PROPAUSE: A SYNTACTICO-PROSODIC SYSTEM DESIGNED TO ASSIGN PAUSES

D*avid* **C***asacuberta*, Lourdes* **A***guilar**,* **R***afael* **M***arín***
Departament de Filosofia*, Departament de Filologia Espanyola**
Universitat Autonoma de Barcelona, Bellaterra, Barcelona, Spain
{david, lourdes, rafa}@liceu.uab.es

ABSTRACT

In this study, a PROLOG-based computational tool designed to assign pauses in Spanish texts is proposed. Our purpose is to develop a prosodic segmentation algorithm suitable to be implemented in a text-to-speech system for Spanish.

By means of the analysis of a corpus of read texts in Spanish, prosodic and syntactic factors guiding the location of orthographically unmarked pauses are identified. These factors are used to design a computational model for assigning pauses in unrestricted texts.

The performance of the system has been assessed by means of a comparison between its suggested segmentation and natural speech. The obtained results indicate that the system is able to capture empirical facts.

1. INTRODUCTION

In this study, we propose a PROLOG-based computational tool for assigning pauses in Spanish texts. Our main purpose is to build a module suitable to be implemented in a text-to-speech (TTS) system.

Efforts in the area of prosodic modeling are being done so as to improve the global quality of TTS systems. To achieve this goal, several prosodic segmentation algorithms have been proposed from different theoretical approaches: from the assumption of a matching between syntactic and prosodic constituents ([1]) to the neglect of the effect of syntax performing a pure morphological analysis ([2], [3], [4]).

Nevertheless, in our opinion, the most efficacious treatment is the one that combines syntactic and non-syntactic factors ([5], [6], [7]). Following this approach, it is hypothesized that suprasegmental phenomena cannot be derived exclusively from one source of information.

We thus assume that pause assignment has to be modeled by means of an adequate interface between, at least, syntax and prosody. However, we do not propose a separate syntactic analysis followed by or complemented with a prosodic analysis. Instead, a procedure including at the same time both syntactic and prosodic information is proposed.

2. MODE*L* ING PAUSE *L*OCATION

To obtain knowledge about the prosodic breaking of read texts, an experimental analysis has been carried out to determine banned and possible positions of orthographically unmarked pauses. Based on the empirical results obtained in [8], an algorithm of pause assignment is developed here.

The main units used in the system are: Phonic Group (PG), which is considered to be, following classical phonetic descriptions, the speech generated between two pauses; Stress Group (SG), to be meant as a stressed word plus the optional unstressed words preceding it; and Categorial Stress Group (CSG), a unit that can be roughly defined as a syntactically labeled SG.

The syntactic label of the SG corresponds to the lexical category of the first element in the group. To illustrate this, a CSG labeling is offered in (1), where *qg* stands for quantifier group, *vg* for verb group, *pg* for prepositional group and *ag* for adjective group.

(1) [Las mujeres]qg [de las casas]pg [inundadas]ag [aparecieron]vg [de repente]pg

'Women from flooded houses appeared suddenly'

In Table I, the list of CSGs with their associated syntactic head is presented:

CSG	*s**y**ntactic* **h***ead*
ag	adjective
adg	adverb
cg	conjunction
ccg	coordinating conjunction
clg	clitic
gg	gerund
ig	infinitive
ng	noun group
pg	preposition
ptg	participle
qg	quantifier
vg	verb

Table I. *List of* **CSG***s and t***h***eir s***y***ntactic* **h***ead.*

It should be noted that some CSGs are formed by just one element (for instance, vg, ng or ag) whereas others have more than one element (clg, ccg or qg). As for the latter, we will refer to the first element as head and to the last one, as modifier. Related to this, it is worth pointing out that some CSGs only accept some lexical categories

as modifiers. Thus, for example, a clg only accepts a verb as modifier (see [8] for a more detailed description).

By means of these basic units, prosodic and syntactic factors guiding pause location have been identified.

2.1. *Length of the sentence*

The length of the sentence, measured in terms of the number of SGs, determines the appearance of a pause. At least six SGs are needed to insert a pause, while the existence of ten or more groups requires the segmentation of the sentence. As a consequence, the pause is optional in a sentence of between seven and nine SGs.

2.2. *Location of pauses in the sentence*

Related to the tendency of speakers to balance the length of phonic groups ([9]), pauses tend to appear in the middle of a sentence. In particular, pauses that generate PGs of unbalanced length have been found in a negligible amount in the analysis of the corpus.

2.3. Type of CSG

It has been found that the type of CSG, in other words, its syntactic properties, favors or blocks the appearance of a pause before it. According to this, data allow us to establish the following hierarchy, representing the probability of a pause to appear before a particular CSG: ccg > vg > cg > gg > clg > adg > pg > qg > ptg > ig > ng > ag.

2.4. Syntactic dependence

There is evidence that some combinations of CSGs cannot be split by a pause. According to this, a battery of syntactic restriction rules interacts with the hierarchy of CSGs. In (2), some of these restriction rules are presented:

(2) a. [H:X, M:n], [H:a]
b. [H:X, M:a], [H:n]
c. [H:v], [H:pt]
d. [H:a], [H:cc, M:a]

Each rule, where *H* stands for head, **M** for modifier and *X* for a wild variable, represents a possible pair of CSGs, between which a pause is impossible.

3. THE SYSTEM

From the empirical results, a PROLOG-based computational tool has been developed to locate orthographically unmarked pauses in unrestricted Spanish texts.

A main module loads all the subprograms needed, asks for a file to be paused, and directs the text inside the file to the different modules they have to work with: a Word Categorizer, a CSG Categorizer, a CSG Counter and a Pause Searcher.

3.1. Word Categorizer

This module takes a text, categorized by an external tagger which uses a large set of grammatical categories ([10]), and for the sake of having a less complex system, translates these categories into those ones used by ProPause, according to the list in Table I.

3.2. CSG Categorizer

The main function of the CSG Categorizer is to automatically divide the text into CSGs. To do this, the program first breaks the text according to the presence of punctuation signs, and afterwards it divides the resulting utterance into CSGs, using the information provided by the Word Categorizer and the predicate *stress***(C***ategor***y**,**S***tress***)**, that gives for each category its properties with respect to stress. The distinction between open and closed categories serves us to determine whether a word is stressed or unstressed.

3.3. CSG Counter

This module decides whether an utterance has pauses or not using length constraints.

If the number of CSGs is fewer than six, no pause is allowed; therefore, no more functions are invoked and the program moves on to the next utterance. As said before, if the number of CSGs is between six and nine, the pause is optional; if it is greater than nine, the pause is mandatory.

Depending on the optional or mandatory nature of the pause, a different hierarchy of CSGs is loaded. The hierarchy for mandatory pauses contains all the CSG types ordered according to their probability to present a pause in front of them; the optional one, however, just contains a subset of CSGs, those that are able to admit a pause when the utterance length is among six and nine CSGs.

3.4. Pause Searcher

The Pause Searcher has the function of assigning reasonable pauses and preventing banned ones. With the CSG hierarchy loaded in the previous module, the program looks for the appropriate place to put a pause. First, it applies the criterion stating that a pause cannot appear before 3 SGs of the beginning of an utterance, and after 3 SGs of its end; and second, it uses the CSG hierarchy to find the best place for the pause. Once a

candidate for a pause is found, the module checks if any of the restriction rules prevents the presence of a pause.

If restriction rules invalidate a suggested pause location, backtracking is applied and the program proposes the next member in the CSG hierarchy. The process is recursively invoked until the utterance is divided into PGs having either less than six CSGs or between six and nine without any possible pause according to syntactic requirements.

3.5. O*utput* M*odule*

Once the text has been paused, the Output Module produces a new version of the text in which the location of pauses is displayed.

4. EVA*L*UATING T*H*E SYSTEM

In order to assess the performance of ProPause, a comparison between its suggested segmentation and natural speech has been made. The latter was produced by a speaker, who read aloud a literary text, including syntactically and prosodically varied sentences, composed of 4979 words. The reading was transcribed with respect to prosodic boundaries, and the deviation of pauses assigned by the system and those made by the speaker has been appraised.

Data referred to punctuation signs have been excluded of the computation. In total, 4201 possible pause locations have been compared: 4979 word boundaries minus 578 orthographically marked pauses.

Results in Table II show the degree of agreement between the human speaker and ProPause, with respect to phrasing.

		Human speaker		
	Pauses	Realized	Not realized	Total
ProPause	Realized	56	73	129
	Not realized	105	—	
	Total	161		

Table II. **D***egree of agreement between t***h***e* **h***uman speaker and* **P***ro***P***ause wit***h** *respect to p***h***rasing***.**

These results, however, are of limited value since a difference between the speaker and the algorithm does not necessarily imply a mistake on the part of the latter. Sentences (3) and (4), in which *$* indicates the presence of a pause, illustrate this:

> (3) Todo cuanto me rodeaba $ parecia haberse transformado mientras me levantaba con la manecilla de oro entre mis dedos.
>
> (4) Todo cuanto me rodeaba parecia haberse transformado $ mientras me levantaba con la manecilla de oro entre mis dedos.
>
> 'Everything around me seemed to have been transformed while I stood with the little golden key in my fingers'

Both (3), paused by the speaker, and (4), paused by the system, are equally acceptable. All cases of discrepancies between the output of the system and naturally produced prosody where the two versions are acceptable have been marked as reasonable by an expert.

agreement	equivalent	non-equivalent
56	49	24
reasonable pauses		non-reasonable pauses
105		24

Table III. **C***ategories found in t***h***e comparison between t***h***e output of t***h***e s***y***stem and naturalp rosod***y**.

Table III gives the results grouped in three categories: agreement between the speaker and the algorithm, difference on segmentation yielding equivalent phrasings, and difference on segmentation resulting on a wrong decision on the part of the system. When phrasing is equivalent, we are dealing with reasonable pauses, that is, pauses that are acceptable to a listener.

The algorithm matches 43.41% of the prosodic boundaries made by a speaker, and predicts 81.31% of reasonable pauses. Besides, 18.6% of the cases remains to be explained.

In order to find the responsible factors, errors in phrasing were verified. It has been found that the majority of cases (11 cases from a total number of 24) appear between a personal pronoun with a subject function and a verb. This evidences a mismatch between syntactic trends as the tendency to locate a pause between a subject and its predicate, and prosodic ones such as length restrictions affecting syntactic constituents, in the prediction of prosodic boundaries.

Additionally, some of the labels used by ProPause have been found insufficient to cover some linguistic behaviors; for instance, the label associated to *que* ('that'), always treated as a conjunction has been revealed inadequate to cover noun phrase behavior (4 cases).

Likewise, a problem that relates text-processing, sentence modality and phrasing arises. In this study, any differences in prosodic segmentation due to the effect of sentence modality (declarative, interrogative or exclamative) were considered. And, nevertheless, in the corpus used for the evaluation, the segmentation done by the speaker in exclamative sentences was inequal than

the one performed in declarative sentences, being an error source (3 cases).

Finally, questions concerning semantic information or complex syntactic structures disentangle the remaining cases.

These results indicate that most ProPause phrasing errors could be solved without changing the global structure of the system. A large proportion of mistakes in the resulting pause location will be solved with a more accurate word labeling, and with the addition of new syntactic restriction rules. On the other hand, it will be worth adjusting the lexical information offered by the tagger with the one used by the algorithm.

With respect to the performance of the system compared to other prosodic segmentation algorithms, it can be said that its degree of accuracy is correct, although lower than those obtained in other works.

Using learning procedures, in [6] 94.2% correct predictions of phrase boundaries for Mexican Spanish are achieved; the complex syntactic system presented in [5] matched 80% of the primary boundaries for English. However, it should be noted that both the methodology used to obtain the algorithm and the evaluation procedure are different, and thus comparisons are only approximative.

5. CONCLUSIONS

Results suggest that the system, although being in a preliminary phase of development, provides an adequate treatment of text prosodic segmentation. Decisions made by ProPause always respect length conditionings and it is in the domain of optionality where some divergences between a phrasing made by a speaker and a phrasing made by the algorithm are found.

On the one hand, the proposed methodology prevents the existence of a certain type of non-reasonable pauses, such as those ones appearing between the article and the noun, or between a clitic and a verb. In this sense, it is worth noting that a pause cannot appear inside a CSG, a fact that accounts for in a natural way the absence of pauses between an unstressed word and a stressed one.

On the other hand, the syntactic labeling of a prosodic unit (SG) allows us to include syntactic effects in the treatment of prosodic segmentation.

By means of CSG, unit that bears certain resemblance to the idea of parsing by chunks proposed in [11], it is possible to build an algorithm that covers in an adequate way the relations between syntax and prosody as far as pause location is concerned, and that avoids problems related to syntactic recursion and complexity. This latter fact makes the system suitable to be incorporated in a TTS system for Spanish.

ACKNOWLEDGEMENTS

We would like to thank Sergio Balari for his valuable comments. R. Marin's work has been partially supported by a grant from Generalitat de Catalunya (AP-LP/95-2704).

REFERENCES

[1] Frenkenberger, S., B. Schnabel, M. Alissali and M. Kommenda (1994). "Prosodic parsing based on parsing of minimal syntactic structures", **The** ***S****econd* **ESCA/IEEE** ***W****orks****h****op on** ***S****peec****h*** ***S****y****n****th****e****sis,*** New Paltz, USA, 143-146.

[2] Emerard, F., L. Mortamet, and A. Cozannet (1992). "Prosodic processing in a text-to-speech synthesis system using a database and learnig procedures". In G. Bailly and C. Benoit (ed.) ***T****alking* ***M****ac****h****ines****:*** ***Th****eories,* ***M****odels and* ***D****esigns,* Amsterdam, Elsevier Science Publishers, 225-254.

[3] Castejon, F., G. Escalada, L. Monzon, M. A. Rodriguez and P. Sanz (1994). "Un conversor texto-voz para el espanol", ***C****omunicaciones de* ***T****elefónica* **I+D**, 5(2): 114-131.

[4] Lopez, E. (1993). ***E****studio de técnicas de procesado lingüístico* **y** *acústico para sistemas de conversión texto-vo****z*** *en español basados en concatenación de unidades,* Doctoral Dissertation, E.T.S.I. de Telecomunicacion, Universidad Politecnica de Madrid.

[5] Bachenko, J. and E. Fitzpatrick (1990). "A computational grammar of discourse-neutral prosodic phrasing in English", ***C****omputational Linguistics,* 16(3): 155-170.

[6] Hirschberg, J. and P. Prieto (1996). "Training intonational phrasing rules automatically for English and Spanish text-to-speech", ***S****peec****h*** ***C****ommunication,* 18(3): 283-290.

[7] Gili, B. and S. Quazza (1996). "A prosodic parser for an Italian text-to-speech system", ***P****roceedings of* ***t****h****e*** *X****II*** ***C****ongreso de la* ***S****ociedad* ***E****spañola para el* ***P****rocesamiento del lenguaje* ***N****atural,* Sevilla, 189-208.

[8] Marin, R., L. Aguilar and D. Casacuberta (1996). "El grupo acentual categorizado como unidad de analisis sintactico-prosodico". In C. Martin Vide (ed.), *Lenguajes* ***N****aturales* **y** *Lenguajes* ***F****ormales,* XII, PPU, Barcelona, 487-494.

[9] Nespor, M. and I. Vogel (1983). "Prosodic Structure above the Word". In A. Cutler and R. D. Ladd (eds.) ***P****rosod****y.*** ***M****odels and* ***M****easurements,* Heidelberg, Springer Verlag, 123-140.

[10] Casacuberta, D., R. Marin and L. Aguilar, (in preparation). "ProTaggeras: A speech processing oriented tagger for Spanish", Universitat Autonoma de Barcelona.

[11] Abney, S. (1992). “Parsing by chunks”. In R. C. Berwick, S. Abney and C. Tenny (eds.), **P***rinciple***-B***ased* **P***arsing***: C***omputation and* **P***s***y***c***h***olinguistics,* 257-278.