A Tagger Environment for Galician

M. Vilares J. Graña T. Araujo D. Cabrero I. Diz

M. Vilares, *J. Graña*
Computer Science Department
University of Corunna
Campus de Elviña s/n
15071 La Coruña
Spain.
E-mail: {vilares, grana}@dc.fi.udc.es

D. Cabrero,*T. Araujo*,*I. Diz*
Ramón Piñeiro
Research Center for Humanities
Estrada Santiago-Noia, Km. 3
A Barcia, 15896 Santiago de Compostela
Spain.
E-mail: {taraujo, dcabrero, idiz}@cirp.es

Abstract

In this paper, we introduce a tagger environment for Galician, the native language of Galicia. Galician belongs to the group of Romance languages which developed from the Latin imposed on the north-west of the Iberian Peninsula by the Romans, with additions from the languages of peoples living here before the colonization, as well as contributions from other languages subsequent to the breaking-up of the Roman Empire.
Various historical circumstances led to its not becoming a State language and although it was relegated to informal usage, our vernacular has managed to survive well into the twentieth century when, parallel to the recovery of the institutions for self-government, Galician was once again granted the status of official language for Galicia, together with the Spanish language.
From an operational point of view, our proposal is based on the notion of finite automaton, separating the execution strategy from the implementation of the tagging interpreter. That facilitates the maintenance at the time that assures the robustness of the architecture. Empirical tests prove the validity of our approach to deal with a language whose morphology is non-trivial.

Key Words: Tagging, User Interface, Maintenance.

1. Introduction

Galician is an inflectional language with a great variety of morphological processes, derived from its Latin origin. We can date the birth of the language at the beginning of the Middle Ages. The first written texts date from the 12^{th} century.

During the second half of the 14^{th} century, their producing splendid literature, Galician evolved owing to historical and political reasons towards two different languages: Galician and Portuguese. Throughout the following five hundred years, Galician went through an obscure stage which ended at the beginning of the 19^{th} century. The Spanish war of independence and the continuous political confrontations, animate the revival of autochthonous literature. Although numerous poetic works and prose date from this era, it is only in the second half of this era when Galician rescues the recognition from society that survives till today.

Galician has been the co-official language of Galicia since 1981, with more that two million Galician-speakers in Spain, there are however Galician-speakers in Latin America, and western areas of different regions such as Asturias, León and Zamora. Although there several dialects exist, the Royal Galician Academy, founded in Havana in 1905, has recently standardized the language. This has allowed us to formally study the linguistic phenomena in order to implement a generation environment for tagging.

Section 2 of this work introduces some of the prominent problems dealt with in this paper, briefly introducing the tagging architecture. Section 3 describes the system at work, introducing the most relevant functionalities available. In section 4 we show some interesting practical tests. Finally, section 5 is a conclusion on the work presented.

2. The tagging architecture

Recently ther has been a renewal in the interest for the finite automaton (FA) model for the designing of taggers [2, 4, 5, 6]. This is due to both the speed and compactness of the representations. The

Work partially supported by the Government of Spain under project HF97-223, and by the Autonomous Government of Galicia under projects XUGA10505B96 and XUGA20402B97.

```
sobre
=> ["sobre", (Vps3s0), Verb, Present, Subjunctive, Third, Singular, Gender NA, "sobrar"]
   ["sobre", (Vps1s0), Verb, Present, Subjunctive, First, Singular, Gender NA, "sobrar"]
   ["sobre", (Scms), Substantive common, Masculine, Singular, "sobre"]
   ["sobre", (P), Preposition. "sobre"]
```

Figure 1: Tagger output for `sobre`.

growing complexity of the tagging systems mean that the space required for implementation together with computational efficiency, are important issues for commercial applications.

Galician contains a great variety of morphological processes, particularly non-concatenative ones. At this point, the work of linguists, who put at computer workers' disposal a set of rules with which the analizer can work, is essential. These rules are not simply morphological, since as they are elaborated for computer work, orthographic aspects must be taken into account. For example, the word `mes` (*month*) takes `-es` in the plural, but in some other words such as `rapaz` (*boy*) the addition of the same morphological suffix produces a change in spelling to give `rapaces`. In essence, this work consists of the elaboration of rules which comprise the inflexion of gender and number of adjectives and nouns, verb inflexion, different types of pronoun, etc. Here, we must take into account some important problems:

1. A highly complex conjugation paradigm, with ten simple tenses, including the Infinitive conjugate, all of which have six different persons. If we add the Present Imperative with two forms, not conjugated Infinitive, Gerund and Participle. Then 65 inflected forms are possible for each verb.

2. Irregularities in both verb stems and endings. Very common verbs, such as `facer` (*to do*), have
   up to five different stems: `fac-er`, `fag-o`, `fa-s`, `fac-emos`, `fix-en`. Approximately 30% of Galician verbs are irregular. We have implemented 42 groups of irregular verbs.

3. Verbal forms with enclitic pronouns at the end. This can produce changes in the stem due to the presence of accents: `deu` (*gave*), `déullelo` (*he/she gave it to them*).

   In Galician the unstressed pronouns are usually suffixed and, moreover, pronouns can be easily drawn together and they can also be contracted (`lle + o = llo`), as in the case of `váitemello buscar` (*go and fetch it for him (do it for me)*). It is also very common to use what we call a *solidarity pronoun*, in order to let the listeners be participant in the action. Therefore, we have even implemented forms with four enclitic pronouns, like `perdéuchellevolo` (*he had lost it to him*). Here, the pronouns `che` and `vos` are solidarity pronouns and they are used to implicate the interlocutor in the facts that are being told. None of them has a translation into english, because this language lacks these kinds of pronouns. So, the analysis has to segment the word and return five tokens.

   When elaborating the rules concerning enclitic pronouns, we had to make new rules for the verbs, since many times the addition of a pronoun to the verbal form might cause a change in graphic stress.

4. A highly complex gender inflection, with words with only one gender as `home` (*man*) and `muller` (*woman*), and words with the same form for both genders as `azul` (*blue*). In relation to words with separate forms for masculine and feminine, we have a lot of models:

   `autor`, `autora` (*author*); `xefe`, `xefa` (*boss*); `poeta`, `poetisa` (*poet*); `rei`, `raiña` (*king*) or `actor`, `actriz` (*actor*).

   We have implemented 33 variation groups for gender. All the linguistic possibilities the language offers for nouns and adjectives are included, even the most infrequent and irregular.

5. The inflexion of number is also highly complex, with words only being presented in singular form, such as `luns` (*monday*), and others where only the plural form is correct, as `matemáticas` (*mathematics*). The construction of different forms does not involve as many variants as is the case for gender, but we can also consider a certain number of models:

   `roxo`, `roxos` (*red*); `luz`, `luces` (*light*); `animal`, `animais` (*animal*); `inglés`,

```
perd'euchellevolo
   => ["perd'eu", (Vei3s0), Verb, Perfect, Indicative, Third, SIngular, Gender NA, 4 pronoun(s), "perder"]
   => ["che", (Rad3as), Pronoun atonic, Dative, Second, Masc & Fem, Singular, "che"]
   => ["lle", (Rad3as), Pronoun atonic, Dative, Third, Masc & Fem, Singular, "lle"]
   => ["vos", (Ral2ap), Pronoun atonic, Accusative & Dative, Second, Masc & Fem, Plural, "vos"]
   => ["o", (Raa3ms), Pronoun atonic, Accusative, Third, Masculine, Singular, "o"]
```

Figure 2: Tagger output for `perdéuchevolo`.

> `ingleses` (*english*); `azul`, `azuis` or `funil`, `funís` (*funnel*)
>
> We have implemented 13 variation groups for number.

This complexity suggests the necessity of interfacing the tagging process in order to facilitate the verification of the properties demanded, as well as their maintenance. It was easier to elaborate linguistic rules for other categories whose number of elements is limited, such articles, demonstratives, possessives, indefinites, relatives, interrogatives and exclamatives, adverbs, conjunctions and interjections. In this case, we used the inflexional rules established for nouns and adjectives. Finally, there had been other elements that a linguistic analizer has to cope with which are not usually considered to be word categories. We have established two groups: the first formed by punctuation marks and the second *peripheral categories*, where we include acronyms and abbreviations, signs and formulas, foreign words and all those elements that could never be included in any of the other groups. In order to deal with these groups we have proposed the fields for Galician tokens, together with their possible values, i.e. the tag set represented in Table 1, which is inspired in the EAGLES proposal.

As an example, let's consider the word `sobre`. This word has three possible meanings in Galician: preposition (*on*, *upon*, *over*, *about*), noun (*envelope*) and verb (*to exceed*, *to be unnecessary*). When it is a verb, there are two possible values for the person: first and third. So, the output of the morphological analyzer should contain four taggings: see figure 1. Another interesting output is that obtained for the word `perdéuchellevolo` before commented, where the system must detect the four enclitic pronouns: figure 2.

However, to deal with tagging, morphological analysis is not sufficient. In effect, a tagger must provide a single interpretation for each word, which requires the incorporation of some kind of disambiguation facility. Here three general approaches are possible: rule-based strategies [6], statistically oriented algorithms [7], although it is not clear what is the best approach [1]. Currently, our tagging environment includes a statistic module based on the hidden Markov Model [3]. This permits us to obtain our initial results, although in the future a mixed strategy involving both rules and statistics will be used.

Field	Values	
Word	*The citation form present in the input text.*	
Lemma	*The canonical form of the word.*	
Category	Adjective	*With no type.*
	Adverb	Exclamative, modifier, nuclear, relative, interrogative *and* nuclear & modifier.
	Article	*With no type.*
	Conjunction	Coordinate *and* subordinate.
	Demonstrative	*With no type.*
	Indefinite	*With no type.*
	Interjection	*With no type.*
	Interrogative	*With no type.*
	Numeral	Cardinal, ordinal, partitive *and* multiple.
	Peripheral	Foreign word, formula, symbol, abbreviation, acronym *and* other.
	Preposition	*With no type.*
	Personal Pronoun	Tonic, proclitic atonic *and* enclitic atonic.
	Possessive	*With no type.*
	Punctuation Mark	Dot, comma, colon, semicolon, dash, quotes, open/close question mark, open/close exclamation mark, open/close parenthesis *and* dots.
	Relative	*With no type.*
	Substantive	Common *and* proper.
	Verb	*With no type.*
Subtype	Determiner, non-determiner *and* both.	
Gender	Masculine, feminine, both, neutral *and* non-applicable.	
Number	Singular, plural, both *and* non-applicable.	
Degree	Comparative *and* non-applicable.	
Person	First, second, third,first & third *and* non-applicable.	
Case	Nominative, accusative, dative, accusative & dative, prepositional case *and* nominative & prepositional case.	
Verbal tense	Present, preterite, co-preterite, future, post-preterite *and* non-applicable.	
Mode	Indicative, subjunctive, imperative, infinitive, gerund *and* participle.	

Table 1: Tag set

3. The system at work

On the basis of a classic compilation process from a set of morphological rules, our goal is to make the generation of these rules transparent for the user. In this way, the user can turn his attention to the linguistic information, leaving to the system to resolve most of the problems imposed by the programming task. From the computational point of view, this implies saving in safety as well as a more user-friendly interface.

To achieve this goal, the whole system can be accessed throught a graphic interface. It includes facilities both for building running and testing the

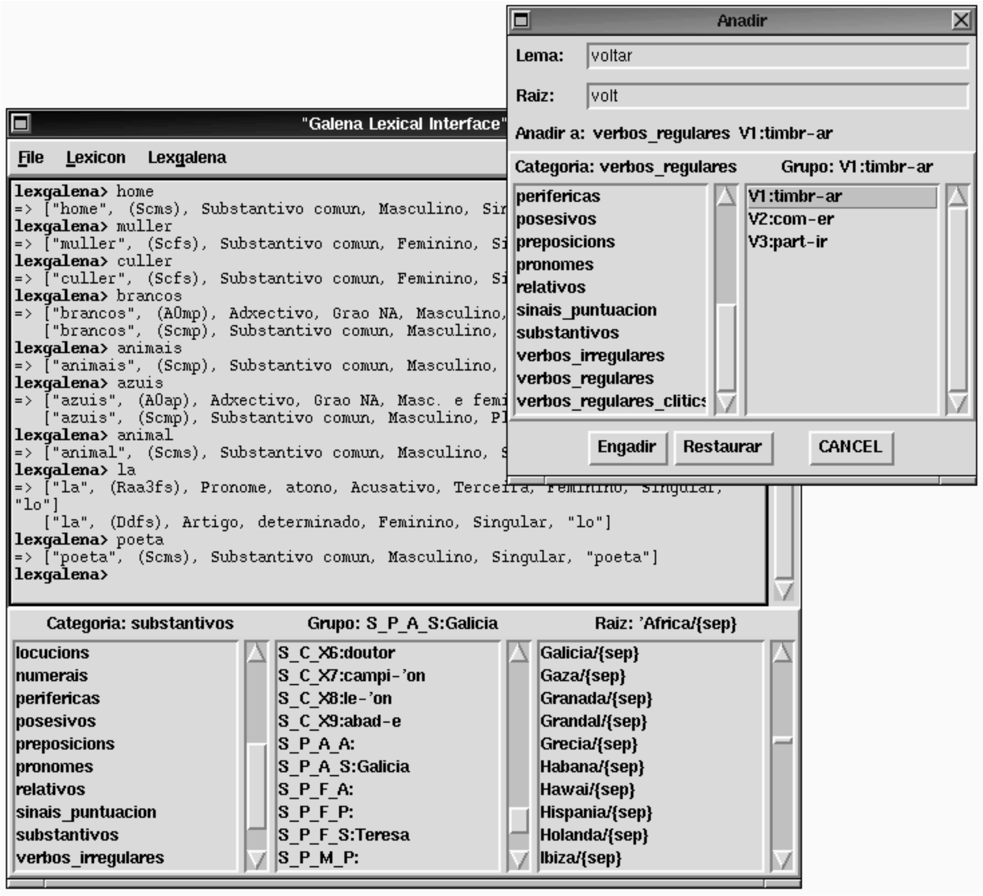

Figure 3: Image of most used interface windows

lexical analyzer, and for adding and removing entries and searching for stems in the lexicon.

In Fig. 3 we show how to add new word-forms to the lexicon (the small window), simply by giving the *lema* and the *stem*[1] and selecting its category and model of inflection from the lists.

The main window shows and allows us to select all the lexicon's categories, the different models of inflection for each category and all the forms included in each model. By selecting one form and clicking the menu option `lexicon/borrar` we can easily remove it.

Other options in the main window include searching all the word-forms whose stems match some prefix and building, from the lexicon we have created, an executable file for the lexical analyzer.

Finally, the text frame redirects what we write to the lexical analyzer and writes back its response. In this way we can use and test the analyzers we build whithout leaving the interface.

As a final remark, Fig. 4 shows that the system also includes a statistical disambiguator if desired. The window allows user to set the parameters of the statistical model by selecting the attributes that are relevant whithin each category. Only attributes which are selected will be used in the trainnig process.

The main improvement in the latter capability is the possibility of fine-tuning the disambiguator, and also of providing a tool for studying the relevance of each attribute in different *corpora*.

4. Experimental results

To illustrate performance we give both information on the current version of our analyzer, and information on the evolution process we expect. As physical support for tests we have taken a *Sun Hyper Sparc Station*.

At present, we are able to recognize and tag the most common 12000 lemmas of Galician. The corresponding automaton has more than 135000 states and the average speed is 1400 words tagged per second. The number of states in the automaton is high, but it will grow slowly because the main inflectional phenomena are already implemented. That is, the only task that remains to be performed is the introduction of more and more stems, and it has been proved that this process yields an average increase of only 2 states for each new lemma.

[1] denoted as *raíz* in the interface.

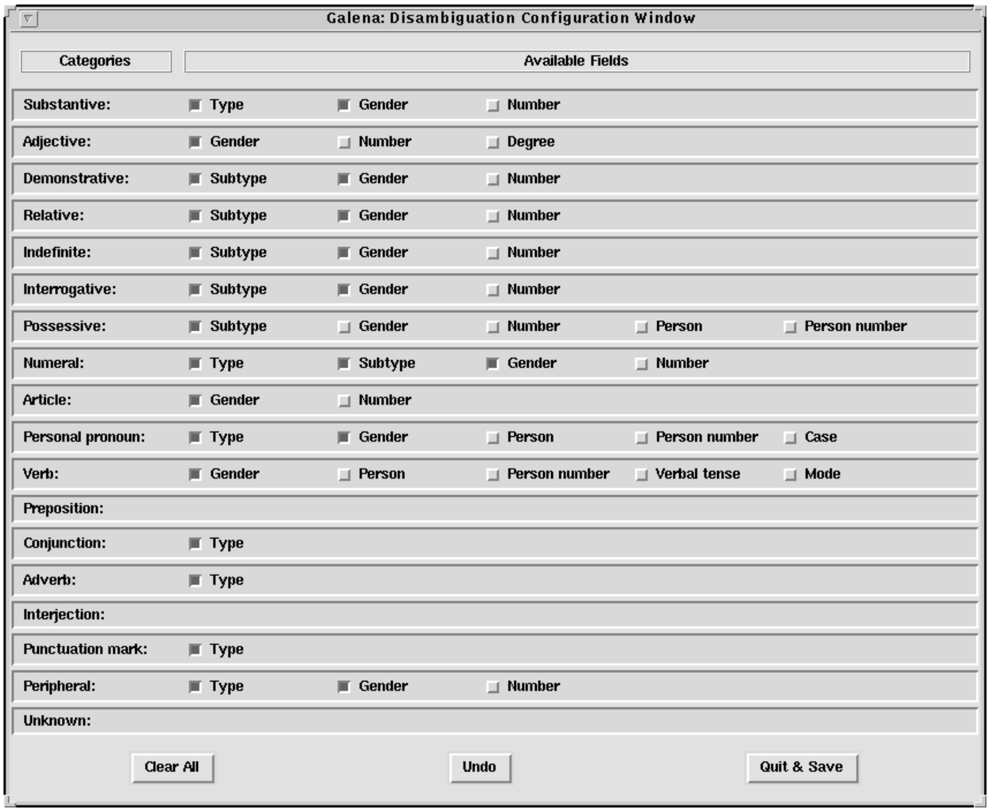

Figure 4: The interfaz for disambiguation

However, unfortunately, compilation time can be very high, which is the price that must be paid when the desired result is high performance. The only way to overcome this obstacle is to implement incremental building processes for automata. This feature is part of our future work.

In order to achieve a better understanding of future sizes and times, we built several analyzers from a large quantity of patterns. These patterns were generated using a random process, but maintaining the same level of ambiguity as in Galician words,and the results are in Fig. 5. These tests show that the proposed architecture for the tagger presents a linear time and space complexity.

5. Conclusion

The design of tagging systems should respond to the constraints of efficiency, safety and maintenance that we have considered from a practical point of view. The choice of the FA model as operational formalism assures computational efficiency. Safety is guaranteed by the separation which exists between this operational kernel and the high-level descriptive formalism.

The work described above is not yet a finished research-line. It represents only an initial approach to the problem of tagging, but preliminary results seem to be promising and the operational formalism well adapted to deal with more complex problems such as the conideration of error recovery algorithms, and the development of disambiguation techniques.

6. References

[1] J.P. Chanod and P. Tapanainen. Creating a tagset, lexicon and guesser for a French tagger. In *ACL SIGDAT Workshop on From Texts to Tags: Issues in Multilingual Language Analysis*, pages 58–64, University College, Dubli, Ireland, 1995.

[2] K. Koskenniemi. Compilation of automata from morphological two-level rules. In *Proc. of the* 5^{th} *Scandinavian Conference of Computational Linguistics*, pages 143–149, Helsinki, Finland, 1985.

[3] J.M. Kupiec. Robust part-of-speech tagging using hidden Markov model. *Computer Speech and Language*, 6:225–242, 1992.

[4] G. Ritchie. On the generative power of two-level morphological rules. In *European Chapter of the ACL*, pages 51–57, Manchester, 1989.

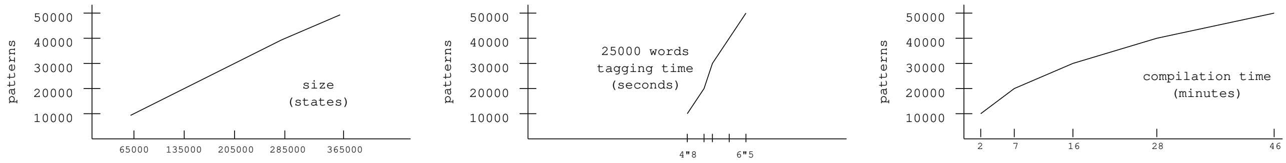

Figure 5: Experimental results

[5] G. Ritchie, D. Pulman, Stephen, A.W. Black, and G.J Russell. *Computational Morphology*. The MIT Press, Cambridge, Massachusetts, U.S.A., 1991.

[6] E. Roche and Y. Schabes. Deterministic part-of-speech tagging with finite-state transducers. *Computational Linguistics*, 21(2):227–253, 1995.

[7] Ralph Weischedel, Marie Meteer, Richard Schwartz, Lance Ramshaw, and Jeff Palmucci. Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19(2):359–382, June 1993.