Управление лексиконом в онтологической семантике

Lexicon management in ontological semantics

Petrenko M. (mpetrenk@gmail.com)

Московский Гуманитарный Институт им. Е. Р. Дашковой. Москва, Россия;
RiverGlass, Inc. Champaign, IL, USA.

В работе рассматриваются способы управления лексиконом — одним из базовых (наряду с онтологией) статических ресурсов в Онтологической Семантике. Описывается общая стратегия освоения лексикона (lexicon acquisition), описывается ряд техник освоения, и, на примере освоения английских глаголов класса с инструментально-субъектной альтернацией, описывается, как решается проблема освоения проблемных лексических единиц.

1. Paper goals

The paper describes how the lexicon — a static knowledge resource — is managed by a human acquirer. The study draws on the methodology, theory and strategy of lexical acquisition outlined in [3] and takes into account the ongoing implementation experience in various applications, as well as recent revisions/improvements. After a brief outline of the lexicon, the general strategy of lexical acquisition will be introduced, and techniques of acquisition described. An example will then illustrate how complex cases are handled through lexical acquisition within the framework of the Ontological Semantic Technology (OST).

2. Ontological Semantics: static knowledge resources

The architecture of Ontological Semantics, also known as Direct Meaning Access, comprises a set of static and dynamic resources. The Ontological Semantics school of thought subscribes to the semantic prerequisite in NLP and is premised on the idea that the full (i. e. human-like) efficiency in natural language processing is only attainable through a solid knowledge resource base, which would (1) model the world as a complex and highly structured conceptual hierarchy and (2) furnish lexical, morphological, and syntactic knowledge essential for parsing a natural language input meaningfully (for a more detailed discussion and support of the need to "do semantics semantically", see [1], [7], and [8]).

2.1. The ontological knowledge resource

A detailed and in-depth description of the ontology is offered in [Taylor et al in this volume], so this subsection will contain only a very brief outline. The concepts of the ontology enter into a large number of relations: the hypero-hyponymic (i. e. class-subclass) relation branches the root concept ALL into EVENT, OBJECT, and PROPERTY. Breaking further into numerous subclasses, EVENT's take a large number of properties (including, but not limited to, case-roles) filled by OBJECT's. Both OBJECT's and EVENT's are, in turn, defined through a broad spectrum of circa one hundred ATTRIBUTE's and RELATION's within and across their branches. As illustrated by the example below, the concept BOX is defined through (i. e. is in the DOMAIN of) the properties MADE-OF, CONTAINS, and SHAPE. The concepts CERAMIC, METAL, PAPER, PLASTIC, WOOD function as fillers (i. e. are in the RANGE) of the property MADE-OF.

(box
  (definition (value("a rectangular container")))
  (is-a (hier(container)))
  (**made-of** (sem(**ceramic metal paper plastic wood**)))
  (shape (value(rectangular square)))
)

Which concepts can fill which properties is regulated by the restrictions on the properties' DOMAIN and RANGE. In other words, the concept BOX can take the property MADE-OF because its ancestor, PHYSICAL-OBJECT, fills the DOMAIN of MADE-OF. The ontological

entry for concept PHYSICAL-OBJECT with hard-coded properties is provided below:

```
(physical-object
   (definition  (value("objects that physically exist")))
   (is-a        (hier(object)))
(subclasses     (hier(surface-feature landscape-object
                animate animate-part animal-artifact
                material artifact celestial-object)))
)
```

The concepts CERAMIC, METAL, PAPER, PLASTIC, and WOOD can fill the RANGE of MADE-OF because their ancestor, MATERIAL, fills the RANGE of MADE-OF, as illustrated by the example of the concept MADE-OF below.

```
(made-of
   (definition  (value("the relation between a thing and
                things made out of it")))
   (is-a        (hier(physical-object-relation)))
   (inverse     (value(material-of)))
   (domain      (sem(physical-object)))
   (range       (sem(material)))
)
```

2.2. The lexical knowledge resource

The lexicon is a language-specific repository of word senses coupled with their morphological and syntactic information (for a detailed description of the template of a lexicon entry see [3]). As a static knowledge resource and a resource used directly by the OST text parser, the Lexicon fulfils two important functions.

In relation to the ontology, the main function of the Lexicon is to map the language-independent ontological knowledge to syntactic and semantic features of a specific language, including semantic idiosyncrasies. The mapping procedure can be either direct anchoring, if the ontology has a concept that exhaustively captures the meaning of a lexical sense (illustrated below by the entry "car-n1") or indicating the semantically nearest (typically a class) concept and specifying its meaning through properties and their restricted fillers (illustrated below by the entry "tourist-n1").

```
(car-n1
      (cat(n))
      (synonyms "")
      (anno(def "an automobile")(comments "")
            (ex "he drives a car"))
      (syn-struc((root($var0))(cat(n))))
      (sem-struc(car)))
(tourist-n1
      (cat(n))
      (synonyms "")
      (anno(def "a person who travels")(comments "")
            (ex "the tourists stayed at the hotel"))
      (syn-struc((root($var0))(cat(n))))
      (sem-struc(human(agent-of(sem(travel))))))
```

In relation to the OST text parser, the main function of the Lexicon is to provide the OST text parser with essential data about the word sense, its syntactic position and semantic information in the sentence so that the machine could (1) retrieve the proper ontological information about the sense, including concepts, their properties and property fillers, and (2) by computing property fillers, accommodate the given sense a text-meaning representation (TMR) of the natural language input.

The TMR is the ultimate product of the OST text parser. It comprehensibly translates a natural language input into a configuration of semantically related concepts, as illustrated by the below example of a TMR of the sentence, "the tourist broke the box", where the concepts HUMAN(agent-of(sem(travel))) and BOX fill the case roles of agent and theme of the clause-forming event DAMAGE. A more in-depth explanation of how TMR are computed within OST is offered in [4], Sections 6 and 8 (see also [3] and [5]).

(110) *The tourist broke the box.*

```
TMR 1:Weight: 4.2 Event: break-v1,
      damage1    agent(value (tourist-n1,
                 human1(agent-of(sem(travel)))))
                 theme(value (box-n1, box1 ))
```

3. Lexical acquisition

The practice of lexical acquisition involves the machine-readable description of every lexical sense in a domain-specific corpus, following the principle of complete coverage stated in [3] (Section 9.3). A well-acquired lexical entry (1) is anchored in an appropriate concept (which is evidenced from the concept's location in the ontological hierarchy, its ancestors, siblings, descendants, and its ontological and prosaic definition), (2) matches syntactic and semantic structures through properly co-indexed variables, and (3) reflects all possible syntactic positions the word may take in the sentence (for more details on steps of lexical acquisition see [3], Section 9.3.4). Procedurally, two lexical acquisition strategies are outlined in [3] (Sections 9.3.2 and 9.3.3). The first strategy, acquisition by rapid propagation, involves covering a large class of semantically and syntactically similar entries by applying, with slight modifications, one "master" lexical template. The degree of modification varies depending on the class size and the homogeneity of its members: while the acquisition of scalar adjectives would mostly require only changing the head concept ATTRIBUTE and its numeric value in the sem-struc, the acquisition of regular nouns like "car-n1", deverbal nouns like "investigation-n1", and deadjectival nouns like "beauty-n1" requires a greater

degree of syntactic and semantic variation. The second strategy is acquisition based on lexical rules of converting grammatical cognates like verbs (e. g. “enjoy-v1”) and their adjectival derivatives (e. g. “enjoyable-adj1”) due to their semantic similarity.

In order to facilitate various aspects of an application, several techniques of lexical acquisition may be defined.

1) Ontology-driven lexical acquisition involves “sliding” down the ontological hierarchy and making sure all concepts of the OBJECT and EVENT branches have minimal representation in the lexicon. While this technique is time-efficient and quickly produces a workable lexicon, its obvious downside is the limited size of the lexical entries, each of which will most likely have only one sense. This technique could be employed at the early stage of ontological acquisition, when the ontology is not yet complete, so that “lexicalizing” the concepts early on would make both resources available for parser-based testing, which is most beneficial in the overall ontology assessment and often points to necessary adjustments.
2) Parser-driven lexical acquisition involves running the OST text parser on a large number of domain-unrestricted corpora. Analyzing the resulting TMR’s allows establishing whether an additional lexical entry needs to be introduced or if it is the existing entry that has not parsed, in which case an adjustment is required. A properly conducted TMR analysis (informally known in the OST community as “blame-assignment”) also helps identifying whether the processing issues are rooted in the ontology, the onomasticon, or dynamic parsing modules.
3) Domain-driven lexical acquisition involves running the OST text parser on a domain-specific corpus. The corpus size and the depth of parsing are largely determined by the application purposes. The application also establishes the focus (e. g. grammatical classes) and the grain size (number of senses per entry) of lexical acquisition. To further fine-tune the acquired corpus to a specific domain, the priming functions can be introduced that prime (a) a lexical sense within the entry based on its general regularity in the language, and (b) a domain-specific lexical sense. This acquisition technique works best when aided by a pre-processing module comprising a tagger, a stemmer and a look-up function, which compares the corpora to the lexicon and identifies missing lexical entries with further part-of-speech sorting.

A usual build out of an application typically involves the interaction of the three lexical acquisition techniques described above. While technique (1) is largely restricted to early development phases, (3) is heavily guided by immediate objectives, (2) constitutes the backbone of lexical acquisition. When applied on a more limited scale and a case-by-case basis, this technique can also be used to test the functionality of every newly acquired or adjusted lexical entry. This is done by running a sample sentence (drawn from a corpus or emulated) with the new entry through the OST text parser, and analyzing the resulting TMR. The example below illustrates a typical lexical acquisition cycle supported by the TMR analysis.

Let us assume that a corpus related to the domain of crimes contains a sentence: “The police arrested the mole for stealing data from federal servers”. The running of the OST text parser returns no TMR for this sentence. For the sake of clarity, let us initially focus on the first clause, “the police arrested the mole”. The analysis starts by looking up the lexicon entries for “arrest-v” and “mole-n”:

```
(arrest-v1
    (cat(v))
    (anno(def "to seize a person by legal authority or warrant")(comments "")(ex "the police arrested the arsonist"))
        (synonyms "")
    (syn-struc(
        (subject((root($var1))(cat(np))))(root($var0))(cat(v))
        (directobject((root($var2))(cat(np))))))
    (sem-struc(arrest
        (agent(value(^$var1)))
        (beneficiary(value(^$var2(should-be-a(sem(human))))))))
```

The lexicon has the following entries for the word “mole”:

```
(mole
    (mole-n1
        (cat(n))(synonyms "")
        (anno(def "a insectivorous mammal living underground")(ex "he noticed a mole in the ground"))
        (syn-struc((root($var0))(cat(n))))
        (sem-struc(rodentia(agent-of(sem(life-event(location(sem(soil))))))))
    (mole-n2
        (cat(n))(synonyms "")
        (anno(def "a spot on the skin")(comments "")(ex "the man injured the mole"))
        (syn-struc((root($var0))(cat(n))))
        (sem-struc(skin(relative-size(less-equal(0.3)))(color(value(black brown)))))))
```

(1) *The police arrested the mole.*

```
TMR 1: Weight: 2.12 Event: arrest-v1,
                           arrest1
                           agent(value (police-n2, police-officer1 ))
                                 beneficiary(value (mole-n3, human1(agent-of (sem(spying)))))
```

(2) *The police arrested the mole for stealing data from federal servers*

```
TMR 1: Weight: 6.31 Event: arrest-v1,
                            arrest1
                            agent(value (police-n2, police-officer1 ))
                                  beneficiary(value (mole-n3, human1(agent-of(sem(spying)))))
                                  precondition(value (steal-v2, larceny
                                                theme(value (data-n1, information (origin(value (server-
                                                      n1, computer(connected-to(sem(network))))))))))
                                                      owned-by(value (government, federal-adj1))
```

None of the head concepts RODENTIA or SKIN in the sem-strucs of "mole-n1" or "mole-n2" can fill the beneficiary case role of ARREST, which is constrained to HUMAN according to the entry "arrest-v1". A lexicon acquirer would then conclude that a lexicon sense of "mole-n3" is needed which would (1) comprehensibly describe the meaning of the word "mole" as "a double agent, spy" and (2) have a head concept in its sem-struc that could fill the beneficiary case role of "arrest-v1"[1]. An ontological lookup will have no direct concept for SPY[2], so the nearest class concept will be listed in the sem-struc with the constraining property (agent-of(sem(spying))). To check whether this description is warranted by the ontology, the concept SPYING will be checked for its AGENT fillers. No restrictions are listed for the AGENT of the concept SPYING, which means that the machine will find the AGENT filler from the RANGE of the property AGENT in the ontology, and this filler is ANIMATE. Since HUMAN is a descendant of ANIMATE in the ontology, the sem-struc (human(agent-of(sem(spying)))) is supported by the ontology. The resulting sense will have the form:

```
(mole-n3
     (cat(n))
     (synonyms "")(anno(def "a double agent") (ex "the
          mole was arrested"))
     (syn-struc((root($var0))(cat(n))))
     (sem-struc(human(agent-of(sem(spying)))))
)
```

Re-running the clause with the OST text parser, would return the TMR [example (2)]:

In case the issues persist (e. g. no TMR is returned, the TMR is not correct, etc.) a more thorough insight into the output of every module would be needed, starting from the pre-processing steps of part-of-speech tagging and stemming.

The processing[3] of the second clause, "for stealing data from federal servers", will involve the clause-merging module of the OST parser. The module will do a lexical lookup of the preposition "for" and locate a sense "for-prep4", which is anchored in the property PRECONDITION (a child of EVENT-RELATION), whose DOMAIN and RANGE, in turn, have EVENT's as fillers. The two clauses will thus be merged into (arrest(precondition(sem(steal)))). The preposition processing module will be activated to process the noun phrase "data from the servers": the entry "from-prep1" will map on the property ORIGIN (a child of OBJECT-RELATION), whose DOMAIN and RANGE will be filled with INFORMATION and COMPUTER (identified through the lexical entries "data-n1" and "server-n1", respectively). The adjective processing module will be called to parse the adjectival phrase "federal servers": the property OWNED-BY (a child of SOCIAL-OBJECT-RELATION) will be located through the lexical entry "federal-adj1", its DOMAIN will be found to match the concept COMPUTER of the modified noun "server-n1", and the range filler GOVERNMENT (a child of ORGANIZATION) from the semantic structure of "federal-adj1" will be copied into the TMR for the concept COMPUTER. The resulting TMR of the whole sentence will have the form [example (2)]:

---

[1] Condition (1) clearly prevails over (2) since efficient processing is the ultimate goal of the system, and if the meaning is described accurately and the ontology cannot accommodate it, ontological adjustment is in order.

[2] The issue of a balanced trade-off in distributing knowledge between the ontology and the lexicon has been discussed in [3] (see also [7]). Whenever a lexical entry has no direct anchoring concept in the ontology, the decision whether a new concept should be added is guided by (1) the considerations of the parsimony of the ontology, which is a language-independent construct; (2) the purposes of a specific application, which defines the grain size of ontological and lexical acquisition.

[3] The author is grateful to the anonymous reviewers for emphasizing the need to illustrate/elaborate on the functionality of the Ontological Semantic Technology based on real-life data. The example contains clause embedment, prepositional phrase and an adjectival modifier, and its parsing would require the deployment and integration of several task-specific modules based on rich ontological and lexical knowledge resources.

4. Handling problematic cases in lexical acquisition in OST

The section below will describe how problematic cases can be acquired with the lexical acquisition inventory. More specifically, the lexical acquisition of verbs with Instrument-Subject alternation will be discussed.

A class of verbs exists where the event can be carried out through an agent or instrument [3].

(3) The man$_{[Agent]}$ broke the window

(4) The hammer$_{[Instrument]}$ broke the window

(5) The asteroid$_{[Instrument]}$ broke the window

(6) The hurricane$_{[Precondition]}$ broke the window

While different solutions were proposed to further stratify the instrument case role into intermediary or facilitating types or relax the notion of subject to include instrumental subjects [2, p. 80], within the framework of OST, the issue translates into the question of how to interchangeably accommodate two distinct case roles of AGENT, INSTRUMENT, and the relation PRECONDITION in one syntactic position indexed by a variable in the syn-struc of an entry like "break-v1".

In the ontology, the case role of AGENT has its RANGE restricted to ANIMATE, which rules out HAMMER (a descendant of ARTIFACT), ASTEROID (a descendant of CELESTIAL-OBJECT), and HURRICANE (a descendant of PHYSICAL-EVENT). On the other hand, the INSTRUMENT case role does not have an animate object in its RANGE, and the PRECONDITION is an EVENT-RELATION, which excludes any OBJECT by definition. Acquiring three separate lexical senses for AGENT INSTRUMENT and PRECONDITION is not entirely justified: all three entries would have shared the same root concept DAMAGE and would have been identical syntactically.

A reasonable solution would be to expand the semstruc of the entries like "break-v1" to include additional case roles that would map on one syntactic variable, which would result in the following entry:

```
(break-v1
    (cat(v))(anno(def "to cause to break")
    (ex "He broke the window. The hammer broke the window. The hurricane broke the window.")(comments ""))
        (synonyms "")
    (syn-struc ((subject((root($var1))(cat(np))))(root($var0))(cat(v))
    (directobject((root($var2))(cat(np))))))
    (sem-struc (damage(agent(value(^ $var1)))
    (instrument(value(^ $var1(should-be-a(sem(artifact animate-part material celestial-object)))
    (precondition(value(^ $var1))))))
    (theme(value(^ $var2(should-be-a(sem(artifact)))))))))
```

Such an entry conforms to the ontological restrictions, because the concepts ARTIFACT, ANIMATE-PART, MATERIAL, CELESTIAL-OBJECT constraining the instrument case role of "break-v1" are within the RANGE of the INSTRUMENT in the ontology, and the concept ARTIFACT constraining the theme case role of "break-v1" is within the RANGE of the property THEME in the ontology. The unconstrained case roles of agent and precondition in "break-v1" will be restricted by the RANGE of the property AGENT (which is ANIMATE) and the RANGE of the property PRECONDITION (which is EVENT).

When reading an entry above during the processing of examples (4–7), the OST text parser would selectively fill (and display in a TMR) the agent case role with HUMAN in (4), the instrument case role with HAMMER in (5), the instrument case role with ASTEROID in (6), and the precondition relation with HURRICANE in (7). The following TMR's will thus be derived:

(7) *The man broke the window*

```
TMR 1: Weight: 4.2200003 Event: break-v1,
                                 damage1
                                        agent(man-n1, human1(gender(value(male))))
                                        theme(value (window-n1, window1 ))
```

(8) *The hammer broke the window*

```
TMR 1: Weight: 4.2 Event: break-v1,
                          damage1
                                 instrument(value (hammer-n1, hammer1 ))
                                 theme(value (window-n1, window1 ))
```

(9) *The asteroid broke the window*

TMR 1: Weight: 4.16 Event: break-v1,
damage1
instrument(value (asteroid-n1 asteroid1))
theme(value (window-n1, window1))

(10) *The hurricane broke the window*

TMR 1: Weight: 4.18 Event: break-v1,
damage1
precondition(value(hurricane-n1 hurricane1))
theme(value (window-n1, window1))

The lexicon thus offers a very versatile toolbox for acquiring complicated word classes comprehensively. The rich ontology allows for a correct representation of semantic multiplicity as separate lexical senses. An exhaustive descriptive vocabulary of syntactic properties helps to accommodate syntactic variation in a single lexical entry, which keeps the lexicon meaningfully parsimonious. At the management level, the three acquisition techniques described above make it possible to calibrate the scope and grain size of acquisition to a specific task and based on a specific application. Lexicon acquisition informed by the OST text parser provides a most balanced and illuminating approach to quality control and improvement.

References

1. *Hempelmann C. F., Raskin V.* Semantic Search: Content Vs. formalism // Rome: Proceedings of Langtech 2008. http://www.langtech.it/en/technical_program/technical_program.html (full paper).
2. *Levin B.* Verb Classes and Alternations: A Preliminary Investigation // London and Chicago. The University of Chicago Press, 1993.
3. *Nirenburg S., Raskin V.* Ontological Semantics // Cambridge, MA: MIT Press, 2004. A prepublication draft, chapter by chapter, can be found on www.ontologicalsemantics.com
4. *Petrenko M.* 2009. Ontological semantics and abduction: parsing ellipsis.// Dialog 2009.
5. *Petrenko M., Raskin V.* Modeling Abduction within Ontological Semantics // Proceedings of Midwestern Computational Linguistics Colloquium 5. East Lansing: Michigan State University, May 2008.
6. *Raskin V.* The how's and why's of ontological semantics. In: I. M. Kobozeva, A. S. Narinyani, and V. P. Selegei (eds.) // Zvenigorod: 2005. Computer Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog 2005".
7. *Raskin V., Hempelmann C. F. and Taylor J. M.* Guessing vs. Knowing: The Two Approaches to Semantics in Natural Language Processing. // In this volume, 2010
8. *Raskin V., Hempelman C. F., Taylor J. M., Petrenko M. S., Trienzenberg K. E., Buck B.* The Why's, How's, and What-of's of Natural Language Ontology // Meaning Computation 1 (forthcoming), 2010.