Open Information Extraction

Dominik Both, Tonio Weidler

Proseminar *Text Mining*
Andrea Zielinski

Institut für Computerlinguistik, Universität Heidelberg, 15.07.2016

Strukturierung

Introduction

OIE - Principles

OIE - Principles

Motivation

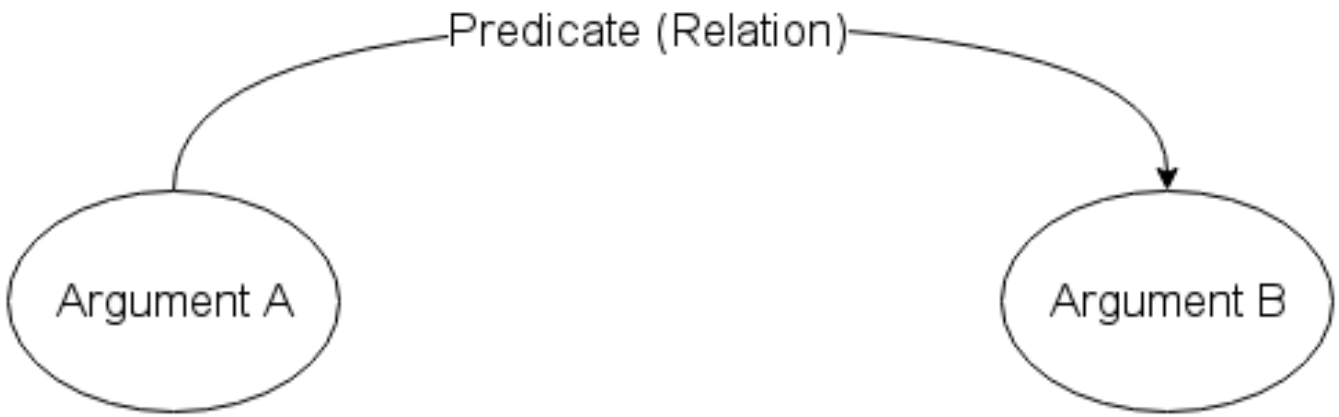
OIE - Principles

Methods

OIE - Principles

Data Representation

Standard Patterns



Argument A is in a directed **relation** to **Argument B**.

Unnormalized Annotation

(argument_a, predicate_x, argument_b)
(argument_a, predicate_y, argument_c)
(argument_a, predicate_y, argument_d)

Problems

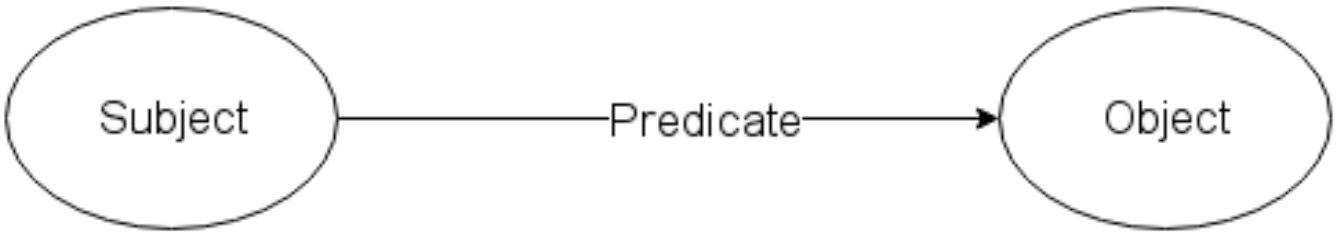
- redundant
- unnormalized
- can only produce binary predicates

RDF and Linked Data

Resource Description Framework

Models propositions by constructing *triples* including **Subjects**, **Objects** and **Predicates**
Generates a directed graph

:subject :predicate :object.



RDF Concepts and Notation

- **URIs**
  identifies ressources (S, R, O) distinctivly and references further informations (triples)
- **Conclusions**
  allows to draw conclusions using rules
- **Turtle**
  allows syntax abbreviations
- **Queries**
  can be searched by querying (eg SPARQL)

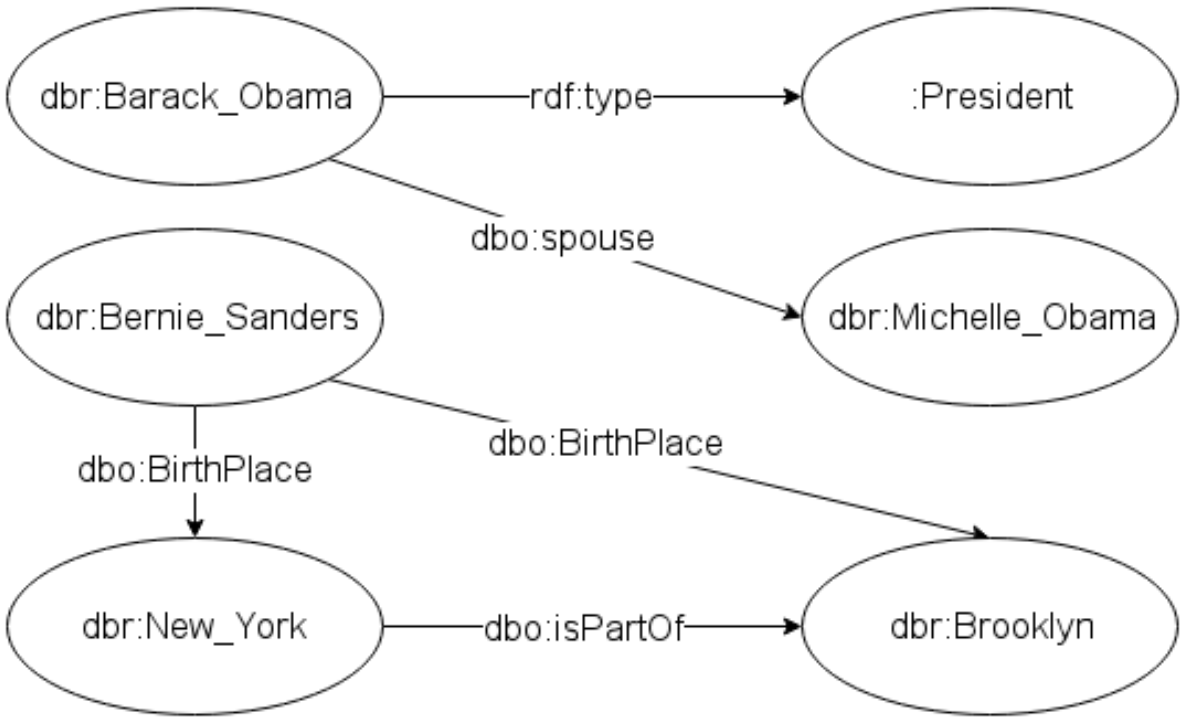
Basic relations (built in)

Relation	**Functionality**
rdf:type (a)	x is of type y
owl:sameAs	x equals y

RDF Syntax

```
dbr:Barack_Obama a foaf:person, :President;
    dbo:spouse dbr:Michelle_Obama.
dbr:Bernie_Sanders dbo:birthPlace dbr:New_York,
                                  dbr:Brooklyn;
dbr:Brooklyn dbo:isPartOf dbr:New_York
```

... as Graph


dbr:Barack_Obama
rdf:type
:President
dbo:spouse
dbr:Bernie_Sanders
dbr:Michelle_Obama
dbo:BirthPlace
dbo:BirthPlace
dbr:New_York
dbo:isPartOf
dbr:Brooklyn

Example: LODifier

LODifier: Generating Linked Data from Unstructured Text (Augenstein et al., 2012

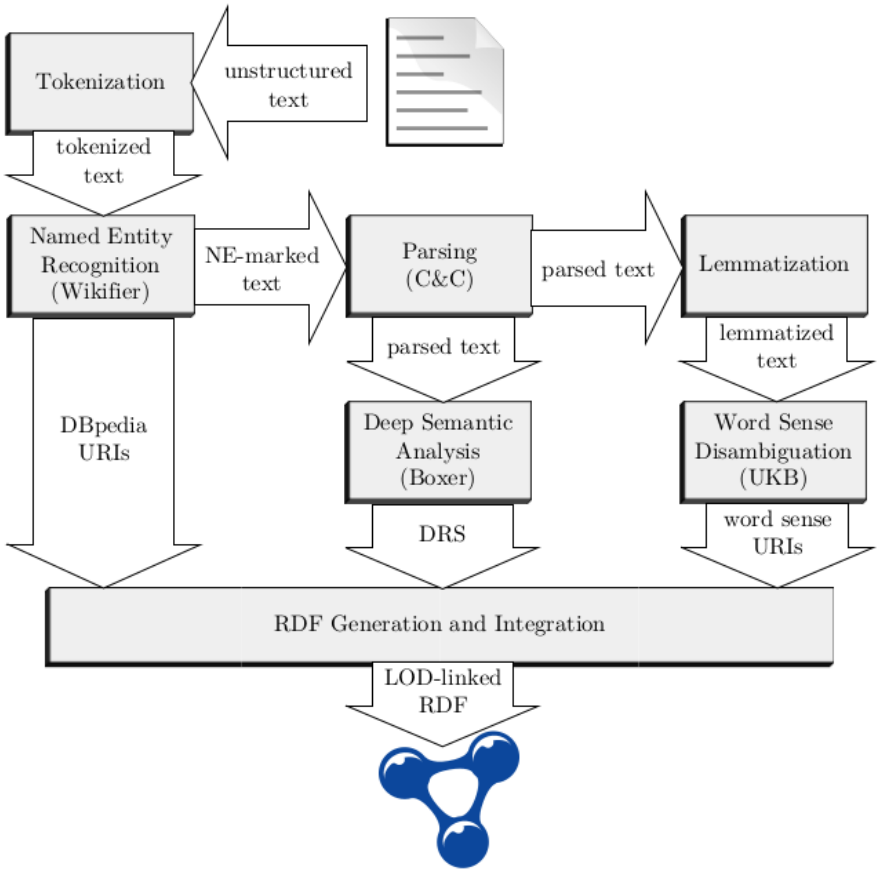
Generate an RDF Graph from unstructured Text

Past Approaches: Use Patterns to trade recall for precision
LODifier: Process the entire text

Example: LODifier

Architecture

Architecture



Approach

1. **Parse** the input text (POS, Treetagging, NER)
2. Apply **Deep Semantic Analysis** to get relations
3. Enrich NEs and words with **URIs** (DBpedia and WordNet)
4. Forge an **RDF Graph** of this information

How does it happen?

Lets go through the process step-by-step!

Example Text:
The New York Times reported that John McCarthy died. He invented the programming language LISP.
example taken from Augenstein et al., 2012

Example: LODifier

Preprocessing

Named Entity Recognition - Wikifier

Wikifier

Recognizes NE and replaces them with the Wikipedia Page Link
Disambiguates by comparing links between pages.

Example Text Output:

[The New York Times] reported that [John McCarthy (computer scientist)|John McCarthy] died. He invented the [Programming language|programming language] [Lisp (programming language)|LISP].

Parsing Syntax - C&C

C&C Parser

Syntactical Parser that tags POS and builds Parse Trees (CCG).

Parsing - Output

```
ccg(1, rp(s:dcl,
    ba(s:dcl,
      lx(np, n,
        t(n, 'The_New_York_Times', 'The_New_York_Times', 'NNS', 'I-NP', 'O')),
      fa(s:dcl\np,
        t((s:dcl\np)/s:em, 'reported', 'report', 'VBD', 'I-VP', 'O'),
        fa(s:em,
          t(s:em/s:dcl, 'that', 'that', 'IN', 'I-SBAR', 'O'),
          ba(s:dcl,
           lx(np, n,
             t(n, 'John_McCarthy', 'John_McCarthy', 'NNP', 'I-NP', 'I-PER')),
           t(s:dcl\np, 'died', 'die', 'VBD', 'I-VP', 'O'))))),
    t(period, '.', '.', '.', 'O', 'O'))).
ccg(2, rp(s:dcl,
    ba(s:dcl,
      t(np, 'He', 'he', 'PRP', 'I-NP', 'O'),
      fa(s:dcl\np,
        t((s:dcl\np)/np, 'invented', 'invent', 'VBD', 'I-VP', 'O'),
        fa(np:nb,
          t(np:nb/n, 'the', 'the', 'DT', 'I-NP', 'O'),
          fa(n,
            t(n/n, 'programming_language', 'programming_language', 'NN', 'I-NP', 'O'),
            t(n, 'LISP', 'LISP', 'NNP', 'I-NP', 'O'))))),
    t(period, '.', '.', '.', 'O', 'O'))).
```

Find Relations - Boxer

Boxer

Creates DRSs from C&C Output

Find Relations - Boxer

Boxer

Creates DRSs from C&C Output

Discours Representation Structure (DRS)

Represents the discourse via *relations* between *entities*
Allows referencing over the entire discourse

Find Relations - Boxer

Boxer

Creates DRSs from C&C Output

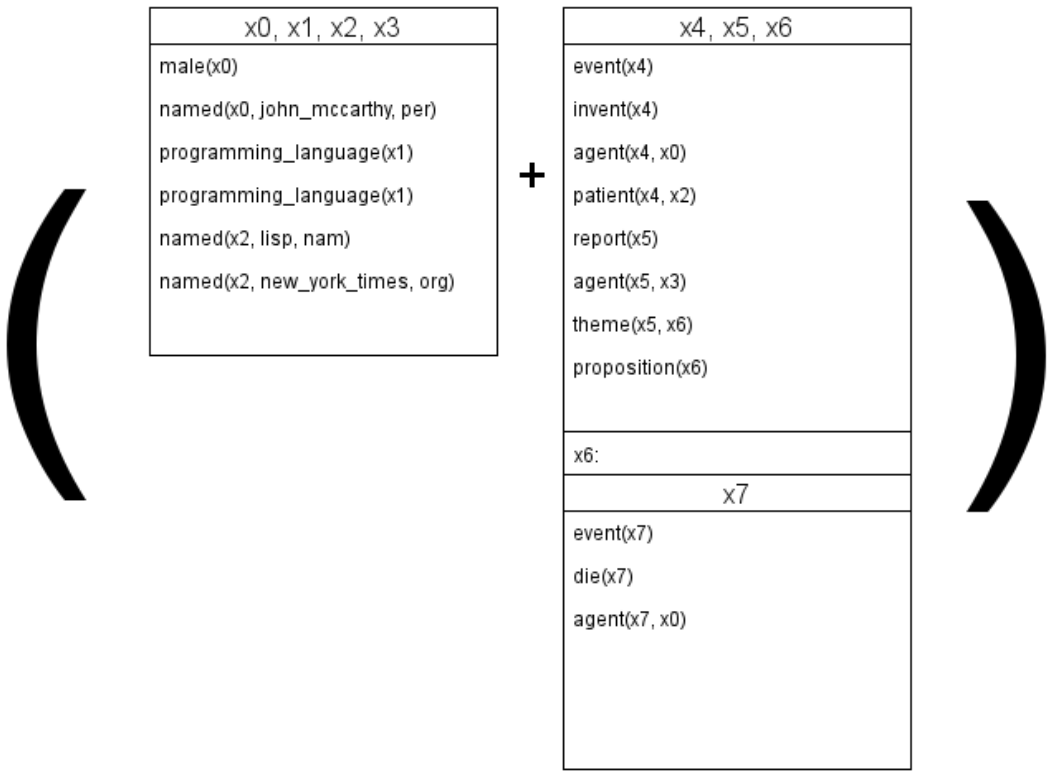
Discours Representation Structure (DRS)

Represents the discourse via *relations* between *entities*
Allows referencing over the entire discourse

Boxers DRS Relations (Conditions):

- **Unary Relations (Classes):** eg. *topic*, *person*, *event*, *male*, ... + all verbs
- **Binary Relations:** agent, patient, ... (semantic roles)

Boxer Output



Assign WordNet URIs

RDF WordNet

WN: Lexicography containing senses linked by semantic relations
RDF WN: LD Representation of WN providing URIs for words

Steps:

1. Lemmatization
2. WSD (UKB)
3. Assign RDF WN URIs to word senses

Preprocessing Result

We now have ...

- URIs for all NEs
- URIs for all (disambiguated) words
- Relations between entities (those URIs)

Example: LODifier

RDF Construction

What now?

Let's now construct the RDF Graph from this information!

Namespaces/Vocabularies

LODifier creates several namepaces:

- drsclass:
- class:
- drsrel:
- ne:
- reify:

And uses standard namespaces:

- rdf:
- owl:

As well as the two ontologies:

- wn30:
- dbpedia:

Example: LODifier

Conclusions

OIE Systems in Context

OIE Systems in Context

Comparison

OIE Systems in Context

Evaluating the Approaches

Conclusion

Conclusion

Problems and Obstacles

Conclusion

Future Opportunities