

The Architecture and Functionality of Ontological Semantics

Author: Max Petrenko

Affiliation: Moscow Dashkova University (Moscow, Russia); RiverGlass, Inc (Champaign, IL, USA).

E-mail: mpetrenk@gmail.com

Abstract

The paper presents static and dynamic resources of the Ontological Semantic Technology (OST) – a meaning-driven and knowledge resource-based NLP system – and illustrates its functionality by discussing cases of ambiguity detection and resolution within the framework of OST. First, the knowledge resources – the ontology and the lexicon – are outlined with pertinent examples provided. Second, an example of OST-based sentence processing is discussed. Third, it is shown how sense multiplicity (homonymy and polysemy) can be resolved in the OST lexicon, how structural ambiguity is detected by correctly deriving two possible input readings, and how lexical ambiguity is resolved by reading contextual clues.

Keywords: ambiguity, computational semantics, ontology, lexicon, natural language processing

The paper introduces the architecture and functionality of the Ontological Semantic Technology. After an outline of static and dynamic Ontological Semantics, it will be illustrated how standard cases of ambiguity are detected and resolved within the Ontological Semantic Technology.

1. Static and dynamic resources of Ontological Semantic Technology

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 language-independent and highly structured conceptual hierarchy and (2) furnish lexical, morphological, and syntactic knowledge essential for parsing natural language input meaningfully (a number of publications have elaborated on the need of “doing semantics semantically” and domain-independent knowledge acquisition as a necessary prerequisite of any NLP system: (Nirenburg, Raskin 2004); (Raskin 2006); (Hempelmann, Raskin 2008); (Raskin et al. 2010). This paper will offer a concise outline of Ontological Semantics resources. A much more detailed description of the methodology, body, purview and applications of the Ontological Semantic Technology (OST) can be found in (Nirenburg, Raskin 2004, (Taylor et al. 2010); (Petrenko 2009).

1.1. The ontological knowledge resource

The ontological knowledge resource comprises thousands of concepts and starts with a top-level ALL → OBJECT, EVENT, PROPERTY branching. As Figure 1 illustrates, the further breakdown of concepts is meant to capture all relevant domains of the world view with the basic coverage, whose grain-size is subject to application requirements.

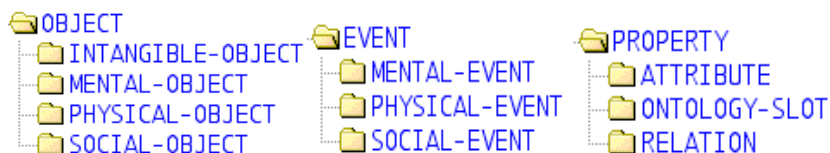


Figure 1. OST basic ontology branching

The structure of the ontology within OST advances far beyond the IS-A-type of relations and features hundreds of properties including case-roles, absolute and relative attributes, mereological and spatial relations between OBJECTs and EVENTs, measurement relations, personal name-relations, etc. The

concept-filler relation follows the triple “property-facet-filler” format and is supported by a flexible set of facets, DEFAULT, SEM, and RELAXABLE-TO, which help prioritize property fillers (see Nirenburg, Raskin 2004; Taylor et al. 2010 on the role of facets in OST).

The example below features the concept STOREFRONT:

```
(storefront
  (definition
    (value("a store")))
  (is-a
    (hier(building-interior)))
  (inside-of
    (default(building)) building-interior)
  (occupied-by
    (sem(retail-organization)))
  (made-by
    (default(human)) artifact)
  (location-of
    (default(sell)))
)
```

The IS-A property links the concept to its parent BUILDING-INTERIOR

The concepts RETAIL-ORGANIZATION (a descendant of GROUP) and SELL (a child of ECONOMIC-EVENT) fill the slots for the properties OCCUPIED-BY and LOCATION-OF, which are inverted versions of the properties OCCUPIES (a child of OBJECT-RELATION) and LOCATION (a child of CASE-ROLE). The facet DEFAULT on the concept SELL allows additional fillers to be added during acquisition or processing.

The italicized properties INSIDE-OF and MADE-BY have been inherited from the parental BUILDING-INTERIOR and the ancestral ARTIFACT. The inheritance consistency mechanism ensures that the property fillers of a concept are either inherited from the concept’s ancestors or specify i.e. are children of, the inherited fillers.

An important feature of OST ontology is the domain/range parameters of ontological properties. An ontological property is defined through its domain, i.e. the class of objects it applies to, and range, i.e. the number of values it takes. For non-literal properties like CASE-ROLE or RELATION, the range is defined through other ontological concepts. For literal properties like ATTRIBUTE, the range is defined through a numeric value or a literal. The consistency and adequacy of restrictions on domain and range fillers of ontological properties is crucial for proper case-role filling during computation and for lexicon acquisition. A too narrow or too broad restriction on the domain or range of a property will lead to a lack or redundancy of incorrect input readings including incorrect case role filing, false positives, etc. For example, the AGENT case role in the ontology has EVENT in its domain and ANIMATE in its range, which means that any EVENT can have an AGENT, and any child of ANIMATE can be an AGENT. If the restriction on the range is relaxed to OBJECT, then for sentences with subject/instrument alternation (see Levin 1993) like “the hammer broke the window” HAMMER would be warranted as both AGENT and INSTRUMENT case role filler, which would result in an incorrect text meaning representation.

1.2. The lexicon

The lexicon knowledge resource provides complete coverage of word senses (including closed classes, non-compositional units, phrasal verbs, idiomatic expressions and proper name databases) in a

specific language including morphological properties and syntactic functions (see Nirenburg, Raskin 2004 on a detailed discussion of lexicon structure, (Petrenko 2010) on lexicon management strategies, and (Taylor et al. 2010) on automated lexicon acquisition toolbox). Lexical entries are linked to ontological concepts either by direct mapping, in case there is a one-to-one concept-entry correspondence (e.g. the entry “human-n1” is hooked up to the concept HUMAN) or by accommodation, i.e. restricting the nearest available parent (e.g. “woman-n1” is mapped onto HUMAN with the FEMALE value of the property HAS-GENDER). While the grain size of lexicon coverage is determined by the needs of an application, a typical lexical entry comprises 2-5 senses capturing grammatical derivatives, polysemy, phrasals and idiomatic expressions.

A lexical entry template is shown below:

```
(head-entry
  (sense-1, 2, 3...
    (cat(n/v/adj/adv/pro/prep/conj))
    (synonyms "")
    (anno
      (def "")
      (comments "Acquired by <acquirer name> on <date> at <time>.")
      (ex ""))
    )
    (syn-struct((root($var0))(cat(n/v/adj/adv/pro/prep/conj))
      (subject/object((root($var#))(cat(np/vp/s))))))
    )
    (sem-struct
      (root-concept
        (property(value(^$var#(should-be-a(default/sem(concept)))))))
      )
    )
  )
)
```

The fields “synonyms”, “anno”, “comments”, and “ex” are of no value to the computer and serve the human acquirer. The fields “cat”, “syn-struct”, and “sem-struct”, meaning “category”, “syntactic structure”, and “semantic structure”, respectively, contain crucial information about the sense used by the parser in meaning computation.

The OST lexicon provides extensive coverage for categorical features, syntactic behavior and the lexical properties of its entries. In every entry, syntactic variables are co-indexed with their semantic interpretations. For noun entries, the root \$var0 in the syntactic structure maps onto the appropriate root concept in the semantic structure, where additional ontological properties are also listed. For verbal, adjectival, adverbial entries, \$var0 also indicates the word’s position in relation to basic syntactic arguments (i.e. subject and object for verbs). Every new argument is assigned a new co-indexed variable pair in syntactic and semantic structures. Additional prepositional, phrasal or infinitival adjuncts can be included with their optionality marked, if necessary. If more than one word order is possible, additional syntactic structures are listed.

The entries below illustrate a noun and verb sense for the entry “store”:

```
(store
  (store-n1
    (cat(n))
    (anno(def "a building where diversified goods are kept"))
```

```

    (comments "")
    (ex "he went to the store across from the street"))
    (synonyms "")
    (syn-struc((root($var0))(cat(n))))
    (sem-struc(storefront))
(store-v1
  (anno(def "to place and leave at a location")
    (ex "he stored guns in a barn")
    (comments ""))
    (synonyms "")
    (cat(v))
    (syn-struc((subject((root($var1))(cat(np))))(root($var0))(cat(v))
      (directobject((root($var2))(cat(np))))
      (pp-adjunct((root(into in inside))(opt(+))(cat(prep))(obj((root($var3))))))))))
    (sem-struc(put(agent(value(^$var1(should-be-a(sem(human))))))
      (theme(value(^$var2(should-be-a(sem(animal-artifact animate-part artifact
material)))))))
      (end-location(value(^$var3(should-be-a(default(building))))))))))
)

```

The variables 1, 2 and 3 for the case roles of agent and theme in the semantic structure of “store-v2” are co-indexed with variables 1, 2 and 3 of the syntactic structure: this allows the OST parser to assign the correct case role to the word based on its syntactic position. Besides the agent and theme case roles, the optional adjunct with prepositions “into/in/inside” provides support for the location of stored objects, captured with the case role END-LOCATION restricted to the concept BUILDING as default filler.

The OST Lexicon is acquired semi-automatically. While the number of lexicon senses per entry, the composition, the number and order of syntactic structures, root concepts, pertinent properties and their fillers are determined by the human acquirer, the consistency of fillers with the ontology, domain/range consistency and format check is fully automatic.

During various implementations of OST, an acquisition strategy was developed which significantly speeds up acquisition rate: developing a master entry and propagating it among other class members. The rapid propagation strategy is most efficient when applied to closed classes with homogeneous members. Adjectives are a good example of such a class: for many adjectives the variation of meaning translates into the variation of numeric values for one attribute (e.g. “horrible – bad – good – excellent” would differ in values of the DESIRABILITY modality). For more details on the adjectives acquisition methodology and the discussion of meaning plasticity adjectives see (Raskin and Nirenbug 1993).

While several lexicon acquisition techniques have proven efficient during lexicon management within OST, the most effective technique is domain-independent and parser-based acquisition, which involves testing the newly acquired entry by the parser and determining its functionality. This provides for good ontology/lexicon synchronization, allows for an efficient quality control and offers fruitful insights into functioning parsing modules. For more details on acquisition strategies and techniques (see Nirenburg, Raskin 2004; Petrenko 2010).

1.3. The dynamic knowledge resource

The core function of the lexicon is to provide the OST parser with full access to appropriate ontological concepts and their property fillers (which guide the meaning computation) restricting those fillers where necessary to avoid generating false positives. The multi-modular OST parser starts with a

lexical lookup on the tagged words of the input. After the head EVENT of the clause has been identified, its case role slots are filled with remaining concepts (identified through respective lexical entries) heeding the restrictions from the lexical sense for the verb entry for the head EVENT or the restrictions in the ontological definition of the concept. Depending on the tagging results, additional modules for processing closed classes (e.g. prepositions), remaining nouns and unattested units are called to build up the ultimate text meaning representation (TMR). For multi-clausal input, clause-merging module is activated, which recursively combines the TMR's for each clause. A most detailed description of TMR generation is given in (Nirenburg, Raskin 2004); various developments were discussed in (Raskin et al. 2008; Raskin et al. 2010, and Hempelmann et al. 2010) including processing number expressions (Taylor et al. 2010), employing abductive reasoning modules (Petrenko, Raskin 2008; Petrenko 2009).

To illustrate, the processing of the example from the lexical entry “store-v1” would produce the following TMR:

(1) *He stored guns in the barn.*

TMR 1: Weight: 5.2 Event: store-v1,

put1

agent(value (he-pro1, human1(gender(value(male))))))

theme(value (gun-n1, gun1))

end-location(value (barn-n1, barn1))

First, the lexical lookup for every word returns available lexicon senses. The TMR building phase then defines concept PUT (identified through “store-v1”) as the head event. The syntactic structure features the “subject – direct object – prepositional adjunct” order, and the semantic structure of “store-v1” features AGENT, a THEME and an optional END-LOCATION case roles. Second, the OST parser then proceeds, observing the syntactic order, to fill the case roles of “store-v2” by matching them with concepts defined in input words: HUMAN(GENDER(VALUE(MALE))) (defined in “he-pro1”) for AGENT, GUN (defined in “gun-n1”) for THEME, and BARN (defined in “barn-n1”) for END-LOCATION. Checking the restrictions on each case role, the OST parser then determines that the restrictions for the case roles in “store-v1” are met by the three concepts. The concept HUMAN(GENDER(VALUE(MALE))) meets the restriction on AGENT. The concept GUN meets the restriction on THEME of PUT by virtue of being a descendant of one of its fillers, the concept ARTIFACT. The concept BARN meets the restrictions on END-LOCATION of PUT by virtue of being a descendant of its filler BUILDING.

2. OST functionality

The section will illustrate the OST in action by discussing how problematic lexicon cases are acquired and how standard cases of ambiguity are detected and resolved.

Among difficult lexicon items that, as per (Zernik 1990), traditionally constitute an NLP bottleneck, this paper addresses two that regularly lead to ambiguity – sense multiplicity and idioms – and illustrates how these cases can be non-controversially acquired in the OST lexicon.

Sense multiplicity traditionally comprises homonymy (e.g. several unrelated senses for one entry) and polysemy (e.g. several semantically motivated senses for one entry). Both constitute traditional culprits in NLP disambiguation issues (see Zernik, 1990, Hirst, 2003, Pustejovsky, Boguraev 2005). Consider standard examples:

(2) a. *He worked at a bank.*

b. *He moored his boat near the bank.*

c. *He waited by the bank.*

(3) a. *He read the book.*

b. *He burned the book.*

c. *He liked the book.*

While cases (a-b) are non-ambiguous and involve polysemous senses for “bank” in (2) (i.e. COMMERCIAL-BUILDING vs. COAST) and homonymous senses for “book” in (3) (i.e. TEXT vs. ARTIFACT), cases (c) conflate both and create ambiguous sentences.

The richness of the resources and the descriptive language of OST allow to effectively differentiate the senses in cases like (a-b) and determine ambiguity in cases like (c).

The senses for “bank” in the lexicon are below:

(bank-n1

```
(cat(n))(synonyms "")
(anno(def "a building occupied by a banking organization")(comments ""))
(ex "he entered the bank")
(syn-struc((root($var0))(cat(n))))
(sem-struc(commercial-building(occupied-by(sem(bank))))))
```

(bank-n2

```
(cat(n))(synonyms "")
(anno(def "a coast over a body of water")(comments "")(ex "he saw ducks on the bank"))
(syn-struc((root($var0))(cat(n))))
(sem-struc(coast)))
```

Based on ontological restrictions for respective events in 2(a-b), COMMERCIAL-BUILDING will be the only legitimate filler for the LOCATION case role of the event WORK in (a), and COAST will only be warranted as a range filler for the relation BESIDE with the concept BOAT in its range. However, the concept WAIT warrants both COAST and COMMERCIAL-BUILDING as fillers for the LOCATION case role, thus causing ambiguity.

The polysemy of “book” in (3) is captured in the lexicon in the following senses:

(book-n1

```
(cat(n))(anno(def "information as text")(ex “he read the book”)(comments ""))(synonyms "")
(syn-struc((root($var0))(cat(n))))
(sem-struc(text)))
```

(book-n2

```
(cat(n))(anno(def "a hard copy")(ex “he burned the book”)(comments ""))(synonyms "")
(syn-struc((root($var0))(cat(n))))
(sem-struc(book)))
```

TEXT in “book-n1” is a child of INFORMATION in the ontology and hence does not meet restrictions for the THEME of BURN but is a legitimate THEME filler for the event READ. On the other hand, BOOK in “book-n2” is a descendant of ARTIFACT in the ontology and hence does not meet restrictions for the THEME of READ but is a legitimate THEME filler for the event BURN. The obvious ambiguity is detected in 3(c), where the concept AWE in “like-v1” allows both INFORMATION and BOOK as fillers for THEME.

Idioms are another recalcitrant case which is non-compositional and requires adequate representation (Zernik 1990). A standard example of idiom-driven ambiguity is 4(c):

- (4) a. *He bought the farm with livestock.*
 b. *He got injured in the fire and soon bought the farm.*
 c. *He bought the farm last week.*

While (a-b) provide contextual clues for disambiguating the non-compositional “buy the farm” and the standard “VP+NP+PP” case, (c) blends both and needs a representation of both readings. The OST lexicon captures the idiom as a verb sense of “buy” with the non-compositional element “farm”

represented only syntactically, replacing the variable for the direct object in the syntactic structure, without any semantic interpretation in the semantic structure, which makes the entry intransitive:

```
(buy-v6
  (cat(v))
  (anno(def "idiomatic: to die")(comments "")(ex "the sick man bought the farm last week"))
  (synonyms "")
  (syn-struct( (subject((root($var1))(cat(np))))(root($var0))(cat(v))
               (directobject((root(farm))(cat(np))))))
  (sem-struct(die (experiencer(value(^$var1(should-be-a(sem(human))))))))))
```

By making the non-compositional “farm” semantically “invisible”, such a format eliminates unnecessary semantic computations “farm” would have entailed during processing and results in a correct interpretation for 4(b): the parser prioritizes “buy-v6” over the regular sense of “buy” by computing from the prior concept INJURY to DIE through the PRECONDITION case role (for more details on idiomatic expressions processing within OST see Taylor et al. 2010). On the other hand, 4(a) carries a full-fledged “farm-n1” by introducing a prepositional phrase “with livestock”, which requires a domain filler for the LOCALE-OF relation defined in the prepositional sense of “with-prep5”:

```
(with-prep5
  (anno(def "serving as a locale of something")(comments "")(ex "he saw a boat with people"))
  (cat(pre))
  (synonyms "")
  (syn-struct((np((root($var1))(cat(np))))(root($var0))(cat(pre))(obj((root($var2))(cat(np))))))
  (sem-struct(^$var1(locale-of(value(^$var2))))))
```

The remainder of the section will discuss how structural ambiguity is detected and resolved within OST. Within the framework of OST, ambiguity detection translates into the issue of deriving several TMR’s with a reasonably high weight score. Ambiguity resolution in turn translates into the issue of re-ranking these “competing” TMR’s based on additional ontological, lexical restrictions. If none are available, contextual clues are sought in order to eliminate ambiguity.

Let us consider a textbook example of a structurally ambiguous sentence:

(5) *He saw a man with binoculars.*

The obvious ambiguity of the prepositional phrase, which can either be dominated by the noun or by the VP is captured within OST as the ambiguity of the concept VIEWING-ARTIFACT either filling the instrument case role for the head event SEE or filling the range of the property OWNER-OF with (human(gender(value(male)))) being the range filler:

(5) *He saw a man with binoculars.*

TMR 1 Weight: 4.7 Event: see-v1,

see1

```
agent(value (he-pro1, human1(gender(value(male))))))
theme(value (man-n1, human2(gender(value(male))))))
owner-of(value (binoculars-n1, viewing-artifact1))
```

TMR 2 Weight: 4.7 Event: see-v1,

see1

```
agent(value (he-pro1, human1(gender(value(male))))))
theme(value (man-n1, human2(gender(value(male))))))
instrument(value (binoculars-n1, viewing-artifact1))
```

Similarly to the human agent, the system would detect ambiguity and return both readings as equally possible. When disambiguating clues are available from the context, the two TMR's would be re-ranked and priorities re-assigned. Consider a traditionally ambiguous example with available contextual clues:

(6) *She has started a new business. You should see her shop.*

In the second sentence, the word “shop” can typically be tagged as a noun or verb and therefore triggers a number of possible readings resulting in several possible TMR's. In (6), however, the sentence is disambiguated by virtue of the “managing a company” reading being primed in the previous clause. OST parser takes advantage of this: it determines that the “prime” BUSINESS is an ancestor of RETAIL-ORGANIZATION in the ontology and assigns a greater weight to the TMR with “shop” tagged as noun and the underlying concept RETAIL-ORGANIZATION filling in the domain of the property HEADED-BY, with (human(gender(value(female)))) being a range filler (see Petrenko 2010 for a more detailed discussion of priming-based disambiguation strategies in OST):

(6) *She has started a new business.*

TMR 1 Weight(TMR): 4.7 Event: start-v1,

create-social-object1 (phase(value(begin)))
agent(value (she-pro1, human1(gender(value(female))))))
theme(value (business-n1, **business1**)))

You should see her shop.

TMR 1 Weight(TMR): 4.7 Event: see-v1,

see1 (desirability(greater-equal(0.5)))
agent(value (you-pro1, human1))
theme(value (shop-n1, **retail-organization1**))
headed-by (value (she-pro1, human1(gender(value(female))))))

References

- Hempelmann, Christian; Victor Raskin . 2008. Semantic search: content vs. formalism *Proceedings of Langtech 2008*. http://www.langtech.it/en/technical_program/technical_program.html (full paper).
- Hempelmann, Christian; Taylor Julia; Victor Raskin, 2010. Application-guided ontology engineering In *Proceedings of the International Conference on Artificial intelligence*. Las Vegas, NV.
- Hirst, Graeme. 2009. Ontology and the Lexicon. In *Handbook on Ontologies. International Handbooks Information System. Part 2. Springer*.
- Levin, Beth. Verb Classes and Alternations: A Preliminary Investigation. 1993. *The University of Chicago Press*.
- Nirenburg, Sergei; Victor Raskin. 2004. Ontological Semantics. *MIT Press*. A prepublication draft, chapter by chapter, can be found on www.ontologicalsemantics.com
- Petrenko, Max. 2009. Ontological semantics and abduction: parsing ellipsis. In *Papers from the Annual International Conference Dialog 2009*. Moscow.
- Petrenko, Max; Victor Raskin. 2008 Modeling Abduction within ontological semantics. In *Proceedings of Midwestern Computational Linguistics Colloquium 5*. East Lansing: Michigan State University.
- Petrenko, Max. 2010. Priming-based disambiguation in ontological semantics. In *Proceedings of the International Conference on Artificial intelligence*. Las Vegas, NV.
- Pustejovsky, James; Branimir Boguraev. 2005. Lexical Semantics in Context. In *Lexical Semantics: The Problem of Polysemy. Oxford University Press*.
- Raskin, Victor; Hempelmann Christian; Julia M. Taylor. 2010. Guessing vs. knowing: the two approaches to semantics in Natural Language Processing”. In *Papers from the Annual International Conference Dialog 2009*. Moscow.
- Raskin, Victor; Sergei Nirenburg. 1998. An Applied Ontological Semantic Microtheory of Adjective Meaning for Natural Language Processing. in *Machine Translation*. Vol. 13. 2/3.
- Taylor, Julia; Hempelmann, Christian; Victor Raskin. 2010. On an automatic acquisition toolbox for ontologies and lexicons. In *Proceedings of the International Conference on Artificial intelligence*. Las Vegas, NV.
- Taylor, Julia; Raskin, Victor., Hempelmann, Christian, Whitney Vandiver. 2010. Computing the meaning of number expressions in English: the common case. In *Proceedings of the International Conference on Artificial intelligence*. Las Vegas, NV.
- Zernik, Uri. 1990. Lexical Acquisition: where is the semantics. In *Machine Translation*. Vol 5. 2. *Lexical Semantics*.