A LARGE VOCABULARY CONTINUOUS SPEECH RECOGNITION SYSTEM FOR HINDI

Chalapathy Neti, Nitendra Rajput, Ashish Verma*

IBM India Research Lab,
Block 1, IIT Campus, Hauz Khas, New Delhi 110016. Ph. 91-11-6861100
*IBM T.J. Watson Research Center, NewYork 10598. Ph. 1-914-9452921
{rnitendr,vashish}@in.ibm.com

ABSTRACT

In this paper we present a Hindi Speech Recognition system which has been trained on 40 hours of audio data and a has a trigram language model that is trained with 3 million words. For a vocabulary size of 65000 words, the system gives a word accuracy of 75% to 95%.

1. INTRODUCTION

In the field of Indian language speech recognition, various researchers have tried to analyze the different aspects of speech. While in [1], the authors aim at detecting the word boundaries for four Indian languages, Shrotriya et.al. in [2] show the characteristics of the consonant clusters for Hindi. In [3] the authors present a technique for identifying the speech into different syllable-like units of sound. Building a complete speech recognition system however utilizes the acoustics as well as the syntax of the language. HMM based statistical speech recognition systems have been popular in the past decade or so, and have been shown to give better results than the other techniques. In a statistical framework for speech recognition, the problem is to find the most likely word sequence $\hat{W}$ given the input speech A [4],

$$\hat{W} = \arg\max_{W} p(W / A) \qquad (1)$$

With a Bayesian approach to solving the above problem, we can write

$$\hat{W} = \arg\max_{W} p(A/W)\,p(W) \qquad (2)$$

Equation 2 above gives the two main components of a speech recognition system, the acoustic model and the language model. Building a speech recognition system involves building an acoustic model that would give the probability of a word sequence being represented by the input speech $p(A/W)$. The second term in equation (2) gives the probability of co-occurrence of such a word sequence and this is computed by a language model. Details of a HMM based statistical speech recognition system can be found in [5][6].

In this paper we present the work on building the acoustic and language models for Hindi language. Section 2 presents the phone set that has been used for building the acoustic models for Hindi. In Section 3 we present a technique for building initial phoneme models by bootstrapping from English phonemic models. The technique to build context-based decision trees in order to build context dependent phoneme (known as allophone) models is presented in Section 4. The language model is described in Section 5 and finally we conclude with the results on the speech recognition system and present the recognition accuracies in Section 6.

2. HINDI PHONE SET

A set of phonemes are required to represent the sounds of the acoustic space which can be formed either from a particular language or from the sounds of a combination of languages in the case of a multilingual speech recognition system. An increase in the number of phonemes in the phone set would result in a lower classification rate. On the other hand lesser number of phonemes in the set may result in a phonetic space that does not cover the whole acoustics of the language/languages.

The IPA [7] has defined phone sets for labeling speech databases for sounds of a large number of languages (including Hindi). But there are some sounds which are not included in the IPA (for example, DN, DXX, AWN) but which are important when building phone models that are to be used for the purpose of automatic speech recognition. In continuous speech recognition tasks, the purpose of defining a phonetic space is to form well-defined, least-overlapping clusters for each phoneme in the acoustic space so that it is easier for the system to recognize the phone to which an input utterance of speech belongs. For the same amount of data and phoneme models, a better phone set is one that gives a higher classification rate and is able to distinguish the words present in the vocabulary of the language.

We define a full-fledged Hindi phone set that covers all different sounds that occur in Hindi. This phone set takes into consideration the fact that even though Hindi is a character based language; from an acoustic point of view some phones such as plosives have different acoustic properties when they occur at the end of the word. Taking these into account we have constructed a Hindi phone set $\Gamma = \{\gamma_i , i=1,2,\ldots,61\}$ consisting 61 phones (including the inter-word silence X and begin/end of sentence silence D$) to represent the sounds in Hindi. It is seen that out of these 61 phones, 39 are common to the English phone set. Table 1 shows the corresponding characters as written in Hindi script.

Hindi Phone (γ)	**Hindi Alph**	**$\mathcal{N}$ (γ)**	**$\mathcal{M}(\gamma)$**	**Hindi Phone (γ)**	**Hindi Alph**	**$\mathcal{N}$ (γ)**	**$\mathcal{M}(\gamma)$**
AA	आ	AA	AA	IYN	ीं	IY	IY+N
AAN	आं	AA	AA+N	JH	ज	JH	JH
AE	,े	AE	AE	JHH	झ	JH	JH+HH
AEN	,ें	AE	AE+N	K	क	K	K
AW	ौ	AW	AW	KD	क्	KD	KD
AWN	ौं	AW	AW+N	KH	ख	KD	KD+HH
AX	अ	AX	AX	L	ल	L	L
AXN	अं	AX	AX+N	M	म	M	M
B	ब	B	B	N	न	N	N
BD	ब्	BD	BD	NG	ङ	NG	NG
BH	भ	BD	BD+HH	OW	ो	OW	OW
CH	च	CH	CH	OWN	ों	OW	OW+N
CHH	छ	CH	CH+HH	P	प	P	P
D	ड	D	D	PD	प्	PD	PD
DD	ड्	DD	DD	PH	फ	P	PD+HH
DDN	ण	DD	DD+R	R	र	R	R
DH	द	DH	DH	S	स	S	S
DHH	ध	DH	DH+HH	SH	श	SH	SH
DN	ण	DX	DX+N	T	ट	T	T
DXX	ढ़	DX	DX+HH	TH	थ	TH	TH
D$	SIL	D$	D$	THH	ठ	TH	TH+HH
EY	े	EY	EY	TX	त	TH	TH
EYN	ें	EY	EY+N	UH	ु	UH	UH
F	फ	F	F	UHN	उ	UH	UH+N
G	ग	G	G	UW	ऊ	UW	UW
GH	घ	GD	GD+HH	UWN	ूं	UW	UW+N
HH	ह	HH	HH	V	व	V	V
IH	ि	IH	IH	X	SIL	X	X
IY	ी	IY	IY	Y	य	Y	Y
				Z	ज़	Z	Z

Table 1: *Hindi phonemes for the characters in Hindi. Mappings are shown using the English phone set.*

As seen, in addition to 10 vowels, Hindi has 9 more vowels (AAN, AEN, AWN, AXN, IYN, OWN, EYN, UHN, UWN) that have some amount of nasal affect embedded in them. Also, each of the plosive phones B, D, K, P and T have an additional phone BD, DD, KD, PD and TD respectively, to represent the acoustic dissimilarity when they occur at the end of a word.

3. BUILDING INITIAL PHONEME MODELS FOR HINDI

The English recognition system in [7] is trained on a phone set Π consisting of 52 phones. Since the English recognition system has been built completely, we can populate the phonetic space of Π by using labeled data in English language. To populate the space represented by Γ, we either need a Hindi recognition system that can produce labeled data or record only the isolated phoneme data that can occur in the different contexts. In this section we present a mapping of English phonetic space to Hindi in order to generate the initial models for the Hindi phonetic space.

Each 60 dimensional cepstral vector (described in Section 6) generated from English speech is labeled with the corresponding phone using Viterbi alignment and the truth. In this 60 dimensional space we form models for Π by representing each phone $\pi \in \Pi$ by a set of mixture Gaussians. We introduce a mapping M that rearranges the English data into 61 clusters, each representing a phone from Γ. The best mapping is the one that produces a space for Γ such that the classification rate in this redefined space is the highest. If $\langle \Phi \rangle$ represents the phone model of Φ, then the mapping M is such that

$$\langle \Pi \rangle \xrightarrow{\mathbf{M}} \langle \Gamma \rangle$$

We initialize M from the acoustic knowledge of the phones in Γ and Π. Each element in Γ is represented by a single element or a combination of elements from Π.

$$\gamma_i = \cup\, \pi_n \,,\ i = 1, 2, \ldots.64,\ \gamma \in \Gamma,\ \pi_n \in \Pi.$$

If an element $\gamma \in \Gamma$ is acoustically similar to an element $\pi \in \Pi$, the model for γ is directly formed from the vectors that created the existing model of π in the English phonetic space. However if the best acoustic closeness for a phone in Γ is achieved by combining more than one sound from Π, the model for that γ is obtained from the vectors that form models for all those elements in Π. As is seen in Table 1, the phone GH is formed from the vectors of GD and HH that belong to Π.

We populate the Hindi phonetic space in the following manner.

1. Using M, find the phones in Π which can generate the acoustically closest sound for γ
2. Take vectors from the subset of phones in Π which form the sound γ. If any $\pi \in \Pi$ appears in more than one γ, randomly divide the vectors labeled by π into each $\gamma \in \Gamma$
3. Create a Gaussian mixture model from these vectors
4. Go to step 1 till all elements in Γ have a model

In Step 2, we can also duplicate rather than divide the data from the space of Π to Γ in cases where $\pi \in \Pi$ appears in more than one γ. But this increased the population in the space and was seen to create models with a lot of overlap thus resulting in less classification. Using this phonetic space mapping we can create the initial phone models for Hindi language using the English recognition system and English data.

4. CONSTRUCTION OF CONTEXT DEPENDENT DECISION TREES

In continuous speech, the pronunciation of a phone is heavily dependent on the context. It is important to model the context dependent variations in the pronunciation of phones. For this purpose, a phonetic decision tree is used to obtain the acoustic realization of the phoneme context [9][10].

A binary decision tree is used to assign classes to objects. The context, i.e. the identities of the K previous phones and K following phones in the phone sequence, denoted as P-K, …P-1, P+1, …, P+K , defines the decision rule based on which the tree is constructed. In our experiments we use five previous and five next contexts and so the value of K is five. The tree consists of nodes that contain the decision rules and leaves (final nodes) which are labeled with the classes. At each node a binary decision criteria (contextual question) assigns the object to the left or right subtree. When the object reaches a leaf, the class label of this leaf is used as the class for the object. The classes are phonologically meaningful groups of phonemes. Each class consists of a subset of the alphabet of phones.

The Hindi phone set we are using from [11] comprised of 64 phones. It comprises of entirely new classes of phonemes like the nasal vowels mentioned in the previous section. Then others like the stressed plosives DHH, DDN, THH etc. also don’t exist in English but are close to some phonemes in English. These are added to the already existing classes. For example DHH is added to the class DH. In building the phoneme models for this particular phoneme two English phonemes (DH and HH) were combined in [6]. The grouping in this case follows naturally. Deletion of phonemes from the English question set comes about in an obvious way. Phonemes that do not exist in Hindi are removed. For example the phones IY, IH, IX belonging to a single class in English get removed. Another example is the phone AO that does not appear in

Hindi. The class AE, EY, AO in English therefore gets modified to exclude AO. As described above, to modify the English question set three things were done. One we added entirely new questions and hence new classes, two we modified some of the existing classes to include/remove phonemes and three we deleted certain classes that are not present in Hindi. This revision resulted in a question set for resolving Hindi context dependency. The resulting question set for Hindi consists of 112 questions for each context.

5. TRIGRAM LANGUAGE MODEL

We built a trigram language model [12] to capture the statistics of Hindi language. We used Hindi text data from various sources to assign bigram and trigram probabilities to facilitate a probability distribution over all possible sentences. The training text data was about 3 million words. A language model vocabulary of 65000 words was used and probability was assigned to unknown words also. Since the amount of data was not huge, we had assigned probabilities to all the bigrams and trigram occurring even once. Deleted interpolation is used to smooth over the held out data.

6. DETAILS OF THE EXPERIMENT

In order to test the initial phone models for Hindi, we perform classification experiments in the Hindi phonetic space generated in the previous section. This section presents the details of a classification experiment that we performed after generating labeled data for Hindi [11].

6.1. Phone Set Modification

The English data used to generate the initial phone models for Hindi consisted of 3000 utterances of 30 different speakers. The English training sentences are chosen from a vocabulary consisting of 96,000 words. This constituted about 7 hours of continuous speech. Hindi data was collected for 7 speakers having a total of 350 utterances totaling around 30 minutes of continuous Hindi speech. We used a Hindi phonetic vocabulary containing 900 Hindi words. A 24-dimensional mel-cepstral coefficient feature vector is formed for each audio frame of duration 10 ms. Linear Discriminant Analysis is used to transform this vector to a 60-dimensional space. The procedure for processing the audio stream is presented in [7] in detail. In order to test the initial phone models for Hindi, we perform classification experiments in the Hindi phonetic space generated in the Section 3. With the help of classification results, we changed the Hindi phone set from 64 phones [11] to 61. The three phones that were dropped (DHD, GD and TXD) were those that are not distinguishable using their models.

6.2. Speech Recognition Engine

To perform the recognition task, we built the acoustic models using about 26000 utterances of 120 speakers. This amounted to 40 hours of speech data. We used viterbi alignment to generate the labeled vectors for building initial phone models. A forward-backward algorithm[4] was used to train the HMMs for each arc of a phone. For acoustic front-end processing, we use 13-dimensional cepstral vector, each representing a 25 msec duration of speech at every 10 msec. First and second-order derivatives are used to capture the dynamics of speech variations and hence a 39-dimensional vector is used to represent speech in the cepstral domain. We also build a parallel engine that has 9 frames (four previous and four forward frames) of cepstral vectors. This forms a 117-dimensional vector on which we apply LDA for dimensionality reduction to form a 39-dimensional vector.

Speaker	Signal Processing	Sentences	Recognition Accuracy
Training	39-dim	300	96.34%
Training	39-dim LDA	300	96.21%
Test	39-dim	200	90.62%
Test	39-dim LDA	200	89.7%

Table 2: *Word recognition accuracies for speakers in the training and test sets.*

As is seen in Table 2, the recognition system performs well over a test set that has not been used in training, emphasizing the speaker-independence of the system. When a speaker from the training set is used, recognition accuracy is seen to improve further. The test sentences are continuous speech sentences of length about eight to ten words that are from a vocabulary of 65000 words.

7. REFERENCES

1. G.V. Rao, J. Srichland, "Word boundary detection using pitch variations", Proc. *ICSLP*, Vol. 2, 813-816, 1996.
2. N. Shrotriya, R. Verma, S.K. Gupta, S.S. Agrawal, "Durational characteristics of Hindi consonant clusters," Proc*.* *ICSLP*, vol. 4, 2427-2430, 1996
3. C. Sekhar, B. Yegnanarayana, "Modular networks and constraint satisfaction model for recognition of stop consonant-vowel (SCV) utterances", *IEEE Neural Network Proceedings*, Vol. 2, 1206-1211, 1998.
4. F. Jelinek, *Statistical methods for Speech Recognition*, MIT Press, Cambridge, 1997
5. L.R. Bahl, P.V. deSouza, P.S. Gopalakrishnan, D Nahamoo, M.A. Picheny, "Robust methods for using context-dependent features and models in a continuous speech recognizer" Proc. ICASSP, I/533 -I/536, 1994.
6. L. R. Bahl, S.V. De Gennaro, P.S. Gopalakrishnan, R.L. Mercer, "A fast approximate acoustic match for large vocabulary speech recognition", *IEEE Transactions on Speech and Audio Processing*, Vol. 1 Issue. 1, 59 –67, Jan. 1993.
7. John Wells, Jill House, *The Sounds of the IPA*, Dept of Phonetics and Linguistics, University College London.
8. L. R. Bahl, S. Balakrishnan-Aiyer, J.R. Bellgarda, M. Franz, P.S. Gopalakrishnan, D. Nahamoo, M. Novak,M. Padmanabhan, M.A. Picheny, S. Roukos, "Performance of the IBM large vocabulary continuous speech recognition system on the ARPA Wall Street Journal task," Proc. *ICASSP*, 41-44, 1995.
9. J. J. Odell, *The use of context in large vocabulary speech recognition*, PhD Thesis, Cambridge University, 1995.
10. L. R. Bahl, P. V. deSouza, P. S. Gopalakrishnan, D. Nahamoo, M. A. Picheny, "Decision trees for phonological rules in continuous speech," Proc. *ICASSP*, Toronto, Canada, 185-188, 1991.
11. N. Mukherjee, N. Rajput, L. V. Subramaniam, A.Verma, "On deriving a Phoneme model for a new language," Proc. *ICSLP*, Beijing, China 2000.
12. L.R. Bahl, P.F. Brown, P.V. deSouza, R.L. Mercer, "A tree-based statistical language model for natural language speech recognition," *IEEE Transaction on Acoustic, Speech, Signal Processing*, Vol. 37, 1001-1008, July 1989.