Domain ontologies and wordnets in OWL: Modelling options

--Extended Abstract--

Harald Lüngen, Angelika Storrer

Justus-Liebig-Univerität Gießen
Angewandte Sprachwissenschaft und Computerlinguistik
harald.luengen@uni-giessen.de

Universität Dortmund
Institut für deutsche Sprache und Literatur
angelika.storrer@uni-dortmund.de

1. Project scenario and goals

Word nets are lexical reference systems that follow the design principles of the Princeton WordNet project (Fellbaum 1998, henceforth referred to as PWN[1]). Domain ontologies (or domain-specific ontologies, e.g. GOLD[2] or the GENE Ontology[3]) represent knowledge about a specific domain in a format that supports automated reasoning about the objects in that domain and the relations between them (cf. Erdmann 2001, 78). Word nets have been used in various applications of text processing, e.g. discourse parsing, lexical and thematic chaining, cohesion analyses, automatic segmentation and linking, anaphora resolution, and information extraction. When these applications process documents dealing with a specific domain, one needs to combine knowlegde about the domain-specific vocabulary represented in domain ontologies with lexical repositories representing general vocabulary (like PWN). In this context, it is useful to represent and interrelate the entities and relations in both types of resources using a common representation language. In our research group "Text-technological Information Modelling[4]" we chose OWL as a common format for this purpose. Since our projects are mainly concerned with German documents, we developed an OWL model that relates the German wordnet GermaNet (henceforth referred to as GN)[5] with domain-specific ontologies in an approach that was inspired by the Plug-In model proposed in Magnini/Speranza (2002). Our approach is decribed in Kunze et al. (to appear); it was evaluated using representative subsets of GN and of the domain ontology TermNet[6] (henceforth referred to as TN) as data and Protégé

[1] http://wordnet.princeton.ed

[2] http://www.linguistics-ontology.org/gold.html

[3] http://www.geneontology.org/

[4] cf. http://www.text-technology.de

[5] http://www.sfs.uni-tuebingen.de/lsd/

[6] cf. Beisswenger u.a. 2003

(version 3.1.1) and RacerPro (version1.9)[7] for editing and consistency checking. In this approach, the technical terms of the two domains, “text technology” and “hypertext research”, which are represented in TN, are modelled as classed related by the OWL `<rdfs:subclassOf>` relationship and other semantic relations of the PWN model. The OWL model for GN, by contrast, follows the design guidelines of the OWL representation of the PWN suggested by van Assem et al. 2006 and represents the synsets of GN as instances of word-class-specific synset classes.

In the present paper, we present an alternative GN model that represents synsets as classes and compare the two modelling options. After some general remarks on the status of word nets in the discussion about ontologies in section 2, we briefly review some modelling decisions taken in approaches concerned with the OWL conversion of the English PWN (section 3). In section 4 we discuss our class-based modelling approach to be used in the the above mentioned application scenarios.

2. Word nets and domain ontologies

The Princeton WordNet was initially not conceived as an ontology, but rather as a psychologically motivated model of lexical knowledge.[8] However, in ontology textbooks and ontology portals, the PWN is often mentioned as a resource. In his ontology glossary, Sowa explicitly lists PWN as an example for a “terminological ontology”. Following his definition, terminological ontologies are ontologies “whose categories need not be fully specified by axioms and definitions” (Sowa 2001). The term is opposed to formal ontologies whose categories “are distinguished by axioms and definitions stated in logic or in some computer-oriented language that could be automatically translated to logic” (Sowa 2001). A similar distinction is drawn in Erdmann (2001): he differentiates between *light weight ontologies* which consist primarily of a representation schema providing means to specify taxonomies and to define additional features and relations, and *heavy weight ontologies* which are specified in a logic-based representation language (cf. Erdmann 2001, 72). In this sense, PWN would be classified as being a light weight ontology – and PWN is, indeed, mentioned in the list of possible ontology resources (Erdmann 2001, 71).

Especially the hierarchical semantic relations of the PWN model (hyponymy and meronymy) have been used in various NLP-related applications. Many approaches concentrate on the noun synsets and interpret hyponymy between noun synsets as a superclass-subclass-relation in the sense that all instances of a hyponym synset (expressing a specific concept) are also instances of its hypernym synset (expressing a superordinate concept with a broader meaning). But since PWM and word nets of other languages were not designed with such an explicit semantic interpretation in mind, one should be aware that reasoning and consistency checking on word-net-based ontologies may produce suprising results (cf. Hirst 2004). However, since the PWN database is freely available and is assumed to be a useful resource for ontology-driven approaches in NLP and Semantic Web applications, several approaches exist for the conversion of the PWN database in OWL or RDFS (see section 3 for details).

[7] http://www.racer-systems.com/

[8] Miller/Hristea (2006,1)

When the description logic foundation of OWL DL is used, this conversion step implies a (at least partial) transformation of a light weight into a heavy weight ontology (in the sense specified above), in which several modelling options are possible. In section 4 of this paper, we will focus on one of these options, namely the question whether synsets should be modelled as classes or as instances.

The choice between one modelling option or the other is highly dependent on the application context in which the ontology is to be used. The modelling approach described in section 4 has been developed with the following application framework in mind:

- The model is designed to be used in text processing applications such as lexical chaining, anaphora resolution, discourse parsing, information extraction, and text classification.
- Since some of these applications may deal with documents in a specific domain, we aim at a common representation format for domain-specific and general ontologies.
- In order to make use of standard OWL reasoning tools, the modelling is restricted to OWL DL, i.e. we do not want to use language constructs that belong to OWL Full.

3. Modelling Wordnets in OWL

In a preliminary study we examined three existing approaches to representing PWN in OWL, focussing on the question of what ontological modelling decisions have been taken: The W3C approach (“W3C”), (a working draft by the Semantic Web Best Practices and Deployment Working Group, cf. van Assem et al. 2006), the “Neuchâtel approach” (“NCH”) (Ciorašcu et al. 2003) and the “Amsterdam approach” (“AMST”) (van Assem et al. 2004). NCH has partly been considered in W3C, and the group of authors of AMST overlaps with that of W3C, thus AMST seems to be a predecessor of W3C. The three approaches differ in their goals: W3C aims at providing a standard conversion of PWN in OWL that can be used directly by Semantic Web applications, and where the converted OWL version should not deviate from the original version, i.e. the PWN data model should be reflected in OWL without further interpretations. The goal of AMST was also to provide an OWL-encoded version of PWN. The main objective of NCH, though, was to create a test domain for the ontology-based information system *knOWLer* and to demonstrate its performance by means of sample queries. A discussion of modelling alternatives did not play a role in this effort. In W3C, WordNet version 2.0 was converted into OWL, in NCH, the version 1.7.1 was converted.

In W3C and NCH, the actual ontology (i.e. the class hierarchy without the instances, cf. Erdmann 2001, p. 74), which is called the “WordNet RDF/OWL schema” in W3C, consists of the class of Synsets (W3C: *Synset*, NCH: *LexicalConcept*) and its subclasses *Noun(Synset), Adjective(Synset), Adverb(Synset),* and *Verb(Synset),* where *Adjective(Synset)* has a further subclass called *AdjectiveSatellite(Synset).* Lexical Units are modelled by the class *WordSense* in W3C and by the class *WordObject* in NCH. In W3C, *WordSense* is further subdivided into part of speech-specific subclasses like *NounWordSense*, in NCH, it is not. Besides, for purely formal units (not associated with a meaning), the class *Word* exists. In NCH, a corresponding class called *StemObject* exists only in an external Ontology which is used for document retrieval. In AMST, the lexical relations are encoded by dint of "helper classes" such as *SynSetVerb*.

The single Synsets as e.g. {horse, nag, steed} are modelled as individuals, i.e. as instances of *NounSynset, VerbSynset* etc. in all three approaches. Likewise, the single lexical units (e.g. *horse*) are modelled as individuals in W3C as well as in NCH. Consequently, the lexicalisation relation (the relation that connects Synsets and Lexical Units) is an OWL ObjectProperty with the domain *Synset* (*LexicalConcept*) and with the range *WordSense* (*WordObject*) (the relation is called *synsetContainsWordSense* in W3C and *wordForm* in NCH), thus connecting a Synset individual to one or more Lexical Unit individuals. In AMST, Lexical Units are modelled as neither classes nor individuals, but as literals which appear as values of the multiple-valued DatatypeProperty *wordForm* (domain: *Synset*). Furthermore, in all three approaches, further ObjectProperties with domain and range = *Synset* exist, which model the PWN conceptual relations (e.g. *hyponymOf, entails*, and partly their POS-specific restrictions) in OWL. In a similar fashion, the PWN lexical relations (e.g. *antonymOf, participleOf*) are represented as ObjectProperties with domain and range = *WordSense* in the OWL versions of W3C and NCH. Moreover, the W3C approach contains instructions how to interpret the PWN hyponymOf relation as a subproperty of the *subclassOf* property in OWL, using a strategy first suggested by Dan Brickley.

The OWL model of the German GermaNet suggested in Kunze et al. (to appear) is based on W3C, but it also includes special features making allowance for certain distinctive features of GermaNet such as the explicit marking of proper names, the non-inverse relationship between Holonymy and Meronymy, and the use of so-called artificial concepts.

In general, the existing models (maybe apart from NCH) were not developed with NLP applications (such as discourse parsing, anaphora resolution, automatic linking) in mind. It seems at first striking and unusual that synsets, i.e. sets of quasi-synonymous units, and their members, the disambiguated lexical units, should be modelled as classes and not as individuals. Unusual, because Synsets are frequently considered as concepts which can be referenced linguistically by the lexical units contained in the synsets; e.g. a synset formed by {horse, nag, steed} denotes the horse concept. This suggests that at least the synsets should be conceived as classes, the instances of which are individual objects (e.g. the horse "Fury"). But in principle, the lexical unit "horse" can also generically refer to the whole class (e.g. in meaning postulates like "A sorrel is a reddish horse"). The decision to model synsets and lexical units as individuals is thus not at all obvious and shall be discussed critically and be compared with possible alternatives in the following section.

4 Discussion of modelling options

Designing an OWL representation for a word-net-style resource implies that one interprets the semantics of the entities and relations used in the original lexical resource with respect to the interpretation that reasoners (e.g. RacerPro[9] or FaCT++[10]) assign to the OWL constructs of the resulting word net model. In this interpretation process there are different modelling options. One of these options, which we will focus on in the following, is concerned with the decision whether the basic entities of the word net model, the synsets, are represented (1) as instances of word-class specific classes (henceforth called "instance model") or whether (2) each synset represents a class of its own (henceforth called "class model"). As discussed in the previous section, most previous approaches to the conversion of the PWN to OWL use the instance model. In the following we want to vouch for the class model for two reasons:

1) The Princeton word net, in its version PWN 2.1, draws an explicit distinction between the relation of hyponymy on the one hand (e.g. the subordinate synset containing "peach" is a hyponym of the superordinate synset containing "drupe") and the class-instance relation on the other (e.g. the proper name "Berlin" is an instance of the synset containing "city")[11]. Over 7.600 PWN synsets were manually classified as being instances and tagged as such. Despite this introduction of the class-instance distinction, the PWN version 2.1 may still be converted to OWL using the instance model, e.g. by ignoring the class-instance distinction or by skipping the synsets tagged as instances. However, Miller/Hristea introduced this distinction with the aim to support ontologists "to distinguish between a concept-to-concept relation of subsumption and an individual-to-concept relation of instantiation" (Miller/Hristea 2006, 1). In our view, this aim implies that synsets (like "peach") are conceived as concepts denoting classes with numerous instances, while proper names (like "Berlin") denote instances of synset classes (in the case of "Berlin" the class synset containing "city"). This perspective seems to be captured more adequately by the class model than by the instance model.

2) Domain ontologies represent taxonomies of technical terms that mirror concept hierarchies established in the field. In the domain-specific word net TN, for instance, there is a set of the technical terms referring to hyperlink types (one-to-one-links, one-to-many-links, many-to-many-links; internal and external links etc.). These terms are defined to form a taxonomy, i.e. a subordinate term like "external link" denotes a class of instances that is included in the class of instances denoted by the superordinate term "link". In a given classification system subordinate terms inherit the features of their superordinate class; this clearly suggests to use the **`<rdfs:subclassOf>`** property and to benefit from its mechanism of feature inheritance. For this reason, we represent technical terms in the TN OWL model as classes which are related by taxonomic relations, i.e. we followed the class model. This design decision has a second advantage: Terms on the same hierarchical level tend to form groups of mutually disjoint concepts. For example, links may be subclassified in internal and external links depending on the position of their target anchor. On the other hand, links may further be subdivided in one-to-one-links, one-to-many-links, many-to-many-links according to the number of anchors involved. The sublasses with the same classification features are mutually disjoint, i.e. an instance of the class *tn:Link* may be simul-

[9] http://www.racer-systems.com.

[10] http://owl.man.ac.uk/factplusplus/

[11] Examples from Miller/Hristea (2006,3).

taneously monodirectional and external, but it cannot be simultaneously be monodirectional and bidirectional. We represent this in our OWL model using the predefined `<owl:disjointWith>` property. Since this property can only be defined for disjoint classes, the instance model cannot capture such restrictions. All in all, the class model seems to be more appropriate to capture domain-specific terminology in OWL than the instance model.

If one chooses the class model for a domain ontology one still may follow the instance model when representing the general vocabulary in PWN or GN. In our approach described in Kunze et al (to appear) we related a subset of TN technical terms with a subset of GN synsets. We defined the three PlugIn relations (cf. Magnini/Speranza, 2002), *attachedToNearSynonym, attachedToGeneralConcept, attachedToHolonym* with domain *tn:Term* and range *gn:synset*. To plug the class *tn:Term_Link* into its corresponding synset *gn:Link*, for example, it is declared to be subclass of a local restriction that assigns every individual of the class *tn:Term_Link* the individual *gn:Link* as the value on the property *plg:attachedToNearSynonym*, using the `<owl:hasValue>` construction.[12]

```
<owl:Class rdf:ID="tn:Term_Link">
  <rdfs:subClassOf rdf:resource="#tn:NounTerm"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:hasValue rdf:resource="#gn:Link"/>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="plg:attachedToNearSynonym"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Note that since in OWL DL, classes cannot be specified as values of properties, no meaningful `<owl:inverseOf>` relations can be declared for the PlugIn relations.

What about developing a class model for GN in OWL just as a class model was used for TN? We designed such an alternative OWL version, again of a section of GermaNet containing hypertext-related lexemes. When modelling the concecptual relations between synsets, our first preference would have been to relate classes pairwise as relation instances of a conceptual relation like *gn:isHyponymOf*. However, when classes are assigned as values of properties, they must function as individuals at the same time, which goes beyond the scope of OWL DL (cf. Smith et al., 2004). Thus, we decided to relate classes with one another by employing local property restrictions using the `<owl:allValuesFrom>` construction such as in the following example, where the synset containing *Webdokument* is declared to be a hyponym of the synset containing *Hypertextsystem*.

[12] cf. Kunze et al. (to appear).

```
<owl:Class rdf:about="#gn:Synset_Webdokument">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom>
        <owl:Class rdf:about="#gn:Synset_Hypertextsystem"/>
      </owl:allValuesFrom>
      <owl:onProperty>
        <owl:TransitiveProperty rdf:about="#gn:isHyponymOf"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:class>
```

When a synset has more than one hypernym, we declare the `<owl:allValuesFrom>` restriction such that all values have to be taken from a *union* of classes:

```
<owl:allValuesFrom>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#gn:Synset_Dokument"/>
      <owl:Class rdf:about="#gn:Synset_Hypertextsystem"/>
    </owl:unionOf>
  </owl:Class>
</owl:allValuesFrom>
```

It is now tempting to model lexical units as the instances of the synsets. However, above we have indicated that we want the instances of synsets to include individuals of a discourse model such as *Berlin* (i.e. named entities), or *Horse_351*. Lexical units do not represent semantic units but linguistic expression types, thus it is adequate to model lexical units as classes, too. Their instances should represent the tokens (occurrences) of lexical units in a text such as `<LUnit_horse>the horse that followed</LUnit_horse>`. Lexical relations are thus be encoded exactly like conceptual relations in OWL, i.e. in declaring `<owl:allValuesFrom>`-restrictions over those properties that represent the lexical relations. For declaring antonymy between the lexical units `<gn:LUnit_Vene>` and `<gn:LUnit_Arterie>`, for example, the following OWL code is used:

```
<owl:Class rdf:about="#gn:LUnit_Vene">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom>
        <owl:Class rdf:about="#gn:LUnit_Arterie"/>
      </owl:allValuesFrom>
      <owl:onProperty>
        <owl:TransitiveProperty rdf:about="#gn:isAntonymOf"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:class>
```

Using the triple store SWI Prolog Semantic Web library[13] in combination with Thea OWL library for Prolog (Vassiliadis 2006), the coding a query predicates that

[13] http://www.swi-prolog.org/

query for the set of direct or transitive hyponyms or hypernyms is as straightforward as it is with the instance model of GN. However, comparative evaluations of the performance of the instance model vs. the class model can be delivered only when the all of GermaNet has been automatically converted in both versions.

5. References

Beißwenger,Michael; Storrer,Angelika; Runte, Maren (2003): Modellierung eines Terminologienetzes für das automatische Linking auf der Grundlage von WordNet. In: Kunze et al. (2003). pp 113-125.

Ciorăscu, Claudia.; Ciorăscu, Iulian; Stoffel, Kilian (2003). KnOWLer – ontological support for information retrieval systems. In: *Proceedings of the 26th Annual International ACM-SIGIR Conference, Workshop on Semantic Web*, Toronto, Canada.

Fellbaum, Christiane (ed.) (1998): *WORDNET: an electronic lexical database*. London.

Erdmann, Michael (2001): *Ontologien zur konzeptuellen Modellierung der Semantik von XML.* Karlsruhe, Books on demand.

Farrar, Scott (to appear): Using ‚Ontolinguistics' for language description. In: Schalley, Andrea C.; Zaefferer, Dietmar (eds..): *Ontolinguistics*. Berlin, Mouton de Gruyter.

Hirst, Graeme (2004): Ontology and the lexicon. In: Staab, Steffen; Studer, Rudi (eds.) (2004): *Handbook on Ontologies*. Springer, pp. 209-229.

Kunze, Claudia (2001): Lexikalisch-semantische Wortnetze. In: Carstensen, Kai-Uwe et al.. (eds.): *Computerlinguistik und Sprachtechnologie: Eine Einführung*. Spektrum Verlag, Heidelberg, pp. 386-393

Kunze, Claudia; Lemnitzer, Lothar; Lüngen, Harald; Storrer, Angelika (2006, to appear): Modellierung und Integration von Wortnetzen und Domänenontologien in OWL am Beispiel von GermaNet und TermNet. In: *Proceedings of Konvens 2006*. Konstanz.

Kunze, Claudia; Lemnitzer, Lothar; Wagner, Andreas (2003): *Anwendungen des deutschen Wortnetzes in Theorie und Praxis.* Sonderheft der Zeitschrift für Computerlinguistik und Sprachtechnologie, LDV-Forum, 19 (1/2).

Magnini, Bernardo, Speranza, Manuela (2002): Merging Global and Specialized Linguistic Ontologies". In: *Proceedings of Ontolex 2002*. Las Palmas de Gran Canaraia, Spain. pp. 34-48.

Miller, George A.; Hristea, Florentina (2006): Word Net Nouns: Classes and Instances. In: *Computational Linguistics*, 32 (1), pp. 1-3

Smith, Michael K.; Welty, Chris; McGuiness, Deborah L. (eds.) (2004): *OWL Web Ontology Language Guide*. W3C recommendation, http://www.w3.org/TR/2004/REC-owl-guide-20040210.

Sowa, John F. (2001): *Glossary*. Online: http://www.jfsowa.com/ontology/gloss.htm (visited 12.7.2006)

van Assem, Mark; Gangemi, Aldo; Schreiber, Guus (eds., 2006a): *RDF/OWL representation of WordNet.* First Public Working Draft of 19 June 2006 produced by the Semantic Web Best Practices and Deployment Working Group. Online: http://www.w3.org/TR/2006/WD-wordnet-rdf-20060619 (visited 12.7.2006).

van Assem, Mark; Menken, Maarten R.; Schreiber, Guus; Wielemaker, Jan; Wielinga, Bob (2004): A method for converting thesauri to RDF/OWL. In: *Proceedings of the 3rd International Semantic Web Conference (ISWC 2004)*. Hiroshima, Japan. Volume 3298 of Lecture Notes in Computer Science.

Vassiliadis, Vangelis (2006) : *Thea. A web ontology language -OWL library for [SWI] Prolog.* Web-published manual, http://www.semanticweb.gr/TheaOWLLib/, visited 15.7.2006.