ACOUSTIC/PHONETIC TRANSCRIPTION USING A POLYNOMIAL CLASSIFIER AND HIDDEN MARKOV MODELS

Alfred Kaltenmeier

AEG Research Institute, Ulm, Germany

ABSTRACT

This paper describes a module for acoustic/phonetic transcription in a continuous speech understanding system. This module segments input utterances into sequences of phone classes which belong to one of six broad phonetic categories. In a higher system level such segment sequences are used to hypothesize possible word candidates from a lexicon.

This module is hierarchically implemented in two stages: a polynomial classifier for a frame-by-frame classification of phone classes followed by a segmentation stage using Hidden Markov Models (HMM) of phone class segments.

INTRODUCTION

This paper describes an acoustic/phonetic module for a continuous speech understanding system which is being developed within the framework of the European Community ESPRIT Project No. 26.

Since continuous speech recognition presupposes an unlimited vocabulary, units smaller than words must be used for recognition. In our system two kinds of small phonetic units are used: phonemes and diphones on the one hand /1/ and phone classes (plosives, fricatives, etc.) on the other. The number of phone classes to be distinguished is low (5 to 10) whereas the number of phonemes and diphones is much higher (100 to 200). Using this double representation of phonetic units, the recognition part of our system can be effectively implemented in three levels (Fig. 1).

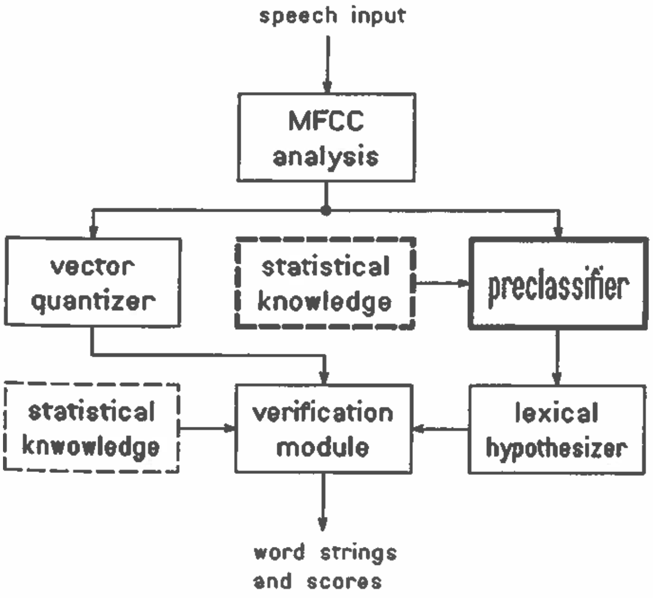

Fig. 1 Block diagram of the recognition stage in the continuous speech understanding system

The data reduction block in the first level computes mel-frequency cepstral coefficients (MFCC) as parametric representations of speech frames /2/.

For the second level, cepstral vectors form the input to both a vector quantizer and a preclassifier which hypothesizes phone classes. Vector quantization reduces the amount of data while preserving all information needed to correctly classify the various sounds. The preclassifier transforms speech signals into broad phonetic categories and in the process computes segment boundaries and likelihoods, too. The statistical knowledge consists of a coefficient matrix for polynomial classification and HMMs of phone class segments, phone class durations, rules, and error models for smoothing/segmentation.

In the third level the preclassifier output is used to extract a reduced number of word candidates from a word lexicon. This reduced set of word candidates is then verified and scored by the verification module which uses HMMs of phonemes and diphones as statistical knowledge.

PRECLASSIFIER MODULE

A hierarchical organization of the acoustic/phonetic transcription can greatly reduce the number of computations required in the word verification module. To this end, the selected set of phone classes must guarantee a high selectivity between the words in the lexicon while at the same time preserving a high reliability in the preclassification. Detailed investigations have shown that these two opposing requirements can be best met using six phonetic categories which are labeled as follows:

pl:	plosives and silence
fr:	fricatives and affricates
ln:	sonorants (liquids and nasals)
fv:	front vowels
cv:	central vowels
bv:	back vowels

The preclassifier is implemented in two stages (Fig. 2). The first stage consists of a polynomial classifier followed by a decision quantizer, both performed frame by frame. The classifier estimates the likelihoods that a cepstral vector belongs to each of the predefined phone classes by evaluating the following matrix product:

$$\underline{d} = A \cdot \underline{x} \qquad (1)$$

where $\underline{d}$ is a decision vector containing estimated likelihoods, A is a coefficient matrix, and $\underline{x}$ is a vector which contains linear, quadratic, and cubic terms of cepstral vector components.

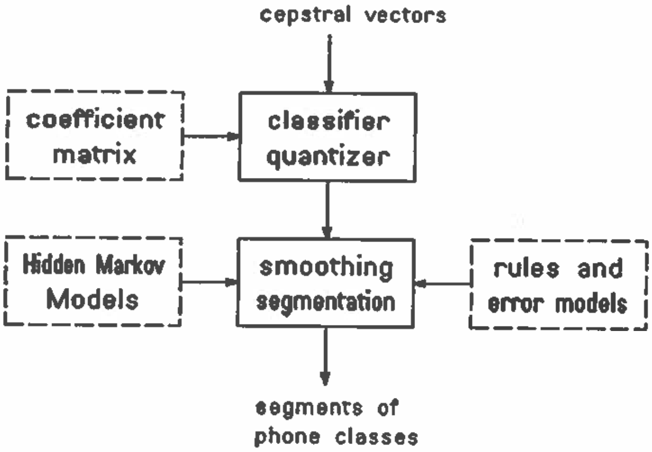

Fig. 2 Block diagram of the preclassifier with its associated statistical knowledge sources

Since determing the coefficient matrix A requires a large amount of computation and storage and a very large speech data base, it is computed off line with automatically labeled speech data . Due to the large computational and storage requirements, this classifier must be speaker-independent if it is to be at all practical.

The classification according to eq. (1) is implemented in a two-level structure. First we estimate the likelihoods of three combined classes (pl+fr, ln+fv, and cv+bv), which are then separated in their respective subclasses in a second level. This hierarchical structure increases performance and requires less computation than a parallel structure which estimates all six classes simultaneously.

Along with the estimated likelihoods the classifier produces a reliability score. This score represents a unique decision for one class, a decision for two of the six classes, or a reject if no reliable decision can be made. According to this score, a decision vector is quantized and transformed into a symbol. We have one symbol for the reject, 6 symbols for unique decisions, and 15 symbols for all possible two-case decisions or 22 symbols in all. Since the first stage of the preclassifier module transforms a cepstral vector into a symbol, it can be viewed as a vector quantizer which incorporates phonetic information. Hence, at the output of this stage an utterance is represented by a sequence of symbols, which then have to be smoothed and segmented by the second stage of the preclassifier.

Such a sequence may contain local irregularities (corresponding to spurious decisions, particularly during transitions) which have to be smoothed out in order to correctly segment an utterance. Using a simple fixed-length majority voting filter for smoothing is not very effective because this does not take statistical information on segment durations into account. Better segmentation results are obtained by statistical decoding using HMMs of phone classes and transitions as well as information on phone class durations.

Fig. 3 illustrates the complete preclassification process. The example used here is the German time phrase 'neun Uhr drei' (9:03) with following phoneme /diphone and phone class descriptions:

phon./diph.: - n o Y n U R d dr r a I -
phone clas.: pl ln bv fv ln bv cv pl ln cv fv pl

The first row of Fig. 3 shows the phoneme/diphone segments which were manually labeled for this example (transition segments are not shown). The second row shows the output of the first preclassifier stage. Dark areas are unique decisions, and shaded areas are two-class decisions. The third row shows the result of the segmentation. Obviously there are two errors in the segmentation (shaded areas). The first back vowel 'o' is split into two short bv and cv segments, and the liquid 'r' is merged with the following central vowel 'a'.

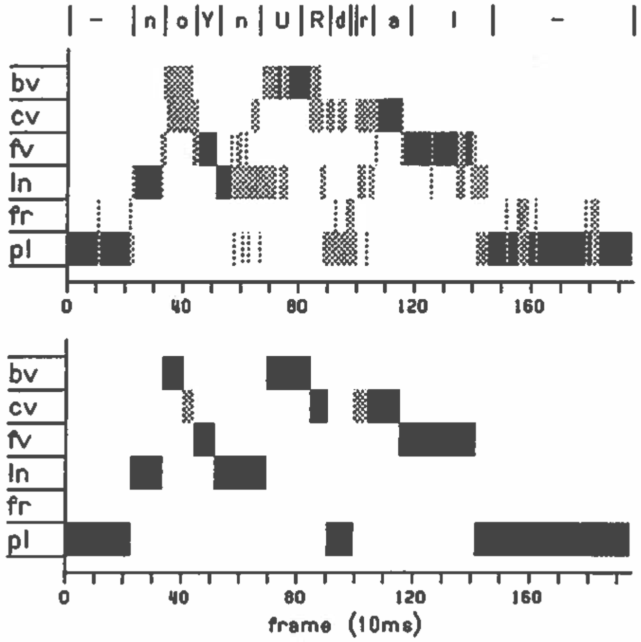

Fig. 3 phone class segmentation of the German time phrase 'neun Uhr drei' (9:03)

In order to reduce the number of such segmentation errors we will implement both a set of rules which directly include the speech signal energy in the segmentation process and also statistical models for the most frequent preclassification errors. A frequent error, for example, is the smoothing out of a short sonorant segment between two vowels. However, such a missing segment can be easily recovered using the energy contour which shows a clear dip in the sonorant segment.

Error models which define the likelihoods of context-dependent preclassification errors will be used to generate alternative segmentations. In order to evaluate error models some experiments with a larger preclassified data base are in progress.

The output of the preclassifier are error modeled phone class sequences forming the input to the next recognition stage. In this stage the lexical module first generates syllabic segments from the phone class sequences. Then syllabic segments are used to select a set of word candidates which are possible in the given part of an utterance.

PRECLASSIFIER PERFORMANCE

This section summarizes the preliminary performance of two preclassifiers which were computed from Italian and German speech data. The classifiers were trained with 720 words from four Italian speakers and 500 words from two German speakers.

The frame-by-frame classifications in the first stage of the preclassifiers have quite low error rates between 3% and 6%. Only segments labeled as stationary phonemes are considered here because its difficult to define an error rate during transitions. Error rates were obtained using the 'k best of six classes' rule, where k = 1 for unique decisions and k = 2 for two-case decisions.

The segmentation is based on the Viterbi algorithm; rules and error models have not yet been implemented. For isolated words we had an segment error rate of about 10%. Using the German preclassifier we made some additional experiments with 100 connected digit strings and 100 five-word sentences. With this material, which did not belong to the training data, we had segment error rates of about 12% for connected digit strings and about 16% for the sentences, respectively. By applying energy information and errors models, error rates can be decreased and the reliability of the preclassifier further improved.

Using the preclassifier approach described above, speech signals can be reliably segmented into six broad phonetic categories which minimize the ambiguity in the lexicon access.

LITERATURE

/1/ Cravero M., Pieraccini R., Raineri F.
Definition and Evaluation of Phonetic Units for Speech Recognition By Phonetic Units
Int. Conf. ICASSP86, April 1986, Tokyo

/2/ Davis S. B. and Mermelstein P., Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences
IEEE Trans. ASSP, Vol. 28, Nr. 4, August 1980