiently manipulate neighbours
Backtrack projection requires

1. A good index
2. A suitable data structure to efficiently manipulate neighbours

(To be used in the context of different optimisations inspired by CSP techniques (Forward Chaining, Back Jump etc.)
Backtrack Projection

• Worse case – exhaustive exploration of the possible mappings

• How to remove big pieces of this graph -> types on the concepts and the relations?
Simple Conceptual Graphs: Concept Hierarchy
Simple Conceptual Graphs: Relation Hierarchy

[Diagram of conceptual graphs showing relationships between terms such as act2, possess, attribute, author, relativeOf, parentOf, siblingOf, father, mother, give, and between.]

[Image of two children wearing hats, possibly related to the context.]
Backtrack projection now requires

1. A good index
2. A suitable data structure to efficiently manipulate neighbours

The indexing could be on types, star graphs, graphs that describe recurrent situations etc.
Graph based Knowledge Representation and Reasoning
Example 1: Projection

• Projection for Conceptual Graphs: labeled graph homomorphism
• Projection for Conceptual Graphs: sound and complete with respect to first order logical deduction

We can now simply do labeled graph homomorphism for performing deduction!
Tomorrow

• Present a real world tool implementing this graph based model: CoGui

• CoGui implemented by the GraphIK group at LIRMM, Montpellier, France