Proceedings of The Institute of Acoustics

THE COMPUTER ANALYSIS OF PROSODY

D. LINDSAY AND W. A. AINSWORTH

DEPARTMENT OF COMMUNICATION & NEUROSCIENCE, UNIVERSITY OF KEELE.

Introduction

One problem in automatic speech recognition is analysing the acoustic correlates of prosody present in the speech stream. The listener can show great perceptual sensitivity to features such as fundamental frequency, vowel duration and loudness, and yet these same features exhibit wide phonetic variability. One way of accounting for this is by modelling the production of prosodic features as being rule-governed behaviour. The description of prosody in language constitutes part of linguistic knowledge; and prosody can be considered as a language system in the sense that phenomena such as pitch, rhythm, loudness, and stress patterns constitute a closed network of patterned relationships.

Until recently little use has been made of this potential knowledge source in automatic speech recognition. One reason for this is the problem of determining just what constitutes a correct linguistic description of prosodic phenomena: there is a long term controversy in linguistics over the characterisation of intonation as contours as opposed to a sequence of pitch levels; also there is the more general task of separating linguistically determined patterning from paralinguistic and emotive features.

Computer analysis of prosody

A technique for analysing prosodic phenomena will be outlined together with a description of the computer implementations of the algorithms, both of work completed and in hand. How this can be used to analyse speech and also to provide an objective means of deciding between rule systems will be discussed. The computer program written can be used as a tool for analysing the regularities of prosodic features, especially fundamental frequency, using a model of British English intonation together with a description of the minimal supra-segmental units of these features. The basic algorithm is a syntactically guided parse of a sequence of short-term analysis feature vectors representing the utterance. There are four main parts to the technique: front-end processor, phonological rule compiler, parser, and template recogniser. The algorithm is implemented in PASCAL on an Apple II microcomputer using the UCSD Pascal system.

The front-end processor

A front-end processor provides the input sequence over which the main algorithm operates. A feature vector is constructed from an analysis of a 51.2ms portion of speech on a frame rate of 10ms. The feature vector contains an estimation of the reliability of the F0 value for the frame in question. The estimation of confidence in the fundamental frequency measure is determined within the analysis algorithm and is designed for use as a weighting

factor by the template matcher. This prevents arbitrary threshold logic from being applied too early in the front-end processor and also preserves voicing information, such as moving in and out of syllabic nuclei, which is lost in systems which are constructed to return an 'unvoiced' label in preference to a low-confidence but specific, fundamental frequency value. At the present stage of development the Lx output waveform from a University College London laryngograph is used to provide a representation of vocal fold movement (1). Analysis is performed on a Computer Automation Alpha sampling simultaneously the Lx waveform and speech low-pass filtered at 2.5kHz.

The phonological rule compiler

The model of the intonation system to be used is explicitly specified by a series of generative rewrite rules (2), for example:

for sequential occurrence:

```
phrase → headP + N + tailP ;
headP  → prehead + head ;
```

for paradigmatic selection:

```
tone → simple, compound, complex ;
```

for optionality:

```
complex → compound + (complex) ;
```

and context-sensitivity is represented by paradigmatic choice under conditions:

```
prehead → PH2 + tail2, PH1 + tail1 ;
```

The compilation of these phonological rules is performed by a small one-pass recursive descent compiler which generates an intermediate parse code. This code is in the form of a context-sensitive network of node structures representing the information contained in the rules and sets of parse state indicators. The pattern templates are specified explicitly within the rewrite rules, for example:

```
tail2 → [ 100.0,
          100.0,
          103.5,
           ...
          125.0,
          130.0  ]
```

The main algorithm

The analysis algorithm is guided by a recursive descent parser operating over the grammar given by the phonological rules and on the feature vector sequence as the input string. The basic algorithm returns the sequence of templates which

collectively return the best overall match score from attempts to match the individual templates with the corresponding portions of the input sequence. As the implementation is not restricted to working in real-time, the complete input sequence is available to the analyser at any state within the parse. Context-sensitivity will be handled by a subordinate parse which seeks a 'best-probable' sequence to include the context node, at which point a decision is made as to whether the context is satisfied and then the main analysis is resumed. The process of matching template to data is an analysis-by-synthesis method in which the synthesis is constrained by a current transformation set and a permitted relaxation space constraining precisely how much change is permitted to the current transformations in the interests of an optimal fit. At present the transformation set is four transformations of shift and dilation in both the time and frequency dimensions. It is envisaged that the result of the algorithm will be a pair consisting of the error score of the optimal fit with a measure of the amount of adaptation required for that score.

The template matcher

The template matcher uses an analysis-by-synthesis method of determining the best fit as measured by a Euclidian distance metric weighted by the estimated confidence in the values of the feature vector. There are two types of control over this algorithm: the initial conditions given by the template and by the current transformations; and the parameters directing the search through the transformation space. The initial transformations specify a guess of what the best fit will involve, based on previous tries on other templates: this involves adjustment of both the range and absolute shift in the fundamental frequency values to the individual speaker and the particular tone of voice used throughout the utterance; adjustment is also made to absolute time shift so that the input sequence need not be assumed to start at a particular point in a breath-phrase, and a linear time dilation is used as an attempt at normalising rate of speech. Control is exercised over how the algorithm searches for the best fit: at present the bounds on iterations and allowable relaxation are fixed by the initialisation routines.

Discussion

Previous analyses of intonation have tended to fall into two main categories: auditory and acoustic. Auditory analysis uses the full perceptual apparatus of the listener to bridge the gap between sound and symbol and provides a phonetic description of categories already assumed. Acoustic analysis starts with the direct measurement of sound, but before the introduction of computers little could be done beyond this stage. Computers have been used in limited ways, as in feature detection, (3) for prehead glide, or attempts at stressed syllable location, (4) for locating the nearest 'loudness' maximum to a prehead glide, but no attempt has been made at analysis across the whole of the breath group although proposals have been made (5), (6). The aim of this research is to determine the effect of using prosodic information as a constraint on a speech recognition process. By limiting the algorithm to operating only over prosodic features the main problem of linking to a conventional segmental or template recogniser is removed; the prosodic system is a small system, yet its domain is the entire phonological phrase. This restriction to only one of the systems of language means that an exhaustive and appropriate phonemic description of

the speech process can be achieved at the non-segmental level.

Nonphonemic differences within a corpus are handled by the series of overlaid transformations mentioned above, so the error score in recognition is a function of both how well the rules characterise the data and how much 'interference' there is from other language systems that make some use of the same basic features, such as tone of voice which is heavily dependent on physiological makeup, and paralinguistic systems such as the signalling of emotions in the speech channel. Part of the present research is to determine just how to approach by automatic analysis the separation of phonemic from non-phonemic differences that listeners can achieve.

References

1. A.J. FOURCIN 1981 Report 11, American Speech-Language-Hearing Association. Laryngographic assessment of phonatory function.

2. Z.S. HARRIS 1957 Language 33, 283-340. Cooccurrence and transformations in linguistic structure.

3. W.A. LEA 1980 in Trends in Speech Recognition, ed. W.A. LEA, Prentice-Hall, 194. Prosodic aids to speech recognition.

4. W.A. LEA 1973 Journal of the Acoustic Society of America 55, 411. An algorithm for locating stressed syllables in continuous speech.

5. W.A. LEA, M.F. MEDRESS and T.E. SKINNER 1975 IEEE Transactions Acoustics, Speech, and Signal Processing ASSP-23, 30-38.

6. J. VAISSIERE 1981 in The Cognitive Representation of Speech, eds. MYERS, LAVER and ANDERSON, Elsevier/North-Holland. Speech recognition programs as models of speech production.