A “toolbox” for tagging the Spanish C-ORAL-ROM corpus

José M. Guirao† Antonio Moreno-Sandoval‡

† Universidad de Granada, Spain
‡ Universidad Autónoma Madrid, Spain
jmguirao@ugr.es sandoval@maria.lllf.uam.es

Abstract

The goals of this paper are to present the tagging procedure for a Spanish spoken corpus, and to show a tool developed for helping human annotators in the process. Some tagging problems especially relevant in spoken corpora, although found also in written texts, will be introduced first. The paper will summarise the experience of the group in tagging one of the currently largest spontaneous speech corpora (over 300.000 transcribed words)

1 Introduction: problems in tagging a spoken corpus

1.1 The multi-word tagging

The Spanish C-ORAL-ROM corpus consists of 312597 tagged tokens. Every tag marks a lexical unit, regardless the number of graphical words it is made of. That is, words and multi-words are considered as a unit or token. For instance, “hola” and “buenos días” are counted (and tagged) as one token each. Accordingly, amalgams of two lexical units in a single graphical word, such as “al” or “del”, are split into two tokens: “a” “el”, “de” “el”. This assumption is important for understanding the tagging procedure and the tagger evaluation. Moreover, a tagger which cannot analyse multi-words will produce very poor results, at least for a Spanish spoken corpus, where very frequent multi-words will be tagged incorrectly:

Correct	Incorrect
“o sea” Discourse Marker	“o” Conj. + “sea” V
“en lugar de” Prep.	“en” Prep + “lugar” N + “de” Prep
“por ejemplo” D M	“por” Prep + “ejemplo” V

Table 1. Multiword tagging.

1.2. A tag for Discourse Markers

Discourse Markers (DM), whose frequency is lower in written texts, are especially relevant in spontaneous speech. Many tagsets and taggers do not include them, and they are usually considered as adverbs, adjectives or nouns. In our annotation, there is a distinction between discourse markers and other POSs. As a consequence, new cases of ambiguity arise:

“bueno” ADJ “Juan es bueno” Juan is *good*	“bueno” DM “bueno / espero que te guste” *Well* I hope you like it.
“hombre” N “Juan es un hombre bueno” Juan is a good *man*	“hombre” DM “hombre / no te enfades” Don’t be mad, man!

Table 2. Discourse Markers and POS ambiguity

Sometimes it is difficult to decide whether the proper tag is a DM or other category. The intonation, or the pragmatic context can help the trained annotator, but it is impossible to formalise in the disambiguation grammar. DM are responsible for a residual uncertainty.

1.3. Tokenization

To segment the stream in tokens presents several differences with respect to the same task in a written corpus. Not only the recognition of multi-words or amalgams, but also the prosodic tags.

Contrary to written texts, where punctuation marks help to delimit analysis units as clauses, sentences and paragraphs, in spoken transcriptions prosodic marks are used instead. Transcriptions are divided in dialogic turns, and turns have tone units, retracting marks, overlapping marks, disfluencies marks, etc. All these types of phenomena fragment the utterance and introduce additional difficulty in tagging: “agrammatical” sequences are quite frequent in spontaneous speech.

1.4. Unknown words

Every tagger will have to deal with words that are not in its lexicon or in its training corpus. In spontaneous speech there are several sources of unknown words:

- **Neologisms**: Spoken language includes words which are not in the dictionaries or in written texts. New words invented by speakers, which are not incorporated yet to the common language.
- **Pronunciation** mistakes: speakers hardly use the proper word. However, the transcription has to reflect the actual use.
- **Derivatives**: The use of appreciative derivation (prefixes or suffixes) is quite common in spontaneous speech. As a result, a common word as “agua” (water) can be said as “agüita” (literally “little water”). Rules for handling derivation are needed.

On the other hand, **proper names** recognition is not a problem in spoken language: since the transcription does not follow the written language rules, only proper names start with a capital letter.

2 The tagger

The main goal is to provide a complete morphological and POS tagging, including lemmatisation. These tasks have been performed automatically and validated by expert annotators. For the automatic tagging, a hybrid rule-based/statistical tagger has been used. The procedure is divided in three steps:

1. **Word analysis**: a morphosyntactic analyser provides all possible tags for a specific token.
2. **Disambiguation phase 1**: a feature-based Constraint Grammar resolves some of the ambiguities
3. **Disambiguation phase 2**: a statistical tagger (the TnT tagger) resolves the remaining ambiguous analyses.

Human annotators have access interactively to the three phases, and can manually change the annotations. In order to validate the human annotation, the whole tagging system is run and the final results are compared against the human-annotated corpus. Evaluation results are reported in Moreno et al. (forthcoming). It is important to stress that the evaluation experiment on a 50.000 words test corpus did show both a few mistakes in the human annotation and some incorrect rules in the disambiguation grammar. The mistakes were fixed while some problems in the grammar are intrinsically unsolved. The precision rate in the evaluation was 95.6. The figure is quite good compared to similar taggers, if we take into account that some of them do not deal with multi-words and discourse markers are not in their tagsets.

The tagging procedure and its evaluation is described in Moreno & Guirao(2003). Here we will briefly provide the main points.

2.1. Word analysis

For the morphological analysis we use GRAMPAL (Moreno 1991; Moreno & Goñi 1995) which is based on a rich morpheme lexicon of over 50.000 lexical units, and morphological rules. GRAMPAL is a symbolic model based on feature unification grammar The system is reversible: same set of rules and same lexicon for both analysis and generation of inflected wordforms. It is designed to allow only grammatical forms. The most prominent feature is its linguistic rigour, which avoids both over-acceptance and over-generation, providing at the same time all the possible analyses for a given word. This system has been successfully used in language engineering applications as ARIES (Goñi, González and Moreno 1997).

With respect to the original system, developed for analysing written language, new modules have been incorporated to handle specific spoken language features:

1. A new tokenizer, for identifying utterance boundaries by means of dialog turns and prosodic tags.
2. A derivative morphology recogniser, including rules and lexicon entries for over 240 prefixes and suffixes.

The analysis procedure consist of five parts:

1. *Unknown words detection*: after the tokenizer segments the transcription in tokens, a quick look-up for unknown words is run. The detected new words are added to the lexicon
2. *Lexical pre-processing:* here the program splits portmanteau words (“al”, “del” → “a” “el”, “de” “el”) and verbs with clitics (“damelo” → “da” “me” “lo”).
3. *Multi-words recognition*: the text is scanned for candidates to multi-words. A lexicon, compiled from printed dictionaries and corpora, is used for the task.
4. *Single words recognition*: every single token is scanned for every possible analysis according to the morphological rules and lexicon entries. Approximately 30% of the tokens are given more than one analysis, and some of them are given up to 5 different analyses.
5. *Unknown words recognition*: those remaining tokens that are not considered new words, pass through the derivative morphology rules. If some tokens still remain without any analysis (because they were not included in the lexicon nor were recognised by the derivative rules), they will wait until the statistical processing, where the most probable tag, according the surrounding context, is given.

2.2. Disambiguation grammar

POS disambiguation has been solved using a rule-based model. In particular, an extension of a Constraint Grammar using features in a Context-Sensible PS. The output of the tagger is a feature structure written in XML. Here is shown the possible tags for the token “la”, as an article and as a pronoun.

```
<pal cat=″ART″ lema=″el″ gen=″fem″ num=″plu″> la </pal>

<pal cat=″P″ lema=″la″ pers=″p3″ gen=″fem″ num=″plu″> la </pal>
```

The formalism allows several types of context sensitive rules. The most basic and frequent rule is as follows:

```
"word" → <cat="X"> / _• <cat ="Y">
"word" → <cat="Z"> / <cat ="W"> •
```

Here are some rules for disambiguating the token “la”:

```
"la" -> <cat="ART"> / •_<cat="N" gen="fem">
"la" -> <cat="P">   / •_<cat="V">
"la" -> <cat="ART"> / • <cat="ADJ">
"la" -> <cat="P">   /  yo _•
"la" -> <cat="P">   /  tú _•
```

The grammar writer tries always to provide as much particular rules as possible for a given ambiguous case. The goal is to get the higher level of precision in disambiguation in this phase.

2.3. Statistical disambiguation

For the remaining unresolved ambiguities, an statistical tagger (the Tnt tagger, Brants 2000) is applied. The statistical model has been obtained from a 50000 words training corpus, which is a subset of the whole spoken corpus. The training corpus has been verified by linguists. The statistical part is applied at the end of the process, when the competence-based knowledge (the grammar and lexicon) is not able to provide a precise and discrete analysis. This way, in case no appropriate analysis is found, always the likeliest tag is assigned.

3. The “toolbox”

In order to help human annotators, an xml-based interface has been developed, which allows the interactive edition of the lexicon, the disambiguation grammar and the annotated text. This “tool box” integrates the different modules of the system: the tokenizer, the morphological analyser, the rule-based and the statistical disambiguation. The interface has resulted to be a useful tool for controlling the complex process of enriching and modifying the mentioned modules.

In this section, we will show some screenshots to give an idea of the benefits of employing such an interface tool. This section will also show how the annotator faces different type of problems.

3.1. Editing the annotated texts

The most basic tool is an editor-concordancer which allows to search for problems and wrong analyses. The experienced annotator usually knows which are the problematic cases. In Spanish the most frequent and hard problem is the disambiguation of “que”, as a RELative and as a Conjunction. “Que” is the most frequent token in spoken Spanish, and it appears in so many contexts that it is impossible to write disambiguation rules for every case, and statistical models do not resolve either (at least in the current state of training). Careful verification by hand is needed.

This option allows to search for occurrences of “que” in some problematic contexts (see Figure 1). After finding a wrong tag, the annotator has the option to directly write the correct tag, and saving the result.

3.2. What if the word is not in the lexicon?

No lexicon (nor a statistical model) is complete. As a consequence, a method for adding new entries is needed. Spontaneous speech presents words which are not usually found in written texts or printed dictionaries. This option edits the GRAMPAL lexicon, allows to introduce and modify entries and saves the enriched lexicon (Figure 2).

3.3. The disambiguation grammar editor

In the interactive process of revising the annotated texts, the linguist wants to add new rules for disambiguation in specific contexts. As a typical grammar writing process, the linguist has to test the new rule. This option allows to edit the grammar file, compile it and try a utterance (Figure 3). The last is especially useful for checking whether the new rule is working properly or not, without running the whole tagging process on the file.

4. Conclusions and future work

Quality tagging of corpus requires, in addition to a good and complete tagger, a human verification of the annotated text. If the corpus is intended to be used as a reference data resource, as it is the case, then a linguist-controlled annotation is a must. A friendly interface that integrates the different modules is a clearly useful tool.

This paper has also shown the experience of tagging an spontaneous speech corpus. In many ways, the procedure is similar to tagging a written corpus, but some differences have also been exposed.

We expect to enrich the current tool box and the current corpus with new layers of annotations, syntactic and semantic information (Alcántara 2003).

5. References

Alcántara, M. (2003). Semantic Tagging System for Corpora. Poster presented at the V International Workshop on Computational Semantics, Tilburg (Holland).

Brants, T. (2000). TNT – a statistical part-of-speech tagger. In Proceedings of ANLP, Seattle, USA.

Goñi, J.M., González, J.C. and Moreno, A. (1997). ARIES: a lexical platform for engineering Spanish processing tools. In Natural Language Engineering, 3(4), pp. 317-345.

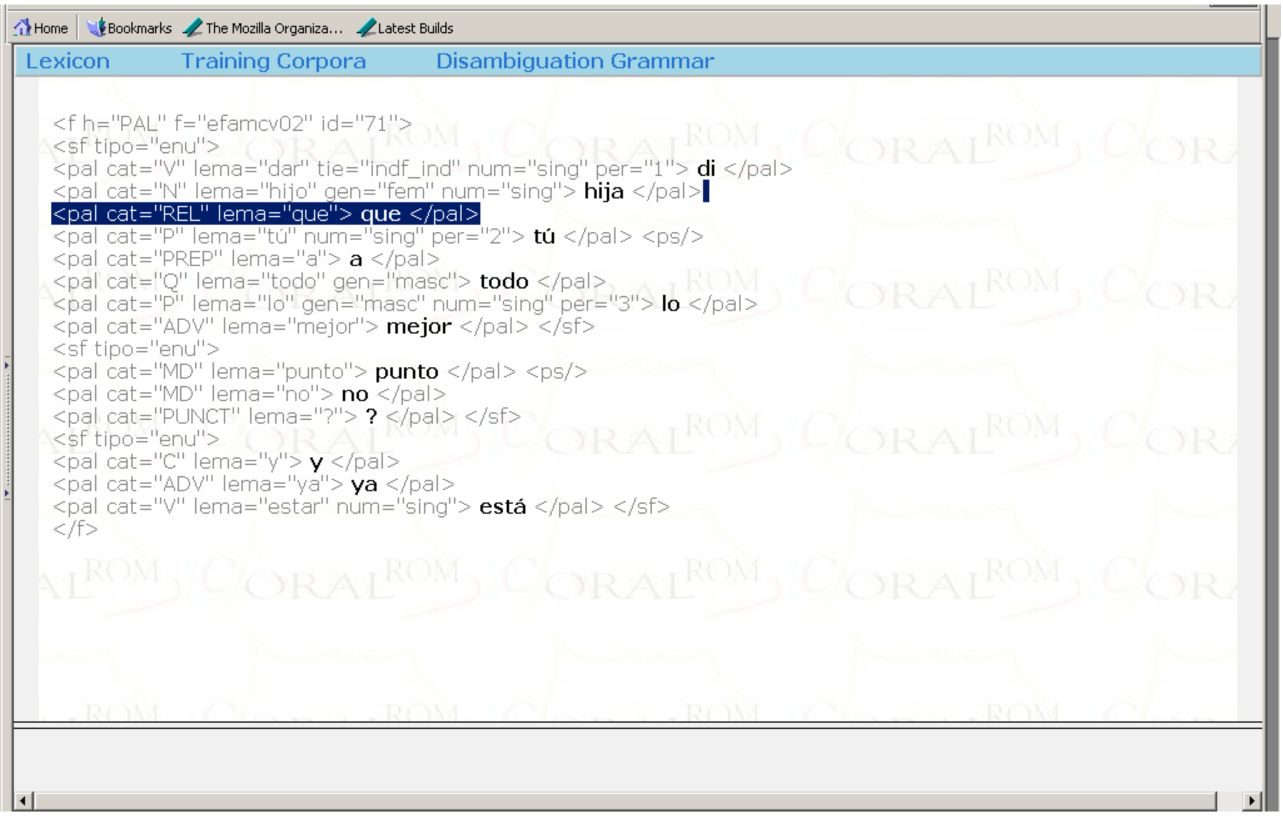
Moreno, A. (1991). Un modelo computacional para el análisis y generación de la morfología del español. Ph.D. Thesis. Universidad Autónoma de Madrid, Spain.

Moreno, A. and Goñi, J.M. (1995). GRAMPAL: a morphological processor of Spanish implemented in Prolog. In Proceedings of Joint Conference on Declarative Programming (GULP-PRODE’ 95). Marina di Vietri, Italy.

Moreno, A. and Guirao, J.M. (2003). Tagging a spontaneous speech corpus of Spanish. In Proceedings of RANLP 2003. Borovets, Bulgaria.

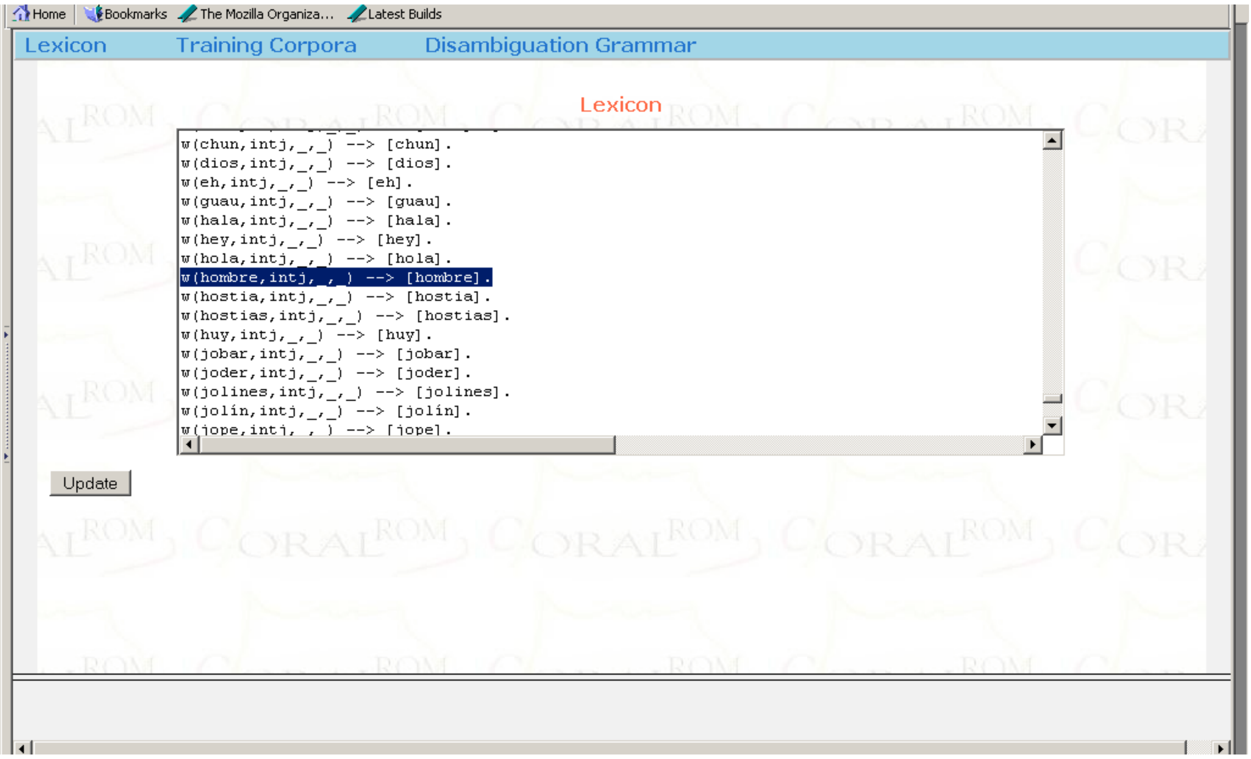
Moreno, A., De la Madrid, G., Alcántara, M., González, A., Guirao, J.M. and De la Torre, Raúl (forthcoming). Notes on the Spanish Spoken corpora and linguistic studies. To be published as a chapter in the C-ORAL-ROM book.

6. Appendix: The Screenshots



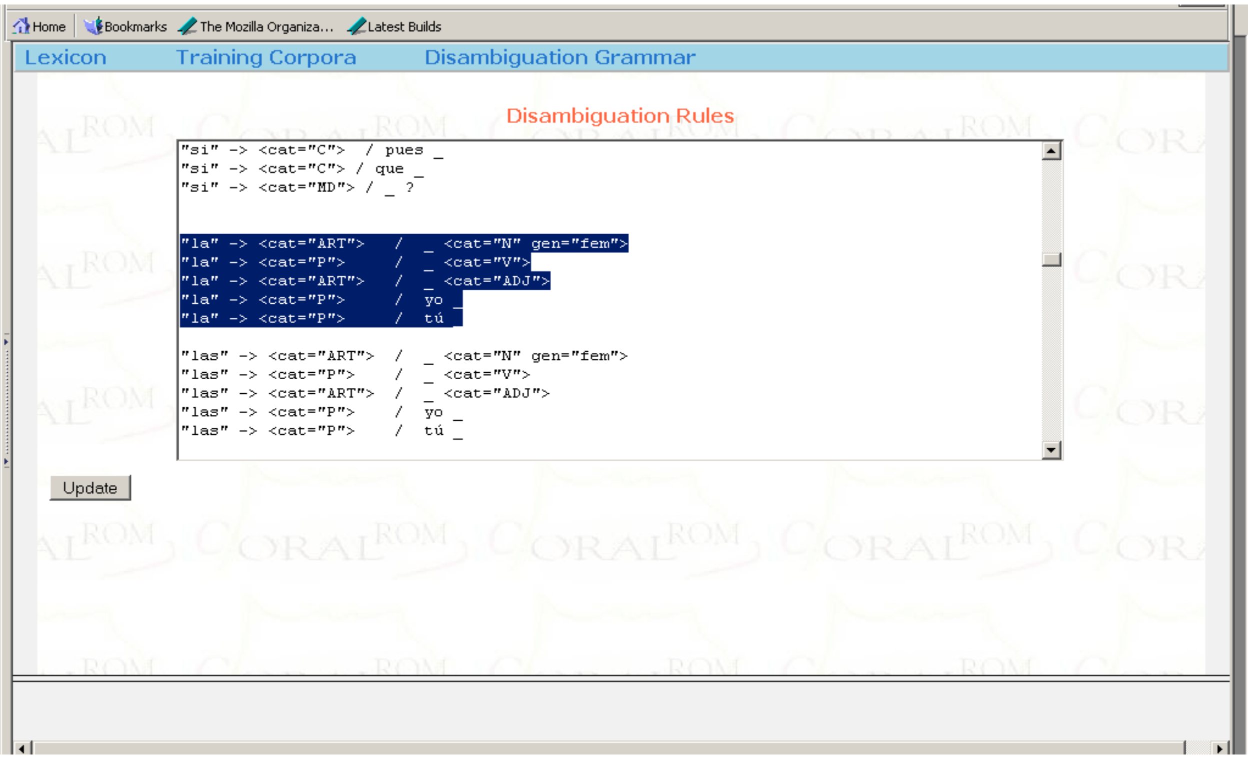
```
<f h="PAL" f="efamcv02" id="71">
<sf tipo="enu">
<pal cat="V" lema="dar" tie="indf_ind" num="sing" per="1"> di </pal>
<pal cat="N" lema="hijo" gen="fem" num="sing"> hija </pal>
<pal cat="REL" lema="que"> que </pal>
<pal cat="P" lema="tú" num="sing" per="2"> tú </pal> <ps/>
<pal cat="PREP" lema="a"> a </pal>
<pal cat="Q" lema="todo" gen="masc"> todo </pal>
<pal cat="P" lema="lo" gen="masc" num="sing" per="3"> lo </pal>
<pal cat="ADV" lema="mejor"> mejor </pal> </sf>
<sf tipo="enu">
<pal cat="MD" lema="punto"> punto </pal> <ps/>
<pal cat="MD" lema="no"> no </pal>
<pal cat="PUNCT" lema="?"> ? </pal> </sf>
<sf tipo="enu">
<pal cat="C" lema="y"> y </pal>
<pal cat="ADV" lema="ya"> ya </pal>
<pal cat="V" lema="estar" num="sing"> está </pal> </sf>
</f>
```

Figure 1. Editing the tagged file



```
w(chun,intj,_,_) --> [chun].
w(dios,intj,_,_) --> [dios].
w(eh,intj,_,_) --> [eh].
w(guau,intj,_,_) --> [guau].
w(hala,intj,_,_) --> [hala].
w(hey,intj,_,_) --> [hey].
w(hola,intj,_,_) --> [hola].
w(hombre,intj,_,_) --> [hombre].
w(hostia,intj,_,_) --> [hostia].
w(hostias,intj,_,_) --> [hostias].
w(huy,intj,_,_) --> [huy].
w(jobar,intj,_,_) --> [jobar].
w(joder,intj,_,_) --> [joder].
w(jolines,intj,_,_) --> [jolines].
w(jolín,intj,_,_) --> [jolín].
w(jope,intj,_,_) --> [jope].
```

Figure 2. Editing the lexicon



```
"si" -> <cat="C">  / pues _
"si" -> <cat="C"> / que _
"si" -> <cat="MD"> / _ ?


"la" -> <cat="ART">   /  _ <cat="N" gen="fem">
"la" -> <cat="P">     /  _ <cat="V">
"la" -> <cat="ART">   /  _ <cat="ADJ">
"la" -> <cat="P">     /  yo _
"la" -> <cat="P">     /  tú _

"las" -> <cat="ART">  /  _ <cat="N" gen="fem">
"las" -> <cat="P">    /  _ <cat="V">
"las" -> <cat="ART">  /  _ <cat="ADJ">
"las" -> <cat="P">    /  yo _
"las" -> <cat="P">    /  tú _
```

Figure 3. Editing the disambiguation grammar