Pronunciation Variation Modeling for Non Native speech Recognition

A major project report submitted

in partial fulfillment of the requirements

for the award of the degree of

Master of Technology

in

Artificial Intelligence

By

PRABANDAPU SHASHI KANTH
(07MCMI09)

Under the Guidance of

Prof. P. N. Girija

Department of Computer & Information Sciences
School of Mathematics & Computer/Information Sciences
University of Hyderabad
Hyderabad -500046, INDIA.

June – 2009.

Certificate

This is to certify that the thesis entitled **"Pronunciation Variation Modeling for Non Native Speech Recognition"** being submitted by **PRABANDAPU SHASHI KANTH** in partial fulfillment for the award of the degree of **Master of Technology** in **Artificial Intelligence**, is a record of the bonafide work carried out by him under my supervision at the University of Hyderabad.

The matter embodied in this project report has not been submitted to any other university for the award of any other degree or diploma.

Prof. P. N. Girija
Supervisor,
Dept. of CIS,
University Of Hyderabad,
Hyderabad.

Head,
Dept. of CIS,
University of Hyderabad,
Hyderabad.

Dean,
School of MCIS,
University of Hyderabad,
Hyderabad.

Acknowledgments

There are a lot of people without whose support: physical, technical and moral, neither the project nor this manuscript could possibly have neared completion. Though I would have liked very much to do so, it is unfortunately not feasible to mention all of them individually here, as that would probably occupy half of this report.

I would like to take this opportunity to thank my guide **Prof. P. N. Girija,** for her guidance, valuable suggestions, support in every task of my work, and freedom in expressing my opinions during my project work.

I would like to specially thank the Head of the department **Prof. Arun Agarwal,** who very kindly provide me the facilities required to proceed with my work.

I would like to thank the Dean of the School of MCIS, **Prof. Amaranath** for giving me valuable suggestions.

My heartful thanks to my **friends,** and thanks to **classmates** for encouragement and support which they given me in completion of my course.

Before closing, I would like to extend special attention to my **Parents,** who gave me this position and providing me inspiration and moral support throughout my studies.

PRABANDAPU SHASHI KANTH

Abstract

In this project we worked with an acoustic and pronunciation model adaptation method for context-independent (CI) and context-dependent (CD) pronunciation variability to improve the performance of a non-native automatic speech recognition (ASR) system. The proposed adaptation method is performed in three steps. First,we perform phone recognition to obtain an n-best list of phoneme sequences and derive pronunciation variant rules by using a decision tree. Second, the pronunciation variant rules are decomposed into CD pronunciation variation. That is, some pronunciation variant rules that are dedicated to the specific phoneme sequences is classified into CD pronunciation variation. It is assumed here that CD pronunciation variabilities are invoked by a different pronunciation space from the mother tongue of a non-native speaker and the sandhi rules effects in a context,respectively. Third, the pronunciation model adaptation is completed by constructing a multiple pronunciation dictionary using the CD pronunciation variability. It is shown from the continuous Telugu-English ASR experiments. ASR system that is trained by native speech i.e., for telugu language.

Table of Contents

TOPIC **PAGE NO**

Chapter 1

Introduction

Considerable progress has been made in speech recognition in the past 15 years. Many successful systems [1-2] have emerged. Each of these systems has attained very impressive accuracy. However, they owe their success to one or more of the constraints they impose. Present project describes SPHINX-III, a speech recognition system that tries to test three of the constraints: 1) speaker dependence/independence, 2) function/content words, and 3) large vocabulary.

Speaker independence has been viewed as the most difficult constraint to overcome. This is because most parametric representations of speech are highly speaker dependent, and a set of reference patterns suitable for one speaker may perform poorly for another speaker. Researchers have found that errors increased by 300-500 percent when a speaker-dependent system is trained and tested in speaker-independent mode. Because of these difficulties, most speech recognition systems are speaker dependent. In other words, they require a speaker to “train” the system before reasonable performance can be expected. This training phase typically requires several hundred sentences.

Continuous speech recognition is significantly more difficult than isolated word recognition. Its complexity is a result of three innate properties of continuous speech. First, word boundaries are difficult to locate. Second, coarticulatory effects are much stronger in continuous speech, causing the same sound to appear differently in various contexts. Third, content words (nouns, verbs, adjectives, etc.) are often emphasized; while function words (articles, prepositions, pronouns, short verbs, etc.) are poorly articulated. Error rates increase drastically from isolated-word to continuous speech. However, in spite of these problems and degradations, it is important to work on continuous speech since continuous speech has many potential applications of man-machine communications.

Large vocabulary typically implies a vocabulary of about 1000 words or more. As vocabulary size increases, so does the number of confusable words also increases. Also, larger vocabularies require the use of sub word models, because it is difficult to train whole word models. Unfortunately, sub word units usually lead to degraded performance because they cannot capture coarticulatory (inter unit) effects as well as word models can. Error rate increased by 200-1000 percent in several studies [3]-[5]. Inspite of these problems, large vocabulary systems are still needed for many versatile applications, such as dictation, dialog systems, and speech translation systems.

In our project, we describe SPHINX, a large-vocabulary speaker-independent, continuous speech recognition system. SPHINX employs discrete hidden Markov models (HMM's) with LPC-derived parameters. To deal with speaker independence, we added knowledge to these HMM's in several ways. We represented additional knowledge through the use of multiple vector quantized codebooks. To deal with co articulation in continuous speech, yet adequately represent a large vocabulary, we introduced two new speech units' like content-words and function-word-dependent phone models [6].

The training step of a statistical recognition system is similar to the learning process of a chila. A child should experience a phenomenon many times and with a wide variability before being able to recognize it. The current ASR technology does not allow real-time implementation of models comparable to human complexity. This means that the variability of speech must be limited to achieve proper results [7].

The databases should contain observations that uniformly cover all the variability of speech. For example, if a phoneme is not included in the training

database, it will never be recognized. In addition, if male and female are not balanced in the training step, the accuracy in recognition may be expected to be insufficient for the less represented sex. Databases must therefore be properly designed.

1.1 Speech Recognition Basics

Speech recognition is the process by which a computer (or other type of machine) identifies spoken words. Basically, it means talking to your computer, and having it correctly recognizes what you are saying.

The following definitions are the basics needed for understanding speech recognition technology.

Utterance

An utterance is the vocalization (speaking) of a word or words that represent a single meaning to the computer. Utterances can be a single word, a few words, a sentence, or even multiple sentences.

Speaker Dependence

Speaker dependent systems are designed around a specific speaker. They generally are more accurate for the correct speaker, but much less accurate for other speakers. They assume the speaker will speak in a consistent voice and tempo. Speaker independent systems are designed for a variety of speakers. Adaptive systems usually start as speaker independent systems and utilize training techniques to adapt to the speaker to increase their recognition accuracy.

Vocabularies

Vocabularies (or dictionaries) are lists of words or utterances that can be recognized by the SR system. Generally, smaller vocabularies are easier for a computer to recognize, while larger vocabularies are more difficult. Unlike normal dictionaries, each entry doesn't have to be a single word. They can be as long as a sentence or two. Smaller vocabularies can have as few as 1 or 2 recognized utterances (e.g.” Wake Up"), while very large vocabularies can have a hundred, thousand or more!

Accuracy

The ability of a recognizer can be examined by measuring its accuracy - or how well it recognizes utterances. This includes not only correctly identifying an utterance but also identifying if the spoken utterance is not in its vocabulary. Good ASR systems have an accuracy of 98% or more! The acceptable accuracy of a system really depends on the application.

Training

Some speech recognizers have the ability to adapt to a speaker. When the system has this ability, it may allow training to take place. An ASR system is trained by having the speaker repeat standard or common phrases and adjusting its comparison algorithms to match that particular speaker. Training a recognizer usually improves its accuracy.

Training can also be used by speakers that have difficulty speaking, or pronouncing certain words. As long as the speaker can consistently repeat an utterance, ASR systems with training should be able to adapt.

1.2. Types of Speech Recognition

Speech recognition systems can be separated in several different classes by describing what types of utterances they have the ability to recognize. These classes are based on the fact that one of the difficulties of ASR is the ability to determine when a speaker starts and finishes an utterance. Most packages can fit into more than one class, depending on which mode they're using.

Isolated Words

Isolated word recognizers usually require each utterance to have quiet (lack of an audio signal) on BOTH sides of the sample window. It doesn't mean that it accepts single words, but does require a single utterance at a time. Often, these systems have "Listen/Not-Listen" states, where they require the speaker to wait between utterances (usually doing processing during the pauses). Isolated Utterance might be a better name for this class.

Connected Words

Connect word systems (or more correctly 'connected utterances') are similar to Isolated words, but allow separate utterances to be 'run-together' with a minimal pause between them.

Continuous Speech

Continuous recognition is the next step. Recognizers with continuous speech capabilities are some of the most difficult to create because they must utilize special methods to determine utterance boundaries. Continuous speech recognizers allow users to speak almost naturally, while the computer determines the content. Basically, it's computer dictation.

Spontaneous Speech

There appears to be a variety of definitions for what spontaneous speech actually is. At a basic level, it can be thought of as speech that is natural sounding and not rehearsed. An ASR system with spontaneous speech ability should be able to handle a variety of natural speech features such as words being run together, "ums" and "ahs", and even slight stutters.

Voice Verification/Identification

Some ASR systems have the ability to identify specific users.

1.3. Uses and Applications

Although any task that involves interfacing with a computer can potentially use ASR, the following applications are the most common right now.

Dictation

Dictation is the most common use for ASR systems today. This includes medical transcriptions, legal and business dictation, as well as general word processing. In some cases special vocabularies are used to increase the accuracy of the system.

Command and Control

ASR systems that are designed to perform functions and actions on the system are

defined as Command and Control systems. Utterances like "Open Netscape" and "Start a new xterm" will do just that.

Telephony

Some PBX/Voice Mail systems allow callers to speak commands instead of pressing buttons to send specific tones.

Wearable

Because inputs are limited for wearable devices, speaking is a natural possibility.

Medical/Disabilities

Many people have difficulty in typing due to physical limitations such as repetitive strain injuries (RSI), muscular dystrophy, and many others. For example, people with difficulty hearing could use a system connected to their telephone to convert the caller's speech to text.

Embedded Applications

Some newer cellular phones include C&C speech recognition that allows utterances such as "Call Home". This could be a major factor in the future of ASR and Linux.

1.4. Inside Speech Recognition

1.4.1. How Recognizer's Work

Recognition systems can be broken down into two main types. Pattern Recognition systems compare patterns to known/trained patterns to determine a match. Acoustic Phonetic systems use knowledge of the human body (speech production, and hearing) to compare speech features (phonetics such as vowel sounds). Most modern systems focus on the pattern recognition approach because it combines nicely with current computing techniques and tends to have higher accuracy.

Most recognizers can be broken down into the following steps:

1. Audio recording and Utterance detection
2. Pre-Filtering (pre-emphasis, normalization, banding, etc.)

3. Framing and Windowing (chopping the data into a usable format)
4. Filtering (further filtering of each window/frame/freq. band)
5. Comparison and Matching (recognizing the utterance)
6. Action (Perform function associated with the recognized pattern)

Although each step seems simple, each one can involve a multitude of different (and sometimes completely opposite) techniques.

(1) Audio/Utterance Recording: can be accomplished in a number of ways. Starting points can be found by comparing ambient audio levels (acoustic energy in some cases) with the sample just recorded. Endpoint detection is harder because speakers tend to leave "artifacts" including breathing/sighing, teeth chatters, and echoes.

(2) Pre-Filtering: is accomplished in a variety of ways, depending on other features of the recognition system. The most common methods are the "Bank-of-Filters" method which utilizes a series of audio filters to prepare the sample, and the Linear Predictive Coding method which uses a prediction function to calculate differences (errors). Different forms of spectral analysis are also used.

(3) Framing/Windowing involves separating the sample data into specific sizes. This is often rolled into step 2 or step 4. This step also involves preparing the sample boundaries for analysis (removing edge clicks, etc.)

(4) Additional Filtering is not always present. It is the final preparation for each window before comparison and matching. Often this consists of time alignment and normalization.

There are a huge number of techniques available for

(5), Comparison and Matching. Most involve comparing the current window with known samples. There are methods that use Hidden Markov Models (HMM), frequency analysis, differential analysis, linear algebra techniques/shortcuts, spectral distortion, and time distortion methods. All these methods are used to generate a probability and accuracy match.

(6) Actions can be just about anything the developer wants.

1.4.2. Digital Audio Basics

Audio is inherently an analog phenomenon. Recording a digital sample is done by converting the analog signal from the microphone to an digital signal through the A/D converter in the sound card. When a microphone is operating, sound waves vibrate the magnetic element in the microphone, causing an electrical current to the sound card (think of a speaker working in reverse). Basically, the A/D converter records the value of the electrical voltage at specific intervals.

There are two important factors during this process. First is the "sample rate", or how often to record the voltage values. Second, is the "bits per sample", or how accurate the value is recorded? A third item is the number of channels (mono or stereo), but for most ASR applications mono is sufficient. Most applications use pre-set values for these parameters and users shouldn't change them unless the documentation suggests it. Developers should experiment with different values to determine what works best with their algorithms.

So what is a good sample rate for ASR? Because speech is relatively low bandwidth (mostly between 100Hz-8kHz), 8000 samples/sec (8kHz) is sufficient for most basic ASR. But, some people prefer 16000 samples/sec (16kHz) because it provides more accurate high frequency information. If you have the processing power, use 16kHz. For most ASR applications, sampling rates higher than about 22kHz is a waste.

And what is a good value for "bits per sample"? 8 bits per sample will record values between 0 and 255, which means that the position of the microphone element is in one of 256 positions. 16 bits per sample divides the element position into 65536 possible values. Similar to sample rate, if you have enough processing power and memory, go with 16 bits per sample. For comparison, an audio Compact Disc is encoded with 16 bits per sample at about 44kHz.

The encoding format used should be simple - linear signed or unsigned.

Using a U-Law/A-Law algorithm or some other compression scheme is usually not worth it, as it will cost you in computing power, and not gain you much.

Chapter-2

Previous Work on Non-Native Speech Recognition

2.1 Speech Recognition for Non-Native Speakers

The advancement in communication has made cultural interactions between different parts of the world easier and more frequent. Although languages such as English, Chinese and some of other languages learned by people around the world at schools and universities, with the development of countries in Asia and other continents, more and more people around the world are embracing languages such as Mandarin, Indian, Arabic, Korean etc. Nowadays, apart from the native language, most people can speak at least one foreign language. Furthermore, people are more interested to travel foreign countries for vacation or business. They also often keep some common phrases with the help of the Internet and travel guides to make the communication easier with the locals.

Speech recognition technology has achieved tremendous advancement in the past decades. However, most of the works in speech recognition in the past focus on native speakers. Non-native speech as we see in previous section is different from native speech in term of phonology, pronunciation, vocabulary and grammars. These differences give rise to what is known as accent of a particular group of non-native speakers. What is the difference between non-native speech and dialects? For dialect speakers, there is no transfer of L1 (first language acquisition) like what happens for non-native speakers, because the dialect is often the first language of the speakers. However, variation from the 'standard' language can still happen in the areas of phonology, pronunciation, vocabulary and grammars. However, unlike non-native speech, dialect has commonly accepted phonology, pronunciation, vocabulary and grammar rules or standard among the speakers. Conversely, there is different degree of accent in non-native speech. The difficulty of non-native speech recognition is worsening by the number of languages available, and the limited amount of non-native resources. Three important components in speech recognition system are affected. They are the acoustic model, pronunciation model and

language model [8].

2.2 Non-native Modeling in Speech Recognition

Non-native speech has different characteristics compared to native speech. Hence, specific non-native models tailored to different non-native speaker groups have to be created to achieve better recognition speech performance. However, the lack of non-native resources implies that many of the conventional techniques proposed for native speakers are unable to be used effectively. Over the past decade, creative approaches have been developed for modeling non-native speech under the constraint of resources, by taking advantage of existing resources.

Automatic speech recognition system for non-native speakers has the same architecture as the conventional system. However, it may have an additional component which determines the accent of the speaker either manually or automatically. With this information, matching models which correspond to the accent of the speaker can be selected for decoding the speech.

2.3. General Adaptation Algorithms

General adaptation algorithms have proven to be effective for creating speaker specific model. By using a few utterances from a speaker, a speaker independent model can be adapted. Adaptation algorithms have also been used for adapting the environment conditions. The flexibility of adaptation algorithms, which are capable to work under limited resource constrains makes them an ideal choice to be employed for creating non-native models.

Two of the most popular adaptation algorithms in automatic speech recognition are Maximum Likelihood Linear Regression (MLLR) [9] and Maximum a Posteriori Estimation (MAP) [10, 11] found that adapting the target language acoustic model using MLLR or MAP with native speech of the speakers does not produce any improvement.

Contrary with this result, the acoustic models created from merging of the target language acoustic model with the target language acoustic model adapted with the native language of the speakers, have shown to be beneficial [12]. On the other hand, significant improvement can be obtained by adapting the target language acoustic model using small amount of non-native speech with.. MLLR or MAP. [14, 15] proposes to apply non-native speech with MAP adaptation and Polyphone Decision Trees Specialization (PDTS). PDTS [16] is a decision tree adaptation algorithm which is used to grow specialized non-native branches from a target language trees by pruning to the point where it can be inserted. The adapted tree represents contexts of the non-native speech data. Other general adaptation algorithm which has been tested on non-native speakers is [17]. It is an unsupervised speaker adaptation algorithm using incremental singular value decomposition (SVD) adaptation technique.

2.4 Pronunciation Modeling

Acoustic model defines elementary speech units using fine phonetic features which are related to mouth, tongue, vocal tract and others from speech. Pronunciation modeling on the other hand consists of creating the bigger word or syllable models using the acoustic units defined in acoustic model. Since in most cases, phoneme or phone is the acoustic unit employed in the acoustic model, a pronunciation dictionary (lexicon) can be built from a typical dictionary, since most of them contain descriptions of how words should be pronounced using International Phonetic Alphabet.[18]

If there is no description on the manner of pronunciation, then rules for converting the graphemes to phonemes have to be created. However, this requires an understanding of the language involved. An automatic grapheme to phoneme tool can be created for generating the ‘standard’ pronunciation models using linguistic rules. A manual verification of the generated pronunciation models is often required to correct words which are exception to the rules. In cases where rules for converting graphemes to phonemes do not exist, and there is limited understanding of the language involved, studies found that using

graphemes (context dependent) as the acoustic units for modeling pronunciation model can produce acceptable speech recognition performance, where it is only slightly worse compared to word modeled using phonemes [19, 20]. Note that, this also means that the grapheme units have to be trained in the acoustic model.

2.5 **Pronunciation Dictionary**

Typically a speech recognition system has a pronunciation dictionary which stores at least the base form representations or standard way for pronunciation of words or syllables. Hence, it i also natural to add the surface form or the variant pronunciation which maybe different from the base form into the pronunciation dictionary as another possible realization of the word [18].

One possible way to add pronunciation variants is through listening to the utterances, and writes down their pronunciations. However, this is time consuming and not necessarily produces better result than the automatic approach. A study shows that manual pronunciation modeling do not necessary outperforms automatic approach [21]. Automatic variants generation can be performed using data-driven approaches. The general procedure for finding pronunciation variants is by aligning the hypotheses obtained from non-native speech against the corresponding reference transcriptions to create phone confusion matrix. Pronunciation variants can be observed from the phone confusion matrix. The unobserved variants can be found by generalizing the variants found according to context by using decision trees, and optionally adding the variant probability from the decision trees for each word into the dictionary [22].

The procedure described above requires the usage of non-native speech. However, in many situations non-native speech is hard to acquire. [23] has attempted to generate pronunciation variants using the native phoneme set of the speaker. It is based on the hypothesis of cross-lingual transfer, where non-native speakers substitute target language phonemes with their native phonemes. The procedures are the same as described before for finding pronunciation variants using non-native speech. The only difference is that the target language speech is decoded by a phoneme recognition system of the source language (native language of the speaker). The phone confusions created from the

alignment are then used to create the decision trees. The pronunciation variants can then be subsequently retrieved from the trees. However, the results show that the new dictionary does not produce a significant improvement. Improvement is only obvious when the dictionary is used in conjunction with MLLR applied to the acoustic model with some speech from the speaker [18].

Different non-native speakers have different pronunciation habits which are specific to that group. [24] has proposed an automatic speaker clustering method for non-native speakers based on a list of manually defined vocalic substitutions. A vector is used to represent a dialogue session from a speaker. It contains the number of times a variant appears. Clustering is carried out using model-based k-means and the vectors are randomly assigned to one of the cluster initially. In the subsequent iteration, the vector is assigned to the cluster which gives the highest likelihood to the pronunciation variants observed. This step is executed until it converges. The pronunciation dictionary created for each cluster has shown to be able to reduce the WER of a speech recognition system for each group [18].

Chapter 3

The Sphinx-III Speech Recognition System

3.1. SPHINX ARCHITECTURE

SPHINX-III is a large-vocabulary, speaker-independent, Hidden Markov Model (HMM)-based continuous speech recognition system, like its predecessor, the original SPHINX system. SPHINX was developed at CMU and was one of the first systems to demonstrate the feasibility of accurate, speaker-independent, large-vocabulary continuous speech recognition [25].

The main blocks in Sphinx-III architecture are front-end, decoder and Knowledge base.

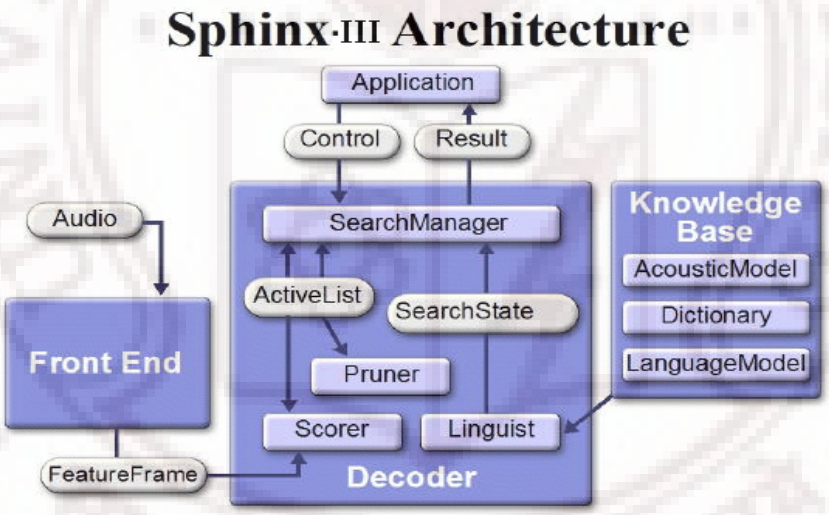

Fig. 3.1 Sphinx-III Architecture

Front End:

It parameterizes an impute signal (e.g. audio) into a sequence of output features. It performs Digital Signal Processing (DSP) on the incoming data.

- ***Feature:*** The outputs of the front end are features, used for decoding in the rest of the system.

***Acoustic Model*:** Contains a representation (often statistical) of a sound, created by training using many acoustic data.

Dictionary: It is responsible for determining how a word is pronounced.

Language Model: It contains a representation (often statistical) of the probability of occurrence of words.

Decoder: It is the main block of the Sphinx-3 system, which performs the bulk of the work. It reads features from the front end, couples this with data from the knowledge base and feedback from the application, and performs a search to determine the most likely sequences of words that could be represented by a series of features.

3.2 THE BASE LINE SPHINX SYSTEM

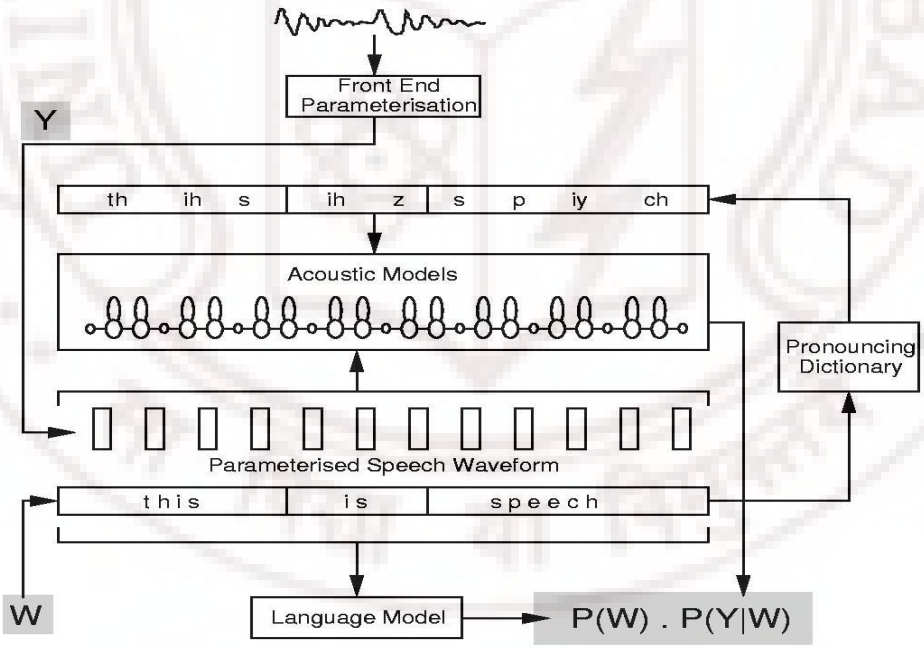

Fig. 3.2 Base line sphinx system

To establish a performance benchmark using standard HMM techniques on

resource database, we began with a baseline HMM system. This system uses standard HMM techniques employed by many other systems [26]. We will show that, using these techniques alone, we can already attain reasonable, accuracies.

A. Speech Processing

The speech is sampled at 16 kHz, and pre-emphasized with a filter whose transform function is $1\text{-}0.97z^{-1}$. The waveform is then blocked into frames. Each frame spans 20 ms, or 320 speech samples. Consecutive frames overlap by 10 ms, or 160 speech samples. Each frame is multiplied by a Hamming window with a width of 20 ms and applied every 10 ms.

From these smoothed speech samples, we computed the LPC coefficients using the autocorrelation method. LPC analysis was performed with order 14. Finally, a set of 12 LPC-derived cepstral coefficients was computed from the LPC coefficients.

The 12 LPC cestrum coefficients for each frame were then vector quantized into one of 256 prototype vectors. These vectors were generated by a variant of the Linde-Buzo-Gray algorithm using Euclidean distance.

B. Phonetic Hidden Markov Models

Hidden Markov Models (HMM) were first described by Baum. Shortly afterwards, they were independently extended to automatic speech recognition by Baker and Jelinek. However, only in the past few years have HMM's become the predominant approach to speech recognition.

HMM's parametric models particularly suitable for describing speech events. The success of HMM's is largely due to the forward-backward re-estimation algorithm, which is a special case of the EM algorithm [27]. Every iteration of the algorithm modifies the parameters to increase the probability of the training data until a local maximum has been reached.

Fig.3.3. The phone HMM used in baseline SPHINX. The label on a transition represents the output pdf to which the transition is tied.

Each phonetic HMM has the topology shown in Fig. 3.1. The three self-loops model three parts of a phone, and the lower transitions explicitly model durations of one, two, or three frames. Instead of assigning a unique output pdf to each transition, each phone is assigned three distributions, representing the beginning, middle, and end of the phone. Each of these three distributions is shared by several transitions.

C. Training

HMM is used to model the specific unit of speech .This Specific unit may be word, a sub word, or a complete form of sentence or paragraph. For a large-vocabulary systems, HMM model the sub word units. These sub word units as phonemes. For the small-vocabulary systems, it is used to model the word itself.

The amount of training data and storage required for word models is (enormous), that why SPHINX-3 based on phonetic models. However it is inadequate to capture the variability of acoustical behaviors for the given phoneme in different contexts, for those particular contexts it will model using the triphones.

Training using sphinx-III:

The training procedure involves optimizing HMM parameters given training data. An iterative procedure, the Baum-Welch or forward backward algorithm is employed to estimate transition probabilities, output distributions, and codebook means and variance under the probabilistic framework.

Recognition using sphinx-III:

For large-vocabulary tasks in the continuous speech recognition, the search algorithm should involved the concepts of acoustic and linguistics in order to maximize the accuracy of the recognition. To apply all these, SPHINX-III uses the Viterbi algorithm.

The SPHINX-III decoder is designed in such a way that it incorporates all the available acoustic and linguistic concepts in several phases. In initial stage, Viterbi beam search produces a single recognition hypothesis as well as a word lattice that includes word segmentations and acoustic scores.

The word lattice is then transformed into a Directed Acyclic Graph (DAG) This DAGs are for quick search for the best hypothesis. DAGs are also used to generate N-best lists for rescoring empirically optimized parameters like the language weight and insertion penalty.

This speech recognition system uses the process of learning the set of sound units which is called as the Training. The process of using the knowledge acquired to deduce the most probable sequence of units in the given signal is termed as the Decoding or simply Recognition.

Chapter 4

Pronunciation Model Adaptation

4.1 Pronunciation variation modeling

For improving the performance of a non-native automatic speech recognition (ASR) system in this, we observe an acoustic and pronunciation model adaptation method for Context Independent (CI) and Context Dependent (CD) pronunciation variability. The adaptation is performed in three steps. First, we do phone recognition to get an n-best list of phoneme sequences and derive pronunciation variant rules by using a decision tree. Second, the pronunciation variant rules are decomposed into CI and CD pronunciation variation on the basis of context dependency. That is, some pronunciation variant rules that are dedicated to the specific phoneme sequences is classified into CI pronunciation variation, but others are classified into CD one. It is assumed here that CI and CD pronunciation variability are invoked by a different pronunciation space from the mother tongue of a non-native speaker and the sandhi rules effect in a context, respectively. Third, the acoustic model adaptation is performed in a state-tying step for the CI pronunciation variability from an indirect data-driven method. In addition, the pronunciation model adaptation is completed by constructing a multiple pronunciation dictionary using the CD pronunciation variability.

By increasing need for non-native automatic speech recognition (ASR), the recognition performance of a non-native ASR system degrades extremely when compared to a system that completely focuses on native speech [28]. There has been considerable research on non-native ASR reported, and they can be categorized into pronunciation modeling, acoustic modeling, and language modeling.First, pronunciation modeling applies the pronunciation variant rules to pronunciation models for non-native speech. For example, several data driven pronunciation modeling methods have been proposed by using a phoneme recognizer and a decision tree. Second, acoustic modeling changes and adapts the acoustic models to include the effect of non-native speech. Third, language modeling handles the grammatical effects or speaking style of nonnative speech [31]. Finally, a hybrid approach combines these three approaches for further

improvement of ASR performance.

In this project, for improving the performance of a non native ASR system we focused on a hybrid approach that combines an acoustic model and a pronunciation model adaptation method. Especially, we analyze the pronunciation variability of non-native speech by using an indirect data-driven method, and adapt acoustic models and pronunciation models depending on the context-dependency of the pronunciation variability. To achieve our task, the pronunciation variability is first investigated with a non-native speech database (DB) in an indirect data-driven method based on a decision tree [32]. That is, we perform phone recognition to obtain an n best phoneme sequences by using a development set, and derive pronunciation variant rules by using a decision tree. Second, pronunciation variability is classified into either CI or CD pronunciation variability on the basis of context dependency. In other words, a pronunciation variant rule that occurs in the specific phoneme sequence is classified as a CI pronunciation variant rule. Otherwise, the pronunciation variant rule is classified as a CD pronunciation variant rule. It is assumed here that CI pronunciation variability reflects a different pronunciation space between a mother tongue and a target language. Conversely, CD pronunciation variability covers sandhi rules effect in a context. Third, an acoustic model adaptation and a pronunciation model adaptation are applied to reduce CI and CD pronunciation variability, respectively.

4.2 Issues in pronunciation variation modeling

This section gives an explanation of the issues that play a role when performing pronunciation variation modeling for ASR. It is intended as an introduction to the main approaches in pronunciation variation modeling:

When we think of pronunciation variation two questions will arise:

1. How we will get the information that is required to describe pronunciation variation?
2. How is this information incorporated in the ASR system?

In the following two sections these questions are addressed. In Section 4.2.1, the approaches to obtaining information are discussed, and in Section 4.2.2 how it is incorporated.

4.2.1 Obtaining information

We can get the Information about pronunciation variation from the data itself or through prior knowledge; also termed the data-derived and the knowledge-based approaches to modeling pronunciation variation. We can classify approaches in which information is derived from phonological knowledge and/or linguistic literature using knowledge-based approaches. Existing dictionaries also fit into this category [33]. In contrast data-derived approaches contains manual transcriptions of the training data are involved to obtain information , or automatic transcriptions are used as the starting point for generating lists of variants (34). Although these methods are useful, to a certain extent, for generating variants, they also have their own drawbacks. In a knowledge-based approach there are differences between theoretical pronunciations and phonetic reality. A drawback of hand-transcribed data is, it laborious, and also expensive. As a consequence, in general there is a little hand-transcribed data. Moreover, manual transcriptions tend to contain an element of quality. Transcriptions written by different transcribers, and even written by the same transcriber, may be differed. The main problem with automatic methods is that phone recognition is not completely reliable either, i.e. it contains errors. This can lead to the generation of erroneous pronunciation variants, which can cause mistakes in the recognition. The options for involving the information into the ASR system are determined by the manner in which the variants are obtained. Using theoretical phonological rules limits the possibilities one has to merely adding variants, whereas a manual or good quality automatic transcription allows for more options. In knowledge based approach obtaining pronunciation variants is investigated. In addition to the knowledge-based approach, a data-driven approach is studied. In this study, the two approaches will produce two different lexica. These different lexica are compared.

4.2.2 Incorporating the information in ASR

After getting the pronunciation variants, the next question which can be answered is how we can put the information into the ASR system. There are different levels at which this

problem can be addressed. In the following sections, pronunciation modeling at each of these levels is discussed. First, adding variants to the lexicon is addressed. This is followed by a discussion of lexical confusability, which is an issue that is closely linked to modeling pronunciation variation in the lexicon. Next, the role of forced alignment in pronunciation modeling is explained, before discussing how pronunciation variation can be incorporated in the acoustic models and how the language models are employed in pronunciation modeling. The final issue that is addressed in this section is the use of articulatory-acoustic features in pronunciation modeling.

Adding variants to the lexicon

When speech recognizers use lexicon, at the lexicon level pronunciation variation is often modeled. Variation that occurs within a word can be dealt with in the lexicon by adding variants of the words to the lexicon. Variants of a single word are different phonetic transcriptions of one and the same word; i.e., substitutions, insertions and deletions of phones in relation to the base-form variant. This type of variation is within-word variation. However, in continuous speech a lot of variation occurs over word boundaries. This is referred to as cross-word variation. Cross-word variation can, to a certain extent, be dealt with in the lexicon by adding sequences of words which are treated as one entity, i.e., multi-words. The variation in pronunciation that occurs due to cross-word variation is modeled by adding variants of the multi-words to the lexicon. An alternative method for modeling cross-word variation in the lexicon is described below: the cross-word variants are coded in the lexicon in such a way that during recognition only compatible variants can be interconnected. In most approaches, the lexicon is static, in the sense that it is not altered during the recognition phase. However, there have also been a few studies in which the lexicon was dynamically altered. For instance, (34) showed that improvements can be found by a dynamic rescoring of n -best lists using a word-based decision tree dictionary. A two-pass approach to modeling pronunciation variation is used in which the recognition lexicon is dynamically adjusted depending on the utterance which is being recognized.

Lexical confusability

By adding variants to the lexicon there is a chance to increase one of the transcriptions of a word will match the corresponding speech signal. However with the addition of variants, the lexical confusability increases. If we simply add variants to the lexicon it does not lead to improvements, and in many cases WER becomes worse. Predicting which pronunciations will be the correct ones for recognition goes hand in hand with dealing with lexical confusability. The dynamic lexica described in the previous section were developed with exactly this problem in mind: dynamically adjusting the lexicon for the utterance that is being recognized should circumvent most of the lexical confusability that is otherwise introduced. Confusability in data-derived approaches is often introduced by errors in phonetic transcriptions. These phonetic transcriptions are used as the information source from which new variants are derived. Consequently, incorrect variants may be created. One commonly used procedure to alleviate this problem is to smooth the phonetic transcriptions by using decision trees (D-trees) to limit the observed pronunciation variation. In a D-tree approach, an alignment between a canonical transcription and an alternative transcription is used as the input to build the D-trees. The context used for decision making can include anything from mere left and right neighboring phone identity to information such as lexical stress, position of a phone within the syllable, or finer-grained feature information. Using the D-trees, finite state grammars (FSGs) are generated for the words in the training material. These FSGs are realigned with the acoustic signal. The resulting phone transcriptions can be used to generate a new lexicon. In this way, mistakes in the transcriptions can be filtered out.

Other approaches struggle confusability by rejecting variants that are highly confusable on the basis of phoneme confusability matrices. A maximum likelihood criterion was used to decide which variants to include in the lexicon. A confusability metric was introduced which was used to discard highly confusable variants. Measures such as absolute or relative frequency of occurrence have also been employed to select rules or variants. Finally, confidence measures have been employed to combat

confusability by augmenting a lexicon with variants using a confidence-based evaluation of potential variants.

In the study of within-word and cross-word variation, lexical confusability is not addressed as such, but an analysis is carried out in an attempt to find tools which can be used to decide which variants to add to a lexicon and which ones to leave out. The D-tree approach is employed to smooth automatically obtained phone transcriptions. In addition, the confusability metric introduced in [34] is further examined as a tool for discarding highly confusable variants.

Forced recognition

Forced recognition can be used in various ways in pronunciation modeling. The main goal of using forced alignment in pronunciation modeling is to "clean up" the transcriptions in the training data, i.e., to obtain a more precise transcription given multiple transcriptions for the words in the lexicon. In the data-derived decision tree approach forced alignment is used to align the FSGs (Finite State Grammars) with the training data, to subsequently select variants on the basis of the output of the alignment. The alignments are also used to obtain priors for the pronunciation variants in the lexicon, or to predict the probabilities in the language model. Finally, the transcriptions can also be involved to retrain the acoustic models.

Variant probabilities

By placing pronunciation variation in the language model, it can be carried out by predicting the probabilities of the variants instead of the probabilities of the words. This is only possible if the pronunciation variants are transcribed in the training data, then the language models are trained on this data. In the intermediate level of modeling pronunciation Variation in the language model is possible in the form of word classes. In particular, this approach is taken to deal with processes of cross-word variation.

Data-driven pronunciation variability analysis

To obtain pronunciation variability for non-native speech, an indirect data-driven method based on a decision tree is used, as shown in Fig. 1. First, each utterance in the development set of non-native

speech is recognized by using a phoneme recognizer. The recognized n-best phoneme sequences are aligned using a dynamic programming algorithm with a reference phoneme sequence of the utterance, where the reference phoneme sequence is automatically obtained by using a CMU pronunciation dictionary [14] for the word of each utterance.

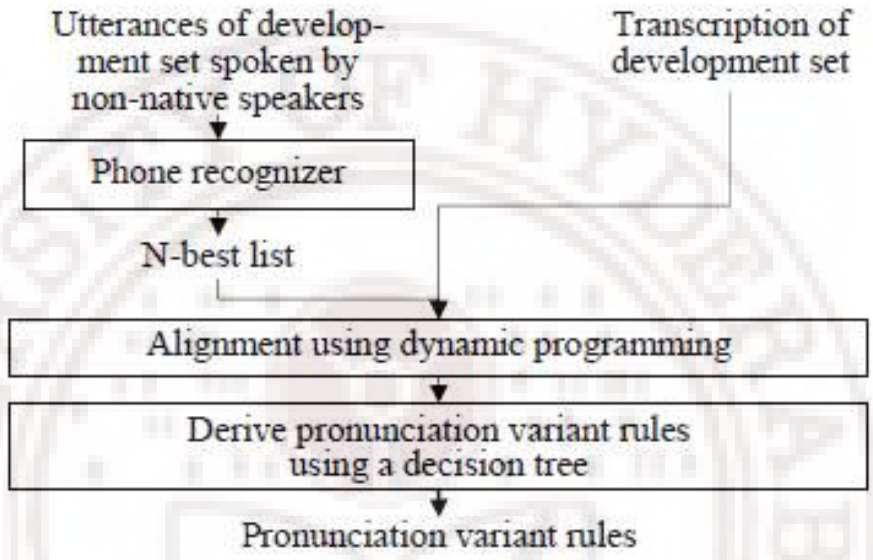

Figure 4.1 : Procedure for obtaining pronunciation variability from non-native speech by using an indirect data-driven method based on a decision tree.

Here we can explain how to classify a pronunciation variability into CI and CD pronunciation variability. CD pronunciation variability is observed only in the limited and specific phoneme sequences such as allophones for a specific phoneme. For example, let us assume that a speaker utters 'this spring.' The pronunciation would be /DH IH S P R IH NG/ instead of /DH IH S S P R IH NG/ because the final phoneme /S/ of 'since' and the initial phoneme /S/ of 'spring' are adjacent. Except for these pronunciation variants due to sandhi rule effects, the phoneme /S/ must be pronounced as /S/.

Chapter 5

Results

Speakers	Non-Native speakers (% of accuracy)	Non-Native (by adding confusion pairs to dictionary) (% of accurac
1	21.45	47.403
2	25.67	49.246
3	28.243	55.671
4	22.90	45.43
5	30.28	57.421

Table 1. Word Recognition Accuracy of Non Native Speakers

Speakers	native speakers (% of accuracy)
1	55.654
2	57.654
3	60.980
4	51.253
5	58.126

Table 2. Word Recognition Accuracy of Native Speakers

Chapter 6

Conclusion

This project mainly concentrates on context dependent(CD) words. speech samples are collected from the native as well as non-native speakers for these CD words.The accuracy for native speech is more than the non-native speech is obtained. For increasing the accuracy of the non-native speech the confusion pairs which were got are added to the lexicon.The accuracy has improved for non-native speech, but it could not reach the accuracy of native speech.

Future work:

For improving the non-native speech recognition accuracy the size of the lexicon need to be increased. And also the no. of speakers need to be increased.

References

[1] B. T. Lowerre, “The HARPY speech recognition aystcm,” Ph.D. dissertation, Comput. Sci. Dep., Carnegie Mellon Univ.. Apr. 1976.

[2] J . G. Wilpon, L. R. Rabiner, and A. Bergh, “Speaker-independent isolated word recognition using a 129-word airline vocabulary,” J. Acoust. Soc. Atnrr. vol. 72. no. 2, pp. 390-396, Aug. 1982.

[3] A. E. Rosenberg. L. R. Rabiner, J. Wilpon. and D. Kahn, “Demisyllable-based isolated word recognition system,” fEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, pp. 713-726, June1983.

[4] Y. L. Chow, R. Schwartz, S. Roucos, 0. Kimball, P. Price, F. Kubala, M. Dunham, M. Krasner, and J. Makhoul, “The role of worddependent coarticulatory effects in a phoneme-based speech recognition system,” presented at the IEEE Int. Conf. Acoust.. Speech, Signal Processing, Apr. 1986.

[5] D. B. Paul and E. A. Martin, “Speaker stress-resistant continuous speech recognition,” presented at the IEEE Int. Conf. Acoust., Speech. Signal Processing, Apr. 1988. (141 P. J. Price, W. Fisher, J . Bernstein, and D. Pallett, “A database for continuous speech recognition in a 1000-word domain,” presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing. Apr. 1988.

[6] “An Overview of the SPHINX Speech Recognition System” Kai-Fu Lee, Member, IEEE, Hsiao-Wuen Hon, And Raj Reddy, Fellow. IEEE, VOL. 38. No. I. January 1990

[7] Lawrence R. Rabiner, Fellow, IEEE “*A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*”, 1989.

[8].Goronzy, S., 2002, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition: Berlin, Springer Verlag.

[9] Leggetter, C. J., and Woodland, P. C., 1995, Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models: Computer Speech and Language, vol. 9,
p. 171-185.

[10] Gauvain, J.-L., and Lee, C.-H., 1994, Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains: IEEE Transactions on Speech and Audio
Processing, vol. 2, p. 291-298.

[11].Tomokiyo, L. M., and Waibel, A., 2001, Adaptation Methods for Non-Native Speech, Workshop on Multilinguality in Spoken Language Processing: Aalborg.

[12].Bartkova, K., and Jouvet, D., 2004, Multiple Models for Improved Speech Recognition for Non-Native Speakers, SPECOM'04: St. Petersburg.
[13].Tomokiyo, L. M., and Waibel, A., 2001, Adaptation Methods for Non-Native Speech, Workshop on Multilinguality in Spoken Language Processing: Aalborg.

[14].Wang, Z., and Schultz, T., 2003, Non-Native Spontaneous Speech Recognition through Polyphone Decision Tree Specialization, Eurospeech'03: Geneva, Switzerland, p. 1449-1452.

[15].Wang, Z., Schultz, T., and Waibel, A., 2003, Comparison of Acoustic Model Adaptation Techniques on Non-native Speech, ICASSP'03: Hong Kong, China, p. 540-543.

[16].Schultz, T., and Waibel, A., 2000, Polyphone Decision Tree Specialization for Language Adaptation, ICASSP'00: Istanbul, p. 1707-1710.

[17].Deng, Y., X., L., Kwan, C., Xu, R., Raj, B., and Williamson, D., 2006, An Integrated Approach to Improve Speech Recognition Rate for Non-Natives Speakers, ICSLP'06: Pittsburgh, p. 1734-1737.

[18].Strik, H., and Cucchiarini, C., 1998, Modeling pronunciation variation for ASR: overview and comparison of methods, ESCA workshop 'Modeling Pronunciation Variation for Automatic Speech Recognition: Rolduc, p. 137-144.

[19].Tan Tien Ping, le 3 juillet 2008, Automatic Speech Recognition for Non- Native Speakers.

[20].Strik, H., and Cucchiarini, C., 1999, Modeling Pronunciation Variation for ASR: A Survey of the Literature: Speech Communication, vol. 29, p. 225-246.

[21].Goronzy, S., Kompe, R., and Rapp, S., 2001, Generating Non- Native Pronunciation Variants for Lexicon Adaptation, ISCA'01: Sophia Antipolis, France, p. 143-146.

[22].Humpries, J. J., and Woodland, P. C., 1997, Using Accent-Specific Pronunciation Modelling for Improved Large Vocabulary Continuous Speech Recognition, Eurospeech'97: Rhodes, Greece, p. 2367-2370.

[23].Goronzy, S., 2002, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition: Berlin, Springer Verlag.

[24].Raux, A., 2004, Automated Lexical Adaptation and Speaker Clustering based on Pronunciation Habits for Non-Native Speech Recognition, ICSLP'04: Jeju Island, Korea, p. 613-616.

[25] H. Satori, M. Harti, and N. Chenfour., "Introduction to Arabic Speech Recognition Using CMUSphinx System", UFR Informatique et Nouvelles Technologies d'Information et de Communication B.P. 1796, Dhar Mehraz Fs Morocco.

[26]Claudio Becchetti & Eucio Prina Ricotti, "speech recognition Theory and C++ implementation", textbook", 2000.

[27] L. E. Baum, "An inequality and associated maximization technique in ststistical estimation of probabilistic functions of Markov pr cesses." Inequalities, vol. 3, pp, 1-8, 1972

[28] D. V. Compernolle, "Recognizing speech of goats, wolves, sheep and non-natives," *Speech Communication*, vol. 35, no. 1, pp. 71-79, Aug. 2001.

[29] R. Gruhn, K. Markov, and S. Nakamura, "A statistical lexicon for nonnative speech recognition," in *Proc. ICSLP*, Jeju Island, Korea, pp. 1497-1500, Oct. 2004.

[30] A. Raux, "Automated lexical adaptation and speaker clustering based on pronunciation habits for non-native speech recognition," in *Proc. ICSLP*, Jeju Island, Korea, pp. 616-616, Oct. 2004.

[31]J. Bellegarda, "An overview of statistical language model adaptation," in *Proc. ISCA Workshop on Adaptation Methods for Speech Recognition*, Sophia-Antipolis, France, pp. 165–174, Aug. 2001.

[32]M. Kim, Y. R. Oh, and H. K. Kim, "Non-native pronunciation variation modeling using an indirect data driven method," in *Proc. ASRU*, Kyoto, Japan, pp. 231-236, Dec. 2007.

[33] Lamel, L. and G. Adda (1996). On designing pronunciation lexicons for large vocabulary, continuous speech recognition. In Proc. of ICSLP '96 , Philadelphia, PA., pp. 6–9.

[34] Fosler-Lussier, E. (1999). Dynamic Pronunciation Models for Automatic Speech Recognition. Ph. D. thesis, University of California, Berkeley, CA.