Towards Cross-checking WordNet and SUMO Using Meronymy

Javier Álvez  German Rigau

LoRea & IXA Groups
Computer Languages and Systems Department
University of the Basque Country (UPV/EHU)
1 Introduction

2 Cross-checking WordNet and Adimen-SUMO

3 Some Experimental Results

4 Conclusions and Future Work
Cross-checking knowledge sources

- This work is an initial study about:
  - Knowledge representation
  - Common Sense (world knowledge)
  - Reasoning

- In particular, we focus on:
  - WordNet (Fellbaum, 1998)
  - SUMO (Niles and Pease, 2001)
  - WN-SUMO Mapping (Niles and Pease, 2003)

- We expect all these knowledge sources to encode correct world knowledge (true knowledge).

- Despite being created manually, they are not free of errors and discrepancies.

- We apply a new Black-box strategy (Álvez et al., 2017b) to the meronymy information encoded in these resources.
SUMO (Niles and Pease, 2001)

- IEEE Standard Upper Ontology Working Group
- SUMO syntax goes beyond first-order logic (FOL)
- SUMO cannot be directly used by FOL Automated Theorem Provers (ATPs) without a suitable transformation
- Different transformations of SUMO into FOL:
  - TPTP-SUMO (Pease and Sutcliffe, 2007)
  - Adimen-SUMO (Álvez et al., 2012)
Adimen-SUMO I

- Following the line of (Horrocks and Voronkov, 2006)
- Obtained by applying a reengineering process to SUMO
  - With the help of ATPs (Automated Theorem Provers)
  - Around an 88% of the core of SUMO (top and middle levels) is translated
  - Domain ontologies are not used (by now)
  - The resulting ontology can be used in tasks that involve reasoning with commonsense knowledge
- The process of manually debugging the ontology is very costly
  - Only 64 manually created tests
  - Example:

\[
( \rightarrow \\
( \text{and} \\
  (\text{instance} \ ?\text{BRAIN} \ \text{Brain}) \\
  (\text{instance} \ ?\text{PLANT} \ \text{Plant})) \\
(\text{not} \\
  (\text{properPart} \ ?\text{BRAIN} \ ?\text{PLANT})))
\]
Adimen-SUMO II

- We have proposed different methodologies for the automatic debugging ontologies like Adimen-SUMO
  - Black-box testing strategies (Álvez et al., 2015, 2017b)
  - White-box testing strategies (Álvez et al., 2017a)
- More than 100 axioms from Adimen-SUMO has been improved
Black-box Testing I

- Introduced in (Álvez et al., 2015) and fully described in (Álvez et al., 2017b)
- Adaptation of the methodology for the design and evaluation of ontologies introduced in (Grüninger and Fox, 1995)
- Based on the use of Competency Questions (CQs):
  - Problems that an ontology is expected to answer
- Its application is automatic by means of the use of ATPs
- Classification of (dual) problems (truth and falsity tests):
  - Passing: the ATPs are able to demonstrate a truth test
  - Non-passing: the ATPs are able to demonstrate a falsity test
  - Unknown: the ATPs produce no answer within a time limit
Black-box Testing II

- CQs are automatically created on the basis of few Question Patterns (QPs) by exploiting WordNet and its mapping into SUMO

- In (Álvez et al., 2017b):
  - antonym and event (agent, instrument and result) relations
  - 11 QPs are proposed
  - More than 7,500 CQs are created
  - More than 43% of CQs are validated
  - Example:

\[
\forall Y \left(\text{instance } Y \text{ MusicalComposition} \Rightarrow \exists X \left(\text{instance } X \text{ ComposingMusic} \land \text{result } X Y\right)\right)
\]
Mapping between WordNet and SUMO

- Described in (Niles and Pease, 2003)
- It connects synsets of WordNet to terms of SUMO using 3 relations:
  - equivalence (=)
  - subsumption (+)
  - instance (@)
- Some examples:

  \[
  \langle \text{calcium}^1 \rangle : [\text{Calcium}_c = ]
  \]

  \[
  \langle \text{calcium}_\text{oxide}^1 \rangle : [\text{CompoundSubstance}_c + ]
  \]

  \[
  \langle \text{police}_\text{officer}^1 \rangle : [\text{PoliceOfficer}_a = ]
  \]

  \[
  \langle \text{police}_\text{force}^1 \rangle : [\text{PoliceOrganization}_c + ]
  \]
Introduction

Cross-checking WordNet and Adimen-SUMO

Some Experimental Results

Conclusions and Future Work
WordNet v3.0 provides 3 part-whole relations (22,187):

- **part**: the general meronymy relation (9,097)
- **member**: it relates particulars and groups (12,293)
- **substance**: it relates physical matters and things (797)

For example:

\[
\langle \text{committee}_n \rangle \rightarrow \langle \text{member} \rangle \rightarrow \langle \text{committee\_member}_n \rangle
\]

\[
\langle \text{wine}_n \rangle \rightarrow \langle \text{substance} \rangle \rightarrow \langle \text{grape}_n \rangle
\]
First, creating a mapping between WordNet and Adimen-SUMO:

\[
\begin{align*}
&[\text{Cooking}_c+] \quad \text{(Top level)} \\
&[\text{subclass}] \\
&\langle \text{frying}_n^1 \rangle : [\text{Frying}_c=] \quad \text{(Food ontology)}
\end{align*}
\]

Propose a formal characterization of the mapping information:

\[
\langle \text{male\textunderscore horse}_n^1 \rangle : [\text{Male}_a+] [\text{Horse}_c+]
\]

- **Literal interpretation:**
  
  \[
  \text{(and} \\
  \text{($instance \ X \ Male)} \\
  \text{($instance \ X \ Horse))}
  \]

- **Precise interpretation:**

  \[
  \text{(and} \\
  \text{(attribute \ X \ Male)} \\
  \text{($instance \ X \ Horse))}
  \]
Four different QPs depending on the used mapping relations (*precise* interpretation):

- equivalence
- subsumption or instance

QPs are instantiated according to the mapping information of the synsets in the WordNet meronymy pairs.
Question patterns for the Creation of CQs (II)

- Applying the first QP (precise interpretation):

\[
\text{(exists} \ (\ ?X \ ?Y) \\
\text{(and} \\
\text{<s\_part} \ ?X> \\
\text{<s\_whole} \ ?Y> \\
\text{(<SUMO\_predicate>} \ ?X \ ?Y))
\]

- to the following WN-SUMO meronymy relation:

\[
\langle\text{genus\_malacosoma}\_1^1 \rangle : \ [\text{Larval}_{a+}] \\
\langle\text{member} \rangle \uparrow \\
\langle\text{member}_r \rangle \\
\langle\text{malacosoma\_americana}\_1^1 \rangle : \ [\text{Insect}_{c+}]
\]
Question patterns for the Creation of CQs (III)

- Creates the following CQ:

\[
\text{(exists } (?X ?Y) \\
\text{(and} \\
\text{($instance \ ?X \text{ Insect})} \\
\text{(attribute \ ?Y \text{ Larval})} \\
\text{(member \ ?X \ ?Y)))}
\]
Question patterns for the Creation of CQs (IV)

- Mapping of WordNet relations to Adimen-SUMO predicates, which have domain restrictions:

  \[
  \langle \text{part} \rangle : [ \text{part}_r(\text{Object}_c \times \text{Object}_c) ] \\
  \langle \text{member} \rangle : [ \text{member}_r(\text{SelfConnectedObject}_c \times \text{Collection}_c) ] \\
  \langle \text{substance} \rangle : [ \text{material}_r(\text{Substance}_c \times \text{CorpuscularObject}_c) ]
  \]

- Many discrepancies arise during the instantiation of question patterns.
- 14,513 part relations from 22,187 (65%) do not hold domain conditions.
  - Example:

    \[
    \langle \text{wine}_1^1 \rangle : [ \text{Wine}_c= ] \\
    \]
    \[
    \langle \text{substance} \rangle \uparrow \quad [\text{material}_r] \\
    \]
    \[
    \langle \text{grape}_1^1 \rangle : [ \text{FruitOrVegetable}_c+ ] \\
    \]

  - Reason: the first argument of \text{material}_r is restricted to be \text{Substance}_c, which is incompatible with \text{FruitOrVegetable}_c
- So, we concentrate on the remaining 7,674 relations (35%)
Some Experimental Results

1 Introduction

2 Cross-checking WordNet and Adimen-SUMO

3 Some Experimental Results

4 Conclusions and Future Work
Creating CQs and applying ATPs

- We apply the 4 QPs to the 7,674 relations allowing to create 2,145 CQs.
- When testing these CQs using ATPs such as Vampire (Kovács and Voronkov, 2013) or E-prover (Schulz, 2002):
  - **Passing**: knowledge validation
  - **Non-passing**: knowledge mismatches
    - WN-SUMO mapping issues
    - WordNet issues
    - SUMO issues
  - **Unknown**: Missing knowledge ... or insufficient execution time?
Knowledge Validation

Example:

\[ \langle \text{police-force}_1 \rangle : [ \text{PoliceOrganization}_{c^+} ] \]

\[ \langle \text{member} \rangle \]

\[ \langle \text{police-officer}_1 \rangle : [ \text{PoliceOfficer}_{a=} ] \]

Reason:
- The resulting CQ is entailed by Adimen-SUMO:

\[
(\forall Y) \\
(\Rightarrow) \\
(\text{attribute } Y \text{ PoliceOfficer}) \\
(\exists X) \\
(\text{and}) \\
(\text{$instance } X \text{ PoliceOrganization}) \\
(\text{member } X Y)))))
\]
Detection of Mapping Mismatches

- Example:

\[
\langle \text{genus\_malacosoma}_n^1 \rangle : [Larval_a^+] \\
\uparrow \quad \uparrow \quad \uparrow \\
\langle \text{member} \rangle : \langle \text{member}_r \rangle \\
\langle \text{malacosoma\_americana}_n^1 \rangle : [\text{Insect}_c^+] 
\]

- Reason:
  - The attribute \( \text{Larval}_a \) cannot be applied to groups in Adimen-SUMO
Detection of WordNet Issues

- Example:

\[
\langle \text{cell}^2_n \rangle : [ \text{Cell}_c = ]
\]

\[
\langle \text{part} \rangle \quad [\text{part}_r]
\]

\[
\langle \text{cell}_nucleus}^1_n \rangle : [ \text{CellNucleus}_c = ]
\]

- Reason:
  - Some cells lack a nucleus, as stated by the following Adimen-SUMO axiom:

\[
(\forall \ ?C) (\Rightarrow

(\$\text{instance} \ ?C \ \text{RedBloodCell})

(\not \ (\exists \ ?N)

(\text{and}

(\$\text{instance} \ ?N \ \text{CellNucleus})

(\text{part} \ ?N \ ?C))))))
\]
Detection of Adimen-SUMO Issues

- Example:

\[
\langle \text{water\_ice}_n^2 \rangle : [\text{Solid}_a+] \\
\langle \text{substance} \rangle \quad [\text{material}_r] \\
\langle \text{water}_n^1 \rangle : [\text{Water}_c=] 
\]

- Problem:
  - The application of subattributes of \text{PhysicalState}_A (as \text{Solid}_a) was restricted to be only! a property of \text{Substance}_c:

\[
(\forall \text{OBJ}) \quad (\equiv) \\
\quad ($\text{instance } \text{OBJ } \text{Substance}) \\
\quad (\exists \text{ATTR}) \\
\quad (\text{and} \\
\quad \quad ($\text{instance } \text{ATTR } \text{PhysicalState}) \\
\quad \quad (\text{attribute } \text{OBJ } \text{ATTR}))))
\]
Some Experimental Results

Summary

- Reported in (Álvez and Rigau, 2018)

<table>
<thead>
<tr>
<th>SUMO relations</th>
<th>CQs</th>
<th>QP #1</th>
<th>QP #2</th>
<th>QP #3</th>
<th>QP #4</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>part&lt;sub&gt;r&lt;/sub&gt;</td>
<td></td>
<td>+599</td>
<td>+56</td>
<td>+162</td>
<td>+8</td>
<td>+825</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-6</td>
<td>-0</td>
<td>-1</td>
<td>-5</td>
<td>-12</td>
</tr>
<tr>
<td>member&lt;sub&gt;r&lt;/sub&gt;</td>
<td></td>
<td>+10</td>
<td>+1</td>
<td>+1</td>
<td>+0</td>
<td>+12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-9</td>
<td>-0</td>
<td>-0</td>
<td>-0</td>
<td>-9</td>
</tr>
<tr>
<td>material&lt;sub&gt;r&lt;/sub&gt;</td>
<td></td>
<td>+17</td>
<td>+1</td>
<td>+2</td>
<td>+0</td>
<td>+17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-0</td>
<td>-2</td>
<td>-0</td>
<td>-0</td>
<td>-2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>+626</td>
<td>+58</td>
<td>+165</td>
<td>+8</td>
<td>+857</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-15</td>
<td>-2</td>
<td>-1</td>
<td>-5</td>
<td>-23</td>
</tr>
</tbody>
</table>

- 857 Passing CQs (39.95% of total) enable to validate the knowledge of WordNet, SUMO and their mapping
- <code>part</code> is better aligned and covered (825 truth-tests, 42.09%) than <code>member</code> (only 12 truth-tests, 9.92%) and <code>substance</code> (17 truth-tests, 26.56%)
- Different issues are detected (23 falsity-tests, 1.07%)
- More than 60% of the total is <code>Unknown</code>. 

Javier Álvez, German Rigau (UPV/EHU)
Introduction

Cross-checking WordNet and Adimen-SUMO

Some Experimental Results

Conclusions and Future Work
Conclusions

- Framework and benchmark for formal commonsense reasoning
- More than 10,000 CQs available (around 60% Unknown)!
- First steps cross-checking of WordNet, Adimen-SUMO and its mapping:
  - Validation of some pieces of knowledge
  - Detection of knowledge mismatches
  - Detection of missing knowledge
- Resources are ready for its application to practical NLP tasks
- http://adimen.si.ehu.es/web/AdimenSUMO
- https://vprover.github.io/
- https://github.com/eprover/eprover
Future Work

- Improving the WN-SUMO mapping
- Extending our proposal to additional WordNet relations
- Automatically derive new SUMO axioms from WordNet knowledge


