A CONCEPTUAL LEXICAL DATABASE IN A NATURAL LANGUAGE PROCESSING SYSTEM

Francisco Arcas Túnez[1], Carlos Periñán Pascual[2]

[1] Departamento de Informática de Sistemas (Phone: 968278898)
[2] Unidad Central de Idiomas (Phone: 968278819)
Universidad Católica de San Antonio, Campus de Los Jerónimos,
30107 Guadalupe, Murcia, Spain
{farcas, jcperinan}@pdi.ucam.edu

Abstract. We present a conceptual lexical database for its application in a natural language processing system. On one hand, this lexical component is more powerful than a simple lexical database; the machine interrelates the senses in the lexicon through a conceptual macrostructure called "semantic density graph". On the other hand, the design of this lexicon is less complex than a standard lexical knowledge base, because it is not necessary to build a hierarchical ontology model beforehand, define a restricted set of semantic primitives as ontological concepts or specify the different mappings between the lexicon and the ontology.

Keywords: NLP, meaning representation, computational lexicography, DAML

Topic: Procesamiento del Lenguaje Natural

Paper Track

A CONCEPTUAL LEXICAL DATABASE IN A NATURAL LANGUAGE PROCESSING SYSTEM

Francisco Arcas Túnez[1], Carlos Periñán Pascual[2]

[1] Departamento de Informática de Sistemas
[2] Unidad Central de Idiomas
Universidad Católica de San Antonio, Campus de Los Jerónimos,
30107 Guadalupe, Murcia, Spain
{farcas, jcperinan}@pdi.ucam.edu

Abstract. We present a conceptual lexical database for its application in a natural language processing system. On one hand, this lexical component is more powerful than a simple lexical database; the machine interrelates the senses in the lexicon through a conceptual macrostructure called "semantic density graph". On the other hand, the design of this lexicon is less complex than a standard lexical knowledge base, because it is not necessary to build a hierarchical ontology model beforehand, define a restricted set of semantic primitives as ontological concepts or specify the different mappings between the lexicon and the ontology.

1 Introduction

In natural language processing (NLP), engineers usually design lexical knowledge bases which allow the mapping from the words in the lexicon, and their syntactico-semantic information, to ontological concepts. The design of this kind of ontologies (e.g. Mikrokosmos[1] or EDR[2]) requires a shared conceptualisation of the world from a particular domain. In these cases, it is necessary to specify a set of entities, attributes and events, as well as their possible interrelations. The representation of this conceptualisation is usually explicit.

We suggest, however, an implicit conceptualisation of the world by using mainly the interaction between the predicate frame and the meaning postulate, resulting in a conceptual framework for knowledge modelling. We intend to provide an NLP system with an enriched lexical database that could allow the machine to build automatically a conceptual macrostructure from the lexicon; consequently, it is not necessary to specify the two well-defined knowledge sources which are typically integrated in most NLP projects: the lexicon and the ontology.

2 S. C. Dik's Functional Grammar

Our lexicon is based on S. C. Dik's Functional Grammar [1] [2] [3] as a model for the formal description of the English language. The lexicon of this

[1] http://crl.nmsu.edu/overviewweb/Kresources/mirkokos.htm
[2] http://www.iijnet.or.jp/edr/

grammar is presented as a highly-structured core component which codifies morphological, syntactic, semantic and pragmatic information. The lexicon is the module of linguistic description, containing lexical items (nouns, verbs and adjectives) with their associated predicate frames and meaning postulates. On one hand, the predicate frame is a structural scheme with the most important syntactico-semantic properties of the headword; on the other hand, the meaning postulate is a formal representation of the word sense. However, and with the aim of turning the lexicon into a lexical knowledge base, it is enriched with a more descriptive meaning postulate. We propose a lexicon which is not only a repository of syntactico-semantic properties but also a component which traces the different lexical relations established in the language from the conceptual units of the meaning postulate.

3 The Meaning Postulate

3.1 Word Sense Representation

S. C. Dik's Functional Grammar uses a relational approach for word sense representation; that is, no abstract metalanguage is used, such as semantic primitives, but lexical items of the language itself. In our case, the meaning postulate is conceived as a lexicographical definition in a machine-tractable format, being manually built by taking into account definition texts in standard dictionaries, e.g. the Collins COBUILD English Language Dictionary [4]. Dictionaries do not only give word sense information, but they also represent common sense knowledge of the world from the lexicographers' point of view. Some language engineers present the problem that lexicographers assume the reader's linguistic competence when writing the dictionary, so lexical entries have just the information which makes readers connect it with their general linguistic knowledge. The solution lies in collecting the information of a word sense which is not found in its lexical entry, but in the lexical entries of other words of the lexicon. Thus, dictionaries can build a huge semantic network through their definition texts [5].

There are many lexicons, e.g. EuroWordNet [6], which adopt another kind of "relational" approach, where the meaning of a word is described by means of the lexico-semantic relations (e.g. hyperonyms, synonyms, troponyms, etc) that are established with other words. In these cases, there is no real word sense representation, since no particular formalism is used.

3.2 Taxonomies and Ontologies

Genus and *differentiae* are clearly separated within our meaning postulate; the *genus* takes part in the automatic design of lexical taxonomies and the *differentiae* contributes to the construction of ontological structures.

In NLP systems, a taxonomy is a lexical hierarchy whose units are described in terms of lexico-semantic relations. With the advent of machine-readable dictionaries, taxonomies are generally used to create semantic networks within the lexicon, where each node represents a *genus* of the *definiens* in at least one definition text. The purpose of these taxonomies is maximal reduction of information redundancy. Since Amsler and White's work [7], and including LINKS [8], many research projects have managed to build coherent, but limited, noun taxonomies. The

problem is that lexical taxonomies do not provide the machine with a deep knowledge on the structuring of the world, which is necessary for a better understanding of linguistic expressions. This problem is solved with the ontology, which is presented as a hierarchical structure whose units are expressed in terms of concepts describing the relations among entities of the world. If we want to understand the usage of this term in the field of language engineering, we should explain the interaction that the system establishes with an ontology, a lexicon and the meaning representation of a text. A NLP system must be able to "understand" the input. With this purpose, it is necessary to shape the output of the semantic analysis into a set of well-formed structures, in which terms are presented as ontological concepts. In this process of the analysis, the system matches lexical units in the input to ontological concepts. Consequently, the lexicon should not only have information concerning the morphosyntactic properties of the lexical unit, but also semantic features in the form of mappings to ontological concepts. In fact, the ontology and the lexicon are usually developed in a parallel way during the process of construction of a knowledge-based system.

A formal ontology is a valuable resource to help the machine represent a word sense. However, the construction of an ontological model of the world implies a great deal of effort. Apart from the fact that it is a time-consuming task, there is no agreement about its size and composition [9]. NLP engineers do not have a clearly-defined methodology for an ontology design, being all this work based on the researchers' experience [10]. Furthermore, the most important problem is found in the own nature of these ontological concepts. They are generally represented by semantic primitives, so it raises the problem of how to express different shades of meaning through a limited set of these primitives. All these factors led us to consider an alternative for the construction of a lexical knowledge base without the explicit design of a hierarchical ontology.

3.3 The Semantic Density Graph

Like some other projects of computational lexicography, our lexicon builds automatically a concept-oriented network, which takes the form of a huge directed graph whose nodes correspond to senses interconnected by means of semantically-labelled arcs. In this respect, the innovation of our approach lies in how this network is developed. Predicate frames and meaning postulates present such a highly-structured formalism that we can apply a more meaningful methodology in order to trace conceptual microstructures out of the lexicon; in this way, we avoid the simplistic strategy of relating two words semantically because they just share a word in their definition texts. As far as the methodology is concerned, the development of our conceptual network is taken as an extension of the semantic density graph (SDG) associated to every headword or "core predicate". The concept of SDG comes in turn from the concept of semantic density list (SDL) [11]. The SDL was conceived of as a list of those lexical items in the meaning postulate which contribute meaningfully to the description of the headword sense. The SDG inherits the main features of the SDL, as well as interrelating all the member items through a directed graph. In the automatic development of the conceptual network, several phases of graph expansion are differentiated:

1- An SDG is created from the meaning postulate of the core predicate from which we want to build a conceptual microstructure.
2- This microstructure is expanded with the semantic preferences in the predicate frame of the core predicate.
3- The conceptual structure is enlarged with the SDG associated to every *genus* in the SDG of the core predicate.
4- Finally, the network is expanded with those SDGs in which at least one node and its corresponding semantic role match a node and the semantic role of one of its arcs in the graph in progress.

The resulting graph represents an efficient way of organizing information. This conceptual network presents the greatest amount of information in the most economical way, giving rise to lexical inference when navigating through it. It is based on a meaningful methodology because every level of expansion has a semantic motivation. In the second level, the integration of the meaning postulate and the predicate frame takes place; selection preferences are incorporated into a conceptual network because they help a lexical item be assigned to a particular concept [12]. The third level is inspired by Dik's principle of *stepwise lexical decomposition* [1] [2] [3], resulting in the construction of hierarchical lexical structures centred on the *genus* in definition texts.[3] In the fourth level, we show that lexical items should not only be interrelated through the *genus*, but also the components within the *differentiae*, establishing in this way other many conceptual relations.

4 DAML

Our conceptual lexical database is implemented in DAML[4] (DARPA Agent Markup Language) [15], an emerging language for knowledge representation supporting reasoning and inference on a semantic model. Appendix 1 shows the lexical entry of "blitz01" as an example to describe the information assigned to any lexical item: a headword, sense number, morphological features, translation equivalent, stylistic information, predicate frame and meaning postulate. On one hand, the predicate frame specifies the part of speech, syntactic pattern and alternations, and quantitative and qualitative valencies of the headword; on the other hand, the meaning postulate contains the *genus* and *differentiae* of the word sense.

DAML is an extension of RDF and RDF schemes, that is why some remarks about them should be made. RDF (Resource Description Framework) [16] is a programming language focused on metadata modelling in web resources. The main objective of RDF is to give interoperativity to applications exchanging metadata in any kind of context, such as resource descriptions, website maps, content statistics, e-mails or collaborative services.

RDF provides a semantic description of objects in a machine-readable format, developing rules that automate decision-making on web resources. A resource description is performed as a collection of properties associated to a specific data type

[3] This is the strategy adopted by WordNet [13] [14].
[4] http://www.daml.org

and a value; in knowledge representation, this corresponds to the standard attribute-value pair. However, RDF provides no mechanism for the declaration of properties, or the definition of relations with other resources. For this reason, we opted for RDF-Schemas (RDF-S) [17] in order to be able to represent classes (e.g. HeadWord, Relation, QualitativeValency…) and properties (morpho_v_01, causer, synonym…) of objects. These RDF-S also contain restrictions on classes and their relations, as well as detecting the cases where these restrictions fail.

RDF is based on XML (Extensible Markup Language) [18]. Consequently, it supports the use of tags to structure information, XML namespaces to identify the scheme where classes and properties are defined, and URI (Uniform Resource Identifier) to address and name any type of web resource. On one hand, RDF-S are very similar to XML Document Type Definitions and XML schema, although both of them give specific restrictions to the structure of an XML document, as we can find in the validation of an expression or value (syntactic rules). On the other hand, RDF-S provide information about a sentence interpretation in a data model (semantic rules).

Finally, DAML, a language based on RDF-S, is used to specify knowledge in a machine-readable way and build reasoning models supported by open web technologies. Figure 1 illustrates the interrelation of all the languages described in this section.

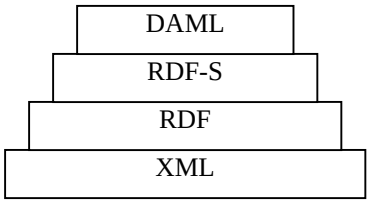


Figure 1

One of the possible applications of our conceptual lexical database is its implementation in a web service to be used by intelligent agents as a response to the users' queries. This is one of the reasons why we have opted for DAML, a language introduced by the World Wide Web Consortium[5] (W3C), who work in the integration of knowledge and web services through the use of semantic models or ontologies.

In the design and construction of lexicons, and as happens in other software projects, the concept of "reusability" takes relevant importance. In this respect, our conceptual lexical database is not only a lexical resource to be used in different fields of research, but it also allows the machine to treat web resources as a huge multimedia corpus, in which knowledge from texts can be used to enrich our own lexicon.

In many language engineering projects, the standard notation for the description of a lexical database is through a set of attribute-value pairs. Although many researches suggest the implementation of these matrices in the form of

[5] http://www.w3c.org

relational tables [20], others state that an object-oriented database is more appropriate for NLP [21]. The relational model presents some problems that can be solved with an object-oriented model [10]:

- Lack of inheritance mechanisms
- Impossibility of representing knowledge hierarchically
- Difficulty of managing non-atomic values

DAML can be easily implemented in an object-oriented database model.

5 UML

UML (Unified Modelling Language) is a general-purpose object-modelling technique used in the development of software projects. We have used UML as a tool which represents diagrammatically the structure of our conceptual lexical database, stating classes and their attributes, relations and restriction specifications (Appendix 2). There are several features of UML which make it a suitable tool for the description of knowledge:

- the semantics of the class diagram
- the use of easily-understandable graphical schemes
- the use of a non-restricted notation
- a high level of standardization in the academic and industrial worlds
- the support of CASE tools

Thanks to its wide acceptance, advanced state of its specification and the possibility of direct conversion to DAML, UML introduces a qualitative change in the development of knowledge management systems. Cranefield and Purvis [19] investigate the use of UML class diagrams to represent ontologies and knowledge representation. The UBOT[6] project (UML Based Ontology Tool-set) demonstrates formally the validity of DAML ontologies and their representation in UML diagrams. The CODIP[7] project (Components for Ontology-Driven Information Push) uses UML to build DAML ontologies for their application in military logistics.

6 Conclusion

In this paper, we have presented a conceptual lexical database for its application in a NLP system. On one hand, we are developing a lexical component more powerful than a simple lexical database, enabling the machine to interrelate the senses in the lexicon through the meaning postulates. On the other hand, the design of this lexicon is less complex than a standard lexical knowledge base, because it is not necessary to build a hierarchical ontology model beforehand, define a restricted set of

[6] http://ubot.lockheedmartin.com
[7] http://codip.grci.com

semantic primitives as ontological concepts or specify the different mappings between the lexicon and the ontology.

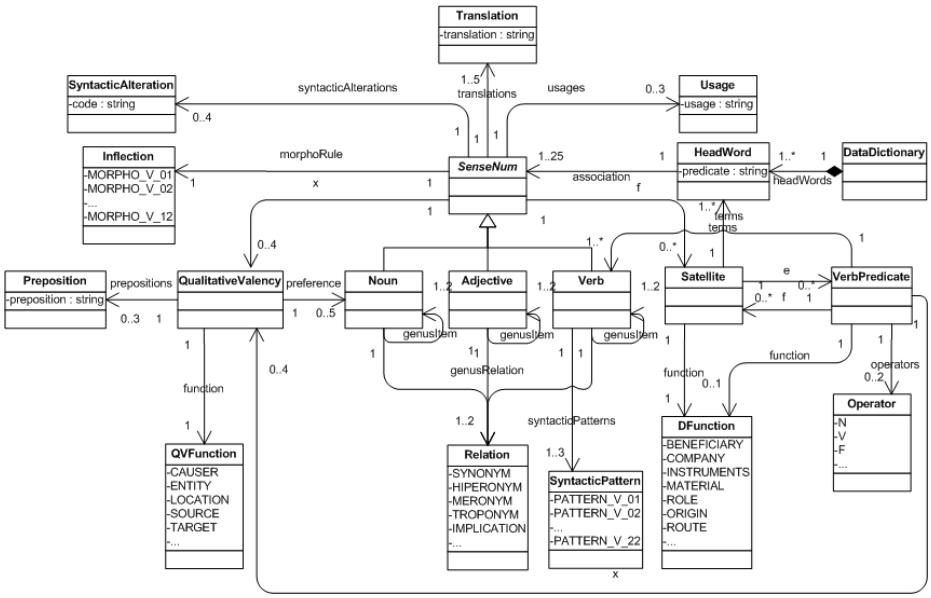
References

1. Dik, S.C.: Functional Grammar. Foris, Dordrecht (1978)
2. Dik, S.C.: The Theory of Functional Grammar. Part I: The Structure of the Clause. Foris, Dordrecht (1989)
3. Dik, S.C.: The Theory of Functional Grammar. Part I: The Structure of the Clause. Mouton de Gruyter, Berlin New York (1997)
4. Collins COBUILD English Language Dictionary. Collins, London (1995)
5. Meijs, W., Vossen, P. : In so many Words: Knowledge as a Lexical Phenomenon. In: Pustejovsky, J., Bergler, S. (eds.): Lexical Semantics and Knowledge Representation: First SIGLEX Workshop. Springer-Verlag, Berlin Heidelberg (1992) 137-153
6. Vossen, P.: Introduction to EuroWordNet. Computers and the Humanities 32, 2-3 (1998) 73-89
7. Amsler, R.A., White, J.S.: Development of a Computational Methodology for Deriving Natural Language Semantic Structures via Analysis of Machine-readable Dictionaries. Final report on NSF project MCS77-01315. University of Texas at Austin, Austin, Texas (1979)
8. Vossen, P.: The Structure of Lexical Knowledge as Envisaged in the LINKS-project. In: Connolly, J.H., Dik, S.C. (eds.): Functional Grammar and the Computer. Foris, Dordrecht (1989) 177-199
9. Onyshkevych, B.A., Nirenburg, S.: Lexicon, Ontology, and Text Meaning. In: Pustejovsky, J., Bergler, S. (eds.): Lexical Semantics and Knowledge Representation: First SIGLEX Workshop. Springer-Verlag, Berlin Heidelberg (1992) 289-303
10. Moreno, A.: Diseño e Implementación de un Lexicón Computacional para Lexicografía y Traducción Automática. Doctoral thesis. Universidad de Córdoba, Córdoba (1998)
11. Periñán, J.C.: Las Listas de Densidad Semántica y la Lexicografía Funcional Computacional, Cuadernos de Investigación Filológica, Logroño (2002)
12. Nirenburg, S., Levin, L.: Syntax-driven and Ontology-driven Lexical Semantics. In: Pustejovsky, J., Bergler, S. (eds.): Lexical Semantics and Knowledge Representation: First SIGLEX Workshop. Springer-Verlag, Berlin Heidelberg (1992) 5-20
13. Miller, G.A. : WordNet: a Lexical Database for English. Communications of the ACM 38, 11 (1995) 39-41
14. Fellbaum, C.: A Semantic Network of English: the Mother of all Wordnets. Computers and the Humanities 32, 2-3 (1998) 209-220
15. Hendler, J., McGuinness, D.: The DARPA Agent Markup Language. IEEE Intelligent Systems 15, 6 (2000) 67–73
16. Lassila, O., Swick R.: Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation (1999) http://www.w3.org/TR/REC-rdf-syntax
17. Brickley, D., Guha, R.: RDF Schema (RDF-S) Specification 1.0. W3C Recommendation (2000) https://www.w3.org/TR/rdf-schema
18. World Wide Web Consortium: Extensible Markup Language (XML) 1.0. W3C Recommendation (1998) http://www.w3.org/TR/REC-xml
19. Cranefield, S., Purvis, M.: UML as an Ontology Modeling Language. Proc. of the Workshop on Intelligent Information Integration, IJCAI-99. Stockholm (1999)
20. Evens, M., Dardaine, J., Huang, Y., Li, S., Markowitz, J., Rinaldo, F., Rinaldo, M., Strutz, R.: For the Lexicon that Has Everything. In: Pustejovsky, J., Bergler, S. (eds.): Lexical Semantics and Knowledge Representation: First SIGLEX Workshop. Springer-Verlag, Berlin Heidelberg (1992) 219-233
21. Ide, N., Le Maitre, J., Véronis, J. : Outline of a Model for Lexical Databases. In: Zampolli, A., Calzolari, N., Palmer, M. (eds.) (1994) Current Issues in Computational Linguistics: Essays in Honour of Don Walker, Kluwer Academic Publishers, Dordrecht (1994) 283-320

Appendix 1

```
<HeadWord rdf:ID="blitz">
   <predicate>
      blitz
   </predicate>
   <senses>
   <Verb rdf:ID="blitz01">
      <genusItem rdf:resource="attack01"/>
      <genusRelation rdf:resource="TROPONYM"/>
      <morphoRule rdf:resource="MORPHO_V_03"/>
      <syntacticPatterns rdf:resource="PATTERN_V_08"/>
      <usages>
            mil
      </usages>
      <translations>
         bombardear
      </translations>
      <x>
         <QualitativeValency rdf:ID="blitz_x01">
            <function rdf:resource="CAUSER"/>
         </QualitativeValency>
      </x>
      <x>
         <QualitativeValency rdf:ID="blitz_x02">
            <function rdf:resource="LOCATION"/>
            <preferences rdf:resource="city01"/>
            <preferences rdf:resource="building01"/>
         </QualitativeValency>
      </x>
      <f><Satellite rdf:ID="blitz_f01">
            <function rdf:resource="MANNER"/>
            <e>
               <VerbPredicate rdf:ID="blitz_e01">
                  <terms rdf:resource="drop01"/>
                  <x>
                     <QualitativeValency rdf:ID="blitz_x03">
                        <function rdf:resource="CAUSER"/>
                        <preferences rdf:resource="aircraft01"/>
                     </QualitativeValency>
                  </x>
                  <x>
                     <QualitativeValency rdf:ID="blitz_x04">
                        <function rdf:resource="ENTITY"/>
                        <preferences rdf:resource="bomb01"/>
                     </QualitativeValency>
                  </x>
                  <f><Satellite rdf:ID="blitz_f02">
                        <function rdf:resource="MANNER"/>
                        <terms rdf:resource="intense"/>
                     </Satellite>
                  </f>
               </VerbPredicate>
            </e>
         </Satellite>
      </f>
   </Verb>
   </senses>
</HeadWord>
```

Appendix 2


Translation
-translation : string
SyntacticAlteration
-code : string
syntacticAlterations
translations
usages
Usage
-usage : string
Inflection
-MORPHO_V_01
-MORPHO_V_02
-...
-MORPHO_V_12
morphoRule
SenseNum
association
HeadWord
-predicate : string
headWords
DataDictionary
terms
Preposition
-preposition : string
prepositions
QualitativeValency
preference
Noun
Adjective
Verb
Satellite
VerbPredicate
genusItem
genusRelation
function
operators
QVFunction
-CAUSER
-ENTITY
-LOCATION
-SOURCE
-TARGET
-...
Relation
-SYNONYM
-HIPERONYM
-MERONYM
-TROPONYM
-IMPLICATION
-...
syntacticPatterns
SyntacticPattern
-PATTERN_V_01
-PATTERN_V_02
-...
-PATTERN_V_22
DFunction
-BENEFICIARY
-COMPANY
-INSTRUMENTS
-MATERIAL
-ROLE
-ORIGIN
-ROUTE
-...
Operator
-N
-V
-F
-...