USING ONTOLOGIES FOR TERMINOLOGICAL KNOWLEDGE REPRESENTATION: A PRELIMINARY DISCUSSION[1]

RICARDO MAIRAL-USÓN
UNED

CARLOS PERIÑÁN-PASCUAL
Universidad Católica San Antonio

EVA SAMANIEGO-FERNÁNDEZ
UNED

ABSTRACT

Within the framework of FunGramKB, the aim of this paper is to offer a preliminary discussion of how terminological knowledge representation could be better formalized along the lines of a robust knowledge base grounded in a deep semantic approach. After presenting the general architecture of FunGramKB, we move on to discuss how its Core Ontology interacts with the various satellite ontologies. A further step is to present a tentative proposal on the methodology to develop these satellite ontologies.

Keywords**:** FunGramKB, terminology, knowledge representation, knowledge acquisition, ontology.

1. INTRODUCTION

When it comes to building terminological resources, we sustain as a methodological credo the fact that terminography (or else lexicography) should never be divorced from advances attained in terminology (or else lexicology). This signifies that, on the one hand, terminological resources should be grounded in theoretical linguistic models that can provide clues as for meaning representation and construction, while, on the other hand, terminological thesauri should be able to comply with the challenges of providing better means for crosslinguistic information retrieval. In connection with this — differences aside, a number of interesting projects have been developed with a view to offering a more comprehensive terminological description, e.g. Ontoterm, EcoLexicon, among many others.

Within this context, FunGramKB[2] is presented as a multifunctional and multilingual knowledge base that integrates a comprehensive model of knowledge representation, including ontological, lexical and even constructional knowledge, being the latter inspired on the Lexical Constructional Model (LCM).[3] Here are some of the most relevant methodological principles that have been discussed elsewhere:[4]

[1] Financial support for this research has been provided by the DGI, Spanish Ministry of Education and Science, grant FFI2008-05035-C02-01/FILO. The research has been co-financed through FEDER funds.
[2] www.fungramkb.com
[3] www.lexicom.es
[4] For a more detailed description of FunGramKB and the LCM, we refer the reader to papers such as Mairal-Usón and Periñán-Pascual (2009), Mairal and Ruiz de Mendoza (2009), Periñán-Pascual and Arcas-Túnez (2007, 2010a,b), Periñán-Pascual and Mairal (2009, 2010, fc), Ruiz de Mendoza and Mairal (2008), as well as the papers you can download from the two URLs stated in the previous footnotes.

a) Drawing on Velardi et al.'s (1991) distinction, FunGramKB follows a deep semantics approach in contrast to the surface semantic approach that defines other projects, e.g. WordNet (cf. Periñán-Pascual and Arcas-Túnez 2007).
b) Every concept in the knowledge base is provided with a number of properties: a meaning postulate and a thematic frame, among others. Consequently, a more fine-grained semantic description is attained if compared with other projects, e.g. SUMO.
c) A concept-oriented interlingua, COREL (COnceptual REpresentational Language), serves to describe the properties of the different modules in the cognitive level (cf. Periñán-Pascual and Mairal-Usón, 2010).
d) A conceptualist rather than a lexicalist approach is maintained. The conceptual level in general, and the ontology in particular, becomes the pivot for the linguistic modules. Consequently, lexical representations in the form of conceptual logical structures now become real language independent representations. A conceptualist approach opens thus the door to the ability to link primes as posited in conceptual logical structures with conceptual units in the ontology, therefore redundancy is minimized while informativeness is maximized (Mairal-Usón, Periñán-Pascual and Pérez Cabello de Alba, 2010).
e) A robust meaning construction model, the LCM, is part of the knowledge base. This model includes, as part of the semantic component, four types of constructional level, including configurations that would be regarded by other theorists as a matter of pragmatic implicature, illocutionary force or discourse structure (cf. Ruiz de Mendoza and Mairal, 2008):

    (i) Level 1 produces core grammar characterizations.
    (ii) Level 2 accounts for heavily conventionalized situation-based low-level meaning implications.
    (iii) Level 3 is concerned with conventionalized illocutionary meaning (situation-based high-level implications).
    (iv) Level 4 deals with very schematic discourse structures.

Within this context, the aim of this paper is to offer a preliminary discussion of the way terminological knowledge can be represented in FunGramKB. In connection with this, we will firstly draw a broad description of the architecture of FunGramKB, which includes a Core Ontology and is intended to comprise a number of different satellite ontologies.

2. THE ARCHITECTURE OF FUNGRAMKB

As shown elsewhere (Periñán-Pascual and Mairal-Usón, 2009, 2010, etc.), the architecture of our knowledge base comprises three major knowledge levels, consisting of several independent but interrelated modules:

Lexical level:

- The Lexicon stores morphosyntactic, pragmatic and collocational information about lexical units. For the actual format of these entries, we follow Role and Reference Grammar's universally based system of predicate decomposition (Mairal-Usón and Periñán-Pascual, 2009; Van Valin, 2005).

- The Morphicon helps our system to handle cases of inflectional morphology.

Grammatical level:

- The Grammaticon is composed of several constructicon modules which are inspired in the four-level distinctions of the LCM (cf. above).

Conceptual level:

- The Ontology is presented as a hierarchical catalogue of the concepts that a person has in mind when talking about everyday situations, that is, the repository of semantic knowledge. Every concept is provided with a thematic frame and a meaning postulate.
- The Cognicon stores procedural knowledge by means of conceptual macrostructures, i.e. script-like schemata in which a sequence of stereotypical actions is organised on the basis of temporal continuity, and more particularly on the basis of Allen's temporal model (Allen, 1983; Allen and Ferguson, 1994).
- The Onomasticon stores information about instances of entities and events, such as Einstein or the Mother's Day. This module stores two different types of schemata (i.e. snapshots and stories), since instances can be portrayed synchronically or diachronically.

The main consequence of this two-level design is that every lexical and grammatical module is language-dependent, while every cognitive module is shared by all languages involved in the knowledge base. Therefore, computational lexicographers must develop one Lexicon, one Morphicon and one Grammaticon for English, one Lexicon, one Morphicon and one Grammaticon for Spanish and so on, but knowledge engineers build just one Ontology, one Cognicon and one Onomasticon to process any language input conceptually.[5] Figure 1 illustrates this architecture of FunGramKB.

[5] For the issue of the actual scope of the term "universal" and the way anisomorphism should be dealt with in FunGramKB, we refer the reader to Periñán-Pascual and Mairal-Usón (fc).

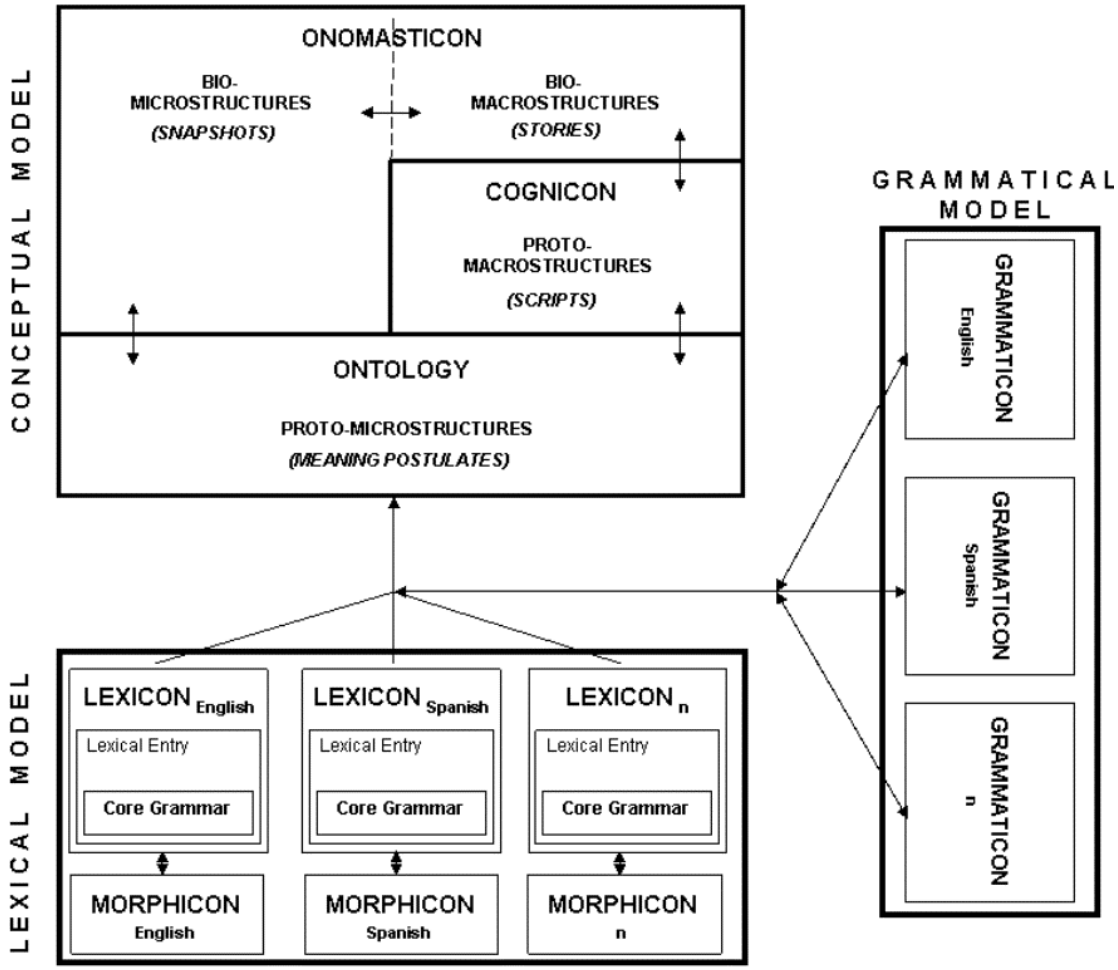

Figure 1. The architecture of FunGramKB.

The general-purpose ontology, or Core Ontology, can be enriched by linking different satellite ontologies that will represent specific terminological domains, i.e. law, finance, medicine, environment, etc.

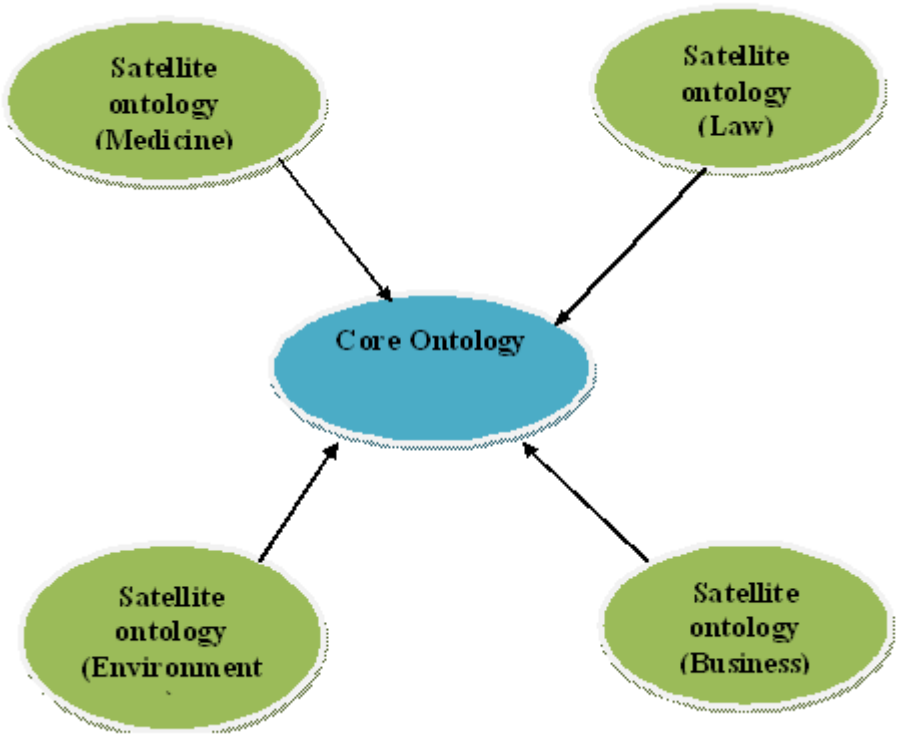

Figure 2. Core Ontology and satellite ontologies.

The FunGramKB Core Ontology, which is conceived as a conceptual IS-A taxonomy, allows multiple non-monotonic inheritance and distinguishes three different conceptual levels, each one of them with concepts of a different type: metaconcepts, basic concepts and terminals.

(i) Metaconcepts, preceded by the "#" symbol, constitute the upper level in the taxonomy and represent cognitive dimensions around which the rest of the conceptual units are organized. The analysis of the upper level in the main linguistic ontologies —SUMO, DOLCE, GUM, Mikrokosmos, SIMPLE etc.— led to a metaconceptual model whose design contributes to the integration and exchange of information with other ontologies, providing thus standardization and uniformity. Some metaconcepts are #ABSTRACT, #MOTION and #TEMPORAL. The result amounts to 42 metaconcepts distributed in three subontologies: #ENTITY, #EVENT and #QUALITY.

(ii) Basic concepts, preceded by the "+" symbol, constitute the intermediate level of the Ontology. These are used in FunGramKB as defining units which enable the construction of meaning postulates for basic concepts and terminals, as well as taking part as selectional preferences in thematic frames.

(iii) Terminal concepts, preceded by the "$" symbol, represent the final nodes in the conceptual hierarchy and lack definitory potential to take part in FunGramKB meaning postulates. Examples of terminal concepts are $ADAPT_00, $FLUCTUATE_00 and $SKYSCRAPER_00.

The next section outlines a proposal for the development of satellite ontologies.

3. DEVELOPING A FUNGRAMKB SATELLITE ONTOLOGY: AN INCIPIENT PROPOSAL

Ontology development can become a tedious and time-consuming task if performed from scratch. Automatic knowledge acquisition, and more particularly ontology learning from texts, seems not to have reached research maturity, despite the relative success of projects such as OntoLearn (Velardi et alii 2005) or Text2Onto (Cimiano and Völker 2005). Nowadays, the most reliable paradigm for ontology learning often makes use of semi-automatic methods with human intervention (Cimiano et alii 2009). The ontology developer's workload varies depending on the expected output from the ontology learning application, as illustrated in the well-known Ontology Learning Layer Cake (Buitelaar et alii 2005).

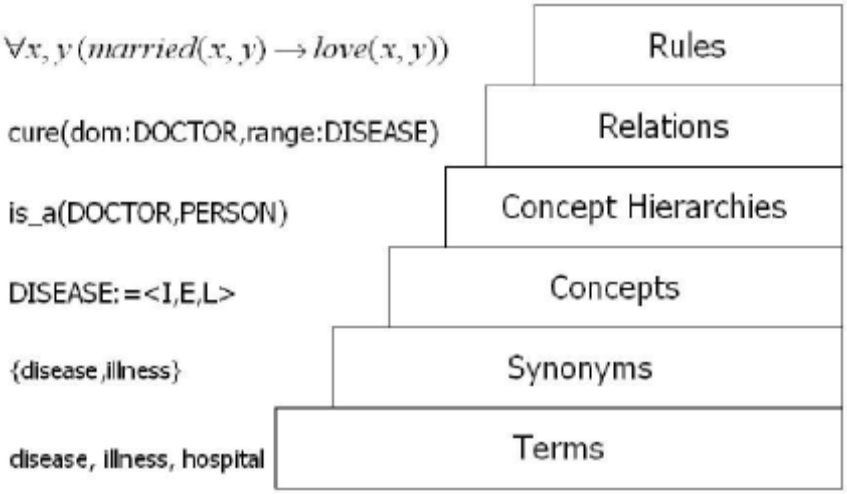

Figure 3. The Ontology Learning Layer Cake.

The length of these layers is usually interpreted as the number of currently-available computer applications which are able to extract that output. However, these layers can also be interpreted as the amount of work to be done by ontology developers. Although not dismissing the possibility of adopting a model based upon lexico-syntactic patterns to detect hyponymy relations for taxonomy induction (i.e. the "Concept Hierarchies" layer), in this section we sketch out a less ambitious methodology for FunGramKB satellite ontology developers, who are usually linguistics-trained researchers. Since many different tools can perform the various tasks comprising our ontology development process,[6] we encourage the reader to preferably focus on the description of the tasks and on the degree of automatization in the processes.

3.1. Bootstrapping specialized corpora from the web

One of the first tasks consists in selecting a corpus from which a terminological repository can be extracted for a given domain. If such an ontology cannot be found, automatic web-based corpus construction will be required. In this case, we can use the BootCaT toolkit — Bootstrapping Corpora and Terms (Baroni and Bernardini. 2004;

[6] Indeed, in this paper, we have opted for freely-available and user-friendly software to illustrate each processing task.

Baroni and Ueyama. 2004),[7] a text-mining tool to automate corpus building through Google and/or Yahoo APIs.

This bootstrapping process is quite interactive, as shown in the following steps:[8]

(i) You provide a set of seed terms which are randomly combined in order to build many tuples used as query strings. The more specialized we want our corpus to be, the more seed terms should be introduced.
(ii) The system itself extracts new terms as seeds from the ongoing document repository to increase the size of the corpus.
(iii) If the resulting corpus is not sufficiently specialized and/or large, the developer can generate a list of keywords with a corpus concondancer, e.g. AntConc (Anthony 2004).[9] Thus, the final output can then be improved by applying again step (i) on the basis of these new keywords.

The final step in this task consists in post-processing the corpus, i.e. cleaning up the plain text file by removing non-alphanumeric characters, multiple whitespaces, URLs, email addresses, etc.

3.2. Extracting named entities from the corpus

Before extracting the domain-based terminology, named-entity recognition, which aims at the identification of proper names, can be applied to the specialized corpus. For instance, the corpus-based named-entity recognition task can be fully automatic through the system ANNIE (A Nearly-New Information Extraction System) in GATE, a computer infrastructure to develop software which can perform NLP tasks.[10] In fact, ANNIE is a controller managing the pipeline of NLP processing components (e.g. tokeniser, lemmatiser, sentence splitter, and so on) which will be run over the corpus. In the end, the system is able to identify named entities from the document collection, as well as assigning their corresponding semantic type, e.g. Person, Location, Organization, etc (Cunningham 2010).

Unluckily, named-entity extractors are not usually topic oriented, so we should sift out those entities closely linked to our domain from those entities which are commonplace. To illustrate, suppose that our corpus has been specialized, or at least we have tried it, in the topic of terrorism, and we have obtained named-entities such as al-Qaida, Hamas, Pakistan or USA, among many others. Instances such as al-Qaida or Hamas clearly fall into this topic, but Pakistan or USA don't.

Finally, the output of this discrimination process is exported to the FunGramKB Onomasticon, where named entities are stored together with their domain. Knowledge representations for these entities in the form of bio-structures (e.g. snapshots or stories) will be automatically constructed by means of mapping rules to the DBpedia knowledge base (Bizer et alii 2009).[11]

3.3. Extracting domain-based terms from the corpus

[7] The tool is available from http://bootcat.sslmit.unibo.it
[8] Sharoff (2006) provides an accurate account on the procedure of construction of a large corpus in which BootCaT is used. The results of his experiment can be browsed in http://corpus.leeds.ac.uk/internet.html.
[9] The tool is available from http://www.antlab.sci.waseda.ac.jp/antconc_index.html.
[10] More precisely, GATE is defined as a "software architecture for language engineering" (Cunningham 2000). This application is available from http://gate.ac.uk.
[11] The DBpedia project is intended to extract structured information from Wikipedia, turn this information into a rich knowledge base, which currently describes more than 2.6 million entities, and make this knowledge base accessible on the Web. The population process of the FunGramKB Onomasticon from the DBpedia knowledge base is briefly described in Periñán-Pascual and Arcas-Túnez (2010b).

Using a text-mining tool, such as TermExtractor (Sclano and Velardi 2007), allows us to parse the corpus, and automatically extract a list of "syntactically plausible" terms (e.g. compounds). Some membership criteria should be applied on the basis of both (i) topicality and (ii) speciality:

(i) As with the named entities, we discriminate those terms belonging to the given topic.
(ii) We also decide if each term is sufficiently specialized so as to belong to the satellite ontology.[12]

3.4. Discovering the IS-A taxonomy

As far as the construction of the ontological taxonomy is concerned, we can work in two different scenarios: (a) reusing an existing ontology, or (b) building our own ontology. Scenario (a) is highly recommendable, since it is easier to populate an existing ontology with terms from our terminological repository than constructing a wholly new ontology from scratch. Indeed, if we find a high-quality fine-grained ontology for our domain, it is pointless to perform all previous processing tasks (i.e. sections 3.1 to 3.3). In that respect, OntoSelect (Buitelaar 2004)[13] can become a very useful resource, where a meaningfully-organized collection of over 850 ontologies covers a wide range of topics.[14] In fact, one of the strengths of this ontology repository lies in its constant updating, since it crawls continuously the web for any newly published ontology.

To a greater or lesser extent, and regardless of the scenario, the ontology developer would have to populate the ontology with concepts from the list of terms, as well as checking the taxonomic relation.[15] These tasks will be performed by means of an ontology editor, being our choice the NeOn Toolkit (Haase et alii, 2008),[16] an ontology engineering environment which provides many plug-ins supporting for the whole ontology life-cycle. In other words, the functionality of the NeOn Toolkit can be easily extended by integrating many different modules. For example, with regard to ontology reuse, the Watson plug-in (d'Aquin et alii 2007) allows the ontology developer to access external information from other ontologies for a given selected entity. Therefore, the NeOn Toolkit is able to guide the ontology developer's decisions by suggesting potential superordinates for every concept in the ongoing ontology. Thus, we conclude that conceptual hierarchization becomes a computer-aided ontology engineering task, where most of the workload lies on the human side but the computer helps us to model our ontological decisions.

Finally, this stage outputs an OWL ontology taking the form of an IS-A taxonomy, where classes are also labelled with a "definition" attribute. It is noteworthy to mention that, in order to export the ontology to the FunGramKB framework, a lexico-conceptual mapping should have occurred along this stage, since terms should be converted into concepts in order to make the resulting ontology compatible with the FunGramKB conceptualist approach to language. In fact, our initial aim is to apply a similar methodology to the one used in the construction of the basic conceptual level in the FunGramKB Core Ontology.[17]

---

[12] For example, those terms included in language-learning dictionaries will be conceptually dealt by the FunGramKB Core Ontology.
[13] The ontology library can be browsed in http://olp.dfki.de/ontoselect.
[14] In OntoSelect, ontologies are published in RDF, DAML and OWL formats.
[15] Subsumption is the only taxonomic relation permitted in the FunGramKB Ontology (Periñán-Pascual and Arcas-Túnez 2010a).
[16] The tool is available from http://neon-toolkit.org/wiki.
[17] For an in-depth account of what we have termed COHERENT methodology, we refer the reader to Periñán-Pascual and Mairal-Usón (fc).

5. Providing conceptual specifications to ontological units

Once the IS-A taxonomy is stored in the FunGramKB satellite ontology, developers should construct the meaning postulate and other conceptual properties for every newly-included concept. This is indeed a computer-aided task, since the FunGramKB Ontology editor is able to check well-formedness and consistency of the input.

CONCLUSIONS

This paper has offered a very preliminary discussion of how terminological knowledge can be better represented within the framework of FunGramKB knowledge base. In connection with this, we have outlined the architecture of FunGramKB and have argued for the development of satellite ontologies that work in close collaboration with the Core Ontology. Moreover, we have shown the steps for a possible methodology for the development of satellite ontologies. In sum, we maintain that such an ontological approach to terminological knowledge offers a sound framework for natural language processing applications. i.e. crosslinguistic information retrieval.

REFERENCES

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM* 26 (11), 832-843.

Allen, J. F. & Ferguson, G. (1994). Actions and events in interval temporal logic. *Journal of Logic and Computation* 4 (5), 531-579.

Anthony, L. (2004). AntConc: A Learner and Classroom Friendly, Multi-Platform Corpus Analysis Toolkit. *Proceedings of IWLeL 2004: An Interactive Workshop on Language e-Learning*, pp. 7-13.

Baroni, M. & Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the web. *Proceedings of LREC 2004.*

Baroni, M. & Ueyama, M. (2004). Retrieving Japanese specialized terms and corpora from the World Wide Web. Proceedings of KONVENS 2004.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R. & Hellmann, S. (2009). DBpedia: a crystallization point for the Web of Data. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 7, pp. 154-165.

Buitelaar, P., Cimiano, P. & Magnini, B. (2005). *Ontology Learning from Text: An Overview* In P. Buitelaar, P. Cimiano & B. Magnini (eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*. Frontiers in Artificial Intelligence and Applications Series, Vol. 123, IOS Press.

Buitelaar, P., Eigner, Th., & Declerck, Th. (2004) OntoSelect: A Dynamic Ontology Library with Support for Ontology Selection. *Proceedings of the Demo Session at the International Semantic Web Conference*, Hiroshima.

Cimiano. P., Mädche, A., Staab, S. & Völker, J. (2009) "Ontology Learning", In *Handbook of Ontologies* (pp. 245-267). Springer Verlag.

Cimiano, P. & Völker, J. (2005) Text2Onto - a framework for ontology learning and data-driven change discovery. *Proceedings of NLDB 2005,*. Lecture Notes in Computer Science, vol. 3513, Springer, Alicante, pp. 227-238.

Cunningham, H. (2000) *Software Architecture for Language Engineering*. Unpublished PhD thesis. University of Sheffield.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Ursu, C., Dimitrov, M., Dowman, M., Aswani, N., Roberts, I., Li, Y., Rin, A.S.& Funk, A. (2010) Developing

Language Processing Components with GATE, from http://gate.ac.uk/sale/tao/tao.pdf

d'Aquin, M., Sabou, M., Dzbor, M., Baldassarre, C.L., Gridinoc, L., Angeletou, S. &. M. E. Watson (2007). A Gateway for the Semantic Web. Poster session of the European Semantic Web Conference, *ESWC* 2007.

Goldberg, A.E. (2006). *Constructions at Work: The Nature of Generalization in Language*. New York: Oxford University Press.

Haase P., Lewen, H., Studer, R. & d'Aquin, M. (2008) The NeOn Ontology Engineering Toolkit. *Demo, WWW2008 - WWW 2008: 17th International World Wide Web Conference*, Beijing, China.

Mairal Usón, R. & Ruiz de Mendoza, F. (2009). Levels of description and explanation in meaning construction. In Ch. Butler & J. Martín Arista (eds.) *Deconstructing Constructions* (pp. 153 – 198). Amsterdam/ Philadelphia: John Benjamins,

Mairal Usón, R. & Periñán-Pascual, C. (2009). The anatomy of the lexicon component within the framework of a conceptual knowledge base. *Revista Española de Lingüística Aplicada* 22 (2009), 217-244.

Mairal Usón, R., Periñán-Pascual, C. & Pérez Cabello de Alba, M.B. (2010). La noción de estructura lógica conceptual. In R. Mairal, L. Guerrero and C. González (fc) .) *El funcionalismo en la teoría lingüística. La Gramática del Papel y la Referencia. Introducción, avances y aplicaciones*. Akal: Madrid (volume in preparation).

Periñán-Pascual, C. & Arcas-Túnez, F. (2005). Microconceptual-Knowledge Spreading in FunGramKB. *Proceedings on the 9th IASTED International Conference on Artificial Intelligence and Soft Computing*. (pp. 239-244). Anaheim-Calgary-Zurich: ACTA Press.

Periñán-Pascual, C. & Arcas-Túnez, F. (2007) Deep semantics in an NLP knowledge base. *Proceedings of the 12th Conference of the Spanish Association for Artificial Intelligence*, Universidad de Salamanca, Salamanca, pp. 279-288.

Periñán-Pascual, C. & Arcas-Túnez, F. (2010a). Ontological Commitments in FungramKB. *Procesamiento del Lenguaje Natural* 44. pp. 27-34

Periñán-Pascual, C. & Arcas-Túnez, F. (2010b). The architecture of FunGramKB. *Proceedings of the Seventh International Conference on Language Resources and Evaluation,* European Language Resources Association (ELRA), pp.2667-2674

Periñán-Pascual, C. & Mairal Usón, R. (2009). Bringing Role and Reference Grammar to natural language understanding. *Procesamiento del Lenguaje Natural*, vol. 43, 265-273.

Periñán-Pascual, C. & Mairal Usón, R. (2010). La Gramática de COREL: un lenguaje de representación conceptual. *Onomazein*. 21 (2010/1), 11-45.

Periñán-Pascual, C & Mairal-Usón, R. (fc). Constructing the FunGramKB basic conceptual level: the COHERENT methodology.

Periñán-Pascual, C & Mairal-Usón, R. (fc). Prototipicality, universality and culturality in an NLP knowledge base.

Ruiz de Mendoza Ibáñez, F. J. & Mairal Usón, R. (2008). Levels of description and constraining factors in meaning construction: an introduction to the Lexical Constructional Model. *Folia Linguistica* 42/2, 355-400.

Sclano, F. & Velardi, P. (2007) TermExtractor: a Web Application to Learn the Common Terminology of Interest Groups and Research Communities . *9th Conference on Terminology and Artificial Intelligence TIA 2007*, Sophia Antinopolis.

Sharoff, S. (2006) Creating general-purpose corpora using automated search engine queries. In M. Baroni and S. Bernardini, (eds), *WaCky! Working papers on the Web as Corpus*.

Van Valin, R. D. Jr. (2005). *Exploring the Syntax-Semantics Interface*. Cambridge: Cambridge University Press.

Velardi, P., Navigli, R., Cuchiarelli, A. & Neri, F. (2005) Evaluation of Ontolearn, a Methodology for Automatic Population of Domain Ontologies. In Buitelaar, P., Cimiano, P., Magnini, B. (eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press: Amsterdam.

Velardi, P., Pazienza, M.T. & Fasolo, M. (1991). How to encode semantic knowledge: a method for meaning representation and computer-aided acquisition. *Computational Linguistics* 17/2, 153-170.