International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 6, No.5 (Sep-2017)

http://dx.doi.org/10.12785/ijcds/0605008

MLLR Based Speaker Adaptation for Indian Accents

Balaji V[1] and G. Sadashivappa[2]

[1] *Research Scholar, Department of Computer Science, Christ University, Bangalore, India*
[2] *Professor and Head, Department of Telecommunication Engineering, R V College of Engineering, Bangalore, India*

Received 28 Feb. 2017, Revised 22 Apr. 2017, Accepted 8 Aug. 2017, Published 1 Sep. 2017

Abstract: Speech Recognition has become an inherent and important feature of today's mobile based apps. Speech input is a very popular option for people with limitations of using the keyboard / mouse in a computer system. Nowadays, more voice messages are used than written text as they also convey the emotions of the speakers. As solutions are developed with native speakers of a language, many of the English input systems have higher accuracy for native speakers than for people with English as their second language (L2), especially for Asian population. The complexity increases since the accent and intonation of Indian speakers are varied from region to region and state to state. This paper analyses an effective speaker adaptation mechanism implemented with Indian speaker profiles and with a very small amount of adaptation data. This research is to facilitate a speaker adaptive system for the speech disabled users with limited disabilities like stuttering and/or unintelligible speech due to illness like cerebral palsy. Experimental results show improvements in the recognition accuracy for speakers speaking small sentences.

Keywords: Automatic Speech Recognition (ASR), Speaker Adaptation, HMM (Hidden Markov Model), Linear Transformations, Maximum Likelihood Linear Regression (MLLR).

1. INTRODUCTION

Speech Recognition is a widely-used mechanism to enable an alternative input to a computerized system. It is comprised of techniques to accept user's voice, convert the analog sound signals to digital form, analyze the digitized voice data, perform pattern matching to identify the underlying task and execute the same. The counterpart of ASR (Automatic Speech Recognition), the text to speech system (TTS) is complementary to the recognition system to facilitate alternative and augmentative communication for the benefit of people with disabilities.

Juri Ganitkevitch, in his paper [1] highlights the three different types of speech recognition systems:

1. Speaker independent (SI) systems – a generic system used in general purpose searches, IVR (Interactive Voice Response) systems, etc.

2. Speaker dependent (SD) systems – a customized system for individual speakers and/or acoustic environments. These require extensive set up and training, which may be difficult in case of users who have disability in speech. However, the benefit of these systems is that once properly trained, the accuracy is very high compared to SI systems.

3. Speaker Adaptive (SA) systems – these start as Speaker Independent systems, use the limited speaker–specific training data and iteratively adjust/adapt the model parameters to improve the accuracy. The expected advantage of these systems is that the training overhead is reduced yet the accuracy is improved over SI systems. There are 2 techniques widely used for speaker adaptation, they are Speaker Normalization and Model Adaptation Techniques [2] and the latter is chosen for this study.

The proposed system uses one of the popular methods of adaptation called Maximum likelihood linear regression (MLLR) to customize the speech recognition capabilities of a desktop system for Indian speakers, speaking English. In phonology, Indian English varies vastly from region to region. The accent of Indians speaking English does lean towards a more vernacular and tinted with their native language. There are about 22 different native languages spoken in India [21] that are grouped under six language families (Indo-Aryan language family, Dravidian language family, Austroasiatic language family, Sino-Tibetan language family, Tai–Kadai language family, Great Andamanese languages) and hence selection of a language or a family itself is an important prerequisite to effectively implement the recognition system.

2. NEED FOR ADAPTATION IN ASR

According to Lawrence Rabiner, who has pioneered signal processing and speech recognition systems, ASR systems hardly achieve 100% accuracy when used across people due to the following reasons:

E-mail address: balaji.v@res.christuniversity.in, sadashivappag@rvce.edu.in

1. Speaking styles and accent of people are variant: The parameters are speaking rate, volume, accent, dialect, pitch and co–articulation.

2. The Speech production anatomy is varying across speakers: Different speakers produce sound waves of different frequencies. In general, adult, male speakers have lower pitch (fundamental frequency) compared to adult–females and children.

3. The pronunciation of various phonemes/words is a strong variant.

4. Placement and quality of the infrastructure (hardware) used – channel conditions, distance of microphone, environmental noise, etc.

5. User specific / Domain specific contents of the language – this is called the language model which is largely used in dictation systems.

By these factors, it is evident that developing a high accuracy, speaker independent recognition system is almost a far–fetched goal. Hence there is a need for developing a speaker independent recognition system with reasonable accuracy and iteratively adapt it using one or more of the above parameters and improve the recognition levels.

3. SPEAKER ADAPTATION TECHNIQUES

In a Speech Recognition System, Adaptation Techniques aim to improve the recognition accuracy for a particular speaker, his/her accent, acoustic environment and transmission channel. To start with, adaptation uses a very small set of sample training data to generate the acoustic feature vectors and improve on the results iteratively. This process enables the Speaker Independent Systems to achieve the accuracy levels of Speaker Dependent Systems without the time consuming and elaborate training process.

There are different approaches followed for speaker adaptation. They are broadly classified into 3:

1. Model based adaptation: Adapt the parameters of the basic acoustic model(s) of the speaker independent system to match the training data. The various techniques for model adaptation are,
    a. Maximum a posteriori (MAP) adaptation of HMM/GMM parameters
    b. Maximum likelihood linear regression (MLLR) of Gaussian parameters
    c. Learning Hidden Unit Contributions (LHUC) for neural networks
2. Speaker normalization: Normalize the speech data collected during testing to reduce the mismatch with the acoustic models. Types of normalization include,
    a. Vocal Tract Length Normalization (VTLN)
    b. Constrained MLLR (cMLLR) model-based normalization
3. Speaker space: Estimate many different sets of acoustic models, which characterizes new speakers. One of the following techniques could be used:
    a. Cluster-adaptive training
    b. Eigen voices
    c. Speaker codes

Adaptation techniques may operate in a number of modes. If the original transcription of the adaptation data (word level) is available and used as input, it is termed as supervised adaptation, whereas if the adaptation data is unlabeled, the adaptation is unsupervised [7]. Situations in which all the adaptation data is available in one block (e.g., from a system enrolment session) and the system is adapted once before use, is termed static adaptation. Alternatively, the data may become available in smaller parts as the system is used and the system will be adapted incrementally [14] [15]. This mode is named dynamic adaptation. The other terms used to describe these two are, block and incremental modes. The MLLR techniques described in this paper concentrate on supervised and dynamic adaptation modes.

Since speech is a continuous signal, if we use Hidden Markov Model for speech recognition, the observations typically form a Gaussian distribution, says Lawrence Rabiner. Adaptation allows Gaussian Mixture Models (GMMs) to be seeded with labeled data. If the unlabeled data also could be incorporated into the model with reasonable accuracy, we will get a more robust model. Earlier work on this domain [3] show that the process of adaptation updates the mean and variances of all the Gaussians so that the new acoustic model fits more closely to the accent of the speaker. In this study, the modified mean value is used to create the mllr matrix and this matrix is used during the recognition phase with test data.

4. EARLIER RESEARCH WORK

Koichi Shinoda, in his paper on survey of adaptation techniques [4], discusses the different scenarios where speaker adaptation is necessary to overcome the mismatches between the training speech data and test speech data. The mismatches may arise due to factors like speaker variability, channel variability, environmental distortion, etc. Ananth Sankar [6] also emphasizes that the accent of the speaker could cause the degradation in recognition accuracy. He states that the accent of the speaker used during the training could be mismatched with that of the test speaker. Thus, the default acoustic models, trained with native speakers of English, will provide low recognition accuracy if the test data is generated by non-native English speakers. There are two ways to reduce this mismatch:

Let us consider an acoustic model Λ_{Trg} trained using speech signal S_{Trg}, which is the (training) set of speech samples collected from native speakers. Now when we use Λ_{Trg} with a different set of speech samples, S_{Test}, recognition quality degrades. This is due to mismatch in

speech utterance D(S), as shown in Figure 1. This mismatch may be reduced by 2 methods.

1. Map the test features X_{Test} to an estimate of original features X_{Trg} and use the original model Λ_{Trg} for recognition – transformation in the feature-space. This is referred to as Speaker normalization in section 3 above.
2. Map the original model Λ_{Trg} to the transformed model Λ_{Test} which will recognize the test utterances X_{Test} with a better accuracy – transformation in the model-space. This is referred to as Model Based adaptation in section 3 above.

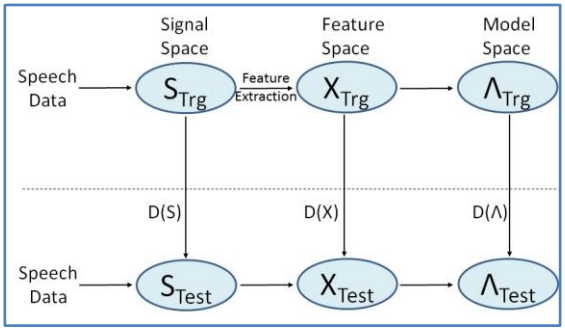

Figure 1. Speech signal mismatches between training & testing

The proposed work is to test the effectiveness of one particular adaptation technique using MLLR in a scenario where environmental distortion creates the mismatch between the training and test data sets. The cause for the same was noisy surroundings and / or using a laptop / desktop mike instead of a head phone mike.

Research work on enabling the human computer interaction for people with speech disabilities have been taken up during recent years and many systems are established for command and control of home appliances and voice–in–voice– out communication aids (VIVOCA). Frank Rudzicz's paper [5] discusses one such system where the acoustics properties of the unintelligible, dysarthric speech signals are transformed using the following techniques: Splicing – correcting dropped and inserted phoneme errors, Tempo morphing and Frequency morphing. This literature has been the motivation behind this work and the change will be to use adaptation mechanisms of the model, instead of modifying the signals themselves.

5. EXPERIMENTAL SETUP

Figure 2 shows the experimental setup for the proposed work. It is based on the architecture of a speech recognizer using Hidden Markov Model (HMM). Input to the system is a recorded speech signal and the decoded text is written to files in ASCII format.

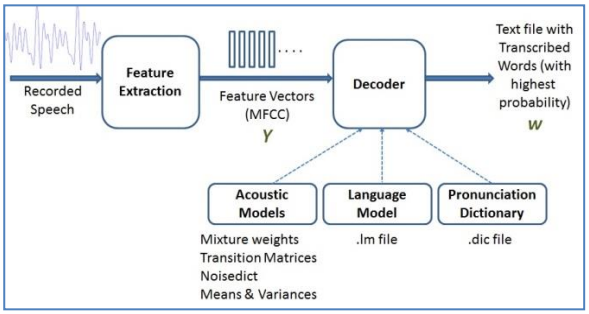

Figure 2. Experimental Setup

The basic objective of a HMM based speech recognition system is to identify the word sequence that has the highest probability of producing the input observation sequence. I.e.,

$$\text{argmax}_{w \in L}\, p(w \mid Y) \qquad (1)$$

Using Bayes' rule on conditional probability, this can be rewritten as,

$$\text{argmax}_{w \in L}\, \frac{p(Y \mid w)\, p(w)}{p(Y)} \qquad (2)$$

p(Y) – the probability of the input acoustic vectors is 1 as they are observed and hence this equation is further reduced to,

$$\text{argmax}_{w \in L}\ (p(Y \mid w) p(w) \qquad (3)$$

Where w is the words in the preferred language L and Y is the observation sequence, which is the acoustic feature vector (X_{Trg} or X_{Test} as in Figure 1). The term p(w) is the apriori probability of the word sequence w. This term is called the language model and contains the probability of each of the words in the speech corpus and their 1-gram, 2-gram and 3-gram probabilities.

1. 1-gram (or unigram): probability of the word appearing in isolation; this model can be treated as one-state finite automata. The 1-gram probability of a word all depends on its own. It is represented as,

$$p(w1\ w2\ w3) = p(w1)p(w2)p(w3) \qquad (4)$$

2. 2-gram (or bigram): this and the next model can be treated as n-gram models with n = 2 and 3 respectively. As an example, consider the sentence,

"a convenient reference for".

The 2-gram probability is,

$$p(a\ convenient\ reference\ for) \approx p(a \mid < sil >) * p(convenient \mid a) * p(reference \mid convenient) * p(for \mid reference) \qquad (5)$$

In general, it is

$$p(w1\ w2\ w3) = p(w1 \mid < sil >) * p(w2 \mid w1) * p(w3 \mid w2) \qquad (6)$$

Where <sil>: silence. Every sentence is assumed to be following and ending with a "silence" syllable.

3. Similarly, a 3-gram (or trigram) model considers every word's probability on the condition of the probability of 2 of its previous words.

In our experiment, the input speech signal is transformed using an FFT (Fast Fourier Transform) with around 20 frequency bins which are non–linearly distributed across the speech spectrum. This non–linear scale is called a mel scale [8]. The log spectrum thus obtained is subject to a truncated discrete cosine transformation (DCT), thus producing the MFCCs (Mel–frequency Cepstral Coefficients).

The adaptation model described in this paper has been implemented in Java for efficiency and portability using the CMUSphinx toolkit for speech recognition [16]. CMUSphinx is an Open Source Speech Recognition Toolkit, developed and maintained by CMU (Carnegie Mellon University) and has the repository of over 20 years of the research materials. It has been rewarded as the project of the week on Sourceforge, dated June 9th, 2016. CMUSphinx also includes SphinxTrain, an acoustic model trainer which is used while configuring the system for speaker adaptation. Sphinx4 is the ASR toolkit from the group of CMUSphinx, chosen for this study. It is written entirely in Java and provides APIs that can be imported into user projects to enable speech recognition and transcription.

The recognition process followed in this project is as follows: The acoustic vectors are composed of 13 MFCC (Mel–frequency Cepstral Coefficients) which are static features, the mel–cepstrum deltas and the delta–deltas (dynamic features) of each coefficient (first and second order derivatives) [9]. This HMM–based, context–dependent acoustic model was trained on 2 corpuses consisting of technical and general English text. The static features are extracted from the wav files and stored with an extension .mfc. The wav files contain the audio of the adaptation data, supplemented with a text file of corresponding transcription.

The adaptation process takes the transcribed data and improves the model that is provided by Sphinx4 for native speakers. This process gives good results even if the training set is very small in size. It is also found to be more robust than training, if speech samples of 5 minutes are only available and enhances the dictation accuracy by adaptation to the particular speaker.

A. *Building the Acoustic Model*

The relationship between the speaker's recorded audio signal and its corresponding phonetic units is represented by the acoustic model. Following are the different types of acoustic models supported by CMUSphinx: continuous, semi-continuous and phonetically tied (PTM). The input audio signal is split into frames of 10ms and the feature vector containing 39 numbers[1] is extracted [16]. While computing the score of each frame, Gaussian mixtures are used and the number of Gaussians makes the model load faster with less accuracy or load slower with more accuracy. In continuous model, the total number of Gaussians is about 1.5 lakhs and hence it is the slowest. The semi-continuous model has very less number of Gaussians compared to continuous model and hence very fast but semi-continuous models are also a bit less accurate. PTM model lies in between the two, uses about 5000 Gaussians and provides a reasonable accuracy.

B. *Building the Language Model*

The language model models the sequence of words in a particular language. It describes what is likely to be spoken in a particular context and uses a stochastic approach. Word transitions are defined in terms of transition probabilities. It restricts the search space during decoding thereby optimizing the recognition process. In this experiment, we have used the locally developed voice corpora and built up 1–gram, 2–gram and 3–gram word sequences and their corresponding probabilities. The language model is combined with the acoustic model to get multiple word sequences and the best one is chosen among them. The default language model is of very large size and so is provided in a binary format (en-us.lm.bin). This model has been created by acquiring archived data over several years by the Linguistic Data Consortium (LDC) at the University of Pennsylvania. An extended or customized language model can be created by using any of the toolkits like CMU language modeling toolkit. There are different file formats to store the language model. They are,

1. ARPA format (pure text format and can be edited) with extension .lm; size of the file is large and hence takes more space and time while loading. This format is chosen for the proposed work.
2. Binary format (cannot be edited) with extension .lm.bin ; size of the file in this format is very compact and hence loads faster than text format file and occupies less space in memory while executing.
3. A binary DMP format, which is obsolete now.

The language model toolkit takes input as the transcribed data and creates the .lm file with n–gram word sequences. The toolkit used in this study is a small online service and is available at http://www.speech.cs.cmu.edu/tools/ lmtool-new.html.

C. *Adaptation with MLLR*

MLLR is one of the most popular techniques that is used in special cases where only a limited amount of data per class / category is available. MLLR trains a linear transform which warps the Gaussian means so as to maximize the likelihood of the data. This method groups the classes that are acoustically close and transform them together. The ML parameter estimation is solved using the expectation–maximization (EM) algorithm to iteratively improve the likelihood. The commonly used parameters are the mean and variance

[1] 39 numbers – 13 static features, 13 deltas and 13 delta–deltas (dynamic features)

which are transformed by optimizing the following equation:

$$Q(M, \hat{M}) = K - \frac{1}{2}\sum_{m=1}^{M}\sum_{\tau=1}^{T}\gamma_m(\tau)\left[K^{(m)} + \log\left(\left|\sum^{(m)}\right|\right) + (o(\tau) - \hat{\mu}^{(m)})^T \sum^{(m)-1}(o(\tau) - \hat{\mu}^{(m)})\right] \quad (7)$$

where,

$\hat{\mu}$ and $\hat{\Sigma}$: the transformed mean and variance for the component m.

M: Total number of components associated with the transform

$$\gamma_m(\tau) = p(q_m(\tau) \mid M, O\tau \quad (8)$$

$q_m(\tau)$: the Gaussian component m at time τ

K: A constant value which depends on the transition probabilities only

K(m): the normalization constant associated with Gaussian component m

{o(1), o(2), … o(T)} : the adaptation data on which the transform is trained.

D. *Pre–processing of input voice data*

Pre-processing of the incoming speech signal has been done in the following way. Recordings were done with a Sony MDRZX770BT Bluetooth Noise Cancelling Stereo Headset microphone. The continuous signal has been sampled at 16 KHz with "mono" option at 16–bit resolution. Steps included in the pre–processing activity are presented in the flow chart, Figure 3.

The speech data captured was continuous sentences with adequate pauses between the words. Analysis of the output was done based on individual words. The accuracy of the recognition was measured on a per sentence basis. While calculating WER / accuracy, the following strategy was used:

Insertion: inclusion of one complete word

Deletion: removal of one complete word

Substitution: replacement of one complete word by another

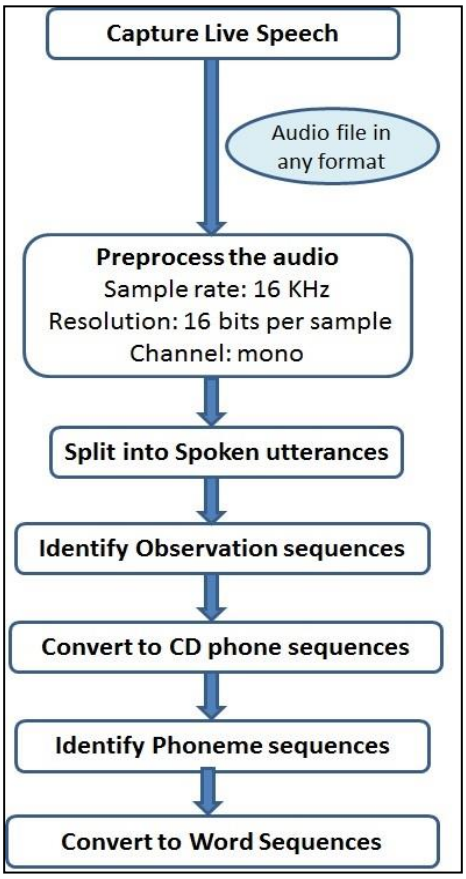

Figure 3. The process of mapping from spoken utterances to word sequences during adaptation

As an example of modeling speech with HMM's, we used 2 sets of vocabularies, a 14–sentence technical content and a 100–sentence general content. We trained the HMMs for 2 speakers, one male and one female.

Today, speech recognition is performed on a wide variety of hardware from a standard desktop PC to very small, handheld devices. Each of them have different limits on the capability of their sound cards, and thus can record at different sampling rates. Typical sampling rates vary between 16 kHz to 48 kHz of audio. They are coded with bit rates of 8 to 16-bits per sample. Using a higher sample rate or bits per sample will yield very good audio quality, however will decrease the speed of the recognition engine. As a tradeoff, the current standard on a desktop application is to use speech audio data recorded at sampling rates of 16 kHz/16bits per sample [18] [19].

E. *Adaptation Process & Observations*

The recognition accuracy for a non–native speaker of the English language with the sample corpus has been measured. The algorithm worked on continuous speech recognition, with the test data containing sentences of size, 10 words or more. The voice files were in the same format and processed the same way as mentioned in Figure 3. The results are presented in the table below:

TABLE I. OBSERVATIONS WITHOUT MLLR ADAPTATION FOR NON–NATIVE SPEAKER PROFILES

Sample Sentence Size	Result	Accuracy %	WER %
10 words (General)	Deletions: 0 Substitutions: 4 Insertions: 4	60.00	80.00
13 words (General)	Deletions: 0 Substitutions: 5 Insertions: 2	61.54	53.85
12 words (General)	Deletions: 2 Substitutions: 1 Insertions: 0	75.00	25.00
12 words (Technical Content)	Deletions: 0 Substitutions: 0 Insertions: 1	100.00	8.33
12 words (Technical Content) – different speaker	Deletions: 0 Substitutions: 5 Insertions: 2	58.33	58.33

Accuracy and Word Error Rate (WER) are the two, popular metrics that are used to benchmark the Speech Recognition algorithms. They are calculated as,

$$Accuracy = \frac{Total\ Number\ of\ words - Deletions - Substitutions}{Total\ Number\ of\ words} \quad (9)$$

$$WER = \frac{Insertions + Deletions + Substitutions}{Total\ Number\ of\ words} \quad (10)$$

To improve the efficiency of the speech recognition model, the Maximum Likelihood Linear Regression (MLLR) technique is applied. Sphinx4 provides a base line acoustic model which is adapted using this MLLR technique with the recorded speech of the target speakers (1 male and 1 female). MLLR reduces the mismatch between the set of trained models and the non–native adaptation data. The steps involved in the adaptation process are detailed out in the below flow chart (Figure 4).

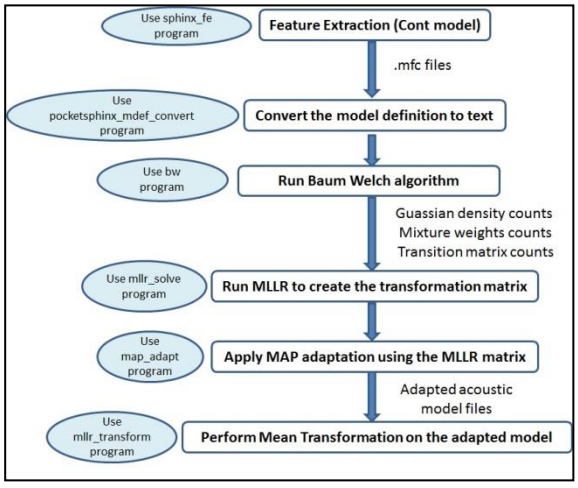

Figure 4. The process of transforming the model to include adaptation

Features are data representation of the input voice which are extracted from the wav files as described in experimental setup. Each .wav file is stored in a .mfc file during this phase. The Baum Welch algorithm is then run on these features to execute the forward and backward processes of generating the transition probability and emission probability matrices. These matrices are the representations of the goodness of the model to recognize the test data.

The output of the last step is a MLLR matrix containing the transformed acoustic features, which is then applied on the recognizer's decoding procedure using the function recognizer.loadTransform(). There are also 3 other inputs provided to the recognizer, which are the adapted acoustic model, newly created Language model and Dictionary containing the words in our corpus. The recognition process is then repeated and the results observed are tabulated below:

TABLE II. OBSERVATIONS AFTER MLLR ADAPTATION FOR NON–NATIVE SPEAKER PROFILES

Sample Sentence Size	Result	Accuracy %	WER %
10 words	Deletions: 0 Substitutions: 1 Insertions: 0	90.00	10.00
14 words	Deletions: 0 Substitutions: 0 Insertions: 0	100.00	0
12 words	Deletions: 0 Substitutions: 1 Insertions: 0	90.00	10.00
9 words	Deletions: 0 Substitutions: 0 Insertions: 1	100.00	11.11

The components of the newly adapted model are,

a) Dictionary, which maps words to pronunciations of the selected speaker. A default dictionary is provided (cmudict-en-us.dict) for various languages that are supported currently. This pronunciation dictionary published by the Carnegie Mellon University is in the machine–readable format and the pronunciations are captured for North American English. There are more than 1,34,000 words and corresponding pronunciations with open access permissions. In the process of adaptation, we have updated the dictionary to contain the specific sentences and their phoneme–based pronunciations by Indian speakers.

b) Language Model: The model built for the small corpus of 14 sentences included 76 1–gram, 126 2–gram and 119 3–gram models. The same for 100 sentences contained 476 1–gram, 885 2–gram and 885 3–gram models.

Sample portions of the language model and the dictionary used in this work are shown in Figures 5 & 6. They have been generated by using the selected speech corpuses of general and technical sentences, which were stored in plain-text format in the transcription files.

This portion of the language model shows the 1-gram probabilities of the words in the training sentences. For example, the probability of the word "ANALYSIS" (on line no.5) appearing independently and not depending on the words preceding or succeeding it. The 2-gram and 3-gram probabilities are also present at a later part of this file.

```
\1-grams:
-1.3310 </s> -0.3010
-1.3310 <s> -0.2472
-1.8751 A -0.2952
-2.4771 ANALYSIS -0.2952
-1.7782 AND -0.2848
-2.4771 APPLIED -0.2952
-2.1761 ARE -0.2788
-2.1761 AS -0.2788
-2.4771 BASIC -0.2967
-2.4771 BE -0.2996
-2.1761 BOOK -0.2772
-2.4771 BRIEF -0.2981
-2.1761 CHAPTER -0.2967
-2.4771 CHAPTERS -0.2803
-2.4771 COMPLETELY -0.2996
-2.4771 CONCEPTS -0.2967
-2.4771 CONSULT -0.2952
-2.4771 CONVENIENT -0.2996
-2.4771 DETAIL. -0.2803
-2.0000 DIGITAL -0.2952
-2.1761 DISCRETE-TIME -0.2937
-2.4771 DISCUSS -0.2967
-2.4771 DOES -0.2996
-2.4771 ESSENTIAL -0.2967
-2.4771 ESTABLISH -0.2923
-2.4771 FIND -0.2981
-2.1761 FOR -0.2981
-2.4771 GOOD -0.2996
-2.4771 HAVE -0.2952
-2.4771 IMPORTANT -0.2937
-2.0000 IN -0.2908
-2.4771 INTENDED -0.2952
-2.4771 INTRINSICALLY -0.2981
-2.1761 IS -0.2981
-2.1761 IT -0.2967
-2.4771 LATER -0.2996
-2.4771 MAY -0.2996
-2.4771 MOST -0.2996
-2.4771 NOT -0.2996
-2.4771 NOTATION -0.2967
-1.8751 OF -0.2632
-2.4771 ON -0.2967
-2.4771 PRESENT -0.2952
-1.7782 PROCESSING -0.2726
-2.4771 PROVIDE -0.2996
-2.4771 READER -0.2996
-2.4771 READERS -0.2996
-2.4771 REFERENCE -0.2981
```

Figure 5. Portion of the Language Model file

The portion of the dictionary given below depicts the way in which each word is pronounced by the training speaker(s). Each word is divided in to phoneme sequences and they are shown, delimited by space. Some words have more than one phoneme sequence, which are marked with (2) in the word list.

```
A                   AH
A(2)                EY
ANALYSIS            AE N AE L IH S IH S
AND                 AH N D
AND(2)              AE N D
APPLIED             AH P L AY D
ARE                 AA R
ARE(2)              ER
AS                  AE Z
AS(2)               EH Z
BASIC               B EY S IH K
BE                  B IY
BOOK                B UH K
BRIEF               B R IY F
CHAPTER             CH AE P T ER
CHAPTERS            CH AE P T ER Z
COMPLETELY          K AH M P L IY T L IY
CONCEPTS            K AA N S EH P T S
CONCEPTS(2)         K AA N S EH P S
CONSULT             K AH N S AH L T
CONVENIENT          K AH N V IY N Y AH N T
DETAIL.             D IH T EY L
DIGITAL             D IH JH AH T AH L
DIGITAL(2)          D IH JH IH T AH L
DISCRETE-TIME       D IH S K R IY T T AY M
DISCUSS             D IH S K AH S
DOES                D AH Z
DOES(2)             D IH Z
ESSENTIAL           EH S EH N SH AH L
ESSENTIAL(2)        IY S EH N SH AH L
ESTABLISH           IH S T AE B L IH SH
FIND                F AY N D
FOR                 F AO R
```

Figure 6. Portion of the Dictionary

6. RESULTS AND OBSERVATIONS

The goal of the undertaken research is to facilitate communication for the speech disabled user community. The work aims to develop an adapted system of ASR and TTS that will act as a surrogate voice for the people who find it difficult to communicate and make presentations in public. The results published here are of an intermediate step in the ASR process.

Table 3 shows the performance of the proposed system on the non-native test set when using the native model (baseline) and non-native (adapted) model. This result provides evidence that the training or adaptation procedure implemented is improving the speaker-independent, baseline model. The new, non-native models created are adequately trained.

TABLE III. COMPARISON OF BASELINE SYSTEM WITH ADAPTED SYSTEM FOR NON–NATIVE SPEAKER PROFILES

Model	WER %
Baseline model	50
Adapted model with non-native speech	8

A similar work had earlier been carried out to cater to non-native English speakers with German accents [11]. The Word Error Rate achieved with native models was 49.3% (comparable with 50% in table 3). However, their non-native, adapted model yielded a reduced WER of

43.5% whereas we could achieve an average of 8% and in some cases, 0%. That is, every word of a sentence has been recognized exactly as it is. This difference could be attributed to the fact that a pooled data set was used in the project [11] which had both native and non-native training data. We have trained our models with non-native speakers' data to improve the accuracy. Both the experiments have modeled the conversational speech instead of isolated words or digits recognition.

The system yields up to 100% accuracy for sample data from the training set but an arbitrary validation set gives lower accuracy rates (accuracy 67% and WER 42% on a 100–sentence corpus). This over fitting problem could be avoided by increasing the length of the adaptation data using a larger corpus whereby accuracy of the recognizer could be improved. Improvements can also be brought in by repeatedly recording the same text by the speaker to accommodate intra–speaker variations.

7. CONCLUSION AND FUTURE WORK

The speech data used to conduct the experiments in this work have been captured locally with direct interactions with people and in noisy environments. This may be one of the factors leading to the low recognition accuracy levels mentioned above. To overcome this, it is proposed to use the professional voice databases developed [12] by research organizations like Indian Institute of Technology, Guwahati. As an extension, the voice data of disabled speakers are also available, published by University of Illinois with the data of 16 speakers with different levels of disability in speech due to dysarthria [10] [20]. It is proposed to include these data sets also for training and testing, in the future enhancement of this work.

As suggested in [22], an M–based approach using MATLAB for the design of filters in the speech analysis could be adopted for the next version of the experiments described here. Akitoshi Matsuda and Shinichi Baba [22] implemented the M–based approach in the domain of image processing. The same could be extended to speech processing, where filter design is an important criterion for the quality of the output voice.

REFERENCES

[1] Juri Ganitkevitch, "Speaker Adaptation using Maximum Likelihood Linear Regression", Seminar on Automatic Speech Recognition, 2005.

[2] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models", Computer Speech and Language (1995) 9, 171–185.

[3] Ngoc Thang Vu, Yuanfan Wang, Marten Klose, Zlatka Mihaylova, Tanja Schultz, "Improving ASR Performance On Non-native Speech Using Multilingual and Crosslingual Information", 15th Annual Conference of the International Speech Communication Association, Singapore, 2014 (Interspeech 2014).

[4] Koichi Shinoda, "Speaker Adaptation Techniques for Automatic Speech Recognition", Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2011 (APSIPA ASC 2011) Xi'an, October 2011.

[5] Frank Rudzicz, "Adjusting dysarthric speech signals to be more intelligible", article in Computer Speech & Language, September 2013.

[6] Ananth Sankar and Chin-Hui Lee, "A Maximum-Likelihood Approach to Stochastic Matching for Robust Speech Recognition", IEEE Transactions on Speech and Audio Processing, Volume: 4, Issue: 3, May 1996.

[7] P.C. Woodland, "Speaker Adaptation for Continuous Density HMMs: A Review", ITRW (ISCA Tutorial and Research Workshop) on Adaptation Methods for Speech Recognition, August 2001.

[8] Mohanty, Sanghamitra, and Basanta Kumar Swain, "Speaker Identification using SVM during Oriya Speech Recognition", International Journal of Image Graphics and Signal Processing, September 2015.

[9] Benjamin Lecouteux ; Michel Vacher ; François Portet, "Distant speech recognition for home automation: Preliminary experimental results in a smart home", 6th Conference on Speech Technology and Human-Computer Dialogue (SpeD), May 2011, DOI: 10.1109/SPED.2011.5940728.

[10] Harsh Vardhan Sharma, "Acoustic Model Adaptation for Recognition of Dysarthric Speech", Ph.D. Dissertation submitted in the Graduate College of the University of Illinois, 2012.

[11] Zhirong Wang, Tanja Schultz, Alex Waibel. "Comparison of Acoustic Model Adaptation Techniques on Non-Native Speech", IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003 Proceedings, (ICASSP '03) April 2003.

[12] Haris B C, G.Pradhan, A.Misra, S.R.M. Prasanna, R.K. Das and R.Sinha, "Multivariability Speaker Recognition Database in Indian Scenario", International Journal of Speech Technology, November 2011.

[13] Mark Gales and Steve Young, "The application of Hidden Markov models in Speech Recognition", Foundations and Trends Signal Processing, Vol.1, No. 3, 2007.

[14] M.J.F. Gales, "Maximum Likelihood Linear Transformations for HMM-based Speech Recognition", Computer Speech and Language, Volume 12, May 1998.

[15] Gales, M.J.F. and Woodland P.C., "Mean and variance adaptation within the MLLR framework", Computer Speech & Language, 1996.

[16] Carnegie Mellon University, "CMUSphinx Tutorial for Developers", 1996.

[17] Audio conversion tool used: http://audio.online-convert.com

[18] T. Sainath et al., "Convolutional neural networks for LVCSR", ICASSP, 2013.

[19] https://en.wikipedia.org/wiki/Acoustic_model

[20] H. Kim, M. Hasegawa-Johnson, A. Perlman, J. Gunderson, T. Huang, K. Watkin, and S. Frame, "Dysarthric speech database for universal access research", Proceedings of Interspeech, Brisbane, Australia, September 2008.

[21] Languages of India, from Wikipedia, (https://en.wikipedia.org/wiki/Languages_of_India).

[22] Akitoshi Matsuda and Shinichi Baba, "M-based Filter Design for Communication and Imaging Systems", International Journal of Computing and Digital Systems, Vol 1, 2012.

Balaji V is a research scholar, affiliated to the Department of Computer Science, Christ University, Bangalore, India. She has been in the field of computer science as a software professional and a teacher for 23 years and her areas of interest include Speech Processing, Software Engineering and Databases.

G Sadashivappa received his BE degree in Electronics Engineering from Bangalore University in 1984 and M.Tech degree in Industrial Electronics from Karnataka Regional Engineering College (NIT-K), Mangalore University in 1991. He worked as Lecturer in AIT Chickmagalore during 1984–86, Engineer trainee in Kirloskar Electric Co Ltd, Unit-IV, Mysore during 1986–87 and as lecturer at JMIT Chitradurga during 1987–89. Since 1992, he is working in R.V.College of Engineering, Bangalore, presently serving as the Professor and Head of Department of Telecommunication Engineering. He obtained his Doctoral degree from VTU Belgaum during 2011 in the area of image processing. His research areas include Image & Video Coding, Biomedical Signal Processing, Underwater Communication and Optical networks/protocols.

International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 6, No.5 (Sep-2017)

E-mail address: balaji.v@res.christuniversity.in, sadashivappag@rvce.edu.in