Tagging and Shallow Processing of Portuguese: workshop notes of TASHA'2003

ANTÓNIO BRANCO
AMÁLIA MENDES
RICARDO RIBEIRO
(editors)

DI-FCUL TR–03–28

OCTOBER 2003

Departamento de Informática
Faculdade de Ciências da Universidade de Lisboa
Campo Grande, 1700 Lisboa
Portugal

Technical reports are available at http://www.di.fc.ul.pt/tech-reports. The files are stored in PDF, with the report number as filename. Alternatively, reports are available by post from the above address.

Foreword

Recent advances on tagging and other shallow processing tools have confirmed the key importance of these applications to enable subsequent stages of efficient deep linguistic processing, and to improve the quality of information retrieval services and support the semantic web.

The Workshop on Tagging and Shallow Processing (TASHA'2003) took place in Lisbon, October 3, 2003, at the Faculdade de Letras de Lisboa. It was an associated event of the XIX Encontro Anual da Associação Portugesa de Linguística and its goal was to provide a forum to discuss experiences and exchange results between researchers working in this area, and to review the current state of the art in this diverse field in what concerns the Portuguese language.

The present collection of notes includes the abstracts of papers selected to be presented at the workshop. These abstracts are extended versions of the abstracts originally submitted and take into account the remarks offered by the reviewers, to the extent that this was possible by the space constraints of the present publication. We would like thus to thank all the colleagues that contributed to the workshop with their submissions to be anonimously appreciated by the program committee. We are also grateful to the colleagues that took part in the program committee and helped in the selection process:

Alina Villalva (Univ. Lisboa, Dep. Linguística, Portugal)
Amália Mendes (Univ. Lisboa, CLUL, Portugal)
António Branco (Univ. Lisboa, Dep. Informática, Portugal)
Caroline Hagège (Xerox Research Centre, France)
Jorge Baptista (LabEL / Univ. Algarve, Portugal)
Fernanda Bacelar Nascimento (Univ. Lisboa, CLUL, Portugal)
Gaël Dias (Univ. Beira Interior, Dep. Informática, Portugal)
Marco Rocha (Univ. Federal de Santa Catarina, Dep. Linguística, Brazil)
Nuno J. Mamede (L2F INESC-ID Lisboa / IST, Portugal)
Nuno Marques (Univ. Nova de Lisboa, Dept. Informática, Portugal)
Renata Vieira (Univ. Vale do Rio dos Sinos, Dep. Informática, Brazil)
Rute Costa (Univ. Nova de Lisboa, Dep. Linguística, Portugal)
Tony Sardinha (Univ. Católica de São Paulo, Dep. Linguística, Brazil)
Vera Strube de Lima (Pontifícia Univ. Católica do Rio Grande do Sul, Fac. Informática, Brazil)

Finally, we would like to thank to São Luís Castro, from the Universidade do Porto, for her kind willingness to contribute with an invited talk.

Lisbon, September 17, 2003.

The editors:

António Branco, Amália Mendes and Ricardo Ribeiro.

Table of Contents

Lexical Learning for Attachment Resolution

Alexandre Agustini, Pablo Gamallo, Gabriel Pereira Lopes

CITI - Center of Informatics and Information Technology
Department of Informatics
Faculty of Sciences and Technology
New University of Lisbon, Portugal
{aagustini, gamallo, gpl}@di.fct.unl.pt

Abstract

This paper describes a procedure based on attachment resolution for evaluating an unsupervised strategy to acquire both nominal and verbal subcategorisation information. The notion of subcategorisation is based on two specific linguistic assumptions. First, it is assumed that two dependent words impose requirements on each other. Second, it is also claimed that a linguistic requirement may be extensionally defined as the set of words that can occur in similar syntactic contexts. The main aim of the learning strategy is to cluster similar syntactic contexts by identifying the words that define extensionally their linguistic requirements. The subcategorisation information acquired is used to constrain attachment heuristics in a parsing task. The evaluation method of this task is described in the article.

1. Introduction

Partially parsed corpora are used for word classes learning. Word classes are associated here to the words that can appear in specific contexts of subcategorisation. In this paper we present how the parsing is used in order to check if the syntactic and semantic information automatically extracted leads to better parses.

The learning method has been accurately described in (Gamallo et al., 2003; Gamallo et al., 2002). For the purpose of this paper, we only outline the basic assumptions on which the acquisition strategy is based, as well as different steps and modules that are involved in it (section 2). In section 3, we describe how the learned information is used to characterise attachment heuristics, and finally, we evaluate the performance of these heuristics.

2. Learning Word Classes from Subcategorisation Contexts

2.1. Basic Assumptions

Our learning method is based on two theoretical assumptions: one is based on word co-composition and the other on context similarity.

First, we consider that in a Head-Modifier syntactic dependency, not only the Head imposes constraints on the Modifier, but the Modifier also imposes linguistic requirements on the Head in return. This idea stems from the Pustejovsky's "co-composition" hypothesis (Pustejovsky, 1995). So, for a particular word, we attempt to learn what type of both modifiers and heads it subcategorises. For instance, consider the compositional behavior of the noun `republic` in a domain-specific corpus. On the one hand, this word appears in the Head position within dependencies such as `republic of Ireland`, `republic of Portugal`, and so on. On the other hand, it plays the role of Modifier in dependencies like `president of the republic`, `government of the republic`, etc. So, given the word `republic`, we attempt to learn both what kind of complements and what kind of heads it specifies (subcategorises). As there are interesting semantic regularities among the words cooccurring with `republic` in the above expressions, the aim of our learning method is to extract two different subcategorisation contexts:

> ***<of,republic,[M]>***, where position *M* must be filled by words referring to particular nations or states. Indeed, in semantic-pragmatic terms, only nations or states can be republics;
>
> ***<of,[H],republic>***, where position *H* must be filled by head words denoting specific parts of the republic: e.g., institutions, organisations, functions, and so on.

The second assumption concerns the procedure for identifying and clustering similar subcategorisation contexts. We assume, in particular, that different contexts are considered to be semantically similar if they have similar word distribution (Faure and Nédellec, 1998). Let's take, for instance, the following set of contexts[1]:

{ <of,[H],republic>,
<of,[H],state>,
<of,delegate,[M]>
<iobj_by,sign,[M]>,
<iobj_on,be_incumbent,[M]> } (1)

These contexts share the same semantic preferences provided they require words denoting the same semantic class. Contexts with the same semantic preferences are likely to possess similar word distribution. Moreover, we also assume that the set of words required by similar subcategorisation contexts represents the extensional description of their semantic preferences.

[1] *iobj_prepname* designates the prepositional complement (*iobj*) of verbs occurring with preposition *prepname*

2.2. Method Overview

Our learning method consists of the following steps. Raw Portuguese text is automatically tagged (Marques and Lopes, 2001) and partially analysed in sequences of basic chunks (Rocio et al., 2001). Then, binary syntactic dependencies are identified on the basis of Right Association attachment heuristics. Then, we extract subcategorisation contexts from the binary dependencies, by following the first assumption outlined above (co-composition). Finally, subcategorisation contexts with similar word distributions are clustered into more general classes (second assumption). Similarity between contexts is calculated by using a particular version of the Lin coefficient (Lin, 1998). For instance, the class illustrated above in (1) is constituted by contexts considered as similar. These contexts have as features those words cooccurring at least once with them, e.g.:

president, assembly, minister, ministry, government, administration.

This is the way we build classes of words representing the semantic preferences of similar contexts subcategorisation restrictions.

3. Application: Attachment Resolution

We ran our learning strategy over a Portuguese corpus with 1,643,579 word occurrences, selected from the P.G.R.[1] text corpora and 16.274 word classes were extracted.

The acquired classes are used to provide the lexicon with subcategorisation information. For the entry `secretário` (*secretary*), for instance, we have learned among other clusters, the information:

`<iobj_a,[H],secretário>=`
`caber,competir,conceder,conferir, confiar,dirigir,…`
(*concern, be-incumbent,concede,confer,trust,send*) (2)

The syntactic and semantic subcategorisation information provided by the lexical entries is used to improve the parsing task. Co-composition is at the centre of attachment resolution. It is used to characterise the main attachment heuristic. This heuristic states that two chunks are syntactically attached only if one of these two conditions is verified: either the *Modifier* is semantically required by the *Head*, or the *Head* is semantically required by the *Modifier* (details of the symbolic grammar with information on linguistic co-composition can be found in (Agustini et al., 2003; Gamallo et al., 2003). Take the expression:

(a) `... compete ao secretário ...`
(*is incumbent on the secretary*)

This expression will be analysed as a VP-PP construction if one of the two following requirements is satisfied:

requirement *M*: context <`iobj_a,competir,[M`]> (*be-incumbent on [M]*) subcategorises a class of nouns to which `secretário` (*secretary*) belongs;

requirement *H*: context <`iobj_a,[H],secretário`> (*[H] on the secretary*) subcategorises a class of verbs to which `competir` (*be-incumbent*) belongs.

According to the lexical information illustred in (2), the expression (a) can be analysed as a VP-PP construction because, at least, requirement *H* is satisfied. Note that, even if we have no information on the verb subcategorisation, the attachment is allowed because of the noun requirements in the *H* position. Co-composition is also used to solve long-distance attachments (Gamallo et al., 2003).

3.1. Evaluating Performance of Attachment Resolution

We evaluated the performance of the parsing strategy based on co-requirements. The general aim of this evaluation is to check whether the subcategorisation information we have learnt is adequate to be used in a parsing task. The degree of efficiency in such a task may serve as a reliable evaluation for measuring the soundness of our learning strategy.

3.1.1. Test Data

Most work on attachment resolution (Hindle and Rooth, 1993; Ratnaparkhi at al., 1994; Collins and Brooks, 1995; Li and Abe, 1998; Niemann, 1998; Pantel and Lin, 2000) uses as test data expressions with three basic phrases (or chunks): *vp-np-pp*. These approaches consider that each expression selected for evaluation can be syntactically ambiguous in two ways. For instance, the partial parse:

(b) [$_{vp}$ cut] [$_{np}$ the potato] [$_{pp}$ with a knife]

can be disambiguated either by the parse:

(c) [$_{vp}$ cut [$_{np}$ the potato [$_{pp}$ with a knife]]]

which represents a syntactic configuration based on proximity (phrase1 is attached to phrase2 and phrase2 is attached to phrase3), or by:

(d) [$_{vp}$ cut [$_{np}$ the potato] [$_{pp}$ with a knife]]

which is here the correct configuration. It contains both a contiguous and a long distance attachment: phrase1 is attached to phrase2 and phrase1 is attached to phrase3.

We consider, however, that the process of attachment resolution should be generalized to other syntactic sequences and ambiguity configurations. Our test data consists of 633 expressions which have been selected randomly from a test corpus. These expressions

[1] P.G.R. (Portuguese General Attorney Opinions) corpora is constituted by case-law documents.

are not only *vp-np-pp* sequences of phrases. They were divided in three groups according to three different syntactic sequences. Moreover, they cannot be reduced to only two syntactic configurations (two parses). They can be syntactically ambiguous in different ways: adjective arguments and sentence adjuncts (see table 1).

Table 1 shows test expressions that cannot be analysed by means of the two standard configurations underlying parses (c) and (d). None of the expressions in that table matches the two standard configurations. For instance, `ao decreto` (to the decree), which is the phrase2 of the first example, is not attached to the head of phrase1 but to the adjective `relativo` (referring). Similarly, in the second expression, `ao citado diploma` (to the cited diploma) is attached to the adjective `anexos` (joined) and not to the head of phrase2. In Latin languages, the subcategorisation of adjectives introduces a new type of structural ambiguity, which makes attachment decisions more difficult to be taken. Finally, in the third expression, `na medida` (in the sense) is the beginning of an adverbial sentence, so it is not attached to one of the individual phrases but to the whole previous sentence. This phenomenon is not specific to Latin languages. In sum, solving structural ambiguity cannot be reduced to a binary choice between the two configurations depicted above in (c) and (d).

[np o artigo relativo] [pp ao decreto] [pp da lei] (*the article referring to the decree of the law*)
[vp publicou] [pp nos estatutos anexos] [pp ao citado diploma] (*published in the statutes joined to the cited diploma*)
[vp tem] [np acesso] [pp na medida] (*has access in the sense*)

Table 1: The three syntactic sequences evaluated

Other important aspect of the evaluation is the over generation of attachments. When the three phrases are semantically related, our method proposes three attachments even if only two of them are syntactically allowed. For instance, take the expression:

(e) [np a remuneração] [pp do cargo] [pp de secretario]
(*the salary of the post of secretary*)

which would be correctly analysed by using the same configuration of parse (c) above, i.e.:

(f) [np a remuneração [pp do cargo [pp de secretario]]]

However, there is also a strong semantic relation between phrase1 (`remuneração`) and phrase3 (`de secretário`), even if they are not syntactically attached in (f). Taking into account the semantic requirements stocked in the dictionary (for instance that presented on (2)), our method is induced to propose, besides the two correct attachments, a long distance dependency, which is not correct in this particular case. We call this phenomenon "attachment over generation". Over generation appears only if an expression contains a semantic relation between two phrases that are actually not syntactically related. Attachment over generation was found in about 10% expressions selected from the test corpus. In order to overcome this problem, we use a default rule based on Right Association. The default rule removes the long distance attachment and only proposes the two contiguous ones. This simple rule has an accuracy of more than 90% with regard to the 10% expressions containing over generation.

3.1.2. Baseline

Concerning the ability to propose correct syntactic attachments, we made a comparison between our method and a baseline strategy. As a baseline, we used the attachments proposed by Right Association. That is, for each expression of the test data, this strategy always proposes the configuration underlying parses (c) and (f), that is: phrase1 is attached to phrase2, phrase2 is attached to phrase3, and phrase1 is not attached to phrase3.

3.1.3. Results

Table 2 reports the test scores concerning the precision and recall of the two comparative experiments performed. We call *precision* the number of correct attachments suggested by the method divided by the number of total suggestions. *Recall* is computed as the number of correct attachments suggested by the method divided by the attachments that are actually correct in the sample.

The baseline scores are very informative concerning the type of syntactic configurations we have found. Precision informs us that about 72% are attachments by proximity. Recall means that about 22% are attachments between phrase1 and phrase3. So, the remainder, about 6%, are other kinds of attachments: adjective subcategorisation, sentence adjuncts, ... The total precision of our method reaches more than 90%, whereas the total recall is about 74%. These results can be hardly compared to related approaches given that:

- there is no related work on Portuguese;
- our test corpus is not restricted to the two standard ambiguity configurations defined above; we also take into account expressions containing adjective attachments and sentence adjuncts;
- we use three types of phrase sequences, and not only the *vp-np-pp* sequence used by most related work.

This makes it difficult to compare the performance of our method to other unsupervised strategies. We consider, however, that the recall we have obtained should be higher. In order to improve it, we need to provide the dictionary with more items of subcategorisation information. One of the main challenges of our current work is to tune the clustering constraints so as to reach high recall by keeping a reasonable precision. This should make the parser more efficient concerning the attachment resolution task.

BASELINE		
Syntactic sequences	Precision (%)	Recall (%)
np-pp-pp	70.81	78.93
vp-pp-pp	71.90	77.83
vp-np-pp	75.49	79.22
Total	**72.74**	**78.66**
CO-COMPOSITION		
Syntactic sequences	Precision (%)	Recall (%)
np-pp-pp	87.15	74.13
vp-pp-pp	92.23	70.36
vp-np-pp	94.23	76.36
Total	**91.20**	**73.62**

Table 2: Evaluation of attachment resolution

4. Conclusion and Future Work

This paper has presented the evaluation of a particular unsupervised strategy to automatically acquire syntactic and semantic subcategorisation requirements. Our strategy is mainly based on two linguistic assumptions: First, it was assumed that not only the syntactic Head imposes restrictions on its Dependent word, but also the latter selects for a specific type of Head. This phenomenon was called "co-requirement". Second, we claimed that similar syntactic contexts share the same selection requirements. So, we measured, not similarity between words on the basis of their syntactic distribution, but similarity between syntactic contexts on the basis of their word distribution. It was assumed that the latter kind of similarity conveys more pertinent information on linguistic subcategorisation than the former one. The learning process allowed us to provide the lexicon with both syntactic and semantic subcategorisation information. This information was used to constrain attachment heuristics.

In future work, our aim is to extend the subcategorised lexicon in order to increase the coverage of the parser. For this purpose, we will use the partial results of the parser to discover new subcategorisation information. That is, the new long distance attachments identified by the parser will serve to learn more syntactic and semantic restrictions. The lexicon will be provided with these new restrictions and thereby the coverage of the parser will be increased. The successive "learning + parsing" cycles will stop as no more new information is acquired and no more new dependencies are proposed.

5. References

Agustini, A., Gamallo, P. and Lopes, G.P., 2003. Selection restrictions acquisition for parsing improvement. In U. Geske, editor, *Selected articles of 14th International Conference of Applications of Prolog (INAP2001)*. Berlin: Springer Verlag.

Collins, M. and Brooks, J., 1995. Prepositional phrase attachment through a backed-off model. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 27–38, Cambridge.

Faure, D. and Nédellec, C., 1998. Asium: Learning subcategorization frames and restrictions of selection. In *ECML98, Workshop on Text Mining*.

Gamallo, P., Agustini, A., and Lopes, G.P., 2003. Learning subcategorisation information to model a grammar with co-restrictions. *Traitement Automatic de la Langue*, 41(1).

Gamallo, P., Agustini, A., and Lopes, G.P., 2002. Using co-composition for acquiring syntactic and semantic subcategorisation. In *ACL-SIGLEX'02*, Philadelphia, USA.

Hindle, D. and Mats, R., 1993. Structural ambiguity and lexical relations. *Computational Linguistics*, 19(1): 103--120.

Li, H. and Abe, N., 1998. Word clustering and disambiguation based on co-occurrence data. In *Coling-ACL'98*, pages 749--755.

Lin, D., 1998 Automatic retrieval and clustering of similar words. In *COLING-ACL'98*, Montreal.

Marqes, N. and Lopes, G.P., 2001. Tagging with small training corpora. In F. Hoffmann, D. Hand, N. Adams, D. Fisher, and G. Guimaraes, editors, *Advances in Intelligent Data Analysis*. LNCS, Springer Verlag, pages 62–72.

Niemann, M., 1998. Determining pp attachment through semantic associations and preferences. In *ANLP Post Graduate Workshop*, pages 25–32.

Pantel, P. and Lin, D., 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *ACL'00*, pages 101–108, Hong Kong.

Pustejovsky, J., 1995. *The Generative Lexicon*. MIT Press, Cambridge.

Ratnaparkhi, A., Reymar, J. and Roukos, S., 1994. A maximum entropy model for prepositional phrase attachment. In *Proceedings of the ARPA Human Language Technology Workshop)*, pages 250--225.

Rocio, V., Clergerie, E., and Lopes, G.P., 2001. Tabulation for multi-purpose partial parsing. *Journal of Grammars*, 4(1).

Flexible Module for Shallow Parsing, Using Preferences

Fernando M. Batista*, Nuno J. Mamede†

*L²F /INESC-ID /ISCTE †L²F /INESC-ID /IST
Spoken Language Systems Lab
Rua Alves Redol 9, 1000-029 Lisboa, Portugal
{Fernando.Batista, Nuno.Mamede}@inesc-id.pt

Abstract

This paper presents a shallow parsing module – SuSAna – that performs efficient analysis over unrestricted text. The module recognizes the boundaries, internal structure, and syntactic category of the syntactic constituents. In addition to the definition of syntactic structures, its grammar supports a hierarchy of symbols and a set of restrictions known as preferences. During the analysis, a directed graph is used for representing all the operations, preventing redundant computation. The algorithm has $O(n^2)$ complexity, where n is the number of lexical units in the segment. SuSAna can be used as a standalone application, fully integrated in a larger system for natural language processing, or in a client/server platform.

1. Introduction

The syntactic analysis of a corpus returns information otherwise hidden, allowing the development of more powerful and complex applications. The syntactic processing of corpora may be applied to areas such as information retrieval, information extraction, speech synthesis and recognition (Marcus Fach, 1999) and automatic translation. Syntactic analysis is also frequently the starting point for semantic processing systems.

The shallow parsing module, SuSAna (Surface Syntactic Analyzer), performs efficient analysis over unrestricted text. The development of the module is based on the work of Caroline Hagège (2000), and recognizes, not only the boundaries, but also the internal structure and syntactic category of syntactic constituents. Its grammar supports a hierarchy of symbols and a set of restrictions known as *preferences* (Tomek Strzalkowski, 1994), in addition to the definition of the syntactic structures. During the analysis, a directed graph is used for representing all the operations, preventing redundant computation. The algorithm has $O(n^2)$ complexity, where *n* is the number of lexical units in the segment. SuSAna can be used as a standalone application, fully integrated in a larger system for natural language processing, or in a client/server platform.

2. The knowledge base

The structures SuSAna identifies, known as *models*, are defined from a set of properties. In the scope of the analysis, morphossyntactic categories are also viewed as models, thus the concepts of *terminal model* and *non-terminal model* are used to distinguish the categories from the models.

The grammar structure defined for SuSAna has been adapted and improved from the grammar used by the shallow parsing prototype AF (Caroline Hagège, 2000). This grammar uses three different structures for representing all the lexical information: block structures define the behavior of models inside other models; *preferences* are used for choosing between different interpretations, according to confidence levels; and a symbol hierarchy, that alllows to define classes and subclasses of models, leading to a clear and reduced number of rules.

Besides preferences, SuSAna makes use of psycholinguistic principles (Daniel Jurafsky and Martin, 2000; Allen, 1995), for choosing between different interpretations that the parser might be able to find. Currently, the module uses the longest model principle, which states that all other things being equal, new constituents tend to be interpreted as being part of the constituent under construction rather than part of some constituent higher in the parse tree. In the future other psycholinguistic principles, such as minimal attachment and right association, may be applied.

3. Algorithm and internal organization

3.1. Architecture

The overall analysis process is performed in two stages. The first stage consists of generating the information concerning the input data and storing it into a repository. The repository will then provide, in a second stage, all the information required for producing the desired output. As shown in Figure 1, the analysis and extraction tasks are performed independently and can be independently parametrized. Besides providing all required data to the extraction module, the repository saves information about previous calculations, thus preventing redundant computation.

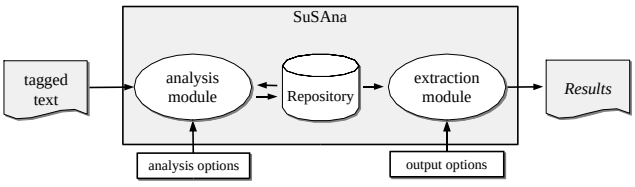

Figure 1 - SuSAna's internal architecture.

3.2. The algorithm

In order to cover unusual linguistic constructions, the algorithm finds all possible sequences for the analysis during the first phase, then selects the most promising ones, either according to preferences or by applying psycholinguistic principles.

The analysis of a given sentence is represented using an in-memory DAG (*Directed Acyclic Graph*). Each vertice of the graph is associated with a lexical unit of the sentence and contains information about the occurrence of a model inside other model, in that position of the sentence. The DAG makes use of two types of edges, one for specifying child vertices and the other for specifying sibling vertices. Each edge has an associated cost, given by the preferences specified in the grammar. The analysis consists of, being at a given vertice, finding all possible child vertices and, when done, finding all sibling vertices. Whenever possible, the algorithm reuses previously calculated analysis fragments, achieving results faster.

Selection of the most promising paths consists of ranking paths from the starting point of the graph, based on the cost associated with each edge and on the longest models principle. The full paper will describe the employed strategy in detail.

The algorithm is robust, in the sense that it can skip unexpected, or out of context, lexical units and reduce as much as possible the number of hypotheses for each analysis, thus providing output suitable for further processing. Special grammar rules may be introduced, in order to increase the robustness.

4. Parametrization

The previously presented architecture allows a flexible way of setting analysis and extraction options. In what concerns analysis options, one of the most important is the possibility of defining the starting model, overriding the default one, during execution. Another important option is the possibility of skipping untreatable lexical units at the beginning and at the end of the analysis, making it possible to find the best solution without considering those words. This option can be used to find large linguistic structures in the segment when boundaries are not feasible. By default, each segment corresponds to a linguistic structure. However, it is possible to search for multiple linguistic structures in a segment, allowing, for example, the identification of sentences in a paragraph. This option can be used simultaneously with the option for skipping models, in order to extract all the linguistic structures of some type in a given segment.

Another interesting option for SuSAna is the ability to process incomplete structures. This is useful when there are no solutions and the user wants to know the largest analysis found. This can also be applied to guess, for an incomplete sentence, the categories that may follow the last lexical unit, so that the sentence remains correct according to the grammar.

5. Evaluation

In what concerns linguistic correctness, at the moment, only small tests have been performed, but they show promising results. The grammar currently in use was written by Caroline Hagège (2000) for extracting noun phrases. Linguistic phenomena, such as verb phrases, are superficially treated, preventing a full linguistic evaluation of the system. Nevertheless, comparisons between SuSAna and AF show better accuracy for SuSAna.

Tests were conducted over a corpus of about 4.6 million words, consisting of two months of the newspaper Público (Batista, 2003). In what concerns performance results in terms of processing time, SuSAna performed all the analyses at an average of about 300 words/second[1]. In what concerns coverage, 61.6% - 97.7% of the lexical units were covered by the analysis process, depending on the performed test. The value 61.6% corresponds to identifying the structure of previously segmented text, considering that each word was correctly placed in the segment. Using SuSAna to segment the corpus, 97.7% of the lexical units were covered.

6. References

Allen, J. (1995). *Natural Language Undestanding*. Benjamin/Cummings, Redwood City, CA, 2nd edition.

Batista (2003). *Análise Sintáctica de Superfície*. MSc Thesis. Universidade Técnica de Lisboa – Instituto Superior Técnico. Lisbon, Portugal. July 2003.

Fach, M. (1999). A comparision between syntactic and prosodic phrasing. In *proceedings of Eurospeech 1999*, volume 1, pages 527–530, Budapest.

Hagège, C. (2000). *Analyse Syntaxique automatique du portugais.* PhD thesis, Laboratoire de Recherche sur le Language, Université Blaise Pascal, Clermond-Ferrand, GRIL.

Jurafsky, D. and Martin, J. H. (2000). *Speech and Language Processing.* Prentice Hall.

Strzalkowski, T., editor (1994). *Reversible Grammar In Natural Language Processing*. Kluwer Academic Publishers, Boston, London.

1 Intel Pentium III processor at 800 Mhz. Linux operating system

Portuguese Specific Issues in the Rapid Development of State of the Art Taggers

António Branco and João Silva

Department of Informatics, University of Lisbon
Faculdade de Ciências, Campo Grande, 1749-016 Lisboa
{ahb,jsilva}@di.fc.ul.pt

Abstract

The application of general purpose machine learning techniques to natural language part of speech tagging has matured to a point where it is now quite rapid to develop new taggers. In the present paper, we report on solutions we adopted for the specific issues that arise when developing a new automatic tagger for Portuguese and are generic enough to be further reused to develop other new taggers for this language, possibly by using other training data.

1. Introduction

Lexemes with the same syntactic distribution are grouped together and assigned the same part-of-speech (POS) tag (e.g. `Noun`, `Adjective`, `Preposition`, etc.). Many lexemes belong to more than one such distributional grouping thus implying that many lexeme-types bear more than one tag in the lexicon and that the correct tag for each of their lexeme-tokens has to be decided given the specific occurrence at stake.

From a computational point of view, the non trivial issue with respect POS tagging consists in deciding for each token of a lexeme in a text, and from the set of admissible POS tags for its type in the lexicon, which tag is the correct one to be assigned to that lexeme in that specific occurrence. Though apparently simple when synthetised under these terms, POS tagging is a very important step in natural language processing inasmuch as it permits to cut down a considerable amount of ambiguity present in natural language utterances at a quite early stage of processing, even before the subsequent, and computationally expensive stages of syntactic and semantic processing.

The application of general-purpose machine learning techniques to natural language POS tagging has matured to a point where it is now quite rapid to develop new taggers. As a matter of fact, when using the applications making use of such techniques to develop a new tagger, the time span needed to set it up is determined basically by the language-specific issues that have to be dealt with. Such issues are found in each of the three major steps involved in automatic tagging raw text: chunking, tokenizing and tagging stricto sensu.

In the present paper, we report on solutions we worked out for the specific issues that arise when developing a new automatic tagger for Portuguese and are generic enough to be further reused with benefit to develop other new taggers for this language from other training data.

2. Chunker

As in other languages with orthographic conventions similar to those adopted for Portuguese, designated punctuation symbols ('`.`', '`?`', '`!`',...) are used to mark the end of sentences. Most sentence boundaries can then be detected when these terminators precede sentence starters, i.e. designated orthographic clues marking the beginning of a subsequent sentence (viz. word beginning with a capital letter) — the expected abbreviation/period ambiguity of '`.`' can be addressed by means of the solutions proposed in the literature for other languages (Mikheev, 2002).

Conventions for sentence bounding that are specific to Portuguese, or at least not found in other close Romance languages or English under exactly the same format, involve the marking of paragraph (turn taking) and sentence boundaries in written dialogue.

The beginning of the first sentence containing a character's turn is easily handled as this starts with a dash ('`-`') immediately followed by the usual sentence starters.

```
<s> - Bom dia! </s>
```

Things get convoluted, however, when it comes to narrator's asides: the beginning of a narrator's aside is always indicated by a dash but its ending is also indicated by a dash only in the cases where the aside does not conclude the sentence.

```
<p><s> - Apetece-me ir ao cinema – anunciou ele. </s></p>
<p><s> - Eu cá - disse ela – também quero. </s></p>
```

Taking narrator's asides into account, it is worth noting that a character's sentence other than the first one in the current turn starts also with a dash exactly in the cases where such sentence follows a character's sentence ending with a narrator's aside.

```
<p><s> - Não - disse ela. </s><s> – Eu não. </s></p>
```

As for termination symbols of character's utterances, only those that are different from a period appear before the beginning of a narrator's aside.

```
<s> - Bom dia! – exclamou. </s>
```

Other hard cases involve the determination of sentence/paragraph boundaries indicated by starters of

enumerated lists and quotation delimiters and by the starter/terminator ambiguity of ellipsis ('. . .').

These issues will be discussed in detail in the presentation and a systematic procedure to handle them will be outlined. For this procedure, we scored a recall of 99.94% and precision of 99.93% when tested on a 12 000 sentence corpus accurately hand tagged with respect to sentence and paragraph boundaries.

3. Tokenizer

For most tokens in a raw text, tokenization is a trivial procedure, consisting in detaching punctuation marks and taking advantage of the whitespace as a delimiter symbol. There are, however, a few non-trivial cases (complete list to be presented at the workshop) that involve tokenization-ambiguous strings, i.e. strings that can be tokenized in more than one way.

`deste -> |deste|` or `deste -> |de|este|.`

In a general setup like ours, where one counts on a tagger trained over previously annotated data, this inevitably introduces circularity that has to be resolved: Although all tagging decisions require previous tokenization, the tokenization of these ambiguous strings requires previous knowledge of the POS tag of the token(s) corresponding to the string. In the example above, we would tokenize `deste` as one token only if it had been tagged as a Verb, but for it to be tagged as a Verb it should have already been tokenized as one token.

To resolve these cases, we used a two-level approach to tokenization where tagging is interpolated into the tokenization process, which has now two stages, one before and another after the tagger has been applied. Accordingly, (i) a pre-tagging tokenizer definitely identifies every token except those related to ambiguous strings: These strings are provisionally identified as one token.

(ii) Subsequently, the tagger assigns a composite or a simple tag to every ambiguous string depending on it being a contracted or a non-contracted form, respectively: The tagger has been trained over a corpus where ambiguous strings are always tokenized as a single token and annotated with single or composite tags.

(iii) Finally, a post-tagging tokenizer handles only ambiguous strings, breaking those that are tagged with a composite tag into two tokens and the corresponding tags.

In our corpus, the ambiguous strings amount to 2% of the tokens. This two-level tokenization approach permitted to successfully resolve 99.4% of these ambiguous cases, against a baseline of 78.2% of success, which is obtained by tokenizing every such ambiguous string as two tokens in every occurrence (as 78.2% of the ambiguous strings were contractions in our text corpus).

4. Tagger

For the development of the Portuguese tagger strictu sensu, we used the TnT software, a Hidden Markov model based application developed and kindly granted to us by Thorsten Brants (Brants, 2000). When using a machine-learning tool like this to develop a new tagger, the critical issues are to be found in the gathering of appropriate training data. Assuming that the consistency and accuracy of the annotation of the general purpose training corpus used as a starting point is ensured, the main concern is directed towards manipulating and relabeling it in accordance with the tag set that needs to be opted for. The design of the latter turns out thus to be the non-trivial aspect that calls to be addressed.

In this respect, one finds the usual tension between increasing the discriminative power of the tagger — by using more tags — and minimizing the data sparseness — by using fewer tags. Looking for the best performance of a POS tagger supported by a suitably tuned balance of these two attractors cannot be reduced, however, to arbitrarily playing around with the number and the assignment of tags: By definition, a syntactic category identifies, under the same tag, tokens with identical syntactic distribution, i.e. tokens that, in any occurrence receiving that tag, can replace each other while preserving the grammaticality of the linguistic construction, modulo the adoption of suitable subcategorisation constraints impinging over them. If the goal is the development of a top-accuracy tagger that optimally supports subsequent syntactic parsing, this is the criterion that we cannot lose sight of in the choice of the tag set.

Accordingly, there are possible "candidate" categories or subcategories that can or should be excluded:

(i) Tags not justified by distributional facts, e.g. those indicating the degree of an adjective (example: `alto_ADJNORM`, `altíssimo_ADJSUP`);

(ii) Tags that tough conveying some distribution-related information can be unequivocally inferred from the form of the token, e.g. those indicating the polarity of an adverb (example: `sim_ADVPOS`; `nem_ADVNEG`), or inferred from its suffixes (example: `alto_ADJMascSing`, `altas_ADJFemPlu`);

(iii) Tags indicating the constituency status of the containing phrase but not a difference in syntactic distribution, e.g. the category of "indefinite pronouns/adjectives" used to mark articles, demonstratives and other prenominals in headless Noun Phrases (example: `li [aquele_DEM livro]`$_{NP}$; `li [aquele_INDPRON Ø]`$_{NP}$) — note that a tag `IN` (Indefinite Nominals) for single word NPs like `tudo` was kept in the tag set.

This rationale, followed to circumscribe the tag set, not only helped to exclude possible tags, but also to isolate and include categories that are usually not taken into account in a more traditional perspective. Though being verbal forms, gerund, past participle and infinitive forms each have a distribution of its own: The tags `GER`, `PTP` and `INF` were thus included in the tag set.

Other non-canonical tags were also included: These may be less interesting from a general linguistic point of view but they are important to enhance the contribution of the tagger for subsequent processing stages, e.g. named entity recognition. We isolated social titles (`Pres.`, `Drª.`, `prof.`,...), part of addresses (`Rua`, `Av.`, `Rot.`,...), months, week days, measurement units (`km`, `kg`, `b.p.m.`,...), etc. as distinct syntactic classes. Our tag set includes also specific tags for roman numerals, denominators of fractions (`meio`, `terço`, `décimo`, `%`,...), and letters.

With the tag set defined (the complete list will be presented at the workshop), we prepared a training corpus by converting and adjusting the initial tagged corpus, a 230 Ktoken, hand tagged corpus kindly granted by CLUL.

With these data and the help of the TnT tool, a tagger for Portuguese was developed with 97.2% accuracy — a value obtained with one run test over a held out evaluation corpus with the 10% not used for training. This result is in line with the state-of-the-art performance obtained for German (96.7%) or English (96.7%) with the same tool over, respectively, the NEGRA Corpus (320 Ktokens) and the Penn Treebank (1.2 Mtokens) corpora, and an accuracy measurement averaged over 10 test runs (Brants, 2000).

5. References

Brants, T., 2000, "TnT-A Statistical Part-of-speech Tagger". *Proc. of ANLP2000*, 224-231.

Brill, E., 1995, "Transformation-based Error-driven Learning and Natural Language Processing: A case study in part-of-speech tagging". *Computational Linguistics*, 21, 543-565

Mikheev, A., 2002, "Periods, Capitalized Words, etc." *Computational Linguistics* 28(3), 289-318.

Rathnaparkhi, A., 1996, "A Maximum Entropy Part-of-speech Tagger". *Proc. EMNLP'96*, 133-142.

Samuelsson, C. and A. Voutilainen, 1997, "Comparing a Linguistic and a Stochastic Tagger". *Proc. ACL'97*, 246-253.

Morphological Tagging and Syntactic Annotation of a Dialectal European Portuguese *Corpus*

Ernestina Carrilho*, Catarina Magro†, Sandra Pereira†

* Faculdade de Letras de Lisboa / Centro de Linguística da Universidade de Lisboa
† Centro de Linguística da Universidade de Lisboa
Av. Gama Pinto, 2 – 1649-003 Lisboa
{e.carrilho, cmm, spereira}@clul.ul.pt

Abstract

This presentation reports on an ongoing project of morphologically tagged and syntactically annotated *corpus* of spoken non-standard European Portuguese. Issues pertaining to the tagging and the annotation processes will be addressed from a linguistic perspective, focused on the structure and application of the tagsets used for annotating this *corpus.*

1. Introduction

The Syntactically Annotated Corpus of Portuguese Dialects (CORDIAL-SIN, from the Portuguese name Corpus Dialectal com Anotação Sintáctica) is an ongoing project of annotated corpus of spoken dialectal European Portuguese (henceforth EP). It started in September 1999 as a first year pilot-study (funded by FCT – PRAXIS XXI/P/PLP/13046/1998), further developed as a three years project (POSI/1999/PLP/33275) by a team of five linguists, under the coordination of Ana Maria Martins at the Centro de Linguística da Universidade de Lisboa (CLUL).

The project main goal is to build up a major resource for linguistic research on dialects. It aims at providing optimal access to precise morphological and syntactic information, ultimately enhancing the study of dialect syntax, a field with no tradition in the Portuguese domain.

The corpus consists of a geographically representative body of selected excerpts of spontaneous and semidirected speech. These materials were drawn from an independently existing rich collection of speech which had been recorded within the scope of several projects of the Variation Research Team of the CLUL, namely, the Atlas Linguístico-Etnográfico de Portugal e da Galiza (ALEPG); the Atlas Linguístico do Litoral Português (ALLP); the Atlas Linguístico e Etnográfico dos Açores (ALEAç); and the Fronteira Dialectal do Barlavento Algarvio (BA).

At the current state, the excerpts of dialectal speech selected for the corpus cover 24 localities within the continental and insular territory of Portugal, amounting to about 300,000 words. The corpus is available via internet (http://www.clul.ul.pt/sectores/cordialsin/projecto_cordialsin.html), under different formats: (i) verbatim orthographic transcripts; (ii) normalized orthographic transcripts; (iii) morphologically tagged versions of the normalized transcripts; (iv) syntactically annotated texts built on the morphologically tagged versions.

Verbatim orthographic transcripts include the marking up of phonetic and morphological variants, and of generalized spoken language phenomena, such as hesitations, filled and empty pauses, repetitions, rephrased segments, false starts, truncated words, speech overlappings, unclear productions, etc. From these verbatim transcripts, normalized orthographic transcripts are automatically obtained by eliminating the marked up features of spoken language and phonetic transcriptions. The tagging and the syntactic annotation apply over the normalized transcripts.

Verbatim transcripts, normalized orthographic transcripts and morphologically tagged texts are gradually made available online as the corpus building up proceeds. Since the syntactic annotation guidelines may not be completely established before the end of the annotation process, the syntactically annotated transcripts will not become available until the project is concluded.

In this paper, we will focus on the tagging and annotation phases of this corpus, which are greatly inspired by the systems used by the Penn-Helsinki Parsed Corpus of Middle English, second edition (henceforth PPCME2, see http://www.ling.upenn.edu/mideng) (Kroch & Taylor, 2000) and the Tycho Brahe Parsed Corpus of Historical Portuguese (henceforth TB, see http://www.ime.usp.br/~tycho/corpus). Collaborative work with the teams developing these corpora has permitted the tuning of already available tagging and annotation tools, in such a way that they could satisfactorily apply to dialectal EP and serve the purposes of the CORDIAL-SIN. Besides accelerating the tagging and annotation phases, this cooperation ensures the ease of linguistic information retrieval (a query tool operating on the annotation system in use is already available – cf. PPCME2 web page).

In the following sections we describe the main guidelines of the tagging and annotation systems adopted from the TB and the PPCME2, emphasizing on the structure and application of the tagsets as developed within the scope of the CORDIAL-SIN.

2. CORDIAL-SIN Morphological Tagging

2.1. The tagging process

The morphological tagging operation is to a great extent facilitated by the use of an automated morphological tagger, created by M. Finger for tagging

the TB *corpus* of Portuguese texts (written by Portuguese authors born from the sixteenth to the nineteenth centuries). After training over a sample of 30,000 hand corrected words of the dialectal *corpus*, the rate of accuracy of this tagger proved to be satisfactory enough to encourage the use of its output as the basis for a hand refined (and corrected) tagged version of the *corpus*. An additional TB tool designed for verifying the tags corrected by hand is used after manual refinement and correction to ensure the precise format of the tags. Thus, CORDIAL-SIN's morphologically tagged transcripts result from a three steps process involving: (i) automatic tagging by the TB tagger; (ii) manual tag correction and refinement using the CORDIAL-SIN's morphological annotation system; (iii) automatic verification of the corrected tags.

2.2. The morphological annotation system

The format of the morphological tags and the basics of the tagset of the CORDIAL-SIN essentially stem from the system designed for the TB automatic tagger (cf. Galves & Britto, 1999, Britto et al., 1999, and *The TB Morphological Annotation System* www.ime.usp.br/~tycho/corpus/manual/tags.html).

Tags have an internal structure consisting of an everpresent main tag (e.g. D, for determiner), and, in certain cases, sub-tags (e.g. F for feminine, P for plural), diacritics attaching different main tags ("+", "!") or main tags to sub-tags ("-"), and figures indicating clusters (see Table 1 for overview).

Tag	**Application**	**Ex.**
/D	singular masculine determiner	*o/D*
/D-P	plural masculine determiner	*os/D-P*
/D-F-P	plural feminine determiner	*as/D-F-P*
/P+D-F	preposition plus singular feminine determiner contraction	*da/P+D-F*
/VB+CL	verb (infinitive) plus enclitic pronoun	*dar-lhe/VB+CL*
/VB-R-1S!CL	verb (future) plus mesoclitic pronoun	*dar-te-ei/VBR-1S!CL*
/P31	first element of a triple prepositional cluster	*por/P31 mor/P32 de/P33*

Table 1: Morphological tags' internal structure

Such structured tags straightforwardly allow for detailed morphological information, which is a highly appealing option when tagging a morphologically rich language such as EP[1]. Indeed, for a number of possible structured tags as high as 1115, the CORDIAL-SIN tagset reduces to 39 main tags plus a smaller set of 25 sub-tags.

Main tags include POS tags and punctuation tags. The complete CORDIAL-SIN main tagset is given in Table 2.

Main Tag	**Application**
SR	verb SER
HV	verb ESTAR
ET	verb HAVER
TR	verb TER
VB	all other verbs
N	common nouns
NPR	proper nouns
PRO	personal pronouns
PRO$	possessive pronouns
CL	clitics in general
SE	clitic SE
D	definite determiner and inflected demonstratives
DEM	invariable demonstratives
ADJ	general adjectives and ordinal numbers
ADV	adverbs and speech connectives
Q	quantifiers
CONJ	coordinating conjunctions
CONJS	subordinating conjunctions
C	complementizer
WPRO	Wh-pronouns
WPRO$	possessive Wh-pronouns
WADV	Wh-adverbs
WD	Wh-determiners
P	prepositions
FP	focus particles
NUM	cardinal numbers
NEG	negative particle
INTJ	interjections and onomatopoeias
OUTRO	the word *outro/a* (all cases)
SENÃO	the word *senão* (all cases)
COISO	the word *coiso/a* (when replacing a word of any category)
MESMO	the word *mesmo/a* (with a determiner and no name)
TAL	the word *tal* (with a determiner and no name)
MAL	the word *mal* (in predicative / transitive constructions, alternating with the adjective or the DO)
BEM	the word *bem* (in predicative / transitive constructions, alternating with the adjective or the DO)
.	final punctuation
,	non-final punctuation
QT	quotation marks
DS	dash

Table 2: Main tagset

1 On the architecture of the TB tagger, especially designed with such a tag system, and on how it permits to increase the degree of accuracy of Brill's (1993, 1995) tagging method on a morphologically rich language, see Finger (1998, 2000).

The set of sub-tags codifies inflectional information – tense/mood and person/number for verbs or gender and number for nominal categories. It also specifies in more detail some morpho-syntactic information (e.g. the –NEG sub-tag to identify negative adverbs, quantifiers, prepositions, focus particles or conjunctions).

The system also allows main tags attachment for contractions or cliticizations and tags and figures combination for multiple words behaving as clusters.

For a detailed description of the tagset and its application, see *CORDIAL-SIN — Manual de Anotação Morfológica* (www.clul.ul.pt/sectores/cordialsin/manual_anotacao_morfologica.pdf).

The enhancements introduced by the CORDIAL-SIN project on the original *TB* tagset are the addition of (i) new word specific main tags; (ii) new person/number inflectional sub-tag for verbs; and (iii) a new NEG sub-tag for negative words. The project also makes a more extensive use of clusters and sub-tag distribution and endorses a wider application of multi-tagging strategy.

This refinement of the initial system, implemented during the phase of manual correction of tags, serves a twofold purpose. Above all, it helps disambiguating morphological information relevant for queries on the current annotated version of the *corpus*. On the other hand, such specific information gives a richer input to the syntactic annotation phase.

3. CORDIAL-SIN Syntactic Annotation

3.1. The syntactic annotation process

Differently from the morphological annotation phase, the process of syntactic annotation is entirely developed by hand. The option for such a time-consuming task is plainly justified by the nature of the CORDIAL-SIN *data* and by the type of rich annotation aimed at.

Manual syntactic annotation is introduced over morphologically annotated texts, with the aid of an annotation tool working in ambient Linux (the tool actually used by the PPCME2 for correcting the output of an automated parser)[1].

As already pointed out, the CORDIAL-SIN syntactic annotation system is highly inspired by the PPCME2 system (see http://www.ling.upenn.edu/~ataylor/ppcmelite.htm). The adoption of this type of rich annotation system for a Portuguese *corpus* required the adaptation of the existing system to a grammar which differs from Middle English in many respects. Accordingly, the initial phase of the CORDIAL-SIN syntactic annotation process has been devoted to the tuning of the basic annotation system, a task which was carried out in strict collaboration with the PPCME2 and the TB teams.[2] Hand annotation of a 10,000 words sample of the *corpus* has served to define and consolidate the main guidelines of the system so as it could apply to Portuguese texts.

As is well known, real *data* annotation itself is usually a very complex task. In the present case, additional complexity was expected, given the spoken and dialectal nature of the *corpus*. Sentences that call for detailed consideration are frequent, even though the basic lines of the system are already defined. Difficult annotations are decided upon after discussion by the whole team, and each new difficult example is added to the annotator's manual, in order to assure consistency. Thus, it is expected that the syntactic annotation guidelines will be progressively enriched during the whole course of the annotation phase, as more data are analysed and as new difficult sentences arise. (See http:/www.clul.ul.pt/english/sectores/cordialsin/manual_syntactic_annotation_system.pdf, for the current version of the *Syntactic Annotation Manual*).

3.2. The annotation system

3.2.1. Main guidelines

The CORDIAL-SIN syntactically annotated transcripts are built on previously tagged texts. The syntactic annotation produces a tree representation in the form of labeled brackets.

```
(IP-MAT  (CONJ e)
         (NP-SBJ *pro*)
         (VB-D-1P andávamos)
         (PP  (P com)
              (NP  (D-F-P as)
                   (N-P redes)
                   (PP    (P+D do)
                          (NP  (N badejo)))
                   (, ,)
                   (CP-REL  (WNP-1   (WPRO que))
                            (IP-SUB   (NP-SBJ *T*-1)
                                      (SR-P-3P são)
                                      (ADJP  (ADV-R mais)
                                             (ADJ-F-P baixas))))))
         (. ...))       [VPA07]
```

Figure 1. CORDIAL-SIN syntactically annotated sentence

As in the PPCME2, the annotation represents quite flat trees, allowing for multiple branching nodes and for some words projecting only a word-level node (e.g. inflected verbs, negation, sentence focus particles).

In addition to constituent boundaries and phrase and clause dependencies, the annotation marks up grammatical relations, clause-types, some empty categories and some transformational relations. At the word level, morphological labels are preserved. Phrase and clause labels indicate category, often specified by an extended label indicating syntactic function (e.g. subject, direct object), clause type (e.g. relative, adverbial, interrogative), or other relevant information (e.g. left dislocation, pragmatic marker).

3.2.2. Labels and extended labels

Even though most labels and extended labels come originally from the PPCME2 system, a restricted number of additional labels were introduced for the EP annotation. In particular, some new extended labels

[1] This tool consists of a task-specific mouse-based package, which is embedded in the GNU Emacs editor.

[2] In particular, with Anthony Kroch and Helena Britto, respectively. A first proposal of the Portuguese system was discussed with A. Kroch in December 2000, and a further extended version of the system was established with H. Britto in April 2002.

were created for the CORDIAL-SIN use, especially adapted to spoken data annotation (e.g. -CON for pragmatic markers, and -ANS, - POL, -TAG, cf. Table 4). Tables 3 and 4 show the main label set used in the CORDIAL-SIN syntactic annotation. (The complete set is available online, see *Syntactic Annotation Manual*).

Label	**Category (and function)**
NP	Noun Phrase
NP-SBJ	Noun Phrase (Subject)
NP-ACC	Noun Phrase (Direct Object or Nominal Predicate)
NP-ADV	Noun Phrase (Adverbial)
NP-VOC	Noun Phrase (Vocative)
NP-DAT	Noun Phrase (Dative)
NP-GEN	Noun Phrase (Dative of Possession)
PP	Prepositional Phrase
PP-ACC	Prepositional Phrase (partitive object)
ADVP	Adverbial Phrase
ADJP	Adjective Phrase
NUMP	Numeral Phrase
INTJP	Interjection Phrase
QP	Quantifier Phrase
WXP	Wh-Phrase (e.g. WNP, WPP)

Table 3: CORDIAL-SIN phrase labels

Label	**Category (and function)**
IP-MAT	Independent or conjoined declarative IP
IP-IND	Independent, non-declarative IP
IP-SUB	Subordinate IP
IP-ADV	Adverbial IP
IP-INF	Infinitival clause
IP-GER	Gerund clause
IP-PPL	Participial clause
IP-SMC	Small clause
IP-ANS	Answer
IP-POL	Reinforcement of an assertion
CP-EXL	Exclamative
CP-IMP	Imperative
CP-QUE	Question
CP-QUE-TAG	Question-tag
CP-INF	Infinitive introduced by *que*
CP-THT	*That* clause
CP-REL	Relative
CP-FRL	Free Relative
CP-CLF	Cleft
CP-ADV	Adverbial clause
CP-DEG	Degree clause
CP-CMP	Comparative clause

Table 4: CORDIAL-SIN clause labels

3.2.3. Adapting the PPCME2 system to EP

Besides the addition of some new extended labels, the adaptation of the PPCME2 annotation system to EP *corpora* essentially required the conception of additional ways of codifying new syntactic constructions, within the possibilities offered by the system (and, consequently, by the annotation tool). For instance, the CORDIAL-SIN/TB system includes unambiguous codification for most clitics, adding information on clitic climbing or exceptional case marking contexts, which was not required for the PPCME2 annotation. Also, the codification of certain types of constructions (such as clefts and topicalization/left-dislocation) implied, for the EP *corpora*, the creation of new variants upon the PPCME2 solutions, given the diversity of related constructions allowed by EP.

The annotation system so designed for the CORDIALSIN is thus compatible with CorpusSearch, a linguistically intuitive query tool, especially developed by Beth Randall for use with the PPCME2[1], which ultimately permits fast and massive information retrieving on relevant aspects of the syntax of the CORDIAL-SIN *data*.

4. References

Brill, Eric, 1993. *A Corpus-Based Approach to language Learning*. PhD thesis, University of Pennsylvania.

Brill, Eric, 1995. Transformation-based error-driven learning and Natural Language Processing: a case study in part-of-speech tagging. *Computational Linguistics* 21: 543-565.

Britto, Helena, Charlotte Galves, Ilza Ribeiro, Marina Augusto, and Ana Paula Scher, 1999. Morphological annotation system for automatic tagging of electronic textual *corpora*: from English to Romance languages. In *Proceedings of the 6th International Symposium of Social Communication*, Santiago de Cuba. Editorial Oriente. 582-589.

Finger, Marcelo, 1998. Tagging a Morphologically Rich Language. In *Proceedings of the First Workshop on Text, Speech and Dialogue (TSD'98)*. Brno, Czech Republic. 39-44.

Finger, Marcelo, 2000. Técnicas de Otimização Empregadas no Etiquetador Tycho Brahe. In Proceedings of V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR 2000). Atibaia, Brazil.

Galves, Charlotte and Helena Britto, 1999. A construção do *Corpus Anotado do Português Histórico Tycho Brahe*: o sistema de anotação morfológica. In I.

Rodrigues and P. Quaresma (eds.) *Proceedings of the IV PROPOR*. Évora. Universidade de Évora. 55-67.

Kroch, Anthony S. and Ann Taylor, 2000. *The Penn-Helsinki Parsed Corpus of Middle English, Second Edition*. Department of Linguistics, University of Pennsylvania.

[1] On this tool, see http://www.ling.upenn.edu/mideng/CSmanual.pdf.

Looking for Similarity among Ontological Structures

Marcirio Silveira Chaves, Vera Lúcia Strube de Lima

Programa de Pós-Graduação em Ciência da Computação
Faculdade de Informática
Pontifícia Universidade Católica do Rio Grande do Sul
Av. Ipiranga, 6681
90619-900 Porto Alegre, RS
{mchaves, vera}@inf.pucrs.br

Abstract

The automatic mapping among Ontological Structures (OSs) has been a continuous concern as a task of integration and reuse of knowledge. Besides, this mapping can support the task of expansion and combination of OSs. However, the manual execution of such task is quite tedious and slow, so it is important to automate, at least partially, the mapping process. This paper describes an ongoing work that employs the similarity measure called String Matching (SM) proposed in (Maedche & Staab, 2002) to compare terms in distinct hierarchies. We apply SM to Portuguese language OSs aiming to finding lexically similar terms. We still present some experiments using the SM measure as well as a stemmer, trying to improve the preliminary results produced by SM.

1. Introduction

Nowadays, studies that focus the mapping among Ontological Structures (OSs) still include a considerable amount of manual work. The more recent proposals (Doan et al., 2002; Noy & Musen, 2001) are described as semiautomatic because they still lack techniques allowing the full automation of this process.

Noy and Musen (2001) assert that the manual work of mapping, merging or aligning OSs is accomplished, most of the cases, by hand. This manual mapping is slow (Uschold, 2001), tedious and susceptible to mistakes (Doan et al., 2002; Noy & Musen, 1999). Besides, this process is difficult to repeat and it is not practical.

In this work, OS is taken as a set of pre-defined terms explicitly connected by semantic relations, in a format readable by humans and machines. This notion includes collections of vocabularies and concepts.

The task of mapping one OS to another reflects a continuous interest on the reuse of available OSs. Ding and Foo (2002) mention that the mapping helps the task of expansion and combination of OSs. For example, on the context of information retrieval, as similar terms are found among OSs, a system can browse through combined OSs. This kind of approach could help improving user queries results.

For Prasad, Peng and Finin (2002) mapping OS_A to OS_B consists of a process where, for each concept in OS_A a correspondent concept with similar semantic has to be found in OS_B. If there is no correspondence in OS_B, the concept is not mapped. To help users or systems find similar concepts between OSs, similarity measures are used.

1.1. Similarity Measures

Similarity between conceptual models is difficult to measure and, to establish an adequate measure of similarity is a quite subjective task (Maedche et al., 2002).

Similarity measures are used in applications such as word sense disambiguation, summarization and text annotation, information retrieval and extraction, and automatic indexing, among others (Budanitsky & Hirst, 2000). Several similarity measures are found in the literature, each one of them applied to a specific situation.

The semantic similarity measures in (Resnik, 1995; Lin, 1998; Jiang & Conrath, 1997), for example, are based on the content of information of each term. This content is defined as the number of occurrences of a term, or any child term, in the same hierarchy in a corpus.

In the present work we do not use corpus but apply the similarity measures to terms belonging to hierarchies of OSs. We work with lexical similarity without concerning about the position of the term in the hierarchy.

We search for the similarity among Portuguese OSs using similarity measures among terms, namely String Match, at the lexical level. We also use a stemmer to improve the results produced by this measure. Some experiments and preliminary results are showed.

This paper is further organized as follows. In section 2, related works are presented. Preliminary experiments are described in section 3. Finally, in section 4 we give an outlook on some future works.

2. Related Works

2.1. Anchor-Prompt

Noy and Musen (2001) developed the algorithm Anchor-Prompt that works on a set of anchorcombinations[1] previously identified (by hand or automatically). The OSs used belong to the library of DAML program[2].

The algorithm receives the anchor-terms that constitute a path in a hierarchy of concepts or terms. After the length of this path is known, a rate is attributed to the similarity between each two terms in

[1] Pair of related terms.
[2] DARPA Agent Markup Languages – *http://www.daml.org/ontologies*

the same position on the path. For example, let A and D be anchor-terms in OS_A and OS_B. In OS_A composed by the terms A-B-C-D the length of path from node A to node D is 3; in OS_B composed by the terms A-M-N-D the length of the path from node A to node D is 3. In this case, the similarity between B and M and C and N will be higher because these terms are in the same relative positions on the path from A to D.

In spite of providing consistent mappings, the approach based on anchors has a strong limitation for OSs with different depths, that is, as an OS is deep (with several levels in the hierarchy) and the other OS is flat (with a few levels in the hierarchy). In this case, Noy and Musen assert that the algorithm does not fit.

The OSs used in our work have distinct depths in most of the cases, so the approach of anchor-terms is not suitable.

2.2. String Matching

Maedche and Staab (2002) present a two layer approach, lexical and conceptual, to measure the similarity between terms of different OSs. At the lexical level, Maedche and Staab considered the Edit Distance (ED) formulated by Levenshtein (1966). This measure considers the minimum number of modifications should occur to change a string into another using a dynamic programming algorithm. For example, ED(*computador*, *computadores*) is 2, because two operations of insertion transform the original string *computador* into *computadores*. The contribution of Maedche and Staab consists of the String Matching (SM) measure given by:

$$SM(T_i, T_j) := \max\left(0, \frac{\min(|T_i|, |T_j|) - ED(T_i, T_j)}{\min(|T_i|, |T_j|)}\right) \in [0,1].$$

The SM measure calculates the similarity between two terms (T_i, T_j). The length of the shortest term is represented by $\min(|T_i|, |T_j|)$. For example, to obtain the similarity between the terms (*computador*, *computadores*) the minimum length is 10 and the value of $ED(T_i, T_j)$ is 2. Thus, the resulting value is 0,8.

The shortest length is considered in the numerator as well as in the denominator of this formula allowing pondering the number of changes appearing in the term with shortest length. In the previous example the value 0,8 corresponds to the similarity between the terms (*computador*, *computadores*). The SM measure always returns a value of similarity between 0 and 1, where one stands for perfect match and zero indicates a bad match. Maedche and Staab used German language OSs, specifically tourism domain, in their experiments.

3. Experiments with Portuguese Language

We apply the SM measure to Portuguese language OSs. These OSs come from two distinct sources, the first from São Paulo University[1] (OS_1) and the second from the Brazilian Senate[2] (OS_2).

[1] Additional information available in *http://www.usp.br/sibi*

[2] Additional information available in *http://webthes.senado.gov.br/thes*

The terms appearing in these OSs can be associated with one of two groups: one word terms and multiword terms.

When calculating the similarity by using the SM measure it is important to establish a threshold in the detection of similar terms. In our experiments were adopt the value 0,75 as a threshold, that is, terms that present values equal or above 0,75 are considered similar, otherwise they are not.

3.1. SM applied to One Word Terms

We first applied the SM measure to terms composed by only one word. Table 1 presents some results for the preliminary tests with Portuguese:

EO_1	EO_2	SM
profissão	procissão	0,89
denúncia	renúncia	0,88
asfalto	assalto	0,86
geoprocessamento	teleprocessamento	0,81

Table 1: Examples of terms considered similar by SM measure.

Despite SM measure has produced good results with one word terms, we can observe in Table 1 unlike terms with values above 0,75.

An alternative solution to this problem is the use of a stemmer. We used a stemmer that was specifically developed for Portuguese language (Orengo & Huyck, 2001) which presented good results when compared to Porter algorithm in (Orengo & Huyck, 2001) and when compared to another algorithm developed also specifically to Portuguese language in (Chaves, 2003).

Some results obtained with the application of this stemmer are shown in Table 2. Column "SM" shows the results to the terms in the first and second columns, while column "SMStem" presents values resulting from the application of the SM to the strings in the two last columns. These strings own a stronger semantic weight, what allows a more reliable result produced by SM and, consequently, by SMStem.

Despite the good results presented in Table 2, we still observe inconsistent values after the application of the stemmer as depicted in Table 3, where SM as well as SMStem present bad results with dissimilar terms.

The extract in Table 3 presents terms with similarity higher than 0,75 for measures SM and SMStem. This indicates that only the use of a stemmer is not enough to solve the similarity problem at the lexical level. In the next section we consider the treatment to multiword terms.

3.2. SM applied to Multiword Terms

For these experiments, ontologies were first preprocessed in order to eliminate blanks. This preprocessing has also been used for other experiments in the literature (Noy & Musen, 2001; Maedche & Staab, 2002).

In the same way that for one word terms, SM generates inconsistent results, some of which can be seen in Table 4.

EO_1	EO_2	SM	SMStem	EO_1	EO_2
acampamento	acabamento	0,89	0,50	acamp	acab
antiguidade	ambiguidade	0.82	0.67	antigu	ambigu
antologia	oncologia	0.78	0.71	antolog	oncolog
funcionalismo	racionalismo	0.75	0.50	funcion	racion

Table 2: Examples of terms considered similar by SM and considered unlike by SMStem.

EO_1	EO_2	SM	SMStem	EO_1	EO_2
tumulos	tumultos	0.86	0.80	tumul	tumult
aceite	azeite	0.83	0.80	aceit	azeit
linho	vinho	0.80	0.75	linh	vinh
metrologia	nefrologia	0.80	0.75	metrolog	Nefrolog
trova	tropa	0.80	0.75	trov	Trop

Table 3: Examples of terms considered similar by SM and SMStem.

EO_1	EO_2	SM
aguasSubterraneas	ruasSubterraneas	0.88
comportamentoPolitico	comportamentoColetivo	0.86
direitoPrevidenciario	direitoPenitenciario	0.85
africaDoSul	americaDoSul	0.82
contratoColetivoDeTrabalho	convencaoColetivaDeTrabalho	0.77

Table 4: Examples of multiword terms considered similar by SM.

Terms can be considered similar if the SM threshold is equal or above to 0,75, as stated in section 3.1, but the terms depicted in Table 4 have low semantic similarity in a human point of view.

So, to improve results like those in Table 4, we calculate the similarity between multiword terms regarding each word individually by means the string returned by the stemmer.

This approach is similar to the one used with one word terms. We apply the stemmer to each word in the term. So, our algorithm process the SM measure for each pair of stems returned. Finally, it returns the minor value found as result of similarity between the multiword terms. For example, SMStem(analiseDoSonho, analiseDoSolo) is changed in SM(analis, analis), SM(do, do) and SM(sonh, sol), (1, 1, 0,33), respectively. So, SMStem(analiseDoSonho, analiseDoSolo) is 0,33. According to SM, the similarity between these terms is 0,84. Considering the threshold 0,75, SMStem points that these terms are not similar, although they could be considered similar if using SM. More results are shown in Table 5.

EO_1	EO_2	SM	SMStem	EO_1	EO_2
pescaIntensiva	pescaExtensiva	0.78	0.67	pescIntens	pescExtens
ecologiaFlorestal	economiaFlorestal	0.75	0.67	ecologFlorest	economFlorest
biologiaDoSolo	ecologiaDoSolo	0.75	0.67	biologDoSol	ecologDoSol
plantasMarinhas	plantasDaninhas	0.75	0.33	plantMar	plantDan

Table 5: Examples of terms considered similar by SM and considered unlike by SMStem.

EO_1	EO_2	SM	SMStem	EO_1	EO_2
veiculosEspeciais	veiculosEspaciais	0.89	0.80	veiculEspec	veiculEspac
acionistaMinoritario	acionistaMajoritario	0,82	0,75	acionMinorita	acionMajorita
turismoDeImportacao	turismoDeExportacao	0.80	0.78	turDeImportaca	turDeExportaca
soloAcido	soloArido	0.80	0.85	solaced	solArid
sociologiaDoRadio	semiologiaDoRadio	0.80	0.75	sociologDoRadi	semiologDoRadi

Table 6: Examples of terms considered similar by SM and SMStem.

Table 5 presents cases where the application of the stemmer improves the results produced by SM. In these cases, similar terms detected by SM are considered unlike by SMStem measure. The reader may notice that these terms are really dissimilar and should not to be related between OSs.

Despite of the improvement with the stemmer, in some cases SMStem measure presented results quite near to SM according to Table 6, which shows terms with low semantic similarity. However, SM as well as SMStem present values allowing these terms to be considered similar. As for the one word terms, we also

found inconsistent results produced by SMStem measure to multiword terms.

Maedche and Staab (2002) assert that SM helps detecting similar lexically similar strings in German. However, regarding the preliminary results, we notice that the SM measure is insufficient to detect similarity of terms in Portuguese. The stemmer algorithm seems to improve the preliminary results, however we still keep some inconsistent examples.

In some cases the stemmer has even introduced some errors, that is, common mistakes like overstemming[1] and understemming[2].

We hope an additional penalty can be set, associated with the changes in the resulting string, that is, changes in the root indicate a higher probability that the words are not similar.

4. Final Remarks and Future Work

In this paper we present an ongoing work that investigates alternatives to detect similar terms in Portuguese language ontologies. We believe that similarity of strings is not completely treated yet, and it can be useful to detect similarities as an initial step in a task of integration of OSs. This integration allows the reuse of information that reflects a concern of research on the semantic web approach.

We apply the SM measure to Portuguese language ontologies and present some preliminary results. It was possible to confirm that this measure alone is not enough to detect similarities. Besides, the use of a stemmer as a complement to SM presents also inconsistent results.

We are conscious that it is necessary to undertake a deeper evaluation in our experiments, once the measures and the stemmer used for this moment do not present completely reliable results.

As a future work we intend to apply a weight to changes accomplished on the root of words and use some heuristics to get more consistent results. Besides, we can use other measures of similarity and compare the results.

5. Acknowledgements

Marcirio Silveira Chaves is supported by the research center HP-CPAD (Centro de Processamento de Alto Desempenho HP Brasil-PUCRS). We would like to thank Viviane Orengo that kindly provided us the stemmer used here.

6. References

Budanitsky, A. & Hirst, G., 2000. Semantic Distance in WordNet: An Experimental, Application-oriented Evaluation of Five Measures. *Workshop on WordNet and Other Lexical Resources*, in the North American Chapter of the Association for Computational Linguistics (NAACL-2000, Pittsburgh, PA.

Chaves, M. S., 2003. Um estudo e apreciação sobre algoritmos de stemming para a língua portuguesa. *IX Jornadas Iberoamericanas de Informática*. Cartagena de Indias - Colômbia, 11-15 agosto de 2003. (CDROM)

Ding, Y. & Foo, S., 2002. Ontology Research and Development Part 2 - A Review of Ontology Mapping and Evolving. *Journal of Information Science*, 28(5): (pp. 375–388).

Doan, A., Madhavan, J., Domingos, P., & Halevy, A., 2002. Learning to Map between Ontologies on the SemanticWeb. In *Proceedings of the World Wide Web Conf. (WWW- 2002)*, Honolulu, Hawaii, USA.

Jiang, J. J. & Conrath, D. W., 1997. Semantic similarity based on corpus statistics and lexical taxonomy. Proceedings of the International Conference on Research in Computational Linguistics (ROCLING), Taiwan.

Levenshtein, I. V., 1966. Binary codes capable of correcting deletions, insertions and reversals. *Cybernetics and Control Theory*, 10(8):(pp. 707–710).

Lin, D., 1998. An information-theoretic definition of similarity. Proceedings of the 15th International Conference on Machine Learning. Morgan Kaufmann, San Francisco, CA, (pp. 296-304).

Maedche, A. & Staab, S., 2002. Measuring Similarity between Ontologies. In *Proceedings of the European Conference on Knowledge Acquisition and Management - EKAW-2002*. Madrid, Spain, October 1-4, (pp. 251–263).

Maedche, A., Motik, B., Silva, N. & Volz, R., 2002. MAFRA - A Mapping Framework for Distributed Ontologies. *Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management EKAW*, Madrid, Spain.

Noy, N. F. & Musen, M., 1999. SMART: Automated Support for Ontology Merging and Alignment. In *Twelfth Banff Workshop on Knowledge Acquisition, Modeling, and Management* - Banff, Alberta, Canada.

Noy, N. F. & Musen, M. A., 2001. Anchor-PROMPT: Using Non-Local Context for Semantic Matching. In Proceedings of the Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001), Seattle, WA.

Orengo, V. M. & Huyck, C., 2001. A *Stemming* Algorithm for Portuguese Language, In: *Eigth Symposium on String Processing and Information Retrieval (SPIRE 2001)*, Chile. (pp. 186-193).

Prasad, S., Peng, Y. & Finin, T., 2002. Using Explicit Information to Map Between Two Ontologies. In Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems - Workshop on Ontologies in Agent Systems (OAS) - Bologna, Italy. 15-19 July.

Resnik, P., 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. *Proceedings of the XI International Joint Conferences on Artificial Intelligence (IJCAI)*. (pp. 448-453).

Uschold, M., 2001. Where is the Semantics in the Semantic Web? In *Workshop on Ontologies in Agent Systems*, Montreal, Canada.

[1] It occurs when the string removed was not a suffix, but part of the stem. For example, *gramática* is reduced to *gramá* and not *gramát*.

[2] It occurs when the suffix is not removed. For example, *sistemático* is reduced to *sistemátic* and not to *sistemát*.

Tira-Teimas: after Shallow Parsing

Luísa Coheur*,+, Nuno J. Mamede*

*L2F INESC-ID/IST - Spoken Languages Systems Laboratory
+GRIL/Université Blaise-Pascal
Rua Alves Redol nº 9, 1000 - 029
{luisa.coheur}{numo.mamede}@l2f.inesc-id.pt

Abstract

In this paper we present Tira-Teimas, which is a program written in XSLT, that checks if a shallow parsed text verifies a set of properties from the 5P paradigm. We show how to code exigency properties in XSLT and we present an example of a model disrespecting a property.

1. Introduction

Tira-Teimas is a program that verifies if a shallow parsed text satisfies a set of properties from the 5P paradigm (Bès, 99; Bès & Hagège, 2001; Hagège, 2000). These properties are used to describe the syntax of a natural language.

Tira-Teimas checks the following properties (concerning a family of models (phrases), labelled M):

- Uniqueness: identifies the elements that cannot occur more than once in a model labelled M;
- Exigency: allows to declare that a occurs in a model labelled M only if b also occurs in it;
- Exclusion: permits to declare that a excludes b in a model labelled M;
- Linearity: declares the linearity relations between the elements occurring in a model labelled M.

These properties can be seen as a repository of linguistic information that can be used according to our needs (Bès & Hagège, 2002). Having nominal phrases extraction as a goal, Hagège developed a shallow parser prototype, AF, that uses information from the 5P properties (Hagège, 2000; Bès & al., 1999). Therefore, 5P properties for nominal models (Hagège, 2000) enriched the information structures used by AF. Nevertheless, these information structures are less expressive than the 5P properties. Therefore, it is not sure that the models identified by AF verify the whole set of 5P properties. As so, **Tira-Teimas** was developed in order to verify if each model identified by AF satisfies (or not) the 5P properties.

2. Tira-Teimas

Tira-Teimas is written in XSLT (W3C-XSL). The following example shows how to code exigency properties in a format that can be easily mapped into XSLT (a similar approach can be applied to other 5P properties).

Exigency properties have the following general syntax:

E_i: $\{a_1, ..., a_n\} \Rightarrow_M \{b_1, ..., b_m\} \mid ... \mid \{c_1, ..., c_k\}$

meaning that if the symbols $a_1, ..., a_n$ occur in a model labelled M, then

$\{b_1, ..., b_m\}$ or ... or $\{c_1, ..., c_k\}$

must also occur in that model.[1]

Given this, **Tira-Teimas** works as follows: suppose that a model labelled M is detected by the shallow parser. Then **Tira-Teimas** checks if it verifies E_i by counting the number of occurrences of every a_j, b_k and c_m in the model. That is, being count(x, X) a function returning the number of occurrences of x in (the model) X, if

count(a_1, M) != 0 and ... and count(a_n, M) != 0

and

[(count(b_1, M) = 0 or ... or count(b_m, M) = 0)

and ... and

(count(c_1, M) = 0 or ... or count(c_k, M) = 0)]

then M does not satisfies E_i.

When a model does not satisfies a property, it is marked with the identification of that property.

As a predicate count is available in XSLT, mapping the previous formulas in XSLT is a trivial task. On the contrary, writing 5P properties directly in XSLT is not an easy task, as 5P properties in XSLT take a very

1 These elements or others subsumed by them .

unfriendly look. In order to solve this situation, 5P properties are written in XML (W3C-XML). Then an extra program **TTT** (Tira-Teimas Translator), maps these properties into XSLT. **TTT** is also written in XSLT.

3. Results

SuSAna (Batista & Mamede, 2002) is, in rough terms, a new implementation of AF, that we used to collect a set of shallow parsed corpus. Experiments with **Tira-Teimas** were made over these corpora. As expected a few (not many) models disrespected 5P properties. The following example describes a situation where a model does not verifies a property.

Consider exigency property E_{15} from (Hagège, 2000), over nuclear nominal models (labelled m-nn)[1]:

E_{15} adj_s $\Rightarrow_{nn}$ det | cada | qualquer | certo1 | algum | nenhum | tal | outro | tanto

The linguistic information that SuSAna uses, accepts that inside an m-nn, *muito* (labelled q3_s) can be followed by an adj1_s (consider for example *Ele comeu muito belo peixe. Tanto que ficou doente.*[2]).

As so, the following model was captured by SuSAna:

($muito_{q3\_s}$ $cansado_{adj1\_s}$)$_{nn}$[3]

Tira-Teimas ran over the same corpus and detected an inconsitency between E_{15} and that syntactic model. In fact, it does not respect E_{15}, because according to E_{15} adj1_s (subsumed by adj_s) requires one of the elements on the right side of E_{15}, and q3_s is not one of them.

4. Conclusions and future work

Although the original motivation for **Tira-Teimas** was to check 5P properties, **Tira-Teimas** is not bounded to this application.

In fact, it can also be used to find differences between what is syntactically correct - supposing that a set of 5P properties describe it - and what is currently practised.

In addition, **Tira-Teimas** could be applied to detect differences between the Portuguese from Portugal and Portuguese from Brazil. For example, the 5P properties from (Hagège, 2000) describing Portuguese (from Portugal) nominal phrases could be applied to a Brazilian shallow parsed text.

Finally, **Tira-Teimas** can be easily extended to other properties.

5. Acknowledgments

Paper supported by FCT (Fundação para a Ciência e Tecnologia).

6. References

Batista, F., Mamede N. 2002. SuSAna: Módulo multifuncional da análise sintáctica de superfície. In J. Gonzalo, A. Penas, and A. Ferràndez (eds.), *Proc. Multilingual Information Access and Natural Language Processing Workshop*, pages 29-37, Sevilla, Sapain, November 2002. IBERAMIA 2002.

Bès, G. G., Hagège, C., 2001. Properties in 5P (soon in the GRIL web page). Technical Report, GRIL, Clermont-Ferrand, France, November, 2001.

Bès, G. G., 1999. La phrase verbal noyau en français. In *Recherches sur le français parlé*.15: 273-358. Université de Provence, France, 1999.

Bès, G. G., Hagège, C., Coheur L., 2001. Des propriétés linguistiques à l'analyse d'une langue. In *VEXTAL*. Venice, Italy, November, 1999.

Hagège, C., 2000. *Analyse Syntatic Automatique du Portugais*. Ph.D. Thesis, Université Blaise-Pascal, Clermont-Ferrand, France, 2000.

Hagège, C., Bès, G. G., 2002. Enconding and reusing linguistic information expressed by linguistic properties. In *Proceedings of COLING'*2002. Taipei, 2002.

World Wide Web Consortium (W3C). *Extensible Markup Language (XML)*. See: www.w3.org/XML.

World Wide Web Consortium (W3C). *The Extensible Stylesheet Language (XSL)*. See: www.w3.org/Style/XSL.

1 adj_s is the label of a category subsuming adj1_s, adj2_s and adj3_s - adjectives of type 1, 2 and 3, respectively. Det stands for determiners, and cada, qualquer, certo1, algum, nenhum, tal, outro, tanto are very particular category labels for the words *cada*, *qualquer*, *certo1*, *algum*, *nenhum*, *tal*, *outro* and *tanto*, respectively.

2 *He ate lots of nice fish. And he became sick.*

3 *Muito cansado* means *very tired*.

Shallow Parsing for Portuguese–Spanish Machine Translation

Alicia Garrido-Alenda, Patrícia Gilabert-Zarco, Juan Antonio Pérez-Ortiz, Antonio Pertusa-Ibáñez, Gema Ramírez-Sánchez, Felipe Sánchez-Martínez, Miriam A. Scalco and Mikel L. Forcada

Departament de Llenguatges i Sistemes Informàtics
Universitat d'Alacant, E-03071 Alacant, Spain.
mlf@ua.es

Abstract

To produce fast, reasonably intelligible and easily corrected translations between related languages, it suffices to use a machine translation strategy which uses shallow parsing techniques to refine what would usually be called *word-for-word* machine translation. This paper describes the application of shallow parsing techniques (morphological analysis, lexical disambiguation, and flat, local parsing) in a Portuguese–Spanish, Spanish–Portuguese machine translation system which is currently being developed by our group and is publicly and freely available at `http://copacabana.dlsi.ua.es`.

1. Introduction

We describe the successful application of shallow parsing techniques in a Portuguese–Spanish, Spanish–Portuguese machine translation (MT) system which is currently being developed by our group and is publicly and freely available at `http://copacabana.dlsi.ua.es`.

The paper is organized as follows: section 2 describes the role of shallow parsing in real-world related-language machine translation. The Portuguese–Spanish MT engine is described in section 3. Lexical disambiguation and structured is discussed with a bit more detail in sections 4 and 5. Section 6 ends the paper with a few concluding remarks.

2. Real Machine Translation and Shallow Parsing

General-purpose MT systems are expected to satisfy the requirements of the two main application modes: *assimilation* or understanding of documents written in another language (fast, intelligible translations) and *dissemination* of documents translated into another language (easily correctable translations).

Real (i.e., working) MT may be seen both as the result of approximations (some of them inevitable) over an ideal, theoretically motivated model based on the *principle of semantic compositionality* and as the result of a set of necessary refinements over a very rudimentary *word-for-word* substitutional system.

On the one hand, real MT may be seen as a set of successive approximations over "ideal MT":

1. Most MT system adopt the approximation that *translating texts is translating sentences*, which, for example, excludes the treatment of some aspects of discourse structure.
2. The *principle of semantic compositionality* (PSC, Radford et al. 1999, p. 359) states that the interpretation (meaning) of a sentence is compositionally built from the interpretation of its words, following the groupings dictated by its parse tree, and also conversely, sentences may be compositionally built from interpretations (Tellier, 2000). Translating a source language (SL) sentence would then mean (a) *fully parsing* it, (b) assigning interpretations to its words, (c) compositionally building an interpretation, (d) analysing this interpretation to obtain target language (TL) words and a TL parse tree from it, and (e) generating a TL sentence from them. This is basically the *modus operandi* of interlingua systems and constitutes the *compositional translation* approximation. Note that this account assumes that *lexical* ambiguity (words having more than one interpretation) and *structural* ambiguity (sentences having more than one parse tree) have been also ideally solved.
3. As is the case with professional translators, MT systems do not always need to completely "understand" (build explicit interpretations of) SL sentences. *Transfer* systems take a shortcut and go from SL parse tree and words directly into TL parse tree and words: they do so by applying parse tree transformations (*structural transfer*) and word substitutions (*lexical transfer*), without building an explicit representation of the interpretation. This constitutes an additional approximation, the *transfer approximation*.
4. When languages are syntactically similar (e.g, when related), full parsing is not performed; lexical transfer is complete, but structural transfer is partial and local and occurs only where required. This could be called the *partial parsing* approximation. *Transformer* systems (Arnold et al., 1994, 4.2), many of them commercial and available on the internet[1], are an example of this approximation.

On the other hand, real MT may be seen as a refinement over what would usually be called *word-for-word* MT (which processes input one word at a time and substitutes it by a constant equivalent independently

[1] For example, SDL Transcend is available through `http://www.freetranslation.com` and Reverso is available as `http://www.reverso.net`.

of context). Taking the previous experience of our research group with the interNOSTRUM (`http://www.interNOSTRUM.com`) Spanish–Catalan MT system (Canals-Marote et al., 2001), used by hundreds of people on a daily basis, we can state that, to produce fast, reasonably intelligible and easily corrected translations between related languages —such as Portuguese (`pt`) and Spanish (`es`)—, it suffices to augment *word-for-word* MT with a robust *lexical* processing (to treat multiword expressions and to adequately choose equivalents for lexically ambiguous words), and a local *structural* processing based on simple and well-formulated rules for some simple structural transformations (reordering, agreement).

These requirements are very well met by *shallow parsing* techniques, which are usually applied sequentially:

1. tokenization and morphological analysis, to be able to build bilingual dictionaries as correspondences between SL and TL lemmas, to be able to identify multiword expressions and to determine the syntactic role of each word in the sentence;
2. categorial disambiguation (to choose among multiple analyses in the case of homographs), and
3. partial, flat parsing of those structures needing treatments that may be applied locally.

The next section illustrates how these operations are integrated into the complete dataflow of a `pt–es` machine translation system.

3. The `pt–es` Machine Translation Engine

As said above, we are currently developing a bidirectional MT system between `pt` and `es` (prototype available at `http://copacabana.dlsi.ua.es`) with emphasis in Brazilian `pt`, based on an existing Spanish–Catalan MT system. The current text coverage surpasses 95%, errors rate below 10%, and speed surpasses 5000 words per second on an desktop PC equipped with an AMD 2100 processor. The system, which already receives thousands of visits a day, (a) translates ASCII, RTF and HTML documents and e-mail messages, (b) translates Internet documents (webpages) during browsing, with link following, and (c) implements a bilingual chat room.

The translation engine is a classical *partial transfer* or *transformer* system consisting of an 8-module *assembly line*; to ease diagnosis and testing, these modules communicate between them using text streams. Five modules are automatically generated from linguistic data files using suitable compilers. The modules (organized as in figure 1) are:

- The *unformatter* separates the text to be translated from the format information. Format information is encapsulated so that the rest of the modules treat it as blanks between words.
- The *morphological analyser* tokenizes the text in surface forms (SF) (lexical units as they appear in texts) and delivers, for each SF, one or more lexical forms (LF) consisting of *lemma*, *lexical category* and morphological inflection information. Tokenization is not straightforward due to the existence, on the one hand, of contractions (e.g., *daquele* = *de* + *aquele* ["of that"]), and, on the other hand, of multiword lexical units (*no entanto* ["in spite of"]), which may inflected (*dava na vista* ["called someone's attention"]). This module is compiled from a SL morphological dictionary (MD) (Garrido et al., 1999; Garrido-Alenda et al., 2002). For example, the `pt` input "`as viagens coletivas`" would give a sequence of four LF's, with the first one being ambiguous: (*o*, article, feminine plural) and (*o*, clitic pronoun, feminine plural), (*viagem*, noun, feminine plural), and (*coletivo*, adjective, feminine plural).
- The *categorial disambiguator* (part-of-speech tagger) chooses, using a hidden Markov model (HMM) trained on representative SL texts, and according to its context, one of the LFs corresponding to an ambiguous SF. Ambiguous SFs are a very frequent source of errors when incorrectly solved. In the example above, the system would choose (*o*, article, feminine plural), (*viagem*, noun, feminine plural), and (*coletivo*, adjective, feminine plural).The *lexical transfer* module is called by the structural transfer module (see below); it reads each SL LF and delivers the corresponding TL LF. This module is compiled from a bilingual dictionary. In the example, the SL LFs are translated to (*el*, article, feminine plural), (*viaje*, noun, **masculine** plural) — note the gender change — , and (*colectivo*, adjective, feminine plural).
- The *structural transfer* module uses finite-state pattern matching to detect (in the usual left-to-right, longest-match way) patterns of LFs (phrases) needing special processing due to grammatical divergences between the two languages (gender and number changes, reorderings, lexical changes, etc.) and performs the corresponding operations. This module is compiled from a transfer rule file (Garrido-Alenda and Forcada, 2001), and generates a `lex` (Lesk, 1975) scanner as an intermediate step during compilation. In the running example, the noun phrase pattern *article–noun–adjective* is detected; this pattern dictates that the article and the adjective should agree with the translation of the noun, producing: (*el*, article, masculine plural), (*viaje*, noun, masculine plural), and (*colectivo*, adjective, masculine plural).
- The *morphological generator* delivers a TL SF for each TL LF, by suitably inflecting it. This module is compiled from a TL MD. In our example, the result would be the text "`los viajes colectivos`".
- The *postgenerator* performs orthographical operations such as contractions (*de* + *el* = *del*, etc.) and is compiled from a rule file.
- The *reformatter* restores the original format information into the translated text.

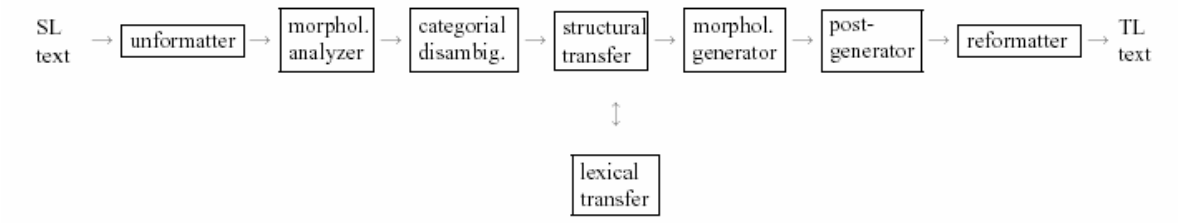

Figure 1: The eight modules of the pt–es machine translation system (see section 3).

The morphological analyser, lexical transfer module, morphological generator, and postgenerator are all based on finite-state transducers (Garrido et al., 1999; Garrido-Alenda et al., 2002).

4. Lexical Disambiguation

Building a lexical disambiguator (part-of-speech tagger) based on HMMs (Cutting et al., 1992) for the SL in a MT system implies: (a) designing or adopting a reduced tagset (set of parts of speech) which groups the finer tags delivered by the morphological analyser into a small set of coarser tags adequate to the translation task; (b) building a representative SL training corpus and manually tagging a portion of it for training (in the case of supervised training) and evaluation; (c) actually training the hidden Markov model on the corpus to obtain the probabilities.

After having used for pt the disambiguator (tagset and probabilities) developed for Spanish–Catalan (a choice which was adequate for initial prototypes), we have just deployed a new pt disambiguator designed as mentioned above.

The tagset used by the pt lexical disambiguator consists of 122 coarse tags (83 single-word and 39 multi-word tags for contractions, etc.) grouping the 2230 fine tags (365 single-word and 1845 multi-word tags) generated by the morphological analyser. The number of different lexical probabilities in the HMM is drastically reduced by grouping words in ambiguity classes (Cutting et al., 1992) receiving the same set of part-of-speech tags: 303 ambiguity classes result. In addition, a few words such as *um* (indefinite article or pronoun) or *ter* (to have, auxiliary verb or lexical verb) are assigned special hidden states. The current disambiguator has been trained as follows: initial parameters are obtained in a supervised manner from a 20,000-word hand-tagged text and the resulting tagger is retrained (using Baum-Welch reestimation as in Cutting et al., 1992) in an unsupervised manner over a 7,800,000-word text. Using an independent 6,600-word hand-tagged text, the observed coarse-tag error rate is 4.89%, with about half of the errors (2.14%) coming from words unknown to the morphological analyser[1].

[1] In the current version, 4.40% of the words were unknown to the morphological analyser

5. Shallow Parsing for Structural Transfer

Many of the structural transfer rules in the Spanish–Catalan system are used without change for pt–es: mainly, all rules ensuring gender and number agreement for about twenty very frequent noun phrases (determinant–noun, determinant–noun–adjective, determinant–adjective–noun, numeral–noun etc.), as in *um sinal vermelho* (pt, masc.) [“a red signal”] ! *una se˜nal roja* (es, fem.). In addition, we have rules to treat very frequent pt–es transfer problems, such as these:

- Rules to choose verb tenses; for example, pt uses the subjunctive future (*futuro do conjuntivo*) both for temporal and hypothetical conditional expressions (*quando vieres* [“when you come”], *se vieres* [“if you came”]) whereas es uses the present subjunctive in temporal expressions (*cuando vengas*) but imperfect subjunctive for conditionals (*si vinieras*).
- Rules to rearrange clitic pronouns (when enclitic in pt when proclitic in es or vice versa): *enviou-me* (pt) ! *me envi´o* (es) [“he/she/it sent me”]; *para te dizer* (pt)!*para decirte* (es) [“to tell you”], etc.
- Rules to add the preposition *a* in some modal constructions (*vai comprar* (pt) ! *va a comprar* (es) [“is going to buy”]).
- Rules for comparatives, both to deal with word order (*mais dois carros* (pt) ! *dos coches m´as* (es) [“two more cars”]) and to translate *do que* (pt) [“than”] as *que* (es).
- Lexical rules, for example, to decide the correct translation of the adverb *muito* (pt) ! *muy*/*mucho* (es) [“very”, “much”] or that of the adjective *primeiro* (pt)! *primer*/*primero* (es) [“first”].

The rules are written in a high-level language (Garrido-Alenda and Forcada, 2001) in the usual *pattern–action* format of lex, where the pattern describes the LFs constituting the chunk which is processed and the action performs the actual transformation of the pattern, with lexical transfer always being implicitly called. The resulting module works left to right, processing always the input prefix of the remaining text which matches the longest pattern, and continuing immediately after the pattern. When input does not match any of the patterns, a LF is translated in isolation and processing continues after it. Left-to-right “state” information may be used to

communicate the information computed during processing of a chunk to other chunks following it.

6. Concluding Remarks

The speed (5600 words/s on a regular desktop PC) and accuracy (around 90%) mentioned above confirm that the shallow-parsing-based strategy previously used by our group to build a Spanish–Catalan MT system is also adequate for `pt–es` MT.

Acknowledgements: Work funded by Portal Universia, S.A. and partially supported by the Spanish Comisión Ministerial de Ciencia y Tecnología through grant TIC2000-1599-CO2-02.

7. References

Arnold, D., Balkan, L., Meijer, S., Humphreys, R., and Sadler, L. (1994). *Machine translation: An introductory guide*. NCC Blackwell, Oxford. available at `http://clwww.essex.ac.uk/~doug/MTbook/`.

Canals-Marote, R., Esteve-Guillen, A., Garrido-Alenda, A., Guardiola-Savall, M., Iturraspe-Bellver, A., Montserrat-Buendia, S., Ortiz-Rojas, S., Pastor-Pina, H., Perez-Antón, P., and Forcada, M. (2001). The Spanish-Catalan machine translation system interNOSTRUM. In *Proceedings of MT Summit VIII: Machine Translation in the Information Age*, pages 73–76. Santiago de Compostela, Spain, 18–22 July 2001.

Cutting, D., Kupiec, J., Pedersen, J., and Sibun, P. (1992). A practical part-of-speech tagger. In Third Conference on Applied Natural Language Processing. Association for Computational Linguistics. Proceedings of the Conference., pages 133–140, Trento, Italy.

Garrido, A., Iturraspe, A., Montserrat, S., Pastor, H., and Forcada, M. L. (1999). A compiler for morphological analysers and generators based on finite-state transducers. *Procesamiento del Lenguaje Natural*, (25):93–98.

Garrido-Alenda, A. and Forcada, M. L. (2001). Morphtrans: un lenguaje y un compilador para especificar y generar módulos de transferencia morfológica para sistemas de traducción automática. *Procesamiento del Lenguaje Natural*, 27:157–164.

Garrido-Alenda, A., Forcada, M. L., and Carrasco, R. C. (2002). Incremental construction and maintenance of morphological analysers based on augmented letter transducers. In *Proceedings of TMI 2002 (Theoretical and Methodological Issues in Machine Translation, Keihanna/Kyoto, Japan, March 2002)*, pages 53–62.

Lesk, M. (1975). Lex—a lexical analyzer generator. Technical Report Technical Report 39, AT&T Bell Laboratories, Murray Hill, N.J.

Radford, A., Atkinson, M., Britain, D., Clahsen, H., and Spencer, A. (1999). *Linguistics: An introduction*. Cambridge Univ. Press, Cambridge.

Tellier, I. (2000). Semantic-driven emergence of syntax: the principle of compositionality upside-down. In *Proc. 3rd Conference on the The Evolution of Language*, pages 220–224, Paris.

Reusing Available Resources for Tagging a Spoken Portuguese Corpus

Amália Mendes, Raquel Amaro, M. Fernanda Bacelar do Nascimento

Centro de Linguística da Universidade de Lisboa
Complexo Interdisciplinar, Av. Prof.Gama Pinto, nº 2, 1649-003 Lisboa
amalia.mendes@clul.ul.pt; ramaro@clul.ul.pt; fbacelar.nascimento@clul.ul.pt

Abstract

This paper discusses the experience of reusing annotation tools developed for written corpora to tag a spoken corpus with POS information. Eric Brill's tagger, initially trained over a written and tagged corpus of 250.000 words, is being used to tag the C ORAL ROM spoken corpus, of 300.000 words. First, we address issues related with the tagset definition as well as the tagger performance over the written corpus. We discuss important options concerning the spoken corpus transcription, with direct impact on the tagging task, as well as the additional tags required. Transcription options allow in some cases for automatic tag identification and replacement, through a post-tagger process. Other cases, like the annotation of discourse markers, are more complex and require manual revision (and eventual listening). Since the final annotation will not only include the POS tag but also the wordform lemma, the paper also addresses issues related to the lemmatisation task. The positive results obtained show that the process of tagging and lemmatising a spoken Portuguese corpus through the reuse of already available resources may constitute an example of how to minimize the costs of such a task, without compromising the results. Finally, we discuss some possible developments to improve the tagger's performance.

1. Introduction

Tagging a spoken corpus with part-of-speech (POS) information presents certain specificities not found in the annotation of written corpora. However, our experience shows that it is possible to attain satisfactory results in spoken texts POS tagging by reusing and adapting resources developed for a written corpus.

The spoken corpus that is actually being tagged has been developed under the project *C-ORAL-ROM: Integrated Reference Corpus for Spoken Romance Languages*[1] – a project of the European Commission addressing spoken speech. This corpus is about 300.000 words and covers several registers: informal, formal, media and phone conversations. Our objective is not only to tag the corpus with POS information, but also to lemmatise the data – increasing the complexity of our task – reusing, whenever possible, already available resources.

We proceeded first by considering the already developed tagset and the training of Eric Brill's tagger over a written and tagged corpus of 250.000 words for the project *Recursos Linguísticos para o Português: um corpus e instrumentospara a sua consulta e análise*[2]. The use of a previously developed resource *Léxico Multifuncional Computorizado do Português Contemporâneo*[3] – LMCPC, a frequency lexicon based on a written 16M words corpus, proved helpful for the lemmatising task, and, hopefully, will also prove to be valuable for the improvement of the tagging task.

2. Tagging a written corpus

We used Eric Brill's tagger (Brill 1993) trained over a written Portuguese corpus of 250.000 words, morphosyntactically annotated and manually revised. Several genres compose this corpus: newspaper (65%), books (20%), magazines (5%) and varia (10%).

The morphosyntactic annotation covered the main POS categories (Noun, Verb, Adjective, etc.) and secondary ones (tense, conjunction type, proper noun and common noun, variable *vs.* invariable pronouns, etc.), but person, gender and number categories were not included, due to limits in time and human resources.

2.1. Some aspects of the tagset definition

The difficult and time-consuming task of deciding between ambiguous categories was avoided by the use of portmanteau tags. Therefore, distinctions such as the one between the indefinite article and numeral for the annotation of the form *um*, *uma*, the one between inflected or non-inflected infinitive verb forms, and the one between some common or proper nouns, were solved by the portmanteau tags /ARTi:NUMc, for the first case, /VB:VBf, for the second, and /Np:Nc for the last.

Some functional distinctions between categories were added when it seemed important for future research. It is the case, for instance, of the distinction between the past participle in compound tenses (/VPP) and the past participle in other contexts (/PPA):

*ele/PES tinha/VAii comprado/**VPP** um/ARTi:NUMc livro/Nc*
*olhos/Nc fechados/**PPA***

[1] *C-ORAL-ROM: Integrated Reference Corpus for Spoken Romance Languages* is being developed by CLUL, under M. Fernanda Bacelar do Nascimento supervising. National C-ORAL-ROM corpora will be distributes by ELDA.

[2] *Recursos Linguísticos para o Português: um corpus e instrumentos para a sua consulta e análise* was developed by CLUL, 2001-2003, under M. Fernanda Bacelar do Nascimento supervising. Corpus available for on-line queries at http://www.clul.ul.pt/sectores/projecto_rld1.html.

[3] The *Léxico Multifuncional Computorizado do Português Contemporâneo* was developed by CLUL, 1997-2000, under M. Fernanda Bacelar do Nascimento supervising. Lexicon available for download at http://www.clul.ul.pt/sectores/projecto_lmcpc.html.

The prepositional, conjunctional, pronominal and adverbial locutions were also tagged, resulting in the following information for each tagged element of the locution: category, element position number and identification number (for cross-reference in an appended list of locutions). The locution identification number is inserted after the tagging process to avoid multiplying the tagset length.

num/LADV1_ 117 instante/LADV1_117
logo/LCONJ1_47 que/LCONJ2_47
à/LPPREP1_003 beira/LPREP2_003
de/LPREP3_003
o/LPRON1_07 qual/LPRON2_07

Since some words sequence constituting a locution may also occur freely, a manual revision was considered necessary for obtaining a maximum success annotation.

The contractions of two lemma were annotated by joining two tags through the sign '+' (dos/**PREP+ARTd**), and the wordforms connected by hyphen received two tags also connected by hyphen (disse-me/**Vppi-CL**). These two tagging options have the effect of expanding the total tagset into an indefinite number (from a minimum of 54 tags, to a maximum of more than 204), by combining several tags that are recognised by the tagger as a new single one.

2.2. Some comments on the results

After the tagger training and after the automatic tagging of a written corpus, the results show two aspects that will have to be considered in the future: first, some difficulties in the automatic tagging of the locutions and, second, the lack of identification of certain words.

In order to respond to the locution tagging problems, two solutions are being studied: on one hand, the inclusion of the locution identification number in the tagset, bearing in mind all the subsequent problems derived from the hudge tagset length increase (note that the current prepositional locutions list alone exceeds 484 locutions); on the other hand, the conception of a post-tagging tool for the locutions annotation is being considered.

In order to respond to the second case, the future development will be to insert in the annotation process the LMCPC, extracted from a 16 million words corpus (considerably larger than the used training corpus – 250.000 words), in which a large set of wordforms occurs (around 140.000).

3. Tagging and lemmatising a spoken corpus

The spoken corpus was tagged with the tool described in the previous section. In spite of having been trained over a written corpus, and surprisingly against our expectations, the results achieved were very satisfactory, with a success rate of 91,5%.

Nevertheless, some post-tagging adaptations had to be made in order to achieve the established spoken corpus annotation.

3.1. Specific spoken language phenomena

Some characteristic spoken language phenomena, such as word repetition and truncated words don't seem to affect the tagger performance, either statistically, either in terms of contextual rules.

However, the tagger identifies and tags the prosodic marks (question marks, slashes, and so on) as punctuation, making it necessary to automatically remove these tags in a following stage.

Besides the previous phenomena mentioned, and due to the specific transcription guidelines used in the C-ORAL-ROM project, there are several other phenomena that required tagset adaptations:

a) extra-linguistic elements;
transcription: hhh; Tag: **EL**
b) fragmented words;
transcription: beginning with &; Tag: **FRAG**
c) words impossible to transcribe (impossible to hear, for example);
transcription: xxx; Tag: **Pimp**
d) paralinguistic elements, such as *hum*, *hã* and onomatopoeias.
Tag: **PL**
e) discourse markers, such as *pá*, *portanto*, *pronto*;
Tag: **MD**
f) discoursive locutions, such as *sei lá*, *estás a ver*, *quer dizer*, *quer-se dizer*.
Tag: **LD**

In the cases described in (a), (b) and (c), the adopted specific transcription allows for automatic tag identification and replacement, through a post-tagger process. The same process is applied in the cases described in (d), since there is a finite list of symbols representing paralinguistic elements.

The discourse markers (cf. (e) an (f)) present a more difficult case, since they correspond to forms that also belong to other word categories: for instance, *não sei* is automatically tagged as *não/ADV sei/Vpi*, making a manual revision (and eventual listening) necessary in order to decide whether the form is a discourse marker or not.

The tagging of proper nouns is, on the contrary, simplified in the spoken corpus tagging process, since proper nouns are the only forms transcribed with initial capital letter.

The development stage that follows consists in the Eric Brill tagger's training over a manually revised spoken corpus, as well as in the exploitation of the tagger contextual rules in order to optimize its performance. Amongst other things, we aim at improving the locution tagging process since locutions account for an increase of around 2% of the error rate.

3.2. Lemmatisation

The final format of the spoken corpus annotation includes, for each form, not only the POS tag, but also the correspondent lemma:

word\LEMMA\tag.

The lemma is extracted automatically from the LMCPC: the form is searched for in the lexicon, the

correspondent(s) lemma(s) is(are) found and placed near the form, in a process parallel to the automatic tagging. So, it is possible for a wordform to be attributed several lemma, requiring thus manual lemma selection.

In the future, with the foreseen improvement of the success rate, we expect to be able to trust the automatic POS tagging in order to cross information with the lexicon POS data and select the proper lemma for each wordform.

In the cases from (a) to (d) described above, and in the proper nouns case, the lemma is considered empty, since it is clear that there is no lemma for that expressions.

In the case of locutions, since the lemma is the locution set, there is no need for the locution identification number. Locution lemmatisation made it necessary to develop a tool to automatically compose the desired lemma format:

o\O_QUAL\LPRON qual\O_QUAL\LPRON

We present next a tagged and lemmatised extract from one of the conversations of the corpus:

```
*FER: e\E\CONJc ela\ELA\PES / <
como\COMO\ADV reagiu\REAGIR\Vppi > ?$
*BEN: [<] < &eh\-\FRAG / &hum\-\FRAG > /
reagiu\REAGIR\Vppi muito\MUITO\ADV
bem\BEM\ADV //$ < hhh\-\EL >$
*FER: [<] < hhh\-\EL >$
*BEN: / reagiu\REAGIR\Vppi
muito\MUITO\ADV bem\BEM\ADV //$
começa\COMEÇAR\Vpi na\EM+A\PREP+ARTd
segunda-feira\SEGUNDA-FEIRA\Nc //$
entretanto\ENTRETANTO\ADV / eu\EU\PES
disse-lhe\DIZER-LHE\Vppi-CL que\QUE\CONJs
/$
*AUG: < então\ENTÃO\ADV e\E\CONJc
as\A\ARTd férias\FÉRIA\Nc > ?$
*BEN: / [<] < que\QUE\CONJs não\NÃO\ADV
sabia\SABER\Vii > se\SE\CONJs / eu\EU\PES
podia\PODER\Vii na\EM+A\PREP+ARTd
segunda-feira\SEGUNDA-FEIRA\Nc /
por\POR\PREP ter\TER\VB os\O\ARTd
conselhos\CONSELHO\Nc //$
de\DE_MANEIRA_QUE\LCONJ
maneira\DE_MANEIRA_QUE\LCONJ
que\DE_MANEIRA_QUE\LCONJ / a\A\ARTd
&senho\-\FRAG / ficou\FICAR\Vppi
então\ENTÃO\ADV combinado\COMBINAR\PPA /
eu\EU\PES telefonar\TELEFONAR\VB:VBf
para\PARA\PREP lá\LÁ\ADV /$
```

4. Final comments

The development of tagged corpora is, definitely, a human resources and time-consuming task.

The process of tagging and lemmatising a spoken Portuguese corpus through the reuse of already available resources here presented may constitute an example of how to minimize the costs of such a task, without compromising the results.

Summing up, this complex process, besides the spoken corpus constitution and transcription, has consisted in:

i) the definition of a suitable tagset and tagging options;

ii) the adaptation of a written tagged corpus to the desired tagset;

iii) the training of Eric Brill's tagger;

iv) the automatic replacement and/or withdraw of the tags, according to the specific spoken language phenomena transcription;

v) the creation of a tool for the automatic lemmatisation of the corpus, using an already existent lexicon;

vi) the creation of a tool for the automatic lemmatisation of the locutions elements;

vii) and, at last, the manual revision of the final result.

The following stage will consist in the Eric Brill's tagger training over the resulting spoken corpus, manually revised.

We hope to achieve significant improvements regarding the performance of spoken corpus automatic tagging.

5. Acknowledgements

We want to thank several colleagues for their help in preparing and revising this paper: João Santos, Rita Veloso, Florbela Barreto, Sandra Antunes and Luísa Alice Santos Pereira.

6. References

Bacelar do Nascimento, M. F. (2001) "Um novo léxico de frequências do português" in Biblos, vol. de *Homenagem ao Professor Herculano de Carvalho* (no prelo).

Brill, E. (1993) *A corpus-based approach to Language Learning*, PhD thesis, University of Pennsylvania, Departement CIS.

Cresti, E., *et al*. (2002) "The C-ORAL-ROM Project. New methods for spoken language archives in a multilingual romance corpus LREC", in M. C. Rodrigues & C. Suarez Araujo (a cura di), Proceedings of the *Third International Conference on Language Resources and Evaluation*, Paris: ELRA, vol. 1, pp. 2-10.

Moreno, A. & J. M. Guirao (2003) "Tagging a spontaneous speech corpus of Spanish" in *Proceedings of RANLP-2003 – Recent Advances in Natural Language Processing* , (forthcoming).

Van Eynde, F., J. Zavrel & W. Daelemans (2000) "Part of Speech Tagging and Lemmatisation for the Spoken Dutch Corpus", in Gavrilidou, M. et al. (eds.) *Proceedings of the Second International Conference on Language Resources and Evaluation. European Language Resources Association*, Paris, 1427-1433.

Easy Automatic Terms Acquisition with ATA and Galinha

Joana L. Paulo, David M. de Matos, Nuno J. Mamede

L2F – Spoken Language System Laboratory
INESC-ID Lisboa / IST, Rua Alves Redol 9, 1000-029 Lisboa, Portugal
{joana.paulo, david.matos, nuno.mamede}@l2f.inesc-id.pt

Abstract

ATA (Paulo, 2002) is a system for Automatic Term Acquisition that takes a text from a specific field and analyses it in order to decide which of the detected nouns and noun phrases ought to be considered terminological units. ATA uses a well known architecture (Daille, 1996), taking advantage of the system's modularity which lets us modify each module independently, thus improving the whole system. Currently, ATA is being evaluated over a Portuguese nautical corpus: in the final version of the article, evaluation results will be discussed. Galinha (Galaxy Interface Handler) (Matos, 2002) is a system that integrates multiple linguistic resources and tools. Galinha enables easy module integration and testing of prototypical configurations, thereby reducing the effort and backtracking usual in the construction of modular applications. Joining ATA and Galinha allowed us to provide a web graphical interface to make it easier to automatically acquire terms while accessing to the intermediate results of each module.

1. ATA

ATA is divided into three main modules (see figure 1): linguistic enrichment and selection of those units that may be terms due to their syntactical categories; enrichment of candidates with corpora-based statistical information; and decision about whether they are terms and should be proposed to the user.

In the linguistic analysis sequence, `SMorph` (Aït-Mokhtar, 1998) lemmatizes and annotates morphologically the text using a dictionary. Then, `PAsMo` (Paulo, 2001) rewrites the text according to recomposition and correspondence rules. `PAsMo` also groups the words in phrases. The syntactic analyser `SuSAna` (Batista, 2002) groups phrase constituents. A filtering tool, `GeTerms`, selects those structures that, given their syntactical features, can be terms. That is, for Portuguese, all noun phrases founded on the text.

After that, in the statistical sequence, `Anota` enriches the selected expressions with their statistical information.

Finally, the `Decision` module evaluates the candidate lists, producing the final results to be presented to the user. This is done, by comparing the occurrence of the candidate term in the specialized text and its occurrence on a newspaper corpora analyzed by the same chain process.

The output is a list of words that can be terms. This list may be divided into two sets, both of which may be empty: the first set contains simple term candidates, identified in the text; the second set contains compound term candidates.

Even though the two types of terms to be detected (simple and compound) have different characteristics, we handle them in the same way, by delegating on the grammar the responsibility for customized processing. In an hybrid system such as this, high-frequency terms will be detected statistically, while low-frequency terms will be detected through the grammar of terms. Afterwards, it will be necessary to review the candidate terms. This step is always necessary since not even human annotators eventually find an agreement about the terms in a text.

For evaluation purposes, we analyzed a 114 thousand-word corpus and asked the system for its terms. Then we compare the given list with a list of terms manually detected by linguists. For now we are still running tests and we are trying to experimentally find the best parameters according to which we will say that a noun phrase or noun is a term: the minimal number of occurrences that a term should have and the multiplicative factor when comparing the occurrence on corpora to the occurrence on the specialized text.

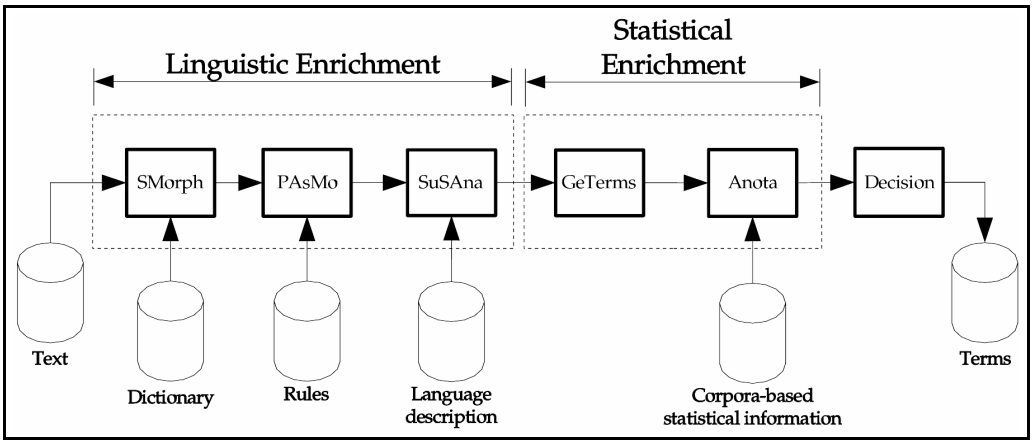

Figure 1: ATA's architecure

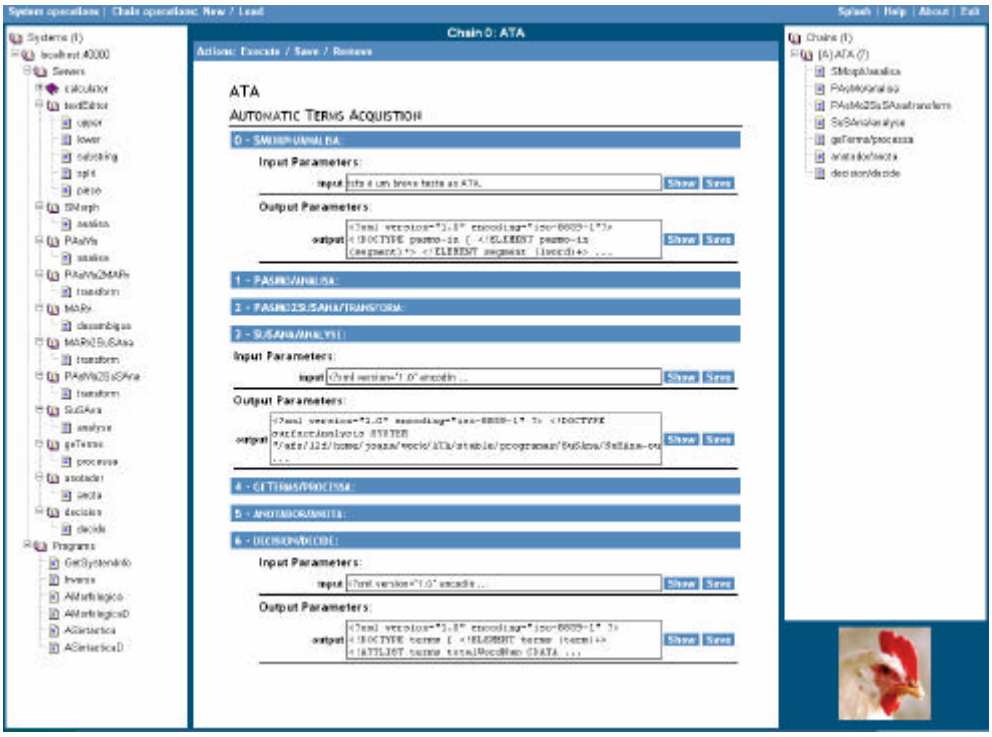

Figure 2: Galinha as a graphical interface for ATA.

2. Galinha

Galinha is a web-based user interface for building modular applications that enables users to access and compose modules using a web browser. Since Galinha works with chains, to include ATA we had to write the corresponding chain and to connect the modules. The main linguistic analysis modules used by ATA were already available through Galinha. Since they accept/produce different data formats, two additional modules were needed to provide data format conversion.

All that was needed to connect the two new modules was to include an existing XSLT (W3C) processor into Galinha. For that, we had only to write a wrapper to call external applications. The wrapper was so simple we were able to generalize it to use any future application we may need.

In figure 2, we show Galinha with ATA's definition: on the left, we have available systems; on the right, we have a chain where all the relevant services are connected and can be executed; in the middle, we can give the text that we want to analyse and - after the results are produced - browse each module's input and output. Since the final result depends on intermediate results, their availability makes evaluation easier.

3. Conclusions

We wanted to automatically extract terms and to create some graphical interface to the system. After designing ATA and its modules, we used Galinha to integrate the modules and provide a graphical interface. As Galinha is easy to use, and adding new modules is also easy, we presented a graphical interface to our automatic terms acquisition system that gives us access to intermediate results and, besides that, can be made available to anyone on the web.

4. References

Aït-Mokhtar, S. (1998). *L'analyse Présyntaxique en une seule étap*. Ph. D. thesis, Université Blaise Pascal, GRIL, Clermont-Ferrand, France.

Batista, F. and Mamede, N. (2002). SuSAna: Módulo Multifuncional de Análise Sintáctica de Superfície. In J. Gonzalo and A. Peñas and A. Ferrández (eds.) *Proc. Multilingual Information Access and Natural Language Processing Workshop*, IBERAMIA 2002, Sevilla, Spain, 29-37.

Daille, B. (1996) *Study and implementation of combined techniques for automatic extraction of terminology*. The balancing act combining symbolic and statistical approaches to language, 49-66.

Matos, D. M. de et al. (2002). Empowering the User: a Data Oriented Application-building Framework. In *Adj. Proc. of the 7th ERCIM Workshop "User Interfaces for All"*. Chantilly, France. European Research Consortium for Informatics and Mathematics, 37-44.

Paulo, J. L. (2001). PAsMo – Pós-Análise Morfológica. *Technical Report*. Lisboa, Portugal.

Paulo, J. L. et al. (2002). Using Morphological, Syntactical, and Statistical Information for Automatic Term Acquisition. In E. Ranchhod and N. Mamede (eds.), *Advances in Natural Language Processing, Third International Conference, Portugal for Natural Language Processing (PorTAL)*. Faro, Portugal. Springer-Verlag, LNAI 2389: 219-227.
See: `www.w3.org/Style/XSL`.

World Wide Web Consortium (W3C). The Extensible Stylesheet Language (XSL).

Reusing Linguistic Resources: a Case Study in Morphossyntactic Tagging

Ricardo Ribeiro†, Nuno J. Mamede*, Isabel Trancoso*

†INESC-ID Lisboa/ISCTE
*INESC-ID Lisboa/IST
Spoken Language Systems Lab
R. Alves Redol, 1000-029 LISBON, Portugal
{Ricardo.Ribeiro, Nuno.Mamede, Isabel.Trancoso}@inesc-id.pt

Abstract

This paper describes several issues concerning the reusability of linguistic resources, with special emphasis on morphossyntactic tagging. Ribeiro (2003) presents a morphossyntactic tagging system with a modular architecture. What are the consequences of changing a module of this system? How difficult would be to integrate the morphossyntactic tagger in other systems? These are some of the questions that are addressed by this paper, where possible approaches to the problems that may appear are also discussed.

1. Introduction

One of the major problems related to natural language processing is the availability of manually annotated resources. In fact, this question can be posed concerning all kinds of resources: corpora, lexica and tools. Yet, nowadays, the relevance of this problem, even for the Portuguese language, seems to be diminishing, but a new one arising: the usability of the existing resources (Matos et al.,2003; Jing and McKeown, 1998; Olsson et al., 1998).

In (Ribeiro, 2003; Ribeiro et al., 2003) is presented a morphossyntactic tagger that followed a modular approach. The strategy adopted by this system, motivated by the fact that neolatin languages, such as Portuguese, are highly inflectional when compared with English, consists of two sequential steps: morphological analysis and ambiguity resolution. Given such architecture, one would expect that replacing one of the modules would not be a difficult task.

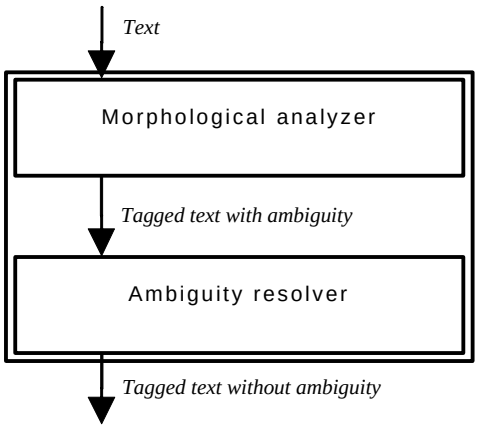

Figure 1. Morphossyntactic tagger architecture.

2. Reusability problem

The reusability problem appeared when we tried to use MARv (Ribeiro, 2003), the morphossyntactic disambiguation module, in the automatic term acquisition (ATA) system presented in (Paulo, 2003). In the ATA system, the morphological analysis is performed by SMorph (Aït-Mokhtar, 1998) and followed by the post morphological analysis tool PAsMo (Paulo, 2001), whilst the morphological analysis module of the morphossyntactic tagging system is Palavroso (Medeiros, 1995). Since there are some conceptual differences between these two systems some adaptations were needed. Two major problems were identified:

- the tokenization performed by the two systems was different;
- the tagsets, besides being different, were ruled by divergent principles.

MARv's architecture comprehends two submodules: a linguistic-oriented disambiguation rules module and a probabilistic disambiguation module. Considering the differences between the two morphological analyzers, substituting Palavroso by SMorph/PAsMo demanded some changes in both modules. Concerning the disambiguation rules module, the focus was on rule adaptation. Concerning the probabilistic disambiguation module, the modifications consisted in the development of new probabilistic models.

3. Used corpus

The corpus used to develop these models was built in the LE-PAROLE European project (Bacelar et al., 1997) in which harmonized reference corpora and generalist lexica were built according to a common model for the 12 European languages involved. This corpus was morphossyntactically tagged using Palavroso and manually disambiguated. The tagset had about 200 tags with information that varied from grammatical category to morphological features that could be combined to form composed tags (resulting in about 400 different tags). This corpus was developed to be part of the core of a set of written language resources for the European Community countries. In other words, its main purpose is to be reused.

4. Adopted approach

In order to develop new models for the probabilistic module of MARv, the LE-PAROLE corpus was used.

But since this corpus was tagged with Palavroso, the tokenization and the tagset problems previously identified arouse in the corpus reuse.

The approach to these problems was a semi-automatic solution that comprehends four steps:

- Tagging of the corpus using SMorph/PAsMo;
- Identification of the situations where occur contraction or expansion of tokens identified by Palavroso. For example, SMorph/PAsMo gives "é sintetizada" or "cidade - campo" as tokens, where Palavroso gives "é", "sintetizada" and "cidade", "-", "campo" as tokens;
- Identification of a mapping between the tagsets;
- Development of an interface based on a rule set obtained from the previously identified situations. Whenever it was not possible to apply a rule the automatic process was interrupted and the user was queried about how to solve that particular situation.

Although effective, this approach was very slow, since the rule set did not cover several situations and it was not possible to define a function from the Palavroso tagset to the SMorph/PAsMo tagset.

5. References

Aït-Mokhtar, S., (1998). *L'analyse présyntaxique en une seule étape*. PhD Thesis, Université Blaise Pascal, Clermont-Ferrand, GRIL.

Bacelar, F., J. Bettencourt, P. Marrafa, R. Ribeiro, R. Veloso and L. Wittmann (1997). LE-PAROLE – Do corpus à modelização da informação lexical num sistema multifunção. In *Actas do XIII Encontro da APL*. Portugal.

Jing, H. and K. McKeown (1998). Combining multiple, large-scale resources in a reusable lexicon for natural language generation. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics* (pp. 607–613).

Matos, D., J. L. Paulo and N. Mamede (2003). Managing Linguistic Resources and Tools. In Mamede, N., J. Baptista, I. Trancoso and M. das Graças Volpe Nunes, editors, *Proceedings of the 6th International Workshop on Computational Processing of the Portuguese Language (PROPOR 2003)*, volume 2721 of *Lecture Notes in Artificial Intelligence* (pp. 135–142). Springer.

Medeiros, J. C. (1995). *Processamento Morfológico e Correcção Ortográfica do Português*. Master's thesis, Instituto Superior Técnico – Universidade Técnica de Lisboa, Portugal.

Olsson, F., B. Gambäck and M. Eriksson (1998). Reusing Swedish Language Processing Resources in SVENSK. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, volume *Workshop on Minimizing the Effort for Language Resource Acquisition* (pp. 27–33). ELRA.

Paulo, J. L. (2001). *PAsMo – Pós-Análise Morfológica*. Technical report, L²F – INESC-ID Lisboa, Portugal.

Paulo, J. L. (2003). *Aquisição Automática de Termos*. Master's thesis, Instituto Superior Técnico – Universidade Técnica de Lisboa, Portugal. (to appear).

Ribeiro, R. (2003). *Anotação Morfossintáctica Desambiguada do Português*. Master's thesis, Instituto Superior Técnico – Universidade Técnica de Lisboa, Portugal.

Ribeiro, R., L. Oliveira and I. Trancoso (2003). Using Morphossyntactic Information in {TTS} Systems: Comparing Strategies for European Portuguese. In Mamede, N., J. Baptista, I. Trancoso and M. das Graças Volpe Nunes, editors, *Proceedings of the 6th International Workshop on Computational Processing of the Portuguese Language (PROPOR 2003)*, volume 2721 of *Lecture Notes in Artificial Intelligence* (pp. 143–150). Springer.