MANAGELEX and the Semantic Web

Monica Gavrila
University of Hamburg
Computer Science Department
Natural Language Systems Division
Vogt-Kölln Str. 30, D-22527
Hamburg, Germany
gavrila@nats.informatik.uni-hamburg.de

Cristina Vertan
University of Hamburg
Computer Science Department
Natural Language Systems Division
Vogt-Kölln Str. 30, D-22527
Hamburg, Germany
vertan@informatik.uni-hamburg.de

Abstract

This paper presents MANAGELEX, “a generic lexicon management tool” (Vertan and von Hahn, 2002) and its possible use for the Semantic Web, namely how ontologies and lexicons managed by the tool can be related. The paper is structured in six sections. The first section offers a general introduction on Semantic Web. The following three sections describe the MANAGELEX tool – structure and functionality - one of its models (LexMod), and one of its modules (StructTool). In the realization of LexMod, Semantic Web technologies (OWL language) are used. In the fifth section it is shown how MANAGELEX can be used in a Semantic Web context, by adding to a concept of an ontology, entries of a lexicon (with all lexical information) that define that concept. Also the concept hierarchy from the ontology attached to a lexicon can be useful for the HLT area. The last section presents the conclusions and further work to be done.

1 Introduction

“The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation. It is based on the idea of having data on the Web defined and linked such that it can be used for more effective discovery, automation, integration, and reuse across various applications” (Jim Hendler et. al, 2002).

The Semantic Web makes this possible by the addition of documents that encode the "knowledge" about a web page, photo, or database, in a publicly accessible, machine-readable form. Driving the Semantic Web is the organization of content into specialized vocabularies – ontologies-, which can be used by Web tools to provide new capabilities.

Due to the development of the Semantic Web, in the last years appeared the problem of combining Semantic Web technologies and resources with HLT elements, and of finding a way for the integration of ontologies in HLT applications. Having lexical information and information given by the hierarchy of an ontology, more power, in both HLT and Semantic Web applications, is gained.

This paper introduces a possible connection (in both directions) between ontologies (Semantic Web resources) and lexicons (HLT elements), using a lexicon management tool in order to have the lexicon structure mapped on some ontology.

2 MANAGELEX

Standard lexicon models and formats solve most of the problems related to the re-usability of lexical resources. They succeed to unify the existing lexical resources and to make them re-usable. However, important problems remain unsolved (existing of procedural elements, structures too complex and too hard to manipulate, not all types of languages taken into consideration, problems in merging lexicons, etc.) and there will be some time until everybody will use the same standard models and formats.

General lexical management tools, which help the user to manipulate and validate lexicons, represent an alternative to standardization. Such a tool is MANAGELEX, in development at Hamburg University, Natural Language Systems Department. This tool is not intended for replacing the present standards, but for managing the already existing lexicons (standard or non-standard).

The MANAGELEX is a tool that permits the user to create, read, convert, and combine lexicons. It is thought in such a way that makes it format, language and platform independent.

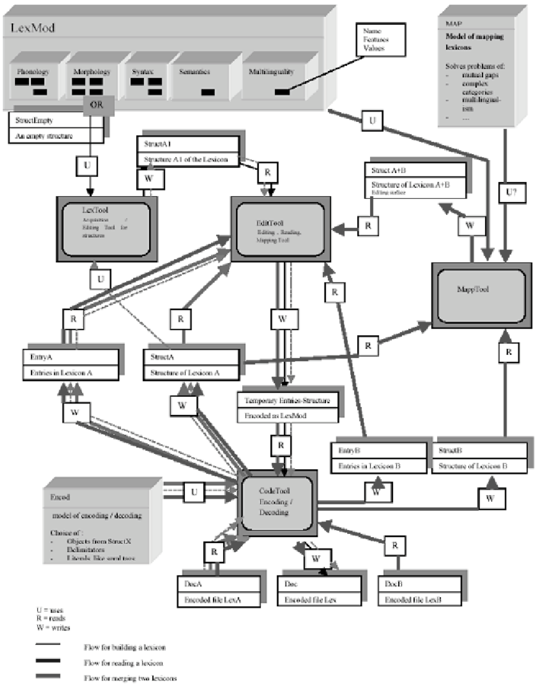


Figure 1. MANAGELEX Structure and Functionality (Gavrila, 2004)[1]

2.1 MANAGELEX Structure and Functionality

Following the ANSI data modeling specification (1999), the MANAGELEX structure is made of three levels of abstraction: the meta-model level, the model level and the real-world level.

The ***real-world level*** contains real objects, their concrete features and the relations that can appear between them. These objects correspond to the encoded **lexicon files** (DocA), **entry files** (EntryA) - lexical contents, and to the **structure files** (StructA) – lexicon structure.

The ***model level*** contains objects and attribute classes (real world objects with their features) and rearranges the relations between them. These objects correspond to the following tools:

- **EditTool**: writes, reads and edits lexicon entries:
- **StructTool**: defines and changes the linguistic specification;
- **EncodTool**: decodes lexicon files and encodes entries into files;
- **MapTool**: merges two lexicons with possible different structure.

The ***meta-model*** level deals with objects that appear in the model level and with the relations between them. There are proposed three models as object of this level:

- **LexMod**: rich model of possible lexical information. The model will be detailed in the next section;
- **Encod/EncodMod**: model that specifies the data structure for a specific entry in a lexicon;
- **Map/MapMod**: specifies the way of mapping two lexicons, taking into account mutual gaps, complex categories, etc.

Regarding the functionality of this tool, it ca do the following:

- Reading a lexicon;
- Updating a lexicon, either at the structure level, or at an entry level;
- Merging two lexicons;
- Creating a new lexicon, having as starting point LexMod, or from scratch.

The exact steps of each operation are illustrated in Figure 1.

[1] The LexTool module that appears in Figure 1 is the old name of the StructTool.

3 LexMod

LexMod is a generic lexicon model, which tries to contain as much lexical information as possible. In this model, linguistic features with possible values that may appear in a language are specified. The model is based on the study of more than 12 machine-readable lexicons (e.g. CELEX, MULTEXT, GermaNet, Verbmobil) and on several standard lexicon models (e.g. PAROLE/SIMPLE and MILE). LexMod has a flexible formal specification and is OWL encoded. The model allows for adding new lexical features, deleting, selecting, renaming, merging or splitting existing ones.

A specific feature of the LexMod is the separation of the linguistic data and the language data (e.g. examples can be added in the entries but not in the lexicon structure). This was done because language data examples can be added at any level.

LexMod can be updated with new linguistic features specific to other languages or linguistic focus. As being a generic lexicon model, LexMod contains no optional grammatical features. It also has no relations between features. If needed (e.g. in case of a relational structure), these can be specified later using StructTool.

The LexMod structure contains the following levels of information (see Figure 2):

- Lexicon Information;
- Entry Information;
    - o Morphological Information;
    - o Phonological Information;
    - o Syntactical Information;
    - o Semantic Information;
    - o Multilingual Information.

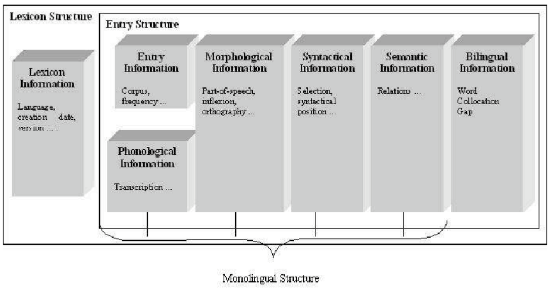


Figure 2. LexMod Structure (Gavrila, 2004)

3.1 LexMod Encoding

LexMod is encoded in OWL (*http://www.w3.org/TR/owl-features/*). The choice for OWL is based on the following considerations:

- Using XML would have introduced a lot of redundant information (e.g. he values of gender' should have been mentioned for every part of speech, etc.).
- RDF/RDFS offers a limited set of relations between classes or properties (e.g. no synonymy relation) and does not allow some specifications (e.g. cardinality restrictions)

Although it is too complex (for the LexMod needs), it permits the user to express all the property constraints he/she needs. For the LexMod encoding it was used OWL Lite. If at a certain point it will be necessary, more complex OWL versions can be used. In encoding the LexMod structure only a subset of the OWL tags were used: (e.g. owl:cardinality, owl:Class, owl:DatatypeProperty, owl:maxCardinality, owl:minCardinality, etc.). There were used 15 tags out of 48. The rdfs:comment tag, although it does not encode linguistic information, is used for elements and data type properties. In this tag, the name that appears on the graphical interface is written.

The main ideas in describing a lexicon structure, and in establishing what should be an element and what a property, are the following:

- If a grammatical feature can have one or several literal values, than it is a data type property.
- If a grammatical feature is described using other features, that it is an element.
- If there are relations between classes, than an object property is used.

Below, in Table 1, some classes and properties that appear in the LexMod encoding, are mentioned.

The LexMod OWL encoding has 34 elements, 88 data type properties, and 23 object properties. A hierarchy of the LexMod classes can be seen in Figure 3. As seen from this figure there is a tree representation.

Class	Property
LexiconStructure	hasLexiconInfo, hasEntryStructure
LexiconInfo	lexiconName, language, version, creationDate, etc.
EntryStructure	hasEntryInfo, hasMonolingualStructure, hasBilingualStructure
EntryInfo	corpus, frequency, workingState, termStatus,
MonolingualStructure	hasMorphologicalInfo, hasPhonologicalInfo, hasSyntacticalInfo, hasSemanticInfo
BilingualStructure	toLanguage, toLexicon, hasCorrespondences
MorphologicalInfo	HasPOS, etc.
PhonologicalInfo	phoneticTranscription, terminalDevoicing, etc
SytacticalInfo	hasSelection, syntacticPosition, etc.
SemanticInfo	hasRelations, ontologyTypes, etc.

Table 1. Some Classes and Properties in LexMod

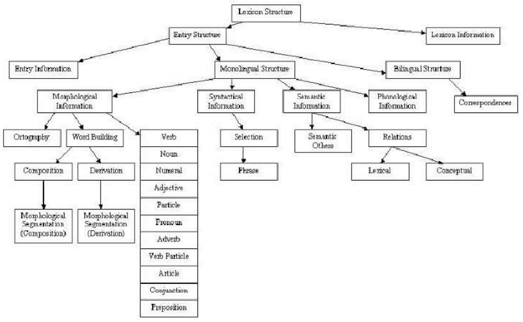


Figure 3. The LexMod Tree Representation (Gavrila, 2004)

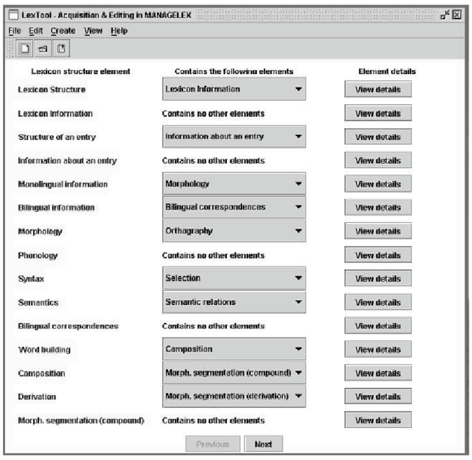
4 StructTool

StructTool is the module that deals with lexicon structures. The aims of this module of MANAGELEX are:

- To be able to read LexMod, and generate a graphical interface with all information, as seen in Figure 4.
- To allow for selecting existing categories and their range of value.
- To allow for defining new categories and save them in LexMod.
- To allow for updating of existing categories.
- To permit the user to define the structure of the lexicon that he/she needs.
- To call the EditTool tool in order to add entries in the lexicon.

The idea behind this module is the following. The StructTool reads the LexMod structure (or another structure similar to LexMod), and generates a graphical user interface that contains all the elements found in this the lexicon structure. The user has the possibility to select what he/she needs, to add, delete, rename, merge or split elements. At the end the new structure is saved in another structure file OWL encoded, similar to LexMod.

Using StructTool the user can create a new lexicon structure – either by using LexMod or from scratch – or can update a lexicon structure. When creating a new lexicon structure, or when changing an existing one, there can be done several operations on the grammatical features (structure elements, properties): adding new ones, deleting, merging, splitting, or renaming some features from an older structure.



Lexicon structure element	Contains the following elements	Element details
Lexicon Structure	Lexicon Information	View details
Lexicon Information	Contains no other elements	View details
Structure of an entry	Information about an entry	View details
Information about an entry	Contains no other elements	View details
Monolingual information	Morphology	View details
Bilingual information	Bilingual correspondences	View details
Morphology	Orthography	View details
Phonology	Contains no other elements	View details
Syntax	Selection	View details
Semantics	Semantic relations	View details
Bilingual correspondences	Contains no other elements	View details
Word building	Composition	View details
Composition	Morph. segmentation (compound)	View details
Derivation	Morph. segmentation (derivation)	View details
Morph. segmentation (compound)	Contains no other elements	View details

Figure 4. Snapshot of LexTool/StructTool Interface

This way the user gets the structure he needs, not being obliged to use a very complicated lexicon structure, or being able to improve/modify the initial structure (LexMod) in case some features were not taken into consideration.

5 Connecting MANAGELEX to the Semantic Web

In the above sections there was presented a tool that manages lexicons: MANAGELEX and two of its components. In this section it will be shown how it can fit in the Semantic Web applications.

One of the connections of this system and the Semantic Web is that it uses in encoding its meta-level component LexMod the OWL language. This means that the lexicon and the ontology are encoded in the same way and this way there are not needed more tools for parsing or manipulating structures.

One of the main challenges in the design of ontologies with multilingual instances is that, very often words in one language overlap concepts in the ontology, and there is no one-to-one mapping to the meaning in the other language

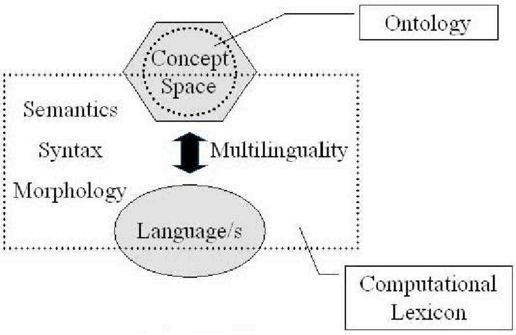


Figure 5. Ontologies and Lexicons (Lenci, 2003)

We can think at MANAGELEX as a plug-in to a Semantic Web application under the following scenario:

- All texts in the Semantic Web refer to an ontology;
- On this ontology we map the lexical information corresponding to the texts. In this way one can access the linguistic information as synonyms, hyperonyms, translation equivalents, as well as other semantic features.

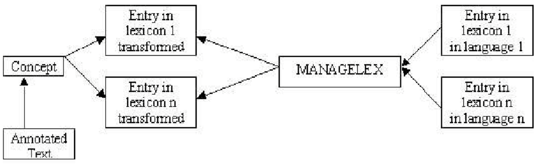

Entry in lexicon 1 transformed
Concept
Entry in lexicon n transformed
Annotated Text
MANAGELEX
Entry in lexicon 1 in language 1
Entry in lexicon n in language n

Figure 6. Connection between Ontologies and Lexicons

MANAGELEX can help by transforming and encoding (in the Semantic Web manner) existing lexicons, so that the connection between lexicons and ontologies is possible. The way MANAGELEX makes the connection between (existing) lexicons and ontologies is shown in the figure above.

6 Conclusions

The paper presents a generic lexicon tool MANAGELX and a way of connecting Semantic Web specific-elements (ontologies) with HLT elements (lexicons) by using it.

For the moment, only European languages are modeled. Among the further operations to be implemented or considered, there are:

- The implementation of all the other tools and models, and of ready-to-start configurations for widely used standards, like PAROLE/SIMPLE and MILE;
- An updating process for LexMod, by including important changes (adding operation) in structure files to LexModshould be considered;
- Analyzing and considering other types of languages.

References

Grigoris Antoniou and Frank van Hermelen. 2004. *A Semantic Web Primer*. The MIT Press Cambridge, Massachusetts, London, England

Nicoletta Calzolari, Francesca Bertagna, Alessandro Lenci and Monica Monachini 2003. *Standards and Best Practice for Multilingual Computational Lexicons &MILE (the Multilingual ISLE Lexical entry)*. Deliverable D2.2-D3.2 ISLE Computational Lexicon Working Group, to be retrieved at http://www.ilc.cnr.it/EAGLES96/ isle/clwg_doc/ISLE_D2.1-D3.1.zip

Monica Gavrila. 2004. *LexMod and LexTool – Lexical Model, Acquisition and Editing in MANAGELEX*. Diploma Thesis, Hamburg University, Computer Science Department, R36887

E. Gius. 2003. *Vergleich maschinenlesbarer deutscher Lexika nach linguistischem Inhalt, Wertebereichen und Kodierung*, Diploma Thesis, University of Hamburg, manuscript.

Emilie Guimier and Antoine Ognowski 1998, *LE-PAROLE Reports on the Morphological and Syntactic Layers*, to be retrieved under: http://www.ub.es/gilcub/SIMPLE/reports/parole

Jürgen Handke, 1995. *The Structure of the Lexicon– Human versus Machine*. Mouton de Gruyter, Berlin-NewYork, 1995

Jim Hendler, Tim Berners-Lee, and E. Miller. 2002 *Integrating Applications on the Semantic Web*, http://www.w3.org/2002/07/swint

Alessandro Lenci. 2003. *Computational Lexicons and the Semantic Web*, Presentation Eurolan Summer School (The Semantic Web and Language Technology. Its Potential and Practicalities), Eurolan CD, Bucharest

Nida Ruimy, Ornella Corazzari, Elisabetta Gola, Antonietta Spanu, Nicoletta Calzolari and Antonio Zampolli 1998, The European LE-PAROLE Project and the Italian Lexical Instantiation, *Proceedings of the ALLC/ACH, 1998, Lajos Kossuth University, Debrecen, Hungary*, July, 5-10 1998, 149:153.

Cristina Vertan and Walther von Hahn. 2002. *Towards a Generic Architecture for Lexicon Management.* Proceedings of the Workshop "International Standards of Terminology and Language Resources Management", LREC, 45:48

Cristina Vertan, Walther von Hahn, Monica Gavrila. 2005. *MANAGELEX – a Tool for the Management of Complex Lexical Structures.* GLDV Workshop "Exchange of Lexical and Terminological Resources in Machine Translation (MT), Computer-Aided Translation (CAT) and Terminology Management Systems (TMS)". Köthen. 17 June. To be published in Proceedings.

Cristina Vertan. 2004. *Language Resources for the Semantic Web – perspectives for Machine Translation.* Proceedings of the Workshop "Language Resources for Translation Work, Research and Training". Coling. Geneva. 37:42.