PHONOLOGICAL PIVOT PARSING

Grzegorz Dogil

Universität Bielefeld
Fakultät für Linguistik
und Literaturwissenschaft
D-4800 Bielefeld
West Germany

There are two basic mysteries about natural language. The speed and ease with which it is acquired by a child and the speed and ease with which it is processed. Similarly to language acquisition, language processing faces a strong input-data-deficiency problem. When we speak we alter a great lot in the idealized phonological and phonetic representations. We delete whole phonemes, we radically change allophones, we shift stresses, we break up intonational patterns, we insert the pauses at the most unexpected places, etc. If to this crippled 'phonological string' we add all the noise from the surroundings which does not help comprehension either, it is bewildering that the parser is supposed to recognize anything at all. However, even in the most difficult circumstances (foreign accent, loud environment, being drunk, etc.) we do comprehend speech quickly and efficiently. There must be then some signals in the phonetic string which are particularly easy to grasp and to process. I call these signals 'pivots' and parsers working with these signals I call 'pivot parsers'.

What are then the pivots in the phonetic string? I am not proclaiming any heresy by saying that the pivots should correspond to the most audible parts of the phonetic string. If we look at the intensity tracing of speech we will notice a fairly regular sequence of peaks. At the lowest prosodic level, the level of the syllable, these peaks correspond to the vowels forming syllabic nuclei. In my view, the parser will orient itself foremostly on these vocalic peaks. That is to say, the parser in my model is a 'jumper' which recognizes the best audible units of speech - vowels building syllabic nuclei - and disregards everything else. Such a parser is definitely very fast but it is also very inefficient. Having recognized just a string of vowels we do not have enough information to find a word which contains these vowels. Or does anyone subconsciously 'know' which word the string of vowels / .a.a.i.e. / corresponds to?! The parser needs definitely more information, but how much more? This is where my hypothesis about 'ideal prosodic types' comes into play.

In Dogil: 1985, I argued that at each level of prosodic organization there exist prototypical, unmarked structures which manifest themselves not only in patterns of all natural languages but are also clearly visible in the areas of external evidence such as language acquisition, language loss, and language change. Here I will argue that these 'ideal prosodic types' play an important role in language processing.

At the lowest prosodic level - the level of the syllable - such an ideal type is constituted by a CV syllable. That is, the prototypical, unmarked syllable consist of a single consonant followed by a vowel. There is plenty of evidence for this prototype (cf. Clements & Keyser: 1983, 28ff., Ohala & Kawasaki: 1984, 115-119). For example:

- there is no language which would not have CV syllables, but there are many languages which have only CV syllables
- phonological rules which obliterate syllabic structure usually spare CV syllables
- CV syllables are acquired as first in the process of language acquisition
- CV syllables are preserved even in the most severe forms of motor aphasia (cf. Dogil: 1985)
- historical syllabic restructuring rules tend towards the creation of CV syllables.

All this evidence clearly illustrates the prototypical character of this unit. I claim that this unit is also essential for pre-lexical parsing. What the parser essentially does is recognize CV syllables in the string. I propose it does this in the following way:

-- The parser searches for the first intensity peak and once it has found it it stops there. As I said before these intensity peaks are coterminous with vowels (most sonorous sound types) forming syllabic nuclei. The parser goes back in 10 msec. steps making a diphone[1] of the vowel and the consonant preceding it. This gives a diphonic representation of CV syllables. The difference between the diphone scanner in my model and in all other models is that my scanner works backwards starting at the peak of the vowel.

-- The parser recognizes the syllable. Strictly speaking it recognizes only the unmarked, prototypical CV part of the syllable. These prototypical CV's are stored as diphones in the diphone dictionary. If the syllable contains other units, for example if it is CCVCC syllable (like in the name 'Planck') these other units will be disregarded, and only the CV (/la/ of /plaŋk/) will be available after the initial parse.

-- Having identified the syllable the parser makes its first hypothesis about the word that this syllable is a part of.

-- The parsing strategy is carried on by jumping to the next intensity peak, i.e. the next vowel.

Consider a simple example of a parse by a syllabic pivot parser of a German sentence "Ich gehe zum Max-Planck-Institut" - I am going to the Max-Planck-Institut:

(1) [ʔç geə tsum maks plaŋk ʔinstitut]

I did some simple speech editing which monitors the

[1] 'Diphones' are defined as transitions from the middle of one phone to the midpoint of the preceding one.

function of my parser. From the phonetic string in (1) I clipped off the parts of the onset and the codas which according to the pivot parser are not processed on the initial parse. The resulting string in (2) was fully recognizable.

(2) [ʔi gə tsu ma la ʔɪ tɨ tu]

Actually it strongly reminded of fast/casual German speech.

When I clipped off these parts of the string which the pivot parser considers relevant - i.e. consonants immediately preceding the vowels - the string was not recognizable any more. Consider the transcription in (3):

(3) [iç eə um aks aŋk ins ɪ ut]

Actually, some of my informants claimed that it was not a sentence of their language. Needless to say the string was not recognizable when the vowels were obliterated.[2]

Given all the grammatical, contextual and background knowledge that we possess when parsing strings, the syllabic pivot parser might be actually sufficient for comprehension. Even if it is insufficient in the form that I have presented it so far, it is fast enough to incorporate a number of repair strategies that can make it sufficient for comprehension. I will just mention some of these possible repair strategies without going into any detail.

1. Phonemic Restoration Strategy - recovers sounds which are adjacent to the CV pivot. For example, in case the syllable /la/ in our example sentence did not contain enough information to recognize the corresponding name 'Planck', the consonant /p/ preceding /la/ and the consonant /ŋ/ following /la/ would have to be recovered by this repair strategy.[3]

2. Pivot parsing at higher prosodic levels - for instance recovering 'ideal types' at the level of the foot or the prosodic word. As I understand it this is exactly what Taft: 1984 has proposed. Another possible method here is finding the patterns of intonational morphemes and pauses and matching these to the dialogue structure, as was proposed in Gibbon: 1985.

3. Taking advantage of the language specific phonotactic constraints - for example, the fact that in a language long vowels may occur only in open syllables takes a great load off the parser which has discovered a long vowel.

4. Allophonic fixing of constituent boundaries. This sort of parsing strategy is central in Church's: 1983 phonological parser, which I will have something to say about later. Obviously, because allophones are a very much language specific matter, the allophonic parser is also language specific.

5. Using higher level representational knowledge (morpho-syntactic and semantic knowledge) in order to repair the result of the prosodic pivot parse - for example, if we parse a word like {export} with an initial (i.e. 'nouny' stress) in a syntactic position of a verb, we will probably not think twice about its prosodic 'nouniness' but interpret it as a verb (cf. Cutler & Clifton: 1984). I guess we use the similar strategy to recover suffixes which are initially not parsed.

Most of these parsing strategies presented above are language specific, and I do not see them as alternatives to my pivot parser but as additions to it. The pivot parser which orients itself on the prototypical linguistic units is obviously universally applicable.

The pivot parser is fast. It is definitely faster than the finite state parser developed by Church: 1983. Church's parser also divides the string of speech into the sequence of syllables (and metrical feet). However, instead of prototypical pivots it uses the constraints that the syllable imposes on the distribution of allophones. It is tuned to the analysis of these phonetic features which are typical of syllable initial and syllable final positions. Church has shown that his method greatly reduces the number of competing syllabic analyses compatible with a given utterance. Nonetheless, some unresolved ambiguity about the correct syllabic segmentation persists despite the effect of the phonotactic constraints. Note that the syllabic pivot parser does not give rise to any ambiguity of this sort. The strings are syllabified to the 'ideal' CV chunks.

Church's parser is slower than the syllabic pivot parser because it has to wait until it reaches the syllable final position in order to fix the boundary of the recognition unit[4]. This, in turn, makes the parser very inefficient and, actually, inadequate given the input-data-deficiency problem that I discussed at the beginning of this paper. The syllable final position that Church's parser critically depends upon is the most vulnerable position for phonological obscuration processes (cf. Dressler: 1984). These processes which weaken, obliterate or even delete syllable final allophones are very operative in natural (particularly fast/casual) speech. Thus, if these processes apply and the positions which Church's parser depends on are not there any more, the parse will break down. I am concluding then that Church's

[2] I did this speech editing using the SPED software on PDP 11. I thank Carla Coenders of the MPI for assisting me in speech editing.

[3] Warren: 1970, who first argued for the Phonemic Restoration Strategy, replaced the first phoneme /s/ in a word like 'legislature' with a coughing sound of about the same intensity as the speech. He then presented this word to subjects, and asked them to indicate where in the word the cough occurred. The subjects were unable to accurately locate the cough. More important, the missing phoneme was completely 'restored'; that is, it was not perceived as missing. The subjects heard the /s/ in 'legislature', and the cough was heard as background noise. Hence, a listener can generate phonemes (given contextual information) that do not exist in the speech string. He can do this, I would predict, only in these positions that are outside of the CV pivot. If we replaced some part of the pivot with noise, the subjects would not be able to restore it - just as it was the case with my example (3).

[4] All psycholinguistic experiments (cf. Frauenfelder: 1985 for an overview) speak against this waiting strategy. Actually the words are recognized long before (2-3 phonemes before) their final segments have been processed.

language specific allophonic parser is slower than my universal syllabic pivot parser and that it also faces a strong inefficiency problem.

Similar problems apply to all the phonemic parsers. As an example let us discuss a parser assumed in the widespread Cohort Model of word recognition. The parser implicit in the Cohort Model is a sequential categorial, correct, phonemic parser (cf. Frauenfelder: 1985). Its purpose is finding the 'uniqueness point' for word recognition. Let us assume (after Marslen-Wilson: 1984, 141-142) that the word to be recognized is "trespass". Given the phonemic information, we can determine the point at which "trespass" becomes uniquely distinguishable. There are many words that begin with /tre/, and at least two that share the initial sequence /tres/ (trestle, tress). But immediately following the /s/ only "trespass" remains. The discrimination point for this word is therefore at the /p/. It is here, and no later, that an optimal system should discriminate the word.

Now, what is the strategy of the pivot parser to recognize a word like "trespass"? First it will find the intensity peak and recognize it as the vowel /e/. Then it will bind the consonant preceding this vowel and recognize it as /tr/. I assume that /tr/ is a monosegmental affricate. The parser will recognize the first syllable as /tre/ and make a first hypothesis about the word. The cohort of compatible words will include all the words in Marslen-Wilson's cohort, plus some more words that have the initial syllable /stre/ (strength, stress, stretch). Then the parser will jump to the next intensity peak and recognize it as the vowel /ə/. It will bind the preceding consonant and recognize it as /p/. Now it possesses two syllables /tre/ and /pə/ for the next hypothesis as to word recognition. This is actually enough as there is just one word in English containing these two syllables in that order - this word is "trespass".

The whole procedure lasts approximately 400 msec., and 4 segments have to be recognized until the 'uniqueness point' has been reached. Hence, my parser is possibly not less efficient than the sequential phonemic parser, and its uniqueness recognition point does not come later than predicted by studies connected with the Cohort Model's phonemic parser (cf. Carlson, Elenius, Granstrom and Hunnicutt: 1985).

Obviously syllabic pivot parser requires a different structuring of the lexicon than the standard phonemic structuring implicit in the Cohort Model parser. Let us imagine a lexicon which is organized according to the CV syllabic pivots. In order to foster our imagination I will compare such a lexicon to a warehouse. Imagine that words are the spare parts that the machines (sentences) are made of. All the spare parts have screws that keep them together. Imagine now that these screws are the prototypical CV syllables. Our warehouse (lexicon) is organized according to which screws (CV syllables) fit which spare parts (words). If you need a spare part (a word), but you know only what type of a screw (CV syllable) you have in it and what type of machine (context and sentence information) it might be used in, the warehouse administration (the parser) will provide you with the spare part you have been looking for. I have been told that warehouses organized according to this principle actually exist (in industry) and that they work much more efficiently than the warehouses which list the details of all of their spare parts.

There is, however, one major advantage which my parser has over any phonemic parser. Phonemic parsers require that all decisions on the sensory input are always made correctly. That is, every single phoneme in the string must be correctly recognized. Given the deficiency of the input string which I kept mentioning through my paper, this correctness requirement may never be fulfilled (except maybe in a psycholinguistic lab).

Even in the most idealized and artificial laboratory situation the acoustic manifestation of many phonemes depends upon the context. For example, the second formant of /d/ in the syllable /di/ has a rising transition, whereas in /du/ it has a falling one. A parser which takes no account of the vowel in the syllable cannot be expected to realize that a rising and a falling transition are cues for the same phoneme.

My parser does not face this sort of problem because the phonological properties it is tuned to are the most salient ones from the perceptual point of view (cf. Marcus: 1981) and are best preserved in phonetic strings.

I have presented to you an idea of what a fast parser which requires the minimum of phonlogically invariant information might look like. This parser works in a sequentially-looping manner and the decisions it makes are non-deterministic. It is universally applicable, it is faster, and it seems to be no less efficient than other phonological parsers that have been proposed.

References:

Carlson, R., K. Elenius, B. Granstrom, & S. Hunnicutt: 1985, Phonetic and orthographic properties of the basic vocabulary of five European languages, Franco-Suedois Symposium, Grenoble, April 20-22.

Church, K.: 1983, Phrase structure parsing: A method of taking advantage of allophonic constraints,Indiana University Linguistic Club.

Clements, G.N. & J. Keyser: 1983, CV Phonology, MIT Press, Cambridge, Massachusetts.

Cutler, A. & Ch. Clifton, Jr.: 1984, The use of prosodic information in word recognition, in H. Bouma & D.G. Bouwhuis (eds.), Attention and Performance X: Control of Language Processes, Erlbaum, Hillsdale, N.J.

Dogil, G.: 1985, Theory of markedness in nonlinear phonology, Habil., Universität Bielefeld.

Dressler, W.: 1984, Explaining Natural Phonology, Phonology Yearbook 1.

Frauenfelder, U.: 1985, Crosslinguistic approaches to lexical segmentation, Linguistics - special issue on Cross-language Psycholinguistics.

Gibbon, D.: 1985, Prosodic parsing above the word. DGfS Hamburg, February 1985.

Marcus, S.: 1981, Acoustic determinants of perceptual center (P-center) location, Perception and Psycholinguistics 30. 247-256.

Marslen-Wilson, W.: 1984, Function and process in spoken word recognition: A tutorial review, in H. Bouma & D.G. Bouwhuis (eds.), Attention and Performance X, Erlbaum, Hillsdale, N.J.

Ohala, J. & H. Kawasaki: 1984, Prosodic phonology and phonetics, Phonology Yearbook 1. 113-127.

Taft, L.A.: 1984, Prosodic constraints and lexical parsing strategies, unpublished Ph.D. dissertation, University of Massachusetts at Amherst.

Warren, R.: 1970, Perceptual restoration of missing speech sounds, Science 167, 393-395.