Sādhanā, Vol. 23, Part 4, August 1998, pp. 313–340. © Printed in India.

A feature-based hierarchical speech recognition system for Hindi

K SAMUDRAVIJAYA, R AHUJA, N BONDALE, T JOSE, S KRISHNAN, P PODDAR, P V S RAO* and R RAVEENDRAN

Computer Systems and Communications Group, Tata Institute of Fundamental Research, Homi Bhabha Road, Bombay 400 005, India
email: {chief, renu, nandini, jose, krish, poddar, rao, ravee}@tifr.res.in

MS received 17 December 1997

Abstract. This paper presents a description of a speech recognition system for *Hindi*. The system follows a hierarchic approach to speech recognition and integrates multiple knowledge sources within statistical pattern recognition paradigms at various stages of signal decoding. Rather than make hard decisions at the level of each processing unit, relative confidence scores of individual units are propagated to higher levels. Phoneme recognition is achieved in two stages: broad acoustic classification of a frame is followed by fine acoustic classification. A semi-Markov model processes the frame level outputs of a broad acoustic maximum likelihood classifier to yield a sequence of segments with broad acoustic labels. The phonemic identities of selected classes of segments are decoded by class-dependent neural nets which are trained with class-specific feature vectors as input. Lexical access is achieved by string matching using a dynamic programming technique. A novel language processor disambiguates between multiple choices given by the acoustic recognizer to recognize the spoken sentence.

Keywords. Speech recognition; hierarchical approach; Hindi; knowledge integration; natural language processing.

1. Introduction

Despite major advances in computer technology, interaction between the computer and the user is still largely confined to keyboard input and screen or printed output. Communication in the speech mode is a very important aspect of multi-modal human–machine interaction

*For correspondence

because speech is a fast and convenient means of communication among human beings. Research in speech synthesis, recognition and coding has been pursued in several countries over the last several decades. Most of this work, for obvious reasons has been for the English language.

The use of computers in the voice mode is particularly important and relevant to a multi-lingual country such as India. First, the facility for information input and output in the speech mode would bring the computer within the reach of a large population of semi-literate users who today are likely to be intimidated by the need to operate a keyboard. Second, a microphone and speaker-oriented terminal is likely to be much cheaper and robust than a keyboard and screen-oriented system. However, Indian languages pose a challenge because of their phonetic richness which requires a detailed study of language-specific features in relation to speech recognition and understanding. This was the primary motivation of the Knowledge-Based Computer Systems project carried out at the Tata Institute of Fundamental Research. An important component of this activity has been the development of the Voice Oriented Interactive Computing Environment (VOICE), in other words, a voice-activated terminal.

The objective of the project was to develop an input/output interface to a computer with a facility for voice and visual feedback. Considering that there was no such voice-oriented system in Indian languages, a system working in a well-defined and strictly delimited task environment was aimed at as a first step. The system was planned to accept *Hindi* sentences clearly spoken by a speaker (with pauses between words, drawn from a vocabulary of about 200 words related to railway reservation enquiries) and to synthesize intelligible speech in Hindi. An overview of the major accomplishments of the project can be found in Rao (1993). The synthesis subsystem of the the speech I/O system is described by Furtado & Sen (1996). The recognition subsystem is described here.

Two general philosophies of speech recognition that are commonly used are the statistical and the knowledge-based approach. The statistical approach primarily addresses the problem of variability in the speech signal with the aim of discovering the underlying structure using the data alone and without making specific *a priori* assumptions about the speech signal (Levinson 1985). At the other end of the spectrum, the knowledge-based approach is based on the premise that a proper understanding of the acoustic-phonetic aspects of speech production and perception is essential for speech recognition (Cole *et al* 1980). In reality, there exists a continuum between these two extremes, leading to an integrated approach (Makhoul & Schwartz 1985). We have focussed on an integrated approach to speech recognition for Hindi.

The rest of the paper is organized as follows. Section 2 deals with special features of the sounds of Indian languages and bring out the need for a comprehensive study of these features. In § 3, the inherent notion of hierarchy of phonemes based on sound features is described. An overview of a database of segmented and labelled Hindi sentences developed for the aforementioned purpose is given in § 4. In § 5, we give a brief discussion on signal processing for extraction of features to represent dynamic as well as static aspects of speech signal. An algorithm for selecting a subset of discriminative features based on an extension of Fisher's criterion to a multi-class situation is also described in the same section. Section 6 deals with the hierarchical approach to classification. It describes classification of speech frames into one of broad acoustic classes by a multi-variate Gaussian, the generation of segments with broad acoustic labels by a semi-Markov model and fine-level (phonemic) labelling of selected segments. In § 7, the modelling of a lexicon in terms of acoustic and durational attributes is described. The domain knowledge is used in the design of a distance metric for comparing lexical templates. An outline of language models utilizing syntactic and/or semantic/pragmatic knowledge of the task domain is given in § 8. The performance of the overall system is presented in § 9. Some proposed improvements and extensions to the system are outlined in § 10. A summary of the work is presented in § 11.

2. Special features of Indian languages

The acoustic-phonetic profile of Hindi (and other Indian languages) differs considerably from European languages. In the context of incorporating language specific features into speech recognition system, it may be worthwhile to delve into these characteristics and point out how they lead to the specific recognition strategy adopted here. In the following, speech sounds in Hindi are compared and contrasted with those of English (due to our familiarity with English).

The Hindi alphabet (in Devanagari script with the corresponding IPA symbols) is shown in table 1. It has three sections: the first section lists the vowels, the second section deals with phonemes whose production involves complete closure of oral tract (plosives, affricates and nasals); the third section lists the semivowels and fricatives.

Table 1. Hindi alphabet and its corresponding IPA symbols.

अ	आ	इ	ई	उ	ऊ	ए	ऐ	ओ	औ
a	aː	*i*	*i*ː	*u*	*u*ː	*e*	ae	*o*	au

क k	ख k^h	ग g	घ g^h	ङ ŋ
च tʃ	छ tʃ^h	ज ʤ	झ ʤ^h	ञ ɲ
ट ʈ	ठ ʈ^h	ड ɖ	ढ ɖ^h	ण ɳ
त t	थ t^h	द d	ध d^h	न n
प p	फ p^h	ब b	भ b^h	म m

य	र	ल	व	श	ष	स	ह
j	*r*	*l*	ω	ʃ	ʂ	*s*	*h*

The second section covers more than half the phonemes in Indian languages; it is this section containing stop consonants which differs most from the ensemble of similar phonemes in English. The phonemes in this section are arranged in a 5 × 5 matrix according to the manner and place of articulation. There are 5 rows corresponding to 5 places of articulation: velar, palatal, retroflex, dental and labial. The plosives are arranged in the first 4 columns according to whether they are voiced and/or aspirated or not. The fifth column contains the nasals. The phonemes in the first column are unvoiced stop consonants. In a given row, a phoneme in the third column is the voiced counterpart of the phoneme in the first column. The phonemes in the second and fourth columns are aspirated counterparts of the sounds in the first and third columns respectively. Aspiration is a phonemic attribute in Hindi. Thus Hindi has 20 phonemes in the plosive category (including the affricates) whereas English has only 8. This abundance of plosives in Hindi is, first, due to the existence of an extra place of articulation (retroflex) and, second, due to its having a set of aspirated plosives (voiced as well as unvoiced). As Hindi and most other Indian languages have phonetic scripts, the alphabet is also organized on the basis of the production mechanisms of phonemes.

Here it may be worth mentioning some of the other special features in Indian languages. The attribute of retroflexion is not restricted to stop consonants alone. Unvoiced retroflexed fricative /ş/ (which appears in the third section in table 1) is a phoneme in most Indian languages. Many others have retroflexed lateral semivowel /l/, although it is not a phoneme in Hindi. Malayalam, a Dravidian language, has retroflexed trill as well. Marathi has an alveolar affricate /c/.

In comparison to stop consonants, Indian languages have a much smaller inventory of fricatives. The labio-dental and alveo-dental fricatives of English are absent. Interestingly, they are often approximated by closest sound of the native tongue while pronouncing English words. Whereas /v/ is substituted by /ω/, /θ/ (/t^h/ as in word "thin") and /∂/ (/d/ as in word "the") are substituted by dental plosives /t^h/ and /d/ respectively. The Marathi language improves upon this approximation by using aspirated /w^h/ in place of /v/.

From the above discussion, it emerges that aspiration has a strong presence as a phonemic attribute in Indian languages. Hence a speech recognition system for an Indian language should take into account the change in the acoustic quality of unaspirated stops (as compared to those of English) due to the extra phonemic dimension of aspiration.

In English, aspiration is used mainly to cue the absence of voicing in word-initial plosives. In fact, aspiration following a burst is a very important factor in distinguishing an unvoiced stop from a voiced one. This advantage is absent in Hindi due to the phonemic character of aspiration. Here, initiation of vocal cord vibrations prior to release is one major cue to identify a voiced stop (Davis 1994).

3. The notion of hierarchy in phoneme classification

An inherent notion of hierarchy exists in describing the phonetic units of a spoken language and, in general, these units are grouped into broader categories (such as vowels, fricatives, plosives etc.) based on their manner of articulation. Also, the organization of dominant stop consonants according to their place and manner of articulation in the Hindi alphabet suggests a hierarchical approach to recognition of phonemes. Phonemes generated by a given manner of articulation generally have similar broad acoustic characteristics. A major advantage of a hierarchical classification strategy over a single stage classifier is that a complex decision surface can be replaced by a combination of relatively simpler decisions. This involves fewer parameters at each stage of classification; these can be estimated better even with limited training data, resulting in better classification accuracy (Poddar & Rao 1993). In addition, features appropriate for the subtask in hand can be used at each stage of classification. This strategy also allows for progressive improvement of the system capabilities by replacing processing modules by better ones, as and when they are available. If the process of decoding a spoken message is viewed as a search process, a hierarchical organization results in pruning the search space at each stage of decoding as well.

The existence of natural organization of different levels of speech perception: e.g. acoustic-phonetic, phonemic, syllabic, lexical etc. is another motivation for looking at the classification problem in a hierarchical framework. The message-decoding process can progress in stages by focusing attention to different levels of abstraction of the signal at successive stages. Ideally, the processing modules in the different strata should interact with each other, each strengthening or weakening the hypotheses generated by other modules in the spirit of a blackboard model (Brachman 1978). We have implemented bottom-up decoding as a simpler but important first step towards this. Nevertheless, we do incorporate some of the features of the more general structure: e.g. by propagating relative confidence scores of individual processing units to higher levels, rather than make hard decisions locally at each level. In the present system, the process of phoneme recognition is

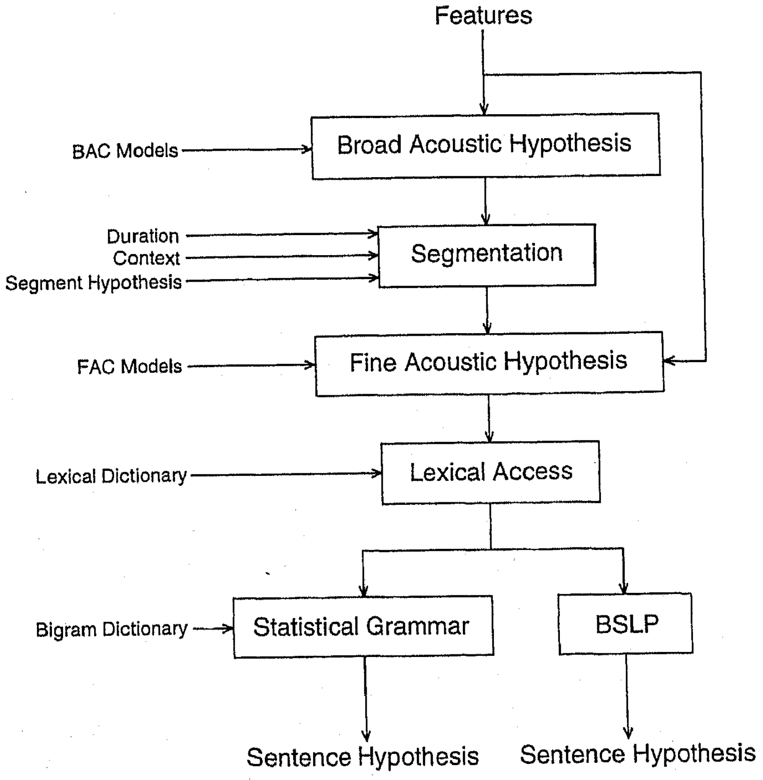

Figure 1. Block diagram of the hierarchical speech recognition system.

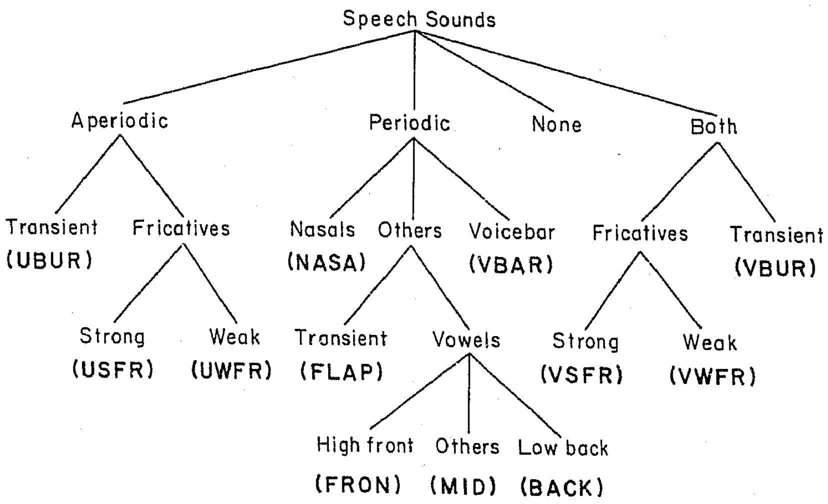

Figure 2. Broad acoustic space of speech sounds.

achieved in two stages: classification of short duration speech segments into Broad Acoustic Classes (BACs) followed by a second stage Fine Acoustic Classification (FAC) into individual phonemes. Figure 1 is a functional diagram of our speech recognition system based on this approach.

Broad acoustic classes are chosen to satisfy the following desirable properties.

(1) Each group should have one or more identifiable acoustic features associated with it.

(2) The categorization of phonemes should help in pruning the search space during recognition phase.

The clustering of the ensemble of phonemes into the BACs is schematically shown in figure 2. This classification scheme has an articulatory basis. The mode of excitation of the resonant cavity is used to divide the set of phonemes into four subsets. Sounds generated by pure aperiodic excitation are classified into transients and continuants. The transient is the release/burst of an unvoiced plosive and is characterized by a narrow peak in the energy contour as well as relatively flat spectrum. The continuant fricatives are subdivided, depending on the strident energy, into two classes: strong and weak unvoiced. Similarly, sounds produced by mixed (i.e., periodic and aperiodic) mode of excitation are grouped into three classes. The interword silence and closure of unvoiced plosives together constitute a separate class. Sounds produced by pure periodic excitation are divided into 3 subclasses depending on the mode of radiation of sound energy. The voicebar, the sound energy radiated from the walls of pharyngeal and oral cavities during the closure of a voiced plosive, forms a class by itself. The production of nasals involves the closure of the oral cavity and lowering of the velum. Sounds radiated only through the mouth are grouped into flap and sonorant classes based on the shape of waveform envelope. The intervocalic /r/ and /d/ in Indian languages often manifest themselves as flaps. This is characterized by a narrow valley in the temporal trajectory of amplitude. The vowels are further subdivided into three classes based on the location of the first two formants in the spectrum.

Table 2. Glossary of the broad acoustic lexicon.

FRON	High front vowels
BACK	Low back vowels
MID	Other vowels
NASA	Nasals
FLAP	Retroflexed flap
UBUR	Unvoiced burst
VBUR	Voiced burst
USFR	Unvoiced strident fricatives
VSFR	Voiced strident fricatives
UWFR	Unvoiced weak fricatives
VWFR	Voiced weak fricatives
VBAR	Voiced closure
SIL	Unvoiced closure

The 13 BACs are listed in table 2. Here, the voiced and unvoiced fricatives are further subdivided into strong and weak ones. This is to take into account the important phonemic role played by the presence or absence of aspiration in Indian languages.

4. Speech database

The choice of phonemes as units of representation demands an inventory of segmented and labelled speech data for training and testing the speech recognition system. Since databases for non-Indian languages cannot be used for Hindi (owing to the language specific effects discussed in previous section), a speech database in Hindi was designed and developed. The vocabulary consists of about 200 words most commonly used in a railway reservation enquiry task. Typical queries in this context were solicited from volunteers. Based on these responses, sentence templates were formed in terms of word categories such as city names, classes of tickets, verb names etc.

4.1 *Speech recording*

The sentences, spoken by a male speaker with pauses between words, were recorded in a sound-treated room in the presence of ambient noise. A Sennheisser directional microphone (Model MD 412 LM) was kept at a distance of about 10 cm from the lips of the speaker. The speech signal was low pass filtered at 7.0 kHz using a Wavetek filter and digitized at 16 kHz rate with 16-bit quantization using a DSC200 A-D/D-A converter.

Energy, zero crossing rate and peak-to-peak amplitude of speech data frames of 9 ms duration were calculated at 3 ms intervals. Energies in the frequency ranges 60–250 Hz, 0–4 kHz and 4–8 kHz were used to capture voicing, vocal tract resonance, and frication respectively. Frequencies and amplitudes of the peaks in the linear prediction spectrum were computed. These parameters were plotted on VAXstation screen to facilitate the segmentation and labelling procedure.

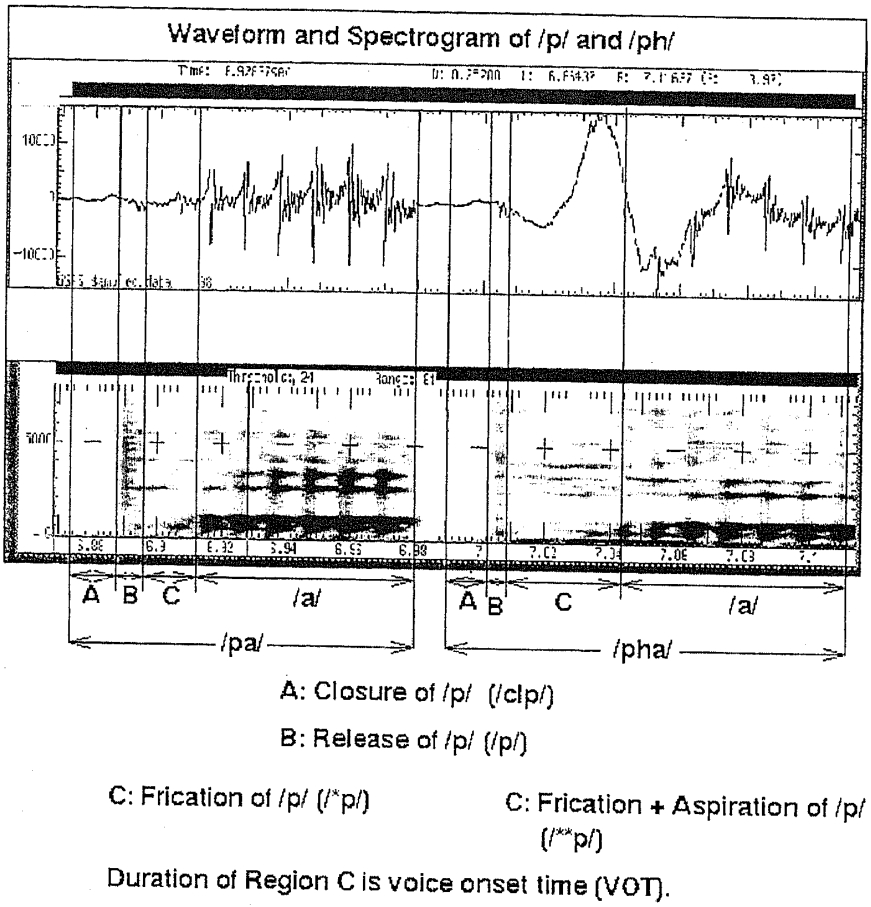

Figure 3. Waveforms with spectrograms of unaspirated and aspirated stops "पल्" /p/ and "फल्" /p^h/ in example words.

4.2 *Units of representation*

Although this database was developed to cater to the needs of the current recognition system, the design goals were of a general nature. The main aim was that the labelling scheme should meet the demands of not only phoneme-based continuous speech recognition systems in different task domains but also those of systems following different design philosophies. A properly designed database would also facilitate studies of acoustic-phonetic correlates of the language.

Phonemes are defined with respect to their linguistic function in a language. A given phoneme need not be an acoustically homogeneous unit. In the context of machine recognition of speech, acoustic homogeneity of the units of representation is important because it is the spectral similarity of different realizations of a unit which enables a recognition system to infer its identity. This was taken into account while choosing the units of segmentation; most of them are phonemes and some of them are sub-phonetic units. Speech spectrograms were used as a guide in this process. For example, a well-articulated stop consonant may be segmented and labelled as four units: closure, burst, frication and aspiration. In fact, the duration and intensity of aspiration differenciates between aspirated and unaspirated plosives. This phenomenon is illustrated in figure 3. Here, the waveform and spectrogram of the phoneme [$/p/$ $/a/$] (as in the word "पल्") is shown along with that of [$/p^h//a/$] (as in the word "फल्").

In order to capture such differences in acoustic characteristics, a system of suffixes and prefixes was employed. Since Roman script was used to label the segments, several symbols were used which can be affixed either before or after the labels representing phonemes. For example, the aspiration of $/p/$ in "फल्" is marked as $/^{**}p/$ whereas the corresponding segment in "पल्" is marked as $/^{*}p/$. In both the cases, there is a frication following the release of the unvoiced stop consonant $/p/$. In the case of aspirated consonant, $/p^h/$, the region marked "C" in figure 3, carries frication as well as aspiration. Intervocalic $/r/$ and retroflex plosive (in non-geminated context) often manifest as taps or flaps. Such instances are recorded by using suffix "F" (i.e., labels $/rF/$ and $/dF/$). The nasalization of vowels, sometimes, makes lexical difference. For example, if the vowel $/ae/$ in the word "hai" is nasalized, it denotes that the subject is plural. Instances of nasalized vowels are recorded by using suffix N to the symbols of vowels. Further details including the inventory of labels, the graphical tools used for segmentation and consistency checks are discussed by Samudravijaya *et al* (1991). 200 sentences were manually labelled. These sentences were used for training the recognition system. The next 50 sentences in the database were used to test the system. An effort is underway to implement a semi-automatic labelling system (Samudravijaya *et al* 1994).

5. Signal processing

The analog speech signal, used for training and testing the recognition system, is passed through an anti-aliasing filter with a cut-off frequency of 7.0 kHz and digitized with 15-bit resolution at 16 kHz sampling rate.

5.1 *Endpointing*

The current design of the system allows for recognition of isolated words. The speech waveform is processed to detect the endpoints of the words constituting the spoken utterance. This avoids the time-consuming computation of the full feature vector during the silence period. Endpoint detection is similar to the one employed by Rabiner & Sambur (1975) and is based on sets of upper and lower thresholds corresponding to Zero Crossing Rate (ZCR), maximum Peak-To-Peak amplitude (P2P) and log energy of the analysis frames of 9 ms duration. These threshold values are updated for every word based on their values during the word.

5.2 *Feature extraction and selection*

The recognition system uses a 30-dimensional feature vector comprising dynamic as well as static features. It primarily consists of cepstral coefficients, their temporal derivatives and features such as zero crossing rate. This section describes the processing of the speech signal to arrive at the feature vector.

The preemphasised (pre-emphasis factor = 0.98) and Hamming windowed signal was analysed in terms of analysis frames of 9 ms duration with a frame shift of 3 ms. The analysis frame size and frame shift interval are chosen to be small so that short duration acoustic events such as bursts are represented sufficiently well. The set of cepstral coefficients was derived from prediction coefficients (order = 16) using the recursion relation between them (Markel & Gray 1976). The cepstral coefficients are weighted by a triangular window to obtain quefrency weighted coefficients.

In this system, the spectral dynamics is represented by the first temporal derivatives of the cepstral coefficients. The rate of change of spectral properties can vary depending on the speaking rate, the phonemic context etc. In order to accommodate a range of spectral changes which can occur in natural speech, it was decided to provide for two sets of cepstral derivatives. Let $C(k, n)$ denote the kth quefrency weighted cepstral coefficient corresponding to the nth frame. The zeroth cepstral coefficient is the log energy of the analysis frame and is normalized in the range (0,1) over the utterance. The Near Context feature NC(k, n) and Far Context feature FC(k, n) are defined as

$$\mathrm{NC}(k, n) = 2 * C(k, n) - C(k, n+2) - C(k, n-2),$$
$$\mathrm{FC}(k, n) = C(k, n+6) - C(k, n-6).$$

While the far context feature captures spectral change over a 39 ms window, the near context feature enhances the detectability of short-term acoustic events such as bursts.

The 17 cepstral coefficients together with features capturing their temporal variations constitute the 51 components of the 55-dimensional feature set. Four additional features which do not depend on cepstral analysis but are known to be useful in speech recognition are also included in the feature set. They are the zero crossing rate, the first reflection coefficient, the normalized residual energy obtained by linear prediction analysis and a special feature called flap enhancer. The motivation in using the feature flap enhancer, is (as mentioned in the section on database) the manifestation of intervocalic $/r/$ and retroflex plosives as flaps. They are characterized by a sharp dip in the amplitude curve. The flap enhancer is defined as

$$FE(n) = A_{\mathrm{left}}(n) + A_{\mathrm{right}}(n) - 2 * P2P(n),$$
$$A_{\mathrm{left}}(n) = \mathrm{MAX}(P2P(n-4), P2P(n-6)),$$
$$A_{\mathrm{right}}(n) = \mathrm{MAX}(P2P(n+4), P2P(n+6)).$$

Here P2P(n) denotes the maximum peak-to-peak amplitude of the nth frame.

The 55-dimensional super feature set is arrived at by considering the need for representing the spectrum and its temporal variations at the frame level. The set also includes features which are expected to be useful from acoustic-phonetic knowledge considerations. However, it is likely that only a subset of these features may actually be useful for classification. A recognizer using a large feature set can be computationally more complex. In addition, it requires a large amount of training data (Kanal & Chandrasekaran 1971). In order to alleviate these problems, an attempt was made to reduce the dimensionality of the feature space.

The Fisher criterion (Duda & Hart 1973) is a well-known method of rating the goodness of an individual feature for pattern recognition based on F-ratio. It is the ratio of the inter-class variance and intra-class variance. Pairwise Fisher's ratio is defined

$$F_{ijk} = (\mu_{ik} - \mu_{jk})^2/(\sigma_{ik}^2 + \sigma_{jk}^2),$$

where μ_{ik} and μ_{jk} denote the cluster means for the kth vector component of classes i and j respectively, and σ_{ik}^2 and σ_{jk}^2 the corresponding cluster variances. We used the following generalized definition of F-ratio to a M-class classification problem

$$F_k = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1}^{M} F_{ijk}; \quad i \neq j.$$

The protocol for feature selection is to find the average class-pair F-ratio for each vector dimension (or feature) and order the F-ratios. The first N features are then selected to get the best recognition accuracy on the broad phonetic categories.

The F-ratios, F_k, were computed for all the components of the 55 dimensional super feature set from the training data. The rank ordering of the 55 features in terms of their F-ratios is listed in table 3.

The F-ratio of the least discriminating feature is about an order of magnitude less than that of the best feature. The incorporation of features with low F-ratios in the feature set may not enhance its discriminatory capability. On the other hand, they may adversely affect the classification performance if training data is not adequate. Thus, a smaller set of features would be preferable. This set can, in principle, be identified by actual classification experiments. However, the process of determining the optimal feature set by exhaustive experimentation is prohibitively expensive. A simpler solution is to assume that the best N-dimensional feature vector consists of the N features with the largest F-ratios. The optimal value of N was decided by a simple experiment as described below.

A Gaussian classifier was trained with 150,000 samples belonging to the 13 classes and frame level recognition accuracy was computed for each class. In each set of experiments,

Table 3. Rank ordering of the 55 features in the super feature set in terms of their F-ratios.

Rank	Feature	F-ratio	Rank	Feature	F-ratio	Rank	Feature	F-ratio
1	ZCR	1.596	2	C_1	1.552	3	RC1	1.474
4	C_0	1.300	5	FE	0.809	6	ER	0.809
7	C_2	0.772	8	C_7	0.732	9	NC_0	0.727
10	C_6	0.714	11	C_9	0.640	12	C_5	0.633
13	FC_0	0.618	14	C_{11}	0.530	15	C_{10}	0.491
16	C_3	0.485	17	C_{15}	0.461	18	NC_2	0.433
19	C_8	0.431	20	C_4	0.428	21	C_{16}	0.389
22	C_{12}	0.375	23	C_{14}	0.357	24	NC_6	0.351
25	FC_6	0.319	26	FC_{11}	0.289	27	FC_2	0.285
29	C_{13}	0.285	28	FC_3	0.262	30	C_5	0.253
31	FC_1	0.247	32	FC_4	0.225	33	FC_{14}	0.218
34	FC_7	0.215	35	FC_{16}	0.209	36	FC_9	0.208
37	NC_3	0.202	38	NC_1	0.187	39	FC_{15}	0.184
40	NC_7	0.184	41	NC_4	0.180	42	NC_{13}	0.170
43	FC_8	0.149	44	NC_9	0.142	45	FC_{13}	0.129
46	FC_{10}	0.117	47	NC_8	0.116	48	NC_5	0.112
49	FC_{12}	0.104	50	NC_{10}	0.100	51	NC_{12}	0.100
52	NC_{15}	0.095	53	NC_{14}	0.087	54	NC_{16}	0.084
55	NC_{13}	0.061						

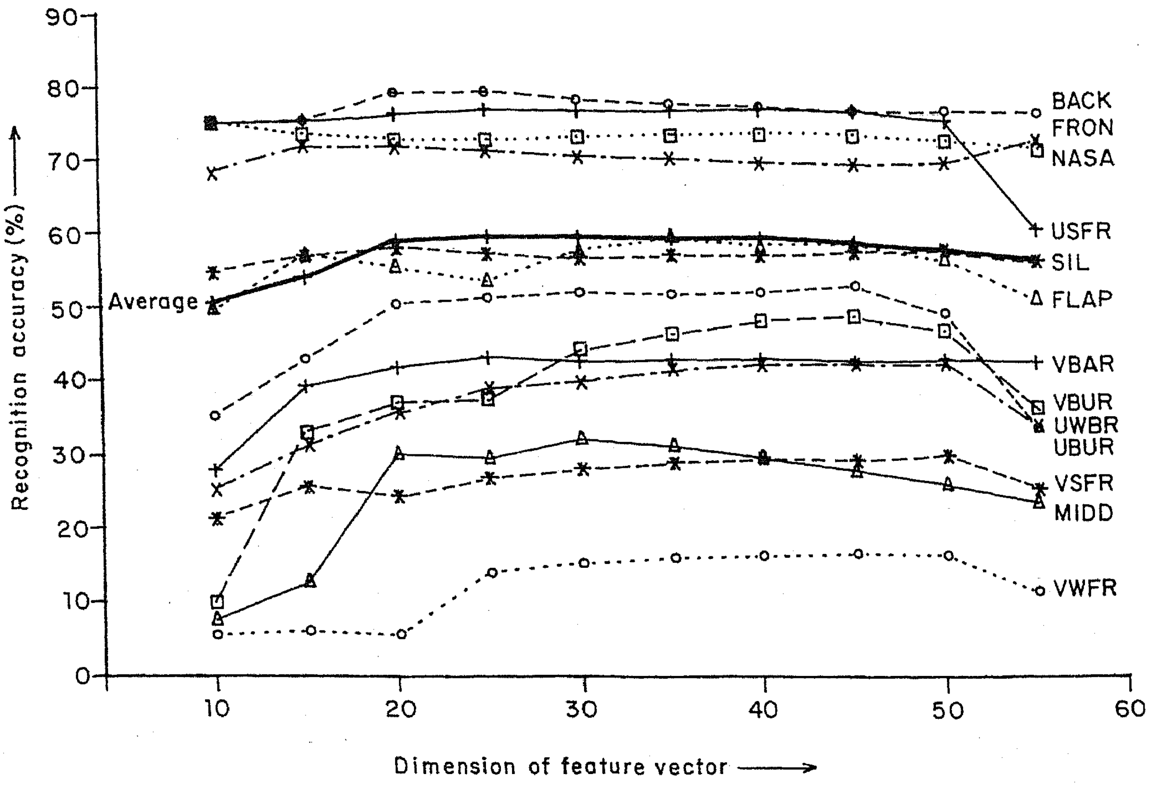

Figure 4. Recognition accuracy of the broad acoustic classifiers as a function of feature vector dimension.

the N best (in terms of highest F-ratio) features were used with N varying from 10 to 55 in steps of 5. The results of these experiments are presented in figure 4. Here, the correct classification rates corresponding to each of the 13 BACs are plotted as a function of the dimensionality of the feature vector. For all classes, the classification accuracy at the frame level generally increases till $N = 30$, remains steady for a while, and then marginally decreases as N increases. Overall classification rate, independent of the class label, is also plotted in the same figure as a solid dark line. Even here, the vector comprising 30 best features appears to offer the best performance with the fewest dimensions. Hence, it was decided to use 30 best features for pattern classification. This vector consists of 17 cepstral coefficients, the first few short- and long-term cepstral differences, and all the non-cepstrum based special features.

The choice of the best feature set seems to be in conformity with acoustic-phonetic considerations. Most of the specialized features appear in the pruned subset as expected. The *Flap Enhancer* feature introduced here, to represent the Broad Acoustic Class *Flap*, has an F-ratio larger than most of the cepstral coefficients. This supports the philosophy of using acoustic-phonetic knowledge to design a comprehensive set of features and the use of a statistical approach to select a proper subset which is as effective or even better than the superset. It is seen that only the first few coefficients of the short- and long-term cepstral differences are members of the pruned feature set. Thus, while speech-specific knowledge motivated our consideration of these dynamic features, the objective measure (F-ratio) guided us in selecting relevant components.

6. Hierarchical model of classification

The classifier architecture described in this section is a partitioning of a monolithic classifier into smaller classifiers. The modularization is guided by the knowledge of class hierarchy. The probability of a given feature vector **x** belonging to a fine acoustic class C_k is computed as

$$P(C_k|\mathbf{x}) = P(C_k|G_m, \mathbf{x})P(G_m|\mathbf{x}),$$

where C_k belongs to the broad acoustic class G_m.

6.1 *Broad acoustic classification and segmentation*

Segmentation of speech in terms of broad acoustic classes can be viewed as a joint optimal search for segments and their labels. A fully connected semi-Markov model (Ferguson 1986) has been used to model the segmentation problem. Here, full connectivity is required as typically any broad acoustic class can precede or follow any other class. The semi-Markov property in the model is employed to exploit duration cues which are especially useful in pruning spurious segments that may arise when acoustic cues are weak. The optimality itself is based on the Maximum Likelihood criterion.

The likelihood of a given feature vector belonging to a broad acoustic class was computed from the multi-variate Gaussian distributions used to model the classes. The initial value of the model parameters were estimated from labelled training data. The intrinsic durations of broad acoustic classes and inter-segment transition data were likewise initialized using labelled data. The parameters were then re-estimated to maximize the likelihood on the training data. The re-estimation was carried out iteratively wherein during each iteration optimal segmentation was first carried out using the existing model, and then computing the new model based on the segments obtained in the current iteration.

As mentioned earlier, whereas the introduction of explicit durations using semi-Markov models can potentially improve segmentation, the computational load increases considerably. If the number of states in the model is N, and the length of the utterance in frames is T, and the maximum duration of a broad acoustic class is T_m, the segmentation complexity is $O(N^2TT_m)$. This is typically about two orders of magnitude more than that of conventional hidden Markov models $O(N^2T)$. To alleviate the problem of the increased computational complexity in semi-Markov models a constrained search approach was proposed. This reduces the search time without unduly sacrificing segmentation performance. The basic idea here is to make use of an acoustic landmark-based candidate boundaries to anchor the search, and at the same time ensure optimality using segmental dynamic programming. More details on the algorithm can be found elsewhere (Samudravijaya *et al* 1994; Krishnan 1994).

In the work being described here, acoustic landmarks based on the multi-level dendrogram proposed by Glass (1988) was employed. This is based on hierarchical clustering of seed segments. However, there is a difference. Whereas Glass generated dendrograms based on the auditory spectrum, we have generated dendrograms based on frame-level *a posteriori* acoustic class probabilities. These probabilities are generated by normalizing the class conditional likelihoods. The primary reason is reduction in computation; the

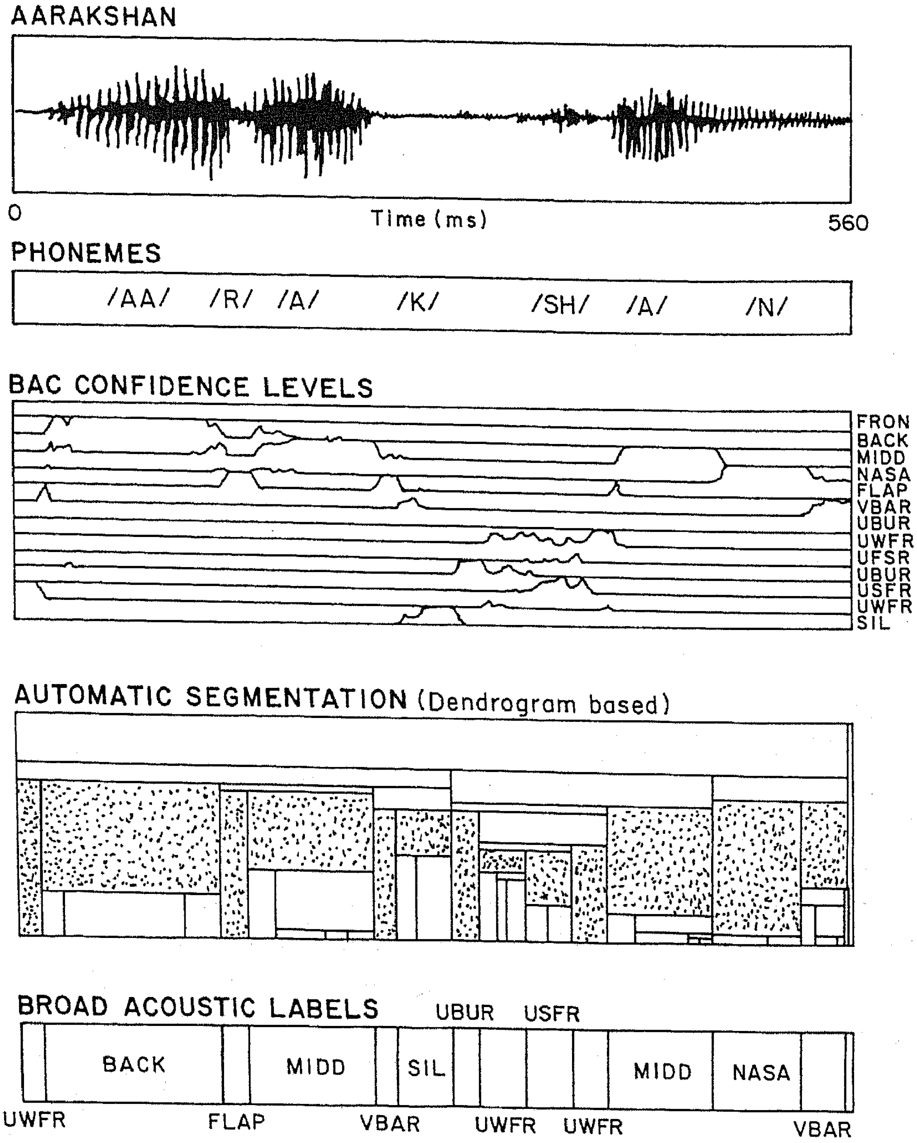

Figure 5. Dendrogram aided segmentation of a word by a semi-Markov model.

dimension of the feature vector is reduced from 40 (in case of the auditory spectrum) to 13 (the number of broad acoustic classes). Another reason is that the feature set used in the present work employs a cepstrum-based representation which tends to be relatively more noisy, thereby rendering the generation of good dendrograms more difficult. Figure 5 illustrates the segmentation achieved by the semi-Markov model using dendrogram based acoustic landmarks for the Hindi word "आरक्षण".

Optimal segmentation of the spoken utterance is performed using the maximum likelihood criterion, as mentioned previously. The constrained search was seen to reduce the computational load by a factor of 50 in comparison with the full search. The method yielded labelling accuracy of 72% using the 13 broad acoustic classes and full search. Substitutions account for the majority (67%) of the errors. This is followed by deletions (24%) and insertion (9%) errors. The substitution errors were seen to occur primarily due to the following confusions: (i) mid and front, and mid and back vowels; (ii) voiced, unvoiced weak fricatives and voiced, unvoiced plosives; (iii) nasals and voiced closure. Many of these confusions are understandable and expected and have been dealt with at the fine classification stage. Insertion errors were primarily seen to occur on flaps. The phone boundary misalignment with reference to the manual boundaries were found to be small and largely *systematic*. The acoustic landmark constrained algorithm performed slightly poorer than the full search algorithm. Here, deletion accounted for the majority of the errors (5.9%), treating the full search segmentation as the reference. The primary source of deletion was seen to be weak unaspirated plosives. This is followed by insertions (1.5%), where the flaps were the major source of error.

6.2 *Fine acoustic classification*

The output of the broad acoustic classification module is a template in the form of a sequence of segments along with their broad acoustic labels, durations, and scores. Following the hierarchical approach, the broad acoustic labels of a segment are then refined to fine acoustic labels using class-specific features. If fine acoustic classification is to be done for all the 13 classes, the amount of training data available will, on an average, be an order of magnitude less than that for broad acoustic classification. Moreover, the actual amount of training data available for certain BACs is much less because of the uneven distribution of BACs in the labelled database. Hence, in the context of the present task, we decided to restrict this stage of processing to some classes based on certain criteria. First, enough data should be available to estimate the parameters of the classifier. Second, resolution of these segments should result in unique templates for all the words in the vocabulary of the task.

The requirement for training data increases with the number of parameters of the classifier. For example, it is essential to capture spectral dynamics of stop consonants well in order to obtain satisfactory recognition performance. Due to paucity of sufficient labelled data, the refinement of broad acoustic labels was limited to steady state segments such as vowels and nasals. Also, the vowels occur more frequently than obstruents such as plosives in any speech database. So, in the current implementation of the system, all segments with either a vowel or nasal broad acoustic label are further processed by the subsequent module to hypothesize their fine acoustic labels.

6.2a *Vowel classification*: The BAC classifier uses a feature vector comprising components suitable for capturing dynamic as well as static characteristics of phonemes. Vowels, being continuants, can be represented by the instantaneous shape of the spectrum and hence features characterizing temporal variations are not essential for their classification. We employed a neural network-based vowel classifier which uses 8 quefrency weighted cepstral coefficients – a proper subset of the 30-dimensional super feature set used for broad acoustic classification – to arrive at the phonemic identity of the vowels.

Each vowel belongs to one of three BACs, each of which have three members – Front: $/i/$, $/e/$, $/ea/$; Middle: $/ae/$, $/\Lambda/$, $/a/$; and Back: $/u/$, $/o/$, $/ao/$. For each of these BACs, a 3-layer perceptron with 8 input, 6 hidden and 3 output nodes was employed to determine the identity of the vowel. Each network was trained with 155 samples and was tested with a dataset of equal size. The vowel classification networks achieved an overall recognition accuracy of 95.2% on the training set and 80.8% on the test set. The considerable difference between classification accuracies on the training and test sets indicates the inadequacy of the training set size. In such a scenario, a single-stage classifier of 9 vowels can be expected to perform worse than the hierarchical classification strategy adopted in the system. In order to verify this hypothesis, the training data comprising all nine vowels was used to train a 8-16-9 single-stage classifier. The number of units in the hidden layer was so chosen that the complexity of hierarchical and direct classifiers are comparable. The recognition accuracies of the latter for training and test sets were 71.2% and 56.3% respectively. The better performance of the 2-stage classifier demonstrates the advantage of hierarchical classification strategy in case of limited data. Also, the hierarchical network needs less time for learning than the equivalent single-stage classifier.

6.2b *Nasal-voicebar classification*: Apart from the BACs associated with vowels, nasals and the voicebar, are two classes whose members are voiced continuants occurring relatively frequently in natural speech. Therefore, it is worthwhile to perform fine classification of these broad groups. A common articulatory characteristic of members of both these groups is that the oral branch is kept closed at one end; hence acoustic energy from the glottis is not transmitted through the oral cavity. In the case of nasals, it passes through the nasal branch

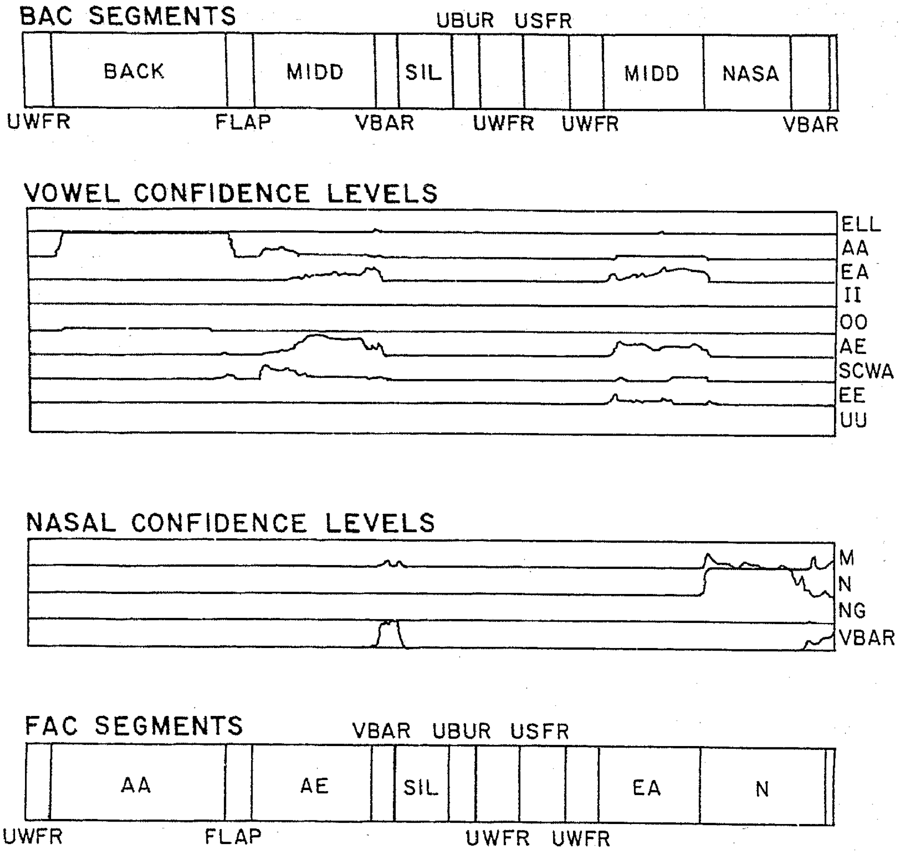

Figure 6. Output of fine acoustic classifiers and the revised labels.

and gets radiated through the nostrils. Consequently, the signature of the corresponding oral cavity configuration is normally evident in the spectrum of a nasal, while it is rarely observable in the voicebar spectrum. However, the energies of word-initial nasals are often so low that the high and medium frequency components are very weak compared to those corresponding to the glottal excitation; this blurs the distinction between such nasals and a voicebar in such situations. Consequently, a significant number of word-initial nasals are classified as voicebars by the broad acoustic classifier. In order to rectify this anomaly, we used a single fine acoustic classifier to discriminate between the three nasals and the voicebar. A 17-6-4 neural network was trained by 1140 patterns with the first 17 components of the super feature vector (q-weighted cepstral coefficients) as the input to the network. The trained network performed with an accuracy of 90% and 88% on training and test corpus respectively.

6.2c *Word template*: The phonemic labels provided by the vowel and nasal fine acoustic classifiers replace the broad acoustic labels of the corresponding segments in the word template. The output of the intra-category classifiers and the revised template of the word आरक्षण are shown in figure 6.

7. Lexical access

In the preceeding sections, we described the process of converting a continuous time-domain signal of an utterance into a sequence of word templates. The processing of these word templates in the symbolic domain to recognize the spoken sentence is described in this and the following sections. The two stages of this symbolic processing are (1) the generation of a sorted list of word candidates corresponding to each test word template (i.e., the lexical access), and (2) the determination of an optimum sequence of words – the recognized sentence – based on a model of grammar of the language, i.e., the syntactic processing.

The goal of lexical processing is to identify within the lexicon the word which best matches the test word. In a general case, the lexical module may yield a sorted list of word candidates together with measures of their similarity with the test word, so that this list may be processed by a syntactic module. The design of lexical module primarily consists of two tasks: (a) deciding a scheme for representing words in terms of their attributes, and (b) defining a measure of similarity between representations of two words in such a manner that the system's performance is optimal for the specified recognition task. Once a representation scheme and a similarity measure are chosen, a dictionary of reference templates of the words can be developed. For each word, one or more reference templates can be generated out of templates corresponding to multiple repetitions of the word. During the recognition phase, the template of a test word is compared with all the reference templates and the word whose reference template is closest to the test template is chosen as the spoken word.

Matching word templates is essentially a string comparison process, since each template is a sequence of acoustic-phonetic segments along with their attributes. Dynamic programming techniques can be used to find a globally optimal path for matching two sequences by making local decisions at each point in the matching grid. The cumulative sum of the local costs along the optimal path provides a measure of dissimilarity of the matching.

The number of segments generated by a classifier-segmenter module corresponding to a test word need not be equal to that in a reference template of the correct word. Poorly articulated short segments may not be detected; extra segments may be introduced by artifacts such as lip smacks or tongue clicks. Also, due to the inherent confusibility of phonemes, a segment may be assigned an incorrect label which corresponds to an acoustically similar phoneme. The consequent insertion, deletion or substitution of segments has to be handled by the matching process to obtain the optimal word label. An extra (missing) segment will invariably lead to a mismatch of a nearby segment in the reference (test) template. While the resultant increase in local distance is difficult to handle, the higher local cost arising from a substitution error can be contained by making use of *a priori* knowledge of acoustic similarities of units in the matching process. It is desirable that the matching process should also take into account intrinsic durational distribution of different units of speech.

7.1 *Inter-segment distance*

There are several ways of defining inter-segment distance which embodies *a priori* knowledge about durational distributions and the inherent confusability of acoustic-phonetic segments in the feature space, in addition to the attributes of the segments comprising the word template. Let $s = (l, t)$ denote a segment where l is the broad acoustic label and t is its duration. A general definition of the inter-segment distance is

$$d(s_i, s_j) = H(g(l_i, l_j), f(t_i, t_j)).$$

Here $g(\cdot)$ and $f(\cdot)$ are the contributions due to the differences in the acoustic labels and the durations respectively and $H(\cdot)$ is a function which combines these two contributions to arrive at an integrated distance measure between the two segments. One simple measure would be to define $g(\cdot)$ as the Kronecker delta function, i.e., $g(l_i, l_j) = 1$, if $l_i = l_j$, and 0 otherwise.

Another way of specifying this contribution would be to take into account the imperfections in the acoustic-phonetic classification. Some of the non-diagonal elements of the confusion matrix may be non-zero due to overlap of the feature distributions corresponding to confusable phonetic units. $\mathrm{CM}[l_i, l_j]$, the (i, j)th element of the confusion matrix (CM), is a measure of the probability of a member of the ith class being classified as of the jth class. Thus, an alternate definition of the function $g(\cdot)$ is given by $g(l_i, l_j) = 1 - \mathrm{CM}[l_i, l_j]$, where CM is the confusion matrix derived from the performance of the acoustic-phonetic classifier with the training data.

The durational distance can be the normalized difference between the two durations, i.e.

$$f(t_i, t_j) = |t_i - t_j|/(t_i + t_j).$$

We know that the duration of an acoustic-phonetic segment depends on several factors such as the phonetic context, speaking rate etc. The extent of the spread of segment duration around the mean value varies considerably for different segments. An alternate definition of $f(\cdot)$ which accounts for this would be

$$f(t_i, t_j) = |t_i - t_j|/[\sigma(l_i) + \sigma(l_j)],$$

where $\sigma(l_i)$ and $\sigma(l_j)$ denote the standard deviations of the durations of segments belonging to the ith and jth acoustic classes respectively.

The two functions can be combined in two different ways:

$$H(g(\cdot), f(\cdot)) = g(\cdot) + f(\cdot),$$
$$H(g(\cdot), f(\cdot)) = g(\cdot) \times (1 + f(\cdot)).$$

In the first form, the contributions from both sources are treated equally, whereas in the second expression, the contribution from the difference in acoustic labels is given more weight than that of the durations.

7.2 *Generation of reference lexicon*

During the training phase, multiple examples of each word from a corpus of 200 sentences are used to generate the reference templates. The set median of all the templates of several exemplars of a given word is used as the reference template for that word. The median template is defined as the template whose total distance from all other members of the set is minimum. During the testing phase, all the templates of words from the sentences of the training set were matched with the reference lexicon of 207 templates. For each word, four best-matched templates were chosen. The performance of the lexical access module is tabulated for each type of distance metric in table 4.

From the experimental results listed in table 3, we observe that:

(i) integration of domain knowledge in the lexical access stage improves recognition performance significantly;

(ii) performance increases consistently when misclassification of phonetic classes is taken into account through the classifier confusion matrix, irrespective of the durational contribution or the way the two contributions are combined;

Table 4. Accuracy of lexical matching for different distance metrics.

Distance measure			Recognition accuracy	
Phonetic contribution $g(\cdot)$	Duration contribution $f(\cdot)$	Combination function $H(\cdot)$	1st choice	Within best 4 choices
δ_{l_i,l_j}	$\|t_i - t_j\|/(t_i + t_j)$	$g(\cdot) + f(\cdot)$	68.6	87.3
δ_{l_i,l_j}	$\|t_i - t_j\|/(t_i + t_j)$	$g(\cdot) \times (1 + f(\cdot))$	70.5	88.6
δ_{l_i,l_j}	$\|t_i - t_j\|/(\sigma l_i + \sigma l_j)$	$g(\cdot) + f(\cdot)$	71.4	89.0
δ_{l_i,l_j}	$\|t_i - t_j\|/(\sigma l_i + \sigma l_j)$	$g(\cdot) \times (1 + f(\cdot))$	72.8	89.7
$1 - \mathrm{CM}[l_i, l_j]$	$\|t_i - t_j\|/(t_i + t_j)$	$g(\cdot) + f(\cdot)$	73.3	89.7
$1 - \mathrm{CM}[l_i, l_j]$	$\|t_i - t_j\|/(t_i + t_j)$	$g(\cdot) \times (1 + f(\cdot))$	75.0	92.2
$1 - \mathrm{CM}[l_i, l_j]$	$\|t_i - t_j\|/(\sigma l_i + \sigma l_j)$	$g(\cdot) + f(\cdot)$	73.0	91.1
$1 - \mathrm{CM}[l_i, l_j]$	$\|t_i - t_j\|/(\sigma l_i + \sigma l_j)$	$g(\cdot) \times (1 + f(\cdot))$	78.2	92.8

(iii) the matching process produces more accurate results when the phonetic identity of the segments is given greater emphasis than their durational differences.

The best recognition accuracy of the lexical access module for an independent test set of 50 sentences (463 words) was 73.7% for the topmost choice and 90.9% within the best 4 choices.

7.3 *Word lattice*

The best distance measure amongst all these was used in subsequent experiments. The distance D_{mn} of the mth ($1 \leq m \leq M$) word in the spoken utterance with the nth ($1 \leq n \leq N$) closest reference template in the lexicon is converted to the corresponding normalized score p_{mn} through the following equation.

$$p_{mn} = \exp(1 - D_{mn}) \Big/ \sum_{n=1}^{N} \exp(1 - D_{mn})$$

This definition ensures that $0 \leq p_{mn} \leq 1$ and $\sum_{n=1}^{N} p_{mn} = 1$. Also, the greater the D_{mn}, the lesser the value of p_{mn}, as desired. The template of each word in the test sentence was compared with all the reference templates and the four best matches were chosen as the most likely candidates for that word in the test utterance. The lattice of N lexical candidates for each spoken word along with the normalized score p_{mn} is processed by a language model.

8. Language modelling

Two language processors have been successfully used. The first one is the conventional statistical grammar. The second, known as Blank Slate Language Processor (BSLP) which was developed as part of the project, uses semantic as well as syntactic knowledge.

As discussed in the section on speech database, the corpus design for the recognition system was based on naturally produced sentences using questionnaires. Consequently, we would expect to draw meaningful conclusions on language modelling using this corpus.

8.1 *Statistical grammar*

A simple method of modelling the syntax of a language is to use the statistical properties of word sequences in the corpus of sentences. The bigram grammar specifies the intrinsic probability of occurrence of ordered word pairs within the vocabulary. The bigram model is represented by an $N \times N$ matrix $|t_{nk}|$ where $t_{nk} = p(w_k|w_n)$ is the probability of the word w_n being followed by w_k. These word transition probabilities are estimated from frequency counts analysis performed on the training sentence set. However, due to the modest size of the corpus, these probabilities cannot be reliably estimated, as not all syntactically allowed word sequences may occur in the training set. Hence, the words are categorized into concept classes (*e.g.* city name, trains, days etc.) and a transition from an exemplar of one class into a member of another is counted as an instantiation of transitions from each member of one class to every member of another. This enabled the assignment of uniform probability to each member of a category to be followed by any member of another category.

Given the site probabilities p_{mn} (from the acoustic level analysis) and transition probabilities t_{nk} (from the bigram language model), the probability of a path is $\Pi_{p_{mn}} t_{nk}$, where the product is carried out over all lattice sites spanned by the path. The maximally probable path in the lattice denotes the most likely sentence satisfying both the acoustic and syntactic level constraints (Bahl 1984). Exhaustive enumeration of all possible paths in the lattice grows exponentially with M as there are N^M possible paths. A dynamic programming based algorithm (linear in M) is applied to find the best path through the lattice (Bellman 1957; Viterbi 1967). The word level recognition accuracy with the bigram module is 85.3%, which is significantly better than that for the top word choice (78.2%) but poorer than that for the best four choices (92.8%).

8.2 *Blank slate language processor*

Rather than use any *a priori* information regarding the language, the Blank Slate Language Processor (BSLP) starts virtually with a blank slate, and acquires most of its knowledge regarding the language structure during a training phase. During this phase, it identifies 'function word' and 'content words' in the corpus. It then groups content words into 'equivalence classes', each containing words belonging to a syntactic category and a semantic sub-category. Words in a class would be mutually interchangeable in a phrase context; each class is identified by an artificially chosen member called Class Exemplar (CE). Phrases in the corpus are identified in terms of CE's, on the basis of the above classification, and are in turn grouped into 'phrase types', such that each type consists of phrases which perform similar semantic functions in the sentence. To facilitate identification of the phrases, each word in the original corpus is replaced by the corresponding CE, reducing the number of distinct word tokens. The resulting corpus is called the reduced corpus. The phrase types are organised in a Phrase Table (PT). Each column of PT contains a distinct phrase type. Any sentence can be generated by concatenating a phrase each from (some of) the columns of this PT. A phrase-output Markovian State Transition Network (STN) is evolved to model the sentences of the language. During the test phase, the STN is used, (a) to parse correct sentences, and (b) to correct words which have been wrongly recognized in the sentences by the Acoustic Level Recognizer. A brief account of the language processor is given below. For details, readers may refer to Rao & Bondale (1992).

The BSLP accepts the word candidates and the associated probabilities from the acoustic level recognizer. Using these, it generates a 'most likely sentence hypothesis', consistent with the constraints imposed by syntax and semantics. This is built up, phrase by phrase, by matching the phrases from the PT columns with the alternatives provided by the acoustic processor, word by word. There is an exact match if alternatives provided in the word lattice include the correct words. Even otherwise, the correct words can be surmised, by using a lenient matching procedure which allows for a mismatch of up to two words (depending on the length of the phrase) between a phrase in the PT and the words in the word lattice. If none of the phrases in a PT column match the word lattice, a null output is assumed for the corresponding transition and match is attempted using phrases in the next column.

This would be straightforward, if at every stage of the matching process, only one phrase from a PT column matches the words in the word lattice. However, two or more phrases might fit in some cases, particularly if the phrases are short and a lenient matching procedure is being used. Whenever this happens, two or more alternative Partial Sentence Hypotheses (PSH) open up; this can lead to a combinatorial explosion. This is prevented by specifying an overall upper limit for the number of alternative PSH that are kept open for consideration at any given point of time.

Whenever this limit is exceeded, hypotheses are discarded from the low probability end. PHS is also discarded whenever it reaches a dead end, i.e. none of the subsequent phrases in the PT can be matched with the remaining words in the word lattice. During this processing each content word in the word lattice is replaced by the corresponding class exemplar. The original words replace the class exemplar when the processing is complete. This phrase matching procedure is carried out until all the columns of the PT and all the words in the word lattice are exhausted/accounted for. The sequence of phrases so matched is the most likely sentence hypothesis.

The elegance of this method lies in the process of classifying content words into equivalence classes and using class exemplars as the constituents of phrases. This gives phrase definitions a high degree of generality. On the other hand, this means that BSLP cannot distinguish between words belonging to the same equivalence class. Consequently, it is insensitive to errors of the acoustic processor which result in confusions between content words in the same equivalence class. It is easy to see that the human listener would also have the same problem in similar situations.

8.3 *Perplexity computation*

Let w_i be the ith word belonging to the word-vocabulary of size N_w. Let $q(\cdot)$ be the true statistics (here, estimated from the training corpus) of a model and $p(\cdot)$ the statistics corresponding to a corpus that is to be evaluated against the true statistics. For the training corpus $p(\cdot) = p_{tr}(\cdot) = q(\cdot)$ and for a testing corpus $p(\cdot) = p_{tst}(\cdot)$. If H is the per-word entropy (computed for logarithm to the base 2) of a corpus then *perplexity* $= 2^H$ (Lee 1988).

For a trigram model, the per-word entropy of a word trigram can be expressed as

$$H = \sum_{i=1}^{N_w} p(w_{i-2}, w_{i-1}, w_i) \log(q(w_i | w_{i-2}, w_{i-1})). \tag{1}$$

For a BSLP model, let the total number of word tokens and sentences that are present in a corpus be N_w and N_{sent} respectively. Let c_j be the jth category belonging to the category-vocabulary of size N_c, and S_{c_j} the number of elements in jth category. Let N_{col} denote the number of columns in a phrase table, N_f^k denote the number of phrases in column k, and f_l^k be the lth phrase in the kth column where $l = 1, 2, \ldots, N_f^k$. The last phrase entry of each column is the null phrase. For a given column, entry for the null phrase denotes the probability of that particular column being skipped while a sentence is parsed.

The phrase probability in a given column is given by

$$p(f_l^k) = n(f_l^k)/N_{sent},$$

where $n(f_l^k)$ is the frequency of occurrence of that phrase in the kth column. The cross entropy associated with the kth column can be expressed as

$$H_k = \sum_{l=1}^{N_f^k} p(f_l^k)\log(q(f_l^k)). \tag{2}$$

The entropy for the entire sentence using the BSLP model with a phrase as the unit is given by

$$H_{sent} = \sum_{k=1}^{N_{col}} H_k. \tag{3}$$

The per-word entropy (for the reduced corpus) will then be

$$H_w = N_{sent}/N_w. \tag{4}$$

For the conversion of the reduced corpus to the actual corpus, i.e., from CE to the actual word, let us assume that all the elements in a given category occur with equal probability ($1/S_{c_j}$: this is a logical assumption in the present task domain). Then, the per-word increase in entropy (for category to word conversion) is given by

$$\Delta H = \sum_{j=1}^{N_c} p(c_j)\log(S_{c_j}). \tag{5}$$

The entropy of the word trigram model is obtained from (1). The entropy and perplexity of the training sentences were 1.01 and 2.0 respectively and those of test sentences were 5.25 and 38.1. The total BSLP entropy at word level is obtained by adding (4) and (5). The total perplexity of training sentences was 15.0 and that of test sentences was 15.9 indicating that the size of the training corpus is inadequate. The perplexity results reflect the relative performance of the models. We discuss these results in the "Language modelling" part of the section on "Discussion".

9. Performance

The complete recognition system is trained on the first 200 sentences (containing 1829 words) and tested on 50 sentences (containing 463 words). The first two rows of table 5 show the percentages of the actual word being identified as the best match or included within the list of best four candidates. The third and fourth rows denote the word level

Table 5. Percentage word accuracy of VOICE speech recognition system.

	Training set Sent: 001-200 (1829 words)	Test set Sent: 201-250 (463 words)
First choice	78.2	73.7
Within best four choices	92.8	90.9
Bigram grammar	87.0	85.3
Language model	93.0	91.4

match between the language processor output sequence and the actual sentence. The bigram language model is constrained to select a word within the first four choices. Hence, the resulting matching score is limited by the actual word being listed within the candidate list. On the other hand, the phrase-structured language processor can supply a word if the available choices do not satisfy the semantic structure of the sentence. This results in an overall accuracy that is even better than that considering the best four choices.

10. Discussion

The current system uses domain knowledge to provide meaningful constraints on statistical models to speech recognition. These are, in turn, better equipped to handle uncertainties in the knowledge about mapping from the acoustic signal to the intended message. Rather than assuming that statistical and knowledge-based approaches are mutually exclusive, our approach recognises their mutual complementarity and integrates their strengths. For instance, acoustic-phonetic knowledge guides the initial choice of a variety of features; statistically motivated criteria determine the final selection of discriminative features for the classification. Similarly, knowledge of the intrinsic durational distributions of different classes of sounds motivates the use of durations (in addition to acoustic likelihoods), while comparing two segments. The optimal method of integrating them in the definition of a distance measure in the segment space is, on the other hand, evolved through an experimental study.

The modular architecture employed in our approach allows us to tap the outputs of the system at different stages of processing. This facilitates the performance evaluation of each module independently. The system can then act as a testbed for comparing various modelling approaches for each module. This provides for constant improvement of system performance by plugging in superior modules as and when they become available. Some of the areas where improved versions of modules may lead to better recognition accuracy at the sentence level are listed below.

10.1 *Feature vector*

There is considerable scope for improvement in the features used for frame level broad acoustic classification. Quefrency-weighted cepstral coefficients derived from linear prediction analysis were chosen as the base set of features for representing the instantaneous spectrum as they perform well in classification experiments. The Euclidean distance in this space is shown to be equivalent to the Itakura–Saito distance which is based on maximum likelihood analysis. However, some studies have highlighted the advantage of using mel-scaled cepstral coefficients (Davis & Mermelstein 1980; O'Shaughsnessy 1987) and performing dimensionality reduction after a principal component analysis (Krishnan 1994). A comparative study of feature representations showed that line spectral-pair frequency representation augmented with spectral amplitudes yields the highest recognition results in a multi-speaker, context independent single frame monophthong vowel recognition task (Krishnan 1994). Usage of such representations can be studied.

Cepstral differences are inadequate for the representation and classification of obstruent sounds such as plosives. Thus, there is a need for features which can capture the dynamics of speech in a more sophisticated form as noted by Yegnanarayana & Sundar (1991). One needs a larger annotated and time-aligned database to train classifiers on such sounds and to facilitate the fine acoustic classifiers of all speech sounds. The performance gap between training and test corpora indeed indicates that the training data is inadequate.

10.2 *Broad acoustic classes*

Although the grouping of phonemes into broad acoustic classes has a strong foundation based on knowledge of the speech production mechanism , certain observations point to a need for re-examining the composition of classes. For example, the class of weak unvoiced fricatives has only one member. There is also a considerable overlap between the class "middle" vowels, and "front" and "back" vowels. The hierarchical classification strategy followed here imposes rather strong constraints on the task decomposition as all realizations of a phonetic unit have to be associated with a single broad category. Consequently, the confusion between voicebar and nasals has been taken care by merging the two classes during fine acoustic classification. A sophisticated extension of this approach would be to allow for a fine acoustic unit to be a member of multiple classes and design fine acoustic classes accordingly.

10.3 *Segmental reliability measure*

Examination of misrecognized words at the output of the lexical access stage showed that a large number of templates contained spurious segments. The current version of the segmentation module has no provision for rejecting a portion of signal. Hence extraneous segments were detected in the transition regions where the activation levels of all the phonetic classes were relatively low. This is because the current matching process provides equal emphasis on each segment, irrespective of its reliability. Segmental features such as the probabilistic output of the classifiers (acoustic likelihood), the probability of transition into and out of a segment, and durational probability can be used to compute a reliability measure of the segment. Incorporation of such a reliability measure for each segment might improve overall accuracy even further. The framework developed for lexical access can easily be upgraded to include such a reliability factor (say, the average activation of the selected phonetic class in the segment) in the definition of inter-segment distance.

An obvious improvement to the existing system would be to perform fine acoustic classification for members of all broad acoustic classes. Modularization of the system allows for such improvements. Acoustic-phonetic knowledge can be used to collect a large set of features for subclassification within each class; these need not be restricted to a subset of the feature set used for broad acoustic classification. The class-specific feature set can be pruned using discriminant criteria such as Fisher's criterion.

10.4 *Lexical access*

Use of multiple reference templates for each word should improve the performance of the lexical module. Currently, the output of the lexical stage is a lattice of word hypotheses comprising a fixed number of choices per word. An alternative would be to adapt this number depending on the confidence level of the lexical processing module. For a given test word, the number of choices can be increased till the cumulative likelihoods of all the choices for that word exceeds a certain threshold. This criterion would limit the number of word alternatives when the top choice has a very high likelihood thereby reducing computational load of the language module. More importantly, it ensures that an adequate number of word candidates are available for the language processor to choose the most likely candidate based on syntactic constraints. This will help in situations where the preceding modules cannot confidently rank the word candidates based on acoustic evidence alone. Such flexibility in the integration of different knowledge sources would enable the system to perform gracefully even when the acoustic information is inadequate.

10.5 *Language modelling*

A better estimation of bigram probabilities can be obtained with an extensive sentence corpus. The statistical approach to syntactical modelling can be extended to N-grams as well. At present, the blank slate language processor supplies a category exemplar when it does not find a word in the candidate set which satisfies the constraints of the language model. This opens up the prospect of going back to the acoustic level recognizer to pick up a suitable lower level choice consistent with the demands of the language model.

The results on perplexity (a gap in performance between the training and test corpora for trigram models) indicate that a word-based model cannot generalize for the existing size of the training corpus. In contrast, for the BSLP model, the difference between the training and test corpus perplexity is very small. Also the BSLP perplexity is much lower than that of the trigram model. This indicates that the BSLP is a superior model compared to the statistical approach and also, that it can generalize optimally for the size of training corpus available.

The design strategies of the system do not impose significant constraints on the extension of the system to continuous speech or to a large vocabulary. Though the detection of word boundaries prior to recognition gave us a convenient unit for distributing computational load in a multi-processor environment, such a segmentation of the utterance is not really needed here because the broad acoustic classifier can easily demarcate the speech from silence regions. The modifications and improvements discussed above and a comprehensive speech database would be expected to enable an implementation of a large vocabulary, multi-speaker, continuous speech recognition system. The planned development (Samudravijaya *et al* 1994) of a general purpose database of segmented and labelled Hindi sentences with extensive coverage of phonemic contexts spoken by many speakers is the first step taken in this direction.

11. Summary

The philosophy and implementation of a speech recognition system for Hindi is described in this paper. The system follows a hierarchic approach to speech recognition and shows the power of integrating speech-specific knowledge with statistical pattern recognition techniques. The modular architecture of the system leaves ample scope for easy replacement of a module at any stage of processing by an advanced version. Also, the architecture has the capability to incorporate the characteristics of the top-down approach. The units of representation and the recognition scheme employed here do not pose any significant constraints for expansion so that the system can be modified for recognition of continuous speech by multiple speakers.

Work reported here has been supported by the Department of Electronics of the Government of India and by the United Nations Development Programme. We ácknowledge the help of T M Ajitha, Bhiksha Raj and A Sen in the development of the annotated and time-aligned speech database for Hindi without which the implementation of this system would not have been possible.

References

Bahl L R 1984 Some experiments with large-vocabulary isolated-word sentence recognition. In *Int. Conf. Acoust., Speech and Signal Processing*

Bellman R 1957 *Dynamic programming* (Princeton: University Press)

Brachman R 1978 A structural paradigm for representing knowledge. Technical Report BBN-3605, Bolt Beranek & Newman Systems, Cambridge, MA

Cole R A, Rudnicky A I, Zue V W, Reddy D R 1980 Speech as patterns on paper. In *Perception and production of fluent speech* (ed.) R A Cole (Hillsdale, NJ: Lawrence Erlbaum Associates)

Davis K 1994 Stop voicing in Hindi. *J. Phonet.* 22: 177–193

Davis S, Mermelstein P 1980 Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans- Acoust. Speech Signal Process.* 28: 357–366

Duda R, Hart P 1973 *Pattern classification and scene analysis* (New York: John Wiley)

Ferguson J D 1986 Variable duration models in speech. In *Proc. Symposium on Applications of Hidden Markov Models to Text and Speech* (ed.) J D Ferguson, pp 143–179

Furtado X A, Sen A 1996 Synthesis of unlimited speech in Indian languages using formant-based rules. *Sadhana* 21: 345–362

Glass J R 1988 *Finding acoustic regularities in speech: applications to phonetic recognition.* Ph D thesis, Department of Electrical Engineering and Computer Science, Massachussetts, Inst. Technol., Boston, MA

Kanal L N, Chandrasekaran B 1971 On dimensionality and sample size in statistical pattern classification. *Pattern Recogn.* 3: 225–234

Krishnan S 1994 *Speech recognition by computer: Spectral temporal redundancy and stochastic segmental models.* Ph D thesis, University of Bombay, Bombay

Lee K F 1988 Large vocabulary speaker-independent continuous speech recognition – The sphinx system. Technical Report CMU-CS-88-148. Computer Science Department, Carnegie-Mellon University, PA

Levinson S E 1985 Structural methods in automatic speech recognition. *Proc. IEEE* 73: 1625–1650

Markel J D, Gray A H 1976 *Linear prediction of speech* (New York: Springer-Verlag)

Makhoul J, Schwartz R 1985 Ignorance modeling. In *Invariance and variability in speech processes* (eds) J S Perkell, D H Klatt (Hillsdale, NJ: Lawrence Erlbaum Associates) pp 344-345

O'Shaughsnessy D 1987 *Speech communication – Human and machine* (Reading, MA: Addison-Wesley)

Poddar P, Rao P V S 1993 Performance of a class of distance metrics for lexical access. In *Workshop on Speech Technology*, Indian Institute of Technology, Madras

Rabiner L R, Sambur M R 1975 An algorithm for determining the end-points of isolated utterances. *Bell Syst. Tech. J.* 54: 297–315

Rao P V S 1993 VOICE: an integrated speech recognition synthesis system for Hindi language. *Speech Commun.* 13: 197–205

Rao P V S, Bondale N 1992 Blank-Slate language processor for speech recognition. In *International Conference on Spoken Language Processing*, Canada, pp 197–205

Samudravijaya K, Krishnan S, Sen A, Rao P 1991 Hindi speech database. In *Workshop on recent trends in speech, music and allied signal processing*, pp s13–s19

Samudravijaya K, Krishnan S, Rao P 1994 *A general purpose phonetically labelled speech database in Hindi* (New Delhi: BPB)

Viterbi A J 1967 Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inf. Theor.* 13:260–269

Yegnanarayana B, Sundar R, 1991 Signal processing issues in realizing voice input to computers. *Asia-Pacific Eng. J.* 1: 197–217