A Syntactical Model of Prosody as an Aid to Spoken Dialogue Systems in Italian Language

Enzo Mumolo

Dipartimento di Elettrotecnica, Elettronica ed Informatica
Universita' di Trieste, Via Valerio 10, 34127 Trieste, Italy
mumolo@univ.trieste.it   http://smartlab.univ.trieste.it

Abstract. An algorithm for modeling and generating prosody, based on a syntactical analysis, is described in this paper. The model provides a common framework for including prosody in spoken dialogue systems. The algorithm has been used to develop a system for natural language interaction with a mobile robot.

1 Introduction

In some applications there is the need to develop a system that can engage in a collaborative spoken dialogue with human operators using natural language. Using the syntactical model of prosody described in this paper, an algorithm is proposed which gives a contribution for solving this problem. In fact, using this algorithm, a system for performing a vocal dialogue with a mobile robot [1] is under way and quite good preliminary results have been achieved, as it will be shown in the paper. The mobile robot was integrated with a word spotting engine and a TTS synthetizer. Simple vocal commands, inserted in a natural language sentence, can be given to the robot. Furthermore, it can answer using vocal messages.

The key idea of this model is to provide a syntactical description of a prosodic profile by a number of simple elementary curves. With respect to other contributions in the field [7, 6], in our case the curves are not described by an analytic function; rather, the curves are directly extracted from spoken utterances. Moreover, our definition of elementary curves is different. In other approaches [5], the linguistically relevant modulations of the fundamental frequency are interpreted into pre-defined configurations of stylized 'pitch movements', derived with a perceptual approach [8]. Finally, the algorithm described here provides a common framework for synthesis and recognition applications using natural language.

Generally speaking, a sentence can be segmented into clauses which are roughly related to the principal, secondary and final meaning of the utterance. Our model is based upon the identification of the clauses in a spoken utterance and on the determination of parameters related to prosody; furthermore the stress points within the sentence are de-emphasized, for getting rid of the events which are not related to the intonation itself. For the sake of simplicity, prosody has been characterized using only fundamental frequency; clearly the algorithm can be easily extended to other parameters.

The declarative sentences have been modelled with the following elementary curves: Principal Initial Profile (PI), Principal Non Initial Profile (PNI), Secondary Initial Profile (SI), Secondary Non Initial Profile (SNI) and Final Profile (FI) while for interrogative sentences Principal Initial Profile (PIN), Secondary Non Initial Profiles (SNIN) and Final Profiles (FPIN) have been used. On the basis of these segments, the syntactical model of declarative sentences can be described as: $(PI)(SNI)^*(FP)|(SI)(SNI)^*(PNI)(SNI)^*(FP)$. Interrogative sentences have been simply modelled as $(PIN)(SNIN)^*(FPIN)$.

It is worth noting that the model for declarative sentences is more complex than that for interrogative sentences, due to some system constraints.

2 Estimation of Basic Prosodic Groups

The estimation of the BPGs starts by using an initial set of spoken utterances in which the breath groups, i.e. the point where a speaker usually interrupts to breath, are taken into account.

A breath group has been considered as a clause and hence it is the basis to derive the BPGs. If the breath group can be grammatically described as (noun, verb, noun) it is called Principal, otherwise it is called Secondary.

Ten utterances are acquired, manually segmented into breath groups and labeled into BPGs using an interactive tool. The estimation of the fundamental frequency is performed using the eSRPD algorithm [3]. To simplify subsequent processing, the pitch has been interpolated using a Lagrange interpolator in order to obtain a continuous profile. Thus, a set of approximated BPGs is obtained; for each BPG, at least two versions are obtained from the ten utterances. The elementary profiles with the same label are averaged together using a Dynamic Programming Algorithm (DPA) in order to filter out stress, microprosodic and idyosincratic events of the spoken utterance. The BPGs obtained so far are the initial seeds of a subsequent iterative refinement procedure. Another set of ten utterances is used for the refinement. Some utterances are in the form (PI)(SNI)(FP) and (SI)(PNI)(FP) for declarative sentences; similarly for the interrogative sentences. Some finite-state syntaxes are built accordingly and a forced recognition is performed using One Stage Dpa where the templates are the seeds obtained as described above and the unknown pattern is the f_0 profile of the utterances of the refinement set. If the number of seeds is K and the number of f_0 points in an utterance is N, the algorithm is described as:

Step 1 $D(1,j,k) = \sum_{n=1}^{j} d(1,n,k),\ k = 1 \cdots K$
Step 2 2a)For $i = 2 \cdots N$ do steps 2b $\rightarrow$ 2e
2b)For $k = 1 \cdots K$ do steps 2c $\rightarrow$ 2e
2c)$D(i,1,k) = d(i,1,k) + min\{D(i-1,1,k), D(i-1,J(k^*),k^*),\ k^* = 1 \cdots K\}$
2d)For $j = 2 \cdots J(K)$ do step 2e
2e)$D(i,j,k) = d(i,j,k) + min\{D(i-1,j,k), D(i-1,j-1,k), D(i,j-1,k)\}$
Step 3 Backtracing with syntactical constraints

This way, the utterances are BPG-segmented according to the DPA optimization criteria, thus giving a new set of BPG starting from the initial seeds. This procedure is repeated for all the utterances.

As a new set of BPG is obtained, the new elementary profiles and the previous ones - the reference templates - are DPA averaged if they have the same label; the weights assigned to the new and previous BPGs are 0.9 and 0.1 respectively. The procedure is iterated until convergence is reached. It can be observed that, at each iteration, both the endpoints and the actual values adapt to the correct values. Convergence is indicated by the decreasing behavior of the DPA score. Some representative results are shown in fig.1 for a declarative sentence.

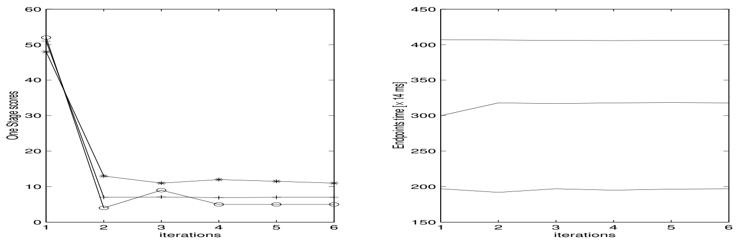

Fig. 1. Convergence of refinement procedure for declarative sentences. Left panel: One Stage DPA score versus iteration number for the three BPGs. Right panel: segmentation points of the three BPG versus iteration number

3 On the use of the syntactical model to help recognizing commands spoken using natural language

There are several possible uses of our model of prosody in a recognition system. The simplest one is to determine if the spoken sentence is a declarative or an interrogative one, and to use this information together with a keyword recognition system. This way, a keykord can be interpreted as being part of an interrogative clause or of a declaration. For example, the word 'localization' can be used in a sentence like 'can you perform localization of my position?' or 'give me all the localization information!'; if the keyword recognition system says that the word 'localization' is present in the input phrase (which was uttered in natural language), by identifying the prosodic model we can perform two completely different tasks. By the way, this can be a real dialogue with the robot.

4 Automatic segmentation of the input text into BPGs with a syntactical approach

In this Section we describe the identification of the BPGs from an input text. The general linguistic framework for the pre-processing and syntactical analysis phases has been derived from the work of [2].

4.1 Pre-processing of the input text

Consider the sets of symbols defined in the following

Definition 1 *Let us define a set of symbols as* $T_1 = \{gv, gn, gnv, gp, gpv, ga, gav, fr, frv, fs, fsv\}$ *where "g" stands for group, "n" for nominal, "v" stands for verbal, "p" for prepositional, "a" for adjunt, "f" for phrase, "r" for relative and "s" stands for secondary. Define moreover* $T_2 = \{< . >, <, >, <:>\}$ *and* $N_2 = \{< PF0 >, < PF10 >, < PF20 >, < PF30 >, < PF40 >, < PF50 >, < PF60 >, < PF70 >, < PF80 >\}$.

The output of the pre-processing module is a sequence of elements of the set T_1. First, a lexical analysis (LA) of the input text is performed. The purpose of the LA is to give the following stages the suitable information on each word of the input text. The lexical analyzer can be built at different levels of complexity but, at least, its output should include the word's grammatical category. Moreover, in case of lexical ambiguities, the LA should furnish all the possible categories. According to [2], we developed an LA composed by a dictionary and an automata. The dictionary contains all the lexicon, using a morphemic approach, while the automata's task is to make a morphological analysis of the input text and to give, as output, the grammatical categories. The dictionary is structured as a graph, which is analyzed with a deph-first strategy. The automata is, as a matter of fact, an Augmented Transition Network (ATN) [4]. The next step of the pre-processing phase concerns the stress assignment. Clearly, stress assignment is fundamental for the correct generation and modeling of the prosody. According to [2], the stress is obtained with a list of rules together with a number of exceptions. The list is searched and as soon as a suitable rule is found, the algorithm returns.

4.2 Syntactical analysis

The goal of the syntactical analysis is to determine high-level soprasegmental characteristics of the sentence, such as the "breath groups" and the prosodic profiles. Moreover, the syntactical information are fundamental to solve ambiguities, or homographs. The input text is divided into "Phonological Words" (PW), defined as a phonetic unit composed by the subset of words which is uttered as it were a single word. The PW contains only one principal stress. The PWs are classified in terms of 'gv' (verbal group), 'gn' (noun group), 'gp' (prepositional group), 'ga' (adjunt group), 'fr' (relative phrase) and 'fs' (secondary phrase).

4.3 Determination of the BPGs in the input text

As introduced before, the meta-text produced so far has to be segmented into BPGs, defined in Section 2. The segmentation has been realized with a grammar which classify a set of PW as the appropriate BPG according to the structure $[< nominalgroup >] < verbalgroup > [< nominalgroup >] < prepositionalgroup >^*$.

We now describe the syntax for the declarative phrases, which has been derived through the analysis of many sentences. Slight modifications are required for the interrogative phrases.

Definition 2 *The grammar used for the segmentation into BPG is a set* $G = \{V_N, V_T, P, S\}$ *where the nonterminal symbols are* $V_N = \{< s_i >, < PF_j >, < p >, < FALL >\}$, *for* $i = 20, 40$ *and* $j = 0, 10 \ldots 80$ *and the set of terminal symbols is* $V_T = T_1 \cup \{PI, PNI, SI, SNI, FP\} \cup \{:, ., , , \{, \}\}$. *The initial symbol is* S. *The set of production rules* P *is given below in BNF.*

1. $S ::= (\{(gav|ga|fs|fsv|gp|gpv|frv|fr) < PF0 >< SI >< s20 >)|(\{(gn|gv|gnv) < PF0 >< PI >< s40 >)$
2. $p ::=, | :$
3. $fr < s20 >::= fr < PF0 >< PSNI >< s20 >$, same for $fsv, frv, fs, ga, gav, gp, gpv$
4. $gn < s20 >::= gn < PF0 >< PNI >< s40 >$, same for gv, gnv
5. $< p >< s20 >::= \{< p > (t, t \in T1) < PF0 >< SNI >< s20 >$
6. $< s40 >::= \{(t, t \in T1$ or $T2) < PF0 >< SNI >< s40 >$
7. $. < FALL >::= .\}$
8. $.\}(< PSI >< s20 > | < PI >< s40 > | < SNI >< s20 > | < SNI >< s40 > | < PNI >< s40 >) ::= .\} < PF >$
9. $< p >< FALL >::=< p >\}$
10. $gv < PF0 >::= gv < PF10 >$
11. $gpv < PF0 >::= gpv < PF20 >$, same for fsv, frv, gav
12. $gn < PF0 >::= gn < PF30 >$, same for fr, fs, ga, gp
13. $< PF10 >::= (gv < PF10 >)|(gp < PF70 >)|((gn|gnv|gpv|ga|gav|fr|frv|fs|fsv) < PF40 >)$
14. $< PF20 >::= (gv < PF10 >)|(fsv < PF20 >)|((gn|gnv|ga|gav) < PF40 >)|(gp < PF70 >)|((sf|fr|gpv|frv) < PF0 >< SNI >< PF70 >))$
15. $< PF30 >::= ((ga|gn|gp) < PF30 >)|((gv|gav|gnv) < PF40 >)|((gpv|fr|frv|fs|fsv) < PF0 >< SNI >< PF10 >))$
16. $< PF40 >::= ((gv|gav|gn) < PF40 >)|(gp < PF50 >)|(\})$
17. $< PF50 >::= ((gv|gav) < PF60 >)|(gp < PF50 >)|(\})$
18. $< PF60 >::= ((ga|gp|gav) < PF60 >)|(\})$
19. $< PF70 >::= ((ga|gp|gav) < PF70 >)|(gn < PF80 >)|(\})$
20. $< PF80 >::= ((ga|gav) < PF60 >)|(\})$
21. $(n, n \in N2 - \{< PFO >\})|(. < PF0 >) ::= . < FALL >$
22. $(n, n \in N2 - \{< PFO >\})|(< p >< PF0 >) ::=< p >< FALL >$

A parser has been realized by means of a small ATN built according to the described grammar.

Example Let us consider, as an example, the following italian phrase: '*Il satellite contiene un orologio ed una stazione trasmittente che permettono di sincronizzare i tempi con grande precisione.*'.This phrase is transformed, at the output of the syntactical analysis of Section 3.2, in the following string: '*gn gv gn frv gp gn gp.*' which is left-to-right analyzed with the grammar described above. The results of the analysis are reported below, where the employed production rules and the output strings are indicated.

$P1, P12 \rightarrow \{gn\, PF30\, PI\, s40$
$P15, P16 \rightarrow \{gn\, gv\, gn\, PF40\, PI\, s40$
$P16, P6 \rightarrow \{gn\, gv\, gn\}PI\{frv\, PF0\, SNI\, s40$
$P19, P20 \rightarrow \{gn\, gv\, gn\}PI\{frv\, gp\, gn\}SNI\, s40$
$P6, P8 \rightarrow \{gn\, gv\, gn\}PI\{frv\, gp\, gn\}SNI\{gp.\}PFIN$

which is the final result.

Of course, the use of the syntactical model in text-to speech systems can be performed by determining the sequence of elementary curves needed to model the text. The suitable BPGs are then concatenated and superimposed to the synthetized signal.

5 Final remarks and Conclusions

Twenty sentences have been used to test the algorithm. Such sentences were available in spoken as well as in written form. All the phrases in the test set were correctly segmented, using One Stage DPA and the BPG derived as described above. A further test was carried out to verify the correctness of the results from a perceptual point of view, giving a success of about 70 %.

The test performed with the written form of the utterances gave good results either: the segmentation into BPG was 50 % correct, but in additional 40% cases it gave acceptable results. Clearly, a refinement of the morphological analyzer would lead to an overall improvement. The algorithm has been integrated into a real time TTS system and it was observed that it improved the naturalness of the spoken utterances.

References

1. E.Mumolo, M.Nolich, G.Vercelli, "Algorithms and Architectures for Acoustic Localization based on Microphone Array in Service Robotics", *IEEE Int. Conference on Robotics and Automation*, Stanford, CA, 2000
2. R.Gretter, G.Mian, R.Rinaldo, M.Salmasi, "Linguistic Processing for an Italian Text-to-Speech System", in Proc. of *Int.Conf. on Speech Tech. VERBA90*, 1990, Rome, Italy
3. P.Bagshaw, S.Hiller, Jack, "Enhanced pitch tracking and the processing F0 contours for computer aided intonational teaching", in *Proceedings of Eurospeech 93*, Berlin
4. W.A.Woods, "Cascaded ATN Grammars", *American Jour. of Comp. Linguistics*, Vol.16, N.1, 1980
5. S.Quazza, P.L.Salza, S.Sandri, A.Spini, "Prosodic control in a text to speech system for italian", in Proc. of *ESCA Workshop on Prosody*, Lund, Sept. 1993
6. L. F. M. ten Bosch, "Automatic Classification of Pitch Movements via MLP-Based Estimation of Class Probabilities", ICASSP 95
7. M.Nakei, H. Singer at al., "Automatic Prosodic Segmentation by F0 Clustering Using Superpositional Modeling", ICASSP 95
8. J.t'Hart, R.Collier, A.Cohen, "A perceptual study of intonation: an experimental-phonetic approach to speech melody", *Cambridge University Press*, 1990