A lexical platform for Spanish*

José M. Goñi, José C. González
E.T.S.I. Telecomunicación
Universidad Politécnica de Madrid
28040 Madrid, SPAIN
Phone: +34 1 336.72.87
Fax: +34 1 336.72.89
{jmg,jcg}@mat.upm.es

Antonio Moreno
Dep. de Lingüística
Universidad Autónoma de Madrid
48049 Cantoblanco, Madrid, SPAIN
sandoval@ccuam3.sdi.uam.es

August, 1995

1 Introduction

This work presents a general purpose lexical platform for the Spanish language. It has been designed and implemented as the core of a set of natural language applications currently under development at *Universidad Politécnica de Madrid.* These applications include a spell checker, a unification-based parser and grammar, morpho-syntactic taggers and other tools for the automatic acquisition of linguistic knowledge and for demonstrations.

The main relevance of the platform comes from two facts:

- First, it has been designed to support well-founded linguistic generalizations. So, it is able to accept and generate any correct Spanish inflected form, but no incorrect ones.
- Second, and simultaneously, it has been developed from a strong practical orientation. Efficiency issues, both in terms of access and storage, have been considered capital.

2 The lexical representation language

2.1 Motivation

The lexical representation language was specially designed searching for the following properties:

*This work has been partially supported by the Spanish *Plan Nacional de I+D*, under the Research Project *An Architecture for para Natural Language Interfaces with User Modeling* (TIC91-0217C02-01).

Expressiveness: All the information needed in the lexical database can be expressed in a structured way. Linguistic generalizations are captured by grouping related entries in *lemmas*, or by using the mechanism of information inheritance. The information related to a lemma is structured as an attached tree-shaped feature bundle.

Versatility: Different applications may have different lexical interfaces, depending on programming languages. In our approach translation to other representation formats and languages is easily done in a non-ambiguous way.

Economy of expression: The syntactic overload needed to structure the information has been reduced to a minimum, without endangering neither the expressive ability nor the non-ambiguity of the syntax of the formalism.

This feature is in permanent conflict with readability, although the latter is not strongly degraded, since source lexical databases are intended to be liable to edition by hand with any text editor.

Non redundancy: Redundant information is kept to a minimum by exploiting default inheritance and the notational abbreviations included, such as value disjunction.

Spanish language strongly relies in inflectional morphology for word formation, particularly for verbs (where different combination of tense, mood, number and person, allows up to 53 different simple inflected forms) and nouns and adjectives (up to 4 forms). So, for any serious natural language applications an account of morphology is needed to group different but related entries in the lexicon, keeping it in a manageable size. We adopted the morphological model for Spanish described elsewhere.

Such morphological model needs that the dictionary contains the different (surface) allomorphs for lexical roots and morphemes, and some lexicalized words for suppletion in irregular cases (or unique word forms). In this model morphological processors only need to concatenate strings, but accounting for the constraints attached to these entries. These restrictions are encoded as feature-value pairs integrated in the feature bundle attached to each entry.

We have merged all the allomorphic variants for a particular lexical entry in the same dictionary entry or *lemma*. The different allomorphs are computed automatically from a representative surface form of the lemma by means of special rules based on regular expressions, according to the morphological model –or paradigm– of the lemma.

2.2 The language details

Each entry in the dictionary has a number of labeled features, that can have an atomic value –a label assigned to that feature–, or a structured one –another feature structure. Value assignment to a feature is achieved by equations:

$$p = v_1\ v_2\ \ldots\ v_n$$

where p is a sequence of one or more blank space separated labels that constitute a path for accessing the feature from the root of the tree. The v_i are the atomic values that this particular feature can take. Paths in the left hand side of the equations are the mechanism provided to define a tree-shaped feature structure, and the multiple-valued features are provided as a notational shorthand for disjunction.

The Source Lexical Base is split into sections, each one headed by a special keyword. An include facility is also provided in order to promote physical division of the Lexical Base into different computer files. The sections that can appear in the base are:

Morphemes: This section contains the entries for the inflectional morphemes, each one with an appropriate feature bundle.

Words: This section contains entries for lexicalized words that need to be *as is* in the dictionary.

Classes: This section contains class definitions, that is, a label attached to a feature bundle. These features are inherited by all the entries –usually lemma entries– that are declared to belong to that class.

Lemmas: This section keeps in the same entry the information of different inflected word forms. It usually maintains for each entry the common information to all the inflected forms, information about how to compute the different allomorphs and their concatenating constraints (usually combined with the inheritance mechanism, so this information is encoded in classes).

Allomorphy rules: This section expresses how to compute all the allomorphs needed for a particular lemma entry. Rules are a sequence of productions that are fired by means of a pattern matching process in its left-hand side. Whenever a production is triggered, the regular expression variables in its left-hand side are instantiated, returning the value computed in the right-hand side.

Type checking: A special section is included to provide a limited type checking for the lexical base entries. All the features have to be declared here, as well as their possible values.

All the details for entries, lemmas, classes, rules, inheritance devices, allomorphy rules and so on will be given in the final version of the paper.

3 The dictionary

The lexicon for this platform has been derived from different sources:

- The concise version of the COLLINS English-Spanish dictionary.
- The work done at New York University by A. Moreno and C. Olmeda in the framework of the PROTEUS project.
- The tag-set proposed by A. Martín at Universidad Autónoma de Madrid for morpho-syntactic tagging of Spanish texts.
- Corpora of Spanish texts compiled by the Kings College of London and by the International Telecommunications Union. (A small part of the first corpus has been tagged manually by A. Martín).
- Collections of texts from Spanish newspapers (ABC, El Mundo, etc.) and lists of words.

Morphological information has been derived in a semi-automatic way:

1. Tools were implemented to classify words (nouns, adjectives and verbs) according to morphological models from their surface form.
2. A Prolog generator was used to produce all the inflected forms (or the most representative ones in the case of verbs) for each dictionary entry. This was devised as a way to check manually the results of the classifiers.
3. Finally, the inflected forms were checked manually by groups of semi-volunteer students.

4 Tools for lexical processing

A number of tools have been –or are being– developed at our site to support the intended properties of the lexical knowledge representation language and the dictionary development process, and to provide efficient access to the dictionary.

The tools developed to support the dictionary development process include:

A verb classifier: a tool that permits to assign a verbal model to a particular verb, according to its infinitive surface form.

A lexical expander: a tool able to compute all the allomorphs and to assign them all the relevant features by following the inheritance links. The result is a full expanded lexicon of morphemes, root allomorphs and lexicalized words.

The GRAMPAL morphological analyzer/generator: a Prolog prototype tool that implements the morphological model adapted. It was used –in combination with the expander– to generate all the inflected forms of our lexicon for evaluation and debugging purposes.

A tokenizer: a set of heuristics to filter a corpus to evaluate the dictionary coverage. These heuristics filter out proper names, acronyms, initials, dates and so on.

For the exploitation of the dictionary some tools were implemented also:

An efficient dictionary access library: The dictionary is indexed by means of a letter-tree (*trie*) that permits fast access and retrieval of the entries.

A morphological analyzer: Its current implementation is a context-free chart based parser, adopted to permit string concatenation. Morphological rules are context-free (PATR-II like) unification rules. This tool interfaces with the dictionary access library to provide full inflected word forms access for NLP applications.

Unification algorithms: Both full unification and pseudo-unification algorithms can be integrated in the morphological processor as well as in the syntactic level processor. Note that in our architecture, syntactic and morphological processing can be integrated in the same processor.

The tools developed for application purposes are coded in –or provide programming interfaces to– C/C++. This election has been made by efficiency and portability considerations.

5 Conclusions

The current version of our dictionary is around 38,000 lemmas covering more than 465,000 inflected forms: 5,200 regular verbs, 2,100 irregular verbs, 9,800 adjectives, 21,000 nouns and 500 entries for prepositions, conjunctions, articles, adverbs, pronouns, etc.

The final version of this paper will provide full evaluation of the coverage and efficiency of the platform for different applications.

References

[Aoe and Morimoto, 1992] Aoe, J. and Morimoto, K. An efficient implementation of trie structures. *Software-Practice and Experience, vol. 22, n. 9, pp. 695–721, September, 1992.*

[Briscoe et. al., 1993] Briscoe, T.; Paiva, V. and Copestake, A. Inheritance, Defaults and the Lexicon. *Studies in Natural Language Processing. Cambridge University Press, 1993.*

[Goñi and González, 1995] Goñi, J.M. and González, J.C. A framework for lexical representation. *Proceedings of AI'95: Fifteenth International Conference. Language Engineering '95, pp. 243–252. Montpellier, June 27–30, 1995.*

[Martín, 1994] Martín, A. Una Propuesta de Codificación Morfosintáctica para Corpus de Referencia en Lengua Española. *Tesis Doctoral. Universidad Autónoma de Madrid, 1994.*

[Moreno, 1992] Moreno, A. Un Modelo Computacional basado en la Unificación para el Análisis y la Generación de la Morfología del Español. *Tesis Doctoral. Universidad Autónoma de Madrid, 1992.*

[Moreno and Goñi, 1995] Moreno, A. and Goñi, J.M. GRAMPAL: A morphological model and processor for Spanish implemented in Prolog. *Paper to be presented at the 1995 Joint Conference on Declarative Programming (GULP-PRODE'95), Marina di Vietri (Salerno, Italy), September, 1995.*

[Russell et. al., 1991] Russell, Graham J.; Carroll, John and Warwick-Armstrong, Susan. Multiple Default Inheritance in a Unification-Based Lexicon. *In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, pp.215–221, 1991.*