ON DERIVING A PHONEME MODEL FOR A NEW LANGUAGE

Niloy Mukherjee, Nitendra Rajput, L. V. Subramaniam, Ashish Verma*

IBM India Research Lab, New Delhi, India

*Indian Instutute of Technology, Kharagpur, India

{rnitendr, lvsubram, vashish}@in.ibm.com

ABSTRACT

We present a method for building an initial phoneme model for training an HMM in a new language using an already trained recognition system in a base language. HMM based phoneme recognition systems are used to model the phonemes in most large vocabulary speech recognition tasks. Mappings between the phonetic spaces of the two languages are generated and are used to populate the phonetic space of the new language. The best possible alignment of the new language data is obtained and initial phone models are built on this labeled data. A classification experiment is performed in the new language to illustrate the goodness of initial phone models.

Experiments are carried out with Hindi as the new language using an English language recognition system to derive the initial phone models for Hindi language.

1. INTRODUCTION

Training of an HMM based speech recognition system involves several iterations over the phone models to finally arrive at a perfect model to represent the acoustic space. A phone based recognition system uses a set of phonemes to cover the whole acoustic space of a particular language. For a large vocabulary speech recognition task, context dependent phone models are used to represent the various contextual subtleties in each phone. When such a system is built from scratch, it is essential to initialize the phone models before carrying out iterations to further refine them. This requires labeled data in the language for which the speech recognition system is to be built.

Previous approaches [1][2] to generate initial phone models include bootstrapping for a multilingual phone set and the use of codebook lookup. The technique mentioned in [1] requires a system already trained in the languages that form a multilingual system. On the other hand [2] requires labeled and segmented data in the language for which the system is to be trained. [3] describes various methods to generate the Chinese phone models by mapping them from the English phone models. This requires the collection of specific utterances of isolated monosyllabic data that is difficult for a language like Hindi and may not be the best means for initializing the phone models that are to be used in large vocabulary continuous speech recognition tasks. Cross-lingual use of recognition systems is also seen in [4] where the aim is to get a crude alignment of words that do not belong to the language of the recognition system. Efforts in Hindi speech recognition include, [5], where a syllable-based identification of isolated utterances of speech is done. [6] presents a method for detecting word boundaries in different Indian languages including Hindi, using pitch variations.

In this paper we present initial steps for developing a phone based continuous speech recognition system for Hindi language using a recognition system and labeled data of English language. The IBM LVCSR system [7] is used as the base recognition model from which we derive the Hindi phone set. We exploit the acoustic similarity between the two languages, to generate the initial phone models for Hindi language, by deforming the English phonetic space such that the Hindi sounds are represented in an efficient manner. We identify a phone set for the Hindi language and use this to define the acoustic vocabulary for the Hindi words. In Section 2 we define the Hindi phone set used and the corresponding Hindi character and sound that is represented by each phoneme. Section 3 presents a method to redefine the English phonetic space in order to populate the Hindi phonetic space with English data. An acoustically close mapping from Hindi to the English phonetic space, for generating the alignment of Hindi data, using the English recognition system is presented in Section 4. Section 5 describes the data used for the experiments and the details of the classification experiment. In this section we also describe a distance measure technique to refine the mapping of phonetic spaces. We conclude by presenting and describing the results 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 [8] 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 but which are important when building phone models that are to be used for the purpose of automatic speech recognition. We define a full-fledged Hindi phone set which can cover all the 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 $\Gamma = \{\gamma_i\ ,\ i=1,2,\ldots,64\}$ consisting 64 phones (including the inter-word silence D$ and long pause silence X) to represent the sounds in Hindi. It is seen that out of these 64 phones 39 are already present in English. Table 1 shows the corresponding characters as written in Hindi script.

Hindi Phone (γ)	**Hindi Alph**	$\mathcal{N}(\gamma)$	$\mathcal{M}(\gamma)$	**Hindi Phone (γ)**	**Hindi Alph**	$\mathcal{N}(\gamma)$	$\mathcal{M}(\gamma)$
AA	आ	AA	AA	IYN	ईं	IY	IY+N
AAN	ऑ	AA	AA+N	JH	ज	JH	JH
AE	ऐ	AE	AE	JHH	झ	JH	JH+HH
AEN	ऐं	AE	AE+N	K	क	K	K
AW	औ	AW	AW	KD	क	KD	KD
AWN	औं	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	ब	BD	BD	NG	ड.	NG	NG
BH	भ	BD	BD+HH	OW	ओ	OW	OW
CH	च	CH	CH	OWN	ओं	OW	OW+N
CHH	छ	CH	CH+HH	P	प	P	P
D	ड	D	D	PD	प	PD	PD
DD	ड	DD	DD	PH	फ	P	PD+HH
DDN	ड़	DD	DD+R	R	र	R	R
DH	द	DH	DH	S	स	S	S
DHD	द	DH	DH+HH	SH	क्ष	SH	SH
DHH	ध	DH	DH+HH	T	ट	T	T
DN	ण	DX	DX+N	TD	ट	TD	TD
DXH	ढ़	DX	DX+HH +R	THH	ठ	TH	TH+HH
DXX	ढ	DX	DX+HH	TX	त	TH	TH
D$	-	D$	D$	TXD	त	TH	TH
EY	ए	EY	EY	TH	थ	TH	TH
EYN	ऍं	EY	EY+N	UH	उ	UH	UH
F	फ़	F	F	UHN	उं	UH	UH+N
G	ग	G	G	UW	ऊ	UW	UW
GD	ग	GD	GD	UWN	ऊँ	UW	UW+N
GH	ध	GD	GD+HH	V	व	V	V
HH	ह	HH	HH	X	-	X	X
IH	इ	IH	IH	Y	य	Y	Y
IY	ई	IY	IY	Z	ज़	Z	Z

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

As is seen, in addition to 10 vowels, Hindi has 9 vowels that have an amount of nasal effect embedded in them. Also, each of the plosive phones has an additional phone to represent the acoustic dissimilarity when they occur at the end of a word. In continuous speech recognition tasks, the purpose of defining a phonetic space is to form well-defined, non-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.

3. CROSS LANGUAGE PHONETIC SPACE MAPPING

The English recognition system in [7] is trained on a phone set Π consisting of 52 phones. Since the English recognition system is completely built, 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 to generate the initial models for the Hindi phonetic space.

Each 60 dimensional cepstral vector (described in Section 5) 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 $\mathcal{M}$ that rearranges the English data into 64 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 $\mathcal{M}$ is such that

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

We initialize $\mathcal{M}$ 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 $\mathcal{M}$, 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. NOVEL LANGUAGE DATA LABELLING

We populate the Hindi phonetic space with the continuous speech sentences of Hindi language using the English speech recognition system. Hindi words are represented by phones from the Hindi phone set Γ to form a phonetic vocabulary of Hindi words. Each word in this vocabulary is known as a lexeme. Since the English recognition system is trained on Π, we use another mapping $\mathcal{N}$ from Γ to Π which is different from $\mathcal{M}$ in the sense that each element in Γ has one and only one corresponding element in Π. Also, $\mathcal{M}$ works on the phonetic space while $\mathcal{N}$ works on each $\gamma\in\Gamma$ such that

$$\gamma \xrightarrow{\mathcal{N}} \pi, \text{ where } \pi\in\Pi.$$

Since more than one element in Γ may map to a single element in Π, $\mathcal{N}$ is a many-to-one mapping and cannot be used in reverse to obtain γ for all the $\pi\in\Pi$. This mapping is used so that the Hindi vocabulary can be represented by an English phone set that can then be recognized by an English recognition system.

As shown in Figure 1, the Hindi phonetic vocabulary is first built using the Hindi phone set Γ. The mapping $\mathcal{N}$ then generates the vocabulary of Hindi lexemes using the English phone set Π.

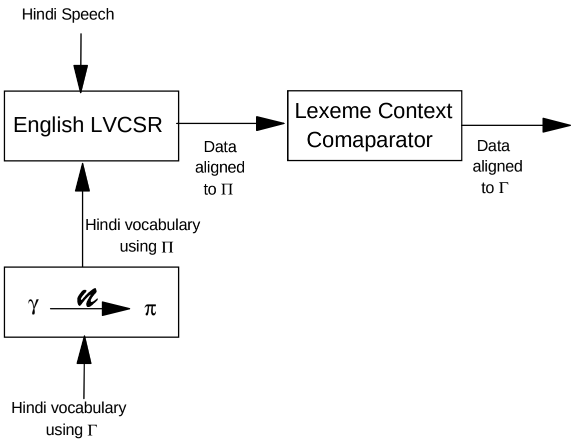

Figure 1: Generating labeled data for Hindi speech from the English recognition system.

This along with the Hindi speech data is fed as input to the English speech recognition system that then aligns the speech to the lexemes in Π. Thus each feature vector has now been labeled by a phone $\pi\in\Pi$ that it represents in the phonetic space of Π. This alignment can be converted to represent the data labeled in accordance with the phone set Γ. For this, the Hindi data aligned to lexemes written in the English phone set Π is fed as an input to the Lexeme Context Comparator. Since the mapping $\mathcal{N}$ cannot be used in its inverse form, the comparator works in the following steps:

1. For a feature vector labeled by a phone $\pi\in\Pi$, form a subset $\Psi\subset\Gamma$ using the inverse mapping $\mathcal{N}$.

2. If Ψ is a singleton, change the label of the feature vector to the element $\gamma\in\Psi$.

3. If not, from the lexeme context of the feature vector, compare the phonetic spelling of the two lexemes (one written with phones in Π and other with phones in Γ) to which this vector belongs. From this remove the disability and choose that phone from Ψ which satisfies the mapping $\mathcal{N}$ for the two lexemes.

This creates the alignment of the continuous speech Hindi data to the Hindi phone set without the help of any Hindi speech recognizer. Although this alignment is not perfect and the phone boundaries found by the above mapping may not be very accurate, they serve as good initial labeled data for a new language. The inaccurate phone boundaries are a result of phonetic space differences in the two languages owing to the different acoustic characteristics of the languages. This depends on the two languages; if the languages are acoustically very similar, we can have very accurate phone boundaries using the above technique.

5. 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 the classification experiment and a method for generating the labeled data for Hindi. A distance measure technique is used to verify the mapping $\mathcal{M}$ and a method for improving the mapping is presented.

5.1. Speech Database

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. The Hindi phonetic vocabulary contains 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.

5.2. Classification and Distance Measure

The Hindi phonetic space is populated from English data using the method described in Section 3. This data forms a training set for the classification experiment. In this phonetic space, we model each phone in Γ by a Gaussian mixture of 5 components. Test data for Hindi is generated using the method described in the previous section. For each phone in Γ we have a set of Hindi vectors representing the acoustics of that particular phone as a point in the 60-dimensional space. Classification is performed by measuring the log probabilities of each vector with the models of Γ. A vector representing the phone $\gamma \in \Gamma$ is said to be misclassified if the most likely model for it is not $\langle\gamma\rangle$. A confusion matrix of size 64×64 is created whose (i, j) entry is the number of vectors representing the phone i being classified to the model of phone j. A diagonal sum represented as a percentage of the total number of vectors gives the phonetic classification rate of the initial phone models generated from the English data.

In order to find a measure of phonetic similarity between the phones in Γ and the phones in Π, we define a similarity measure that finds the distances between the two phones. One method is to find the distance between the two phones in the LDA space. For this, we model each phone in the phonetic space of Π by a univariate normal distribution and the phonetic distance of a phone $\gamma \in \Gamma$ from the phone $\pi \in \Pi$ is defined as

$$D(\gamma,\pi) = \left| \sum_{i\in\gamma} (v_i - m_\pi)^2 \right|$$

where, v_i represents a 60-dimensional vector in the LDA space, m_π is the mean of the vectors in π. Another distance measure is that based on log likelihood over each of the 52 models of phones in Π for each test vector in $\gamma \in \Gamma$. The mean square of the sum of likelihoods is taken as the measure of acoustic similarity between the phones in the two languages.

This measure is calculated for each phone $\gamma \in \Gamma$ and the phones in Π that are close to γ are used to create a model for γ from the English phonetic space. This may change the mapping $\mathcal{M}$. The latter distance measure described above is seen to give better results. The described method of generating the phone model and performing the classification experiment are used to get an improved classification rate as reported in the next section.

6. RESULTS AND DISCUSSIONS

Table 2 shows the results for classification of the Hindi data over the Hindi phonetic space generated from the English data. Normally the phonetic classification rate is seen to be around 40-50% for most of the languages [2] with a trained system. The rate of 27% obtained for Hindi language without using context dependent models is a promising reason for using the phone models generated by the method described.

The distance measure technique provides an insight into the measure of closeness between the phone sets of the two languages. This is used to modify the mapping in order to create a better phonetic representation of the Hindi phones in the English data space. This modified mapping provides a 13% relative improvement in the rate of classification. Also the use the lexeme context to generate the Hindi data is a very fast way of generating the labeled data for a new language. Its advantage is reflected by an improved classification rate of 23.82% over not using lexeme context information and randomly distributing the phones that had a one-to-many mapping in $\mathcal{N}$.

Hindi Phonetic Space Method	Hindi Data Labelling Method	Classification Rate
Context based	Random	16.23%
Random	Random	21.29%
Random	Lexeme context	23.82%
Modified with Dist	Modified with Dist	26.99%

Table 2: Phonetic Classification rates for Hindi data using the Hindi phone models created by the English data.

We see that phone models for a new language can be created using the cross language phonetic space mapping. These can be used as good initial models to perform further iterations for training the phone based HMM for speech recognition. A distance measure feedback to the mapping can help in optimizing the cross language phonetic space mapping. We also showed that context dependent data could be labeled automatically with such a system. The method presented in this paper can be directly used to build a complete speech recognition system for large vocabulary continuous speech in a new language.

7. REFERENCES

1. Kohler, J., “Multi-lingual phoneme recognition exploiting acoustic-phonetic similarities of sounds”, *ICSLP*, 2195-2198, 1996.

2. Anderson, O., Dalsgaard, P., Barry, W., “On the use of data-driven clustering technique for identification of poly- and mono-phonemes for four European languages,” *ICASSP* , 1/121-1/124, 1994.

3. MA Chi Yuen and Pascale Fung "Adapting English Phoneme Models for Chinese Speech Recognition," *ISCSLP*, 80-82, Dec 1998.

4. T. A. Faruquie, C. Neti, N. Rajput, L. V. Subramaniam, A. Verma, "Translingual Visual Speech Synthesis," *IEEE International Conference on Multimedia and Expo (ICME 2000)*, New York, USA, July 30-August 2, 2000.

5. Sekhar, C., Yegnanarayana, B. “Modular networks and constraint satisfaction model for recognition of stop consonant-vowel (SCV) utterances”, *IEEE Neural Network Proceedings*, pp. 1206-1211 vol.2, 1998.

6. Rao, G.V., Srichland, J., “Word boundary detection using pitch variations”, *ICSLP* 1996.

7. L. R. Bahl et. al., “Performance of the IBM large vocabulary continuous speech recognition system on the ARPA Wall Street Journal task,” *Proc. ICASSP*, 41-44, 1995.

8. John Wells, Jill House, *The Sounds of the IPA*, Dept of Phonetics and Linguistics, University College London.