Lessons learned (and questions raised) from an interdisciplinary Machine Translation approach

Lightening talk at the W3C Workshop on the Open Data on the Web, 23 - 24 April 2013, Google Campus, Shoreditch, London

Timm Heuss
University of Plymouth, Plymouth, United Kingdom
Motivation
Usual problems in NLP: Ambiguities

- The problem in many areas of Natural Language Processing (NLP) is the **ambiguity** of natural language on various levels, from word level to sentence level.
- In the NLP-subfield of Machine Translation (MT), it is often **crucial to understand** the source text correctly – otherwise, ambiguities may result in **incorrect target translations**.

*Pages by Apple is a word processor like Word by MS.*
Motivation

Usual problems in NLP: Ambiguities

- The problem in many areas of Natural Language Processing (NLP) is the ambiguity of natural language on various levels, from word level to sentence level.
- In the NLP-subfield of Machine Translation (MT), it is often crucial to understand the source text correctly – otherwise, ambiguities may result in incorrect target translations.

Pages by Apple is a word processor like Word by MS.

- Fruit?
- Apple Inc.?
- English honorific?
- Motor ship prefix?
- Microsoft Corporation?
Motivation
Usual problems in NLP: Ambiguities

• The problem in many areas of Natural Language Processing (NLP) is the **ambiguity** of natural language on various levels, from word level to sentence level

• In the NLP-subfield of Machine Translation (MT), it is often **crucial to understand** the source text correctly – otherwise, ambiguities may result in **incorrect target translations**.

  
  **Pages by Apple** is a word processor like **Word by MS**.

• In this cases, strings can be only disambiguated on the basis of **world or expert knowledge**

• A typical solution: creation of **dedicated dictionaries**
A new Machine Translation approach
Semantic Web based Machine Translation (SWMT)

• Idea: use knowledge that already exists in form of LOD to enhance a Machine Translation task
  ▪ Evaluate multi-language labels of RDF triples:

    dbpedia:Microsoft_Word dbp:developer
dbpedia:Microsoft .
dbp:developer rdfs:label "by"@en,
    "von"@de, "entwickelt von"@de.
A new Machine Translation approach

Semantic Web based Machine Translation (SWMT)

- Idea: use knowledge that already exists in form of LOD to enhance a Machine Translation task
  - Evaluate multi-language labels of RDF triples:

    - English: Word by MS
      - `dbpedia:Microsoft_Word` as an example
      - `dbpedia:Microsoft`.
      - `dbp:developer rdfs:label "by"@en, "von"@de, "entwickelt von"@de`.

    - German: Word von Microsoft
      - Word entwickelet von Microsoft
A new Machine Translation approach
Semantic Web based Machine Translation (SWMT)

• Idea: use knowledge that already exists in form of LOD to enhance a Machine Translation task
  ▪ Evaluate multi-language labels of RDF triples:

    dbpedia:Microsoft_Word dbp:developer
    dbpedia:Microsoft .
    dbp:developer rdfs:label "by"@en,
    "von"@de, "entwickelt von"@de .

  ▪ Integrate retrieved phrases into a MT process
• Good news: it works 😊 (see github.com/heuusd)
• Not-so-good news: Some issues and conceptual mismatches emerged during development
Lessons learned and questions raised

Statistics

• In NLP, applications usually utilize various statistics.
• In the Web of Data, the Open World Assumption does not allow us the creation of statistics – it even “make[s] counting difficult”

Dean Allemang and James A. Hendler, _Semantic Web for the Working Ontologist_

• Is there a chance for statistics in LOD?
Lessons learned and questions raised

Restriction of RDF

- RDF triples are the backbone of the approach, which might resolve to small phrases of three or four words
- Real-world natural language sentences are more complex

- Does LOD claim to carry the complete sense of the human language?
- Is a seamless conversion possible?

→ follow-up lightening talk: David Lewis
   *Interoperability Challenges for Linguistic Linked Data*
Lessons learned and questions raised

The prototype cannot work live with DBpedia data (iSPARQL), because the query is too complex for the endpoint policies:

“The estimated execution time 7219 (sec) exceeds the limit of 3000 (sec)”

Can the world’s largest pooled collection of LOD only be queried with simple queries?

What is the best practice in this case? Obviously, additional program logic is required to create a application-specific cache for LOD
Lessons learned and questions raised

Performance

- Production of the translation phrases takes considerable time (about 1 second) – this might be too long for real-time NLP
- A triple store could speed this up
- A more specialized storage form could be even faster

- If LOD would just be the input for an Extraction, Transfer and Load (ETL) process – wouldn’t that be against the LOD vision?
- What is the most suitable storage form for RDF?
Thanks for your attention!
Do you have any questions?

What do you think?

Contact me:
http://heussd.github.io

Some rights reserved. This work is published under the Creative Commons AttributionNonCommercial-ShareAlike 3.0 License. Commercial distribution of this work requires a prior written permission of the author. Non-commercial distribution is permitted. Derived work is permitted with some limitations. See http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode for the full license statement.