Development of Indian Language Speech Databases for Large Vocabulary Speech Recognition Systems

Gopalakrishna Anumanchipalli, Rahul Chitturi, Sachin Joshi, Rohit Kumar, Satinder Pal Singh, R.N.V. Sitaram*, S P Kishore*

International Institute of Information Technology, Hyderabad, India
{gopalakrishna, rahul_ch, sachin_sj}@students.iiit.net, {rohit, kishore}@iiit.net

* Hewlett Packard Labs India, Bangalore, India
{satinder, sitaram}@hp.com

Abstract

In this paper, we discuss our efforts in the development of Indian language speech databases in Tamil, Telugu and Marathi for building large vocabulary speech recognition systems. We have collected speech data from about 560 speakers in these three languages. We discuss the design and methodology of collection of speech databases. We also present preliminary speech recognition results using the acoustic models created on these databases using Sphinx 2 speech tool kit.

1. Introduction

Most of the Information in digital world is accessible to a few who can read or understand a particular language. Language technologies can provide solutions in the form of natural interfaces so that digital content can reach to the masses and facilitate the exchange of information across different people speaking different languages. These technologies play a crucial role in multi-lingual societies such as India which has about 1652 dialects/native languages. While Hindi written in Devanagari script, is the official language, the other 17 languages recognized by the constitution of India are: 1) Assamese 2) Tamil 3) Malayalam 4) Gujarati 5) Telugu 6) Oriya 7) Urdu 8) Bengali 9) Sanskrit 10) Kashmiri 11) Sindhi 12) Punjabi 13) Konkani 14) Marathi 15) Manipuri 16) Kannada and 17) Nepali. Seamless integration of speech recognition, machine translation and speech synthesis systems could facilitate the exchange of information between two people speaking two different languages. Our overall goal is to develop speech recognition and speech synthesis systems for most of these languages. Applications of speech synthesis and recognition systems include speech interface for Universal Digital Library [11], PCtvt [12] and Reading Aid for Visually Impaired (RAVI).

In the context of developing large vocabulary continuous speech recognition systems, there has been effort by IBM to build a large vocabulary continuous speech recognition system in Hindi [8] by bootstrapping existing English acoustic models. A more recent development was a Hindi recognition system from HP Labs, India [7] which involved Hindi speech corpus collection and subsequent system building. In this paper, we discuss the development of speech databases in Telugu, Tamil and Marathi languages for building large vocabulary speech recognition systems. We also discuss some preliminary speech recognition results obtained on these databases. We hope that this work will serve as a benchmark and impetus for the development of large speech corpora and large vocabulary continuous speech recognition systems in other Indian languages.

This paper is organized as follows: Section 2 describes our methodology of collecting the text corpora to build speech databases. Section 2.3 describes the optimal text selection algorithm adopted to generate the text corpus. Section 3 discusses the speech data collection effort and the statistics of the collected data. Section 4 deals with the building of acoustic models and the Section 5 describes the recognition results obtained on these acoustic models. The subsections discuss the gradual improvement in the performance with the application of various tuning techniques on the models. The section 6 describes a simple speech-to-speech application built by exploiting a constrained language model. The concluding remarks and future work are stated in the section 7.

2. Text Corpora for Speech Databases

The first step we followed in creating a speech database for building an Automatic Speech Recognizer (ASR) is the generation of an optimal set of textual sentences to be recorded from the native speakers of the language [7]. The selected sentences should be minimal in number to save on manual recording effort and at the same time have enough occurrences of each type of sounds to capture all types of co-articulation effects in the chosen language. In this section, the various stages involved in the generation of the optimal text are described.

2.1. Text Corpus Collection

To select a set of phonetically rich sentences, one of the important decisions to be made is the choice of a huge text corpus source from which the optimal sub-set has to be extracted. The reliability and coverage of the optimal text and of the language model largely depends on the quality of the text corpus chosen. The corpus should be unbiased and large enough to convey the entire syntactic behavior of the language. We have collected the text corpora for the three Indian languages we are working on, from commonly available error free linguistic text corpora of the languages wherever available in sufficient size. We have mostly used the CIIL Corpus [6] for all the languages. Incase of Tamil and

Telugu where the available linguistic corpus contained a lot of errors, we have manually corrected some part of the corpora and also collected representative corpus of the language by crawling content from websites of newspapers of the respective languages. The table below shows the size of the corpora and some other statistics.

Table 1: Analysis of the text Corpora

Language	No. of Sentences	No. of Unique Words	No. of Words
Marathi	155541	184293	1557667
Tamil	303537	202212	3178851
Telugu	444292	419685	5521970

2.1.1. Font Converters

To ease the analysis and work with the text corpus, it is required that the text is in a representation that can be easily processed. Normally the text of a language from any web-sources will be available in a specific font. So, font converters have to be made to convert the corpus into the desired electronic character code. The font will be reflective of the language and will have as many unique codes for each of the characters in the language. It is necessary to ensure that there are no mapping and conversion errors.

2.2. Grapheme to phoneme Converters (G2P)

The Text corpus has to be phonetized so that the distribution of the basic recognition units, the phones, diphones, syllables, etc in the text corpus can be analyzed. Phonetizers or the G2P converters are the tools that convert the text corpus into its phonetic equivalent. Phonetizers are also required for generating the phonetic lexicon, an important component of ASR systems [7]. The lexicon is a representation of each entry of the vocabulary word set of the system in its phonetic form. The lexicon dictates to the decoder, the phonetic composition, or the pronunciation of an entry. The process of corpora phonetization or the development of phonetic lexicons for the western languages is traditionally done by linguists. These lexicons are subject to constant refinement and modification. But the phonetic nature of Indian scripts reduces the effort to building mere mapping tables and rules for the lexical representation. These rules and the mapping tables together comprise the phonetizers or the Grapheme to Phoneme converters. The Telugu and the Marathi G2P were derived from the HP Labs Hindi G2P [5]. Special ligatures have been used to denote nasalization, homo-organic nasals and dependent vowels. Two rules were added to address the issues specific to Marathi. A rule was added to nasalize the consonant that follows an *anuswara*. Another rule was added to substitute the Hindi *chandrabindu* with the consonant ‘*n’*. The Tamil G2P is a one to one mapping table. Certain exceptions arise in the case of the vowels ‘*o’, ‘oo’* and ‘*au’* vowels where the vowel comes before the consonant in the orthography but the pronunciation has the vowel sound following the consonant. Such combinations have been identified and have been split accordingly so as to ensure the one to one correspondence between the pronunciation and the phonetic lexicon.

2.3. Optimal Text Selection

As mentioned in the earlier sections, the optimal set of sentences is extracted from the phonetized corpus using an Optimal Text Selection (OTS) algorithm which selects the sentences which are phonetically rich. These sentences are distributed among the intended number of speakers such that good speaker variability is achieved in the database. Also it is attempted to distribute the optimal text among speakers such that each speaker gets to speak as many phonetic contexts as possible, The best-known OTS algorithm is the greedy algorithm as applied to set covering problem [4]. In the greedy approach, a sentence is selected from the corpus if it best satisfies a desired criterion. In general, the criterion for the optimal text selection would be the maximum number of diphones or a highest score computed by a scoring function based on the type of the diphones that the sentence is composed of. A threshold based approach has been implemented for OTS of the Indian language ASRs. This algorithm performs as well as the classical greedy algorithm in a significantly lesser time. A threshold is maintained on the number of new diphones to be covered in a new sentence. A set of sentences is thus selected in each iteration. The optimal text, hence selected, has a full coverage of the diphones present in the corpus. These sentences are distributed among the speakers. The table 2 gives the diphone coverage statistics of the optimal texts of each language.

Table 2: Diphone coverage in the three languages

Language	Phones	Diphones covered
Marathi	74	1779
Tamil	55	1821
Telugu	66	2781

3. Speech Data Collection

In this section, the various steps involved in building the speech corpus are detailed. Firstly, the recording media is chosen so as to capture the effects due to channel and microphone variations. For the databases that were built for the Indian language ASRs, the speech data was recorded over calculated number of landline and cellular phones using a multi-channel computer telephony interface card.

3.1. Speaker Selection

Speech data is collected from the native speakers of the language who were comfortable in speaking and reading the language. The speakers were chosen such that all the diversities attributing to the gender, age and dialect are sufficiently captured. The recording is clean and has minimal background disturbance. Any mistakes made while recording have been undone by re-recording or by making the corresponding changes in the transcription set. The various statistics of the data collected for the Indian Language ASRs are given in the next subsection.

3.2. Data Statistics

Speakers from various parts of the respective states (regions) were carefully recorded in order to cover all possible dialectic variations of the language. Each speaker has recorded 52 sentences of the optimal text. Table 3 gives the number of speakers recorded in each language and in each of the recording modes – landline and cellphones. To capture different microphonic variations, four different cellphones were used while recording the speakers. Table 4 gives the age wise distribution of the speakers in the three languages. Table 5 shows the gender wise distribution of the speakers in each of the three languages.

Table 3: Number of speakers in the three languages

Language	Landline	Cellphone	Total
Marathi	92	84	176
Tamil	86	114	200
Telugu	108	75	183

Table 4: Age wise speaker distribution

Language	18-30	30-40	40-50	50-60	> 60
Marathi	77	56	26	12	5
Tamil	80	28	31	16	45
Telugu	52	52	32	32	15

Table 5: Gender distribution among speakers

Language	Male	Female
Marathi	91	85
Tamil	118	82
Telugu	93	90

3.3. Transcription Correction

Despite the care taken to record the speech with minimal background noise and mistakes in pronunciation, some errors have crept in while recording. These errors had to be identified manually by listening to the speech. If felt unsuitable, some of the utterances have been discarded.

In the case of the data collected in the three Indian languages, the transcriptions were manually edited and ranked based on the goodness of the speech recorded. The utterances were classified as "Good", "With Channel distortion", "With Background Noise" and "Useless" whichever is appropriate. The pronunciation mistakes were carefully identified and if possible the corresponding changes were made in the transcriptions so that the utterance and the transcription correspond to each other. The idea behind the classification was to make the utmost utilization of the data and to serve as a corpus for further related research work.

4. Building ASR systems

Typically, ASR systems comprise three major constituents- the acoustic models, the language models and the phonetic lexicon. In the subsections that follow, each of these components is discussed.

4.1. Acoustic Model

The first key constituent of the ASR systems is the acoustic model. Acoustic models capture the characteristics of the basic recognition units [3]. The recognition units can be at the word level, syllable level and or at the phoneme level. Many constraints and inadequacies come into picture with the selection of each of these units. For large vocabulary continuous speech recognition systems (LVCSR), phoneme is the preferred unit. Neural networks (NN) and Hidden Markov models are the most commonly used for acoustic modeling of ASR systems. We have chosen the Semi-Continuous Hidden Markov models (SCHMMs) [1] to represent context-dependent phones (triphones). At the time of recognition, various words are hypothesized against the speech signal. To compute the likelihood of a word, the lexicon is referred and the word is broken into its constituent phones. The phone likelihood is computed from the HMMs. The combined likelihood of all the phones represents the likelihood of the word in the acoustic model. The triphone acoustic models built for the Indian language ASRs are 5 state SCHMMs with states clustered using decision trees. The models were built using CMU Sphinx II trainer.

Two different models were trained from Landline and Cellphone data. The training data was carefully selected to have an even distribution across all the possible variations. For each of the systems, the speech data collected was divided into 'training' and the 'test' sets based upon the demographic details. 70% of the speakers were included in the training set and the rest 30% was used to test the respective systems.

4.2. Language Model

The language model attempts to convey the behavior of the language. By the means of probabilistic information, the language model provides the syntax of the language. It aims to predict the occurrence of specific word sequences possible in the language. From the perspective of the recognition engine, the language model helps narrow down the search space for a valid combination of words [3]. Language models help guide and constrain the search among alternative word hypotheses during decoding. Most ASR systems use the stochastic language models (SLM). Speech recognizers seek the word sequence W_b which is most likely to be produced from the acoustic evidence A.

$$P(W_b \mid A) = \max_w P(W|A) = \max_w P(A|W)\, P(W)/P(A)$$

P(A), the probability of acoustic evidence is normally omitted as it is common for any word sequence. The recognized word sequence is the word combination W_b which maximizes the above equation. LMs assign a probability estimate P(W) to word sequences $W=\{w_1,w_2,\ldots\ldots w_n\}$. These probabilities can be trained from a corpus. Since there is only a finite coverage of word sequences, some smoothing has to be done to discount the frequencies to those of the lower order. SLMs use the N-gram LM where it is assumed that the probability of occurrence of a word is dependent only on the past N-1 words. The trigram model [N=3], based on the previous two words is particularly powerful, as most words have a strong dependence on the previous two words and it can be estimated reasonably well with an attainable corpus. The trigram based language model with back-off was used for recognition. The LM was created using the CMU statistical LM toolkit [2]. For the baseline systems developed, the LM was generated from the training and the testing transcripts. The unigrams of the language model were taken as the lexical entries for the systems.

5. Baseline System Performance

This section presents the performance of the recognition systems. Two systems were built in each language using the landline and the mobile speech data separately. The tests on these recognition systems were done using the speech data of the untrained 30% of the speakers' data. "Good" utterances of these speakers were decoded using the Sphinx II decoder. Appropriate tuning was done on the decoder to get the best performance. The evaluation of the experiment was made according to the recognition accuracy and computed using the word error rate [WER] metric which aligns a recognized word string against the correct word string and computes the number of substitutions (S), deletions (D) and Insertions (I) and the number of words in the correct sentence (N).

$$W.E.R = 100 * (S+D+I) / N$$

In the following subsections, the gradual improvement in the performance of the ASRs is shown. The subsection 5.1 discusses the improvements obtained on iterations of Forced alignment. The subsection 5.2 deals with the use of an extended phoneme set.

5.1. Forced Alignments

This is a well known technique to improve the HMM of the Acoustic Model. The technique sounds quite simple but the performance of the ASR dramatically improves by nearly 10% at times. The method involves insertion of silence at appropriate places in the training transcription and retraining the models. Using existing acoustic models, the silence is detected by aligning the speech signal against the trained acoustic models and the corresponding transcription. The new transcriptions have silence markers at appropriate places, the intermediate tied states and hence the overall models are improved. Multiple Forced alignment iterations could be done to further improve the model. The performance eventually converges after a few iterations. The table 6 shows the improvement in the performance of the recognition systems built from the landline data of each language. The vocabulary sizes of the various systems built is shown in the table 7.

Table 6: Improvements in the WER of the landline recognition systems through a couple of Forced Alignment iterations.

Languag e	Initial Performance	1st iteration	2nd iteration
Marathi	27.3	24	23.2
Tamil	27.9	23.6	20.2
Telugu	30.3	28.6	28

Table 7 Vocabulary size of each ASR (Number of lexical entries)

ASR System	Vocabulary
Marathi	21640
Tamil	13883
Telugu	25626

5.2. Significance of Extended Phone Set

As mentioned earlier, lexicon defines the pronunciation of each lexical entry. The performance of the ASR system depends on the phone set chosen. New sub word units can be identified and be introduced into the phone set. The choice, however, should be such that, the variability of the phones and the complexity of the HMM is reduced. It is also to be ensured that there are sufficient instances of all the phones. The phonetic split in the lexicon will change accordingly as does the phone set.

The Initial experiments were done using the standard ISCII phone sets of the three languages [9]. To improve the phone set, we have added the geminates or the stressed consonants as standard phones. Stressed consonants, though represented as repetition of a consonant twice in the orthography, do not follow the same pattern in pronunciation [5]. The consonant is instead stressed, attributing some amount of plosiveness to it. The spectrogram of a consonant uttered twice would display two similar observations coarticulated due to one another. The spectrogram of a stressed consonant, on the other hand, would vividly display a blob of energy due to the stress. This clearly indicates that stressed consonants need to be treated separately and not as variants of the parent consonant for acoustic modeling. The table 8 shows the improved performances of all systems employing an extended phoneset. It also shows the size of the lexicon of the systems.

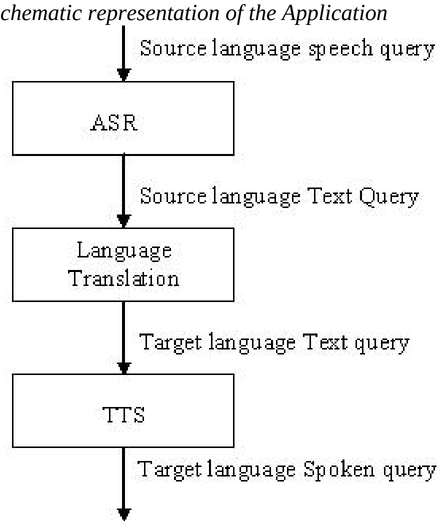
Table 8: Improvements in %WER by including the stressed consonants.

ASR System	Initial	Improved	Vocabulary
Marathi Landline	23.2	20.7	21640
Marathi Mobile	25.0	23.6	18912
Tamil Landline	23.9	19.4	13883
Tamil Mobile	22.2	17.6	16187
Telugu Landline	18.5	15.1	25626
Telugu Mobile	20.1	18.3	16419

6. Speech –to-Speech Application

The speech-speech demo was an attempt to develop an interesting application using the above systems and integrating it with existing systems. It is a tourist aid application with a 145 word vocabulary and 1000 legitimate sentence templates within the tourist domain. The application converts a spoken source language query into the target language utterance. A template based language translation module has been simulated to convert the recognized text query into the target language. The Telugu and the Tamil Text-to-Speech (TTS) [10] engines were integrated to synthesize and speak out the translated query in the target language.

Figure1: Schematic representation of the Application


Source language speech query
ASR
Source language Text Query
Language Translation
Target language Text query
TTS
Target language Spoken query

As illustrated in the schematic representation, the application has three components, the recognition module (ASR), the translation module and the synthesis module (TTS). The Speech query of the user is taken by the ASR and is decoded into a text query of the source language. This goes as an input to the simulated language translation system. This identifies the template of the source language text query and converts it into the corresponding template of the target language. The TTS engine converts the output into spoken speech query of the target language. With the limitation of the available ASR and TTS engines, we could render the translation within the following source-target language pairs.

(i) Telugu to Hindi
(ii) Tamil to Hindi
(iii) Tamil to Telugu

The finite size of the legitimate utterances has been exploited, giving a significant bias to the language model. Given minimal background disturbances, the performance of the application was close to 100%.

7. Conclusion and Future work

In this paper, we discussed the optimal design and development of speech databases for three Indian languages. We hope the simple methodology of database creation presented will serve as catalyst for the creation of speech databases in all other Indian languages. We also created speech recognizers and presented preliminary results on these databases. We hope the ASRs created will serve as baseline systems for further research on improving the accuracies in each of the languages. Our future work is focused in tuning these models and test them using language models built using a larger corpus.

8. Acknowledgements

We would like to acknowledge the help and support of Naval, Gaurav, Pavan, Natraj, Santhosh and Jitendra for speech data collection. Our special thanks to Dr. Ravi Mosur, Dr. James K Baker and Dr. Raj Reddy of Carnegie Mellon University for their support and suggestions.

9. References

[1] L.Rabiner., "*A Tutorial on Hidden Markov models and Selected Applications in Speech Recognition",* Proc. Of IEEE, Vol. 77 No. 2, 1989.

[2] Rosenfeld Roni, "*CMU statistical Language Modelling (SLM) Toolkit (Version 2)".*

[3] X. Huang, A. Acero, H. Hon, "*Spoken Language Processing: A Guide to Theory, System and Algorithm Development*", New Jersey, Prentice Hall, 2001

[4] T.Cormen, C. Leiserson and R. Rivest,. *"Introduction to Algorithms."* The MIT Press, Cambridge, Massachusetts, 1990.

[5] Kalika Bali, Partha Pratim Talukdar, , *Tools for the development of a Hindi Speech Synthesis System*, 5th ISCA Speech Synthesis Workshop, Pittsburgh, pp.109-114,2004

[6] "http://tdil.mit.gov.in/corpora/ach-corpora.htm#tech" Marathi CIIL corpus.

[7] Singh, S. P., et al *"Building Large Vocabulary Speech Recognition Systems for Indian Languages"*, International Conference on Natural Language Processing, 1:245-254, 2004.

[8] M Kumar., et al "A Large Vocabulary Continuous Speech recognition system for Hindi", IBM Research and Development Journal, September 2004.

[9] IS13194:1991, Indian Script Code for Information Interchange– ISCII-91, Bureau of Indian Standards, April,1999

[10] Rohit Kumar et al. "Unit Selection Approach for building Indian Language TTS", Interspeech 2004, Jeju Island, Korea.

[11] "Digital library of India, http://dli.iiit.ac.in", 2005

[12] Raj Reddy, "PCtvt: A multifunction information appliance for the illiterate people, http://www.rr.cs.cmu.edu/pctvt.ppt" in *ICT4B retreat at UC Berkeley, August 26, 2004*