ISSN No. 0976-5697

Volume 2, No. 3, May-June 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

# Projected Features for Hindi Speech Recognition System

R. K. Aggarwal*
Department Of Computer Engineering
National Institute of Technology
Kurukshetra, INDIA
rka15969@gmail.com

M. Dave
Department Of Computer Engineering
National Institute of Technology
Kurukshetra, INDIA
mdave67@gmail.com

*Abstract:* Automatic speech recognition (ASR) is a technology that allows a computer to identify the words uttered by a person using a microphone or telephone. The processing required is divided into two parts: the signal processing front end and statistical framework of hidden Markov model (HMM) at back-end for pattern classification. This paper presents a comparative study of different types of feature reduction techniques in the context of Hindi language. Experimental results show a significant improvement in ASR performance by using extended MF-PLP (PLP derived from Mel scale filter bank) feature extraction technique at front-end with the help of Hetroscedastic linear discriminant analysis (HLDA) projection scheme. All the investigations are based on the experiments conducted in typical field conditions using standard close talking microphone.

***Keywords:*** ASR; feature reduction; Hindi; PCA; LDA; HLDA

## I. Introduction

Automatic speech recognition (ASR) involves in the recovery of information into textual form from the physical speech signals. It has wide variety of application such as interactive voice response system (IVRS), dictation writing and voice controlled house hold appliances. The major application of ASR is to build speech based interfaces for human computer interaction, providing a convenient access to the information technology and its applications for illiterate and physically challenged persons [1].

The two main components, normally used in ASR, are signal processing component at front-end and pattern matching component at back-end. Speech signal is converted into discrete sequence of feature vectors, which is assumed to contain only that information about given utterance that is important for its correct recognition. These feature vectors are decoded into linguistic units like word, syllable, and phones using hidden Markov models [2].

In this paper we review and present a comparative study of various reduction techniques proposed so far such as principal component analysis (PCA), linear discriminant analysis (LDA), and Hetroscedastic linear discriminant analysis (HLDA) with their merits and demerits. These methods are applied at the front-end to obtain the uncorrelated and reduced size feature vector. At front-end extended MF-PLP (PLP Derived from Mel scale Filter Bank) is used for feature extraction and these features are evaluated using the statistical techniques of acoustic models (HMM-GMM). All the experiments are conducted in the context of Hindi languages. The rest of the paper is organized as follows: Section 2 describes the architecture and working of ASR. Feature extraction techniques are given in section 3. Section 4 describes the feature reduction methods normally used in pattern recognition. The techniques used for pattern classification like Gaussian mixture models are discussed in section 5. In section 6, an experimental comparison of ASR performance with various reduction methods is presented. Finally, the paper concludes with a brief discussion of the experimental results in section 7.

## II. Working of ASR

The main modules of ASR are preprocessing and feature extraction, acoustic and language model generation, and decoding (classification) as shown in Figure 1. Preprocessing covers the signal acquisitions, conditioning and segmentation. Speech signal is captured through a close talking microphone, i.e., 10 cm distance between lips and transducer. Next step involves analog to digital conversion as well as digital filtering to emphasize the important components of the speech. This signal is blocked into overlapped frames (frame size 20-40 ms and frame shift 10 ms) using a Hamming window. Further speech activity detection is performed as a front end step having a positive impact on the ASR system in terms of both CPU usage and ASR accuracy. This is due to the fact that the decoder is not required to operate on non speech segment, thus reducing the processing effort and word insertion error rate. These frames are further processed using the signal processing techniques like Mel-frequency cepstrum coefficient (MFCC), perceptual linear prediction (PLP), MF-PLP and wavelets [3].

At back end, speech database and text corpus are required to generate acoustic and language models respectively which are used as knowledge sources during decoding. On the basis of generated models, ASR uses hidden Markov model framework to provide acoustic match score between observations and proposed strings of words. For large vocabulary task, it is impractical to create a separate acoustic model for every possible word since it requires too much training data to measure the variability in every possible context. A word model is formed by concatenating the models for the constituent subword sounds in the word, as defined in a word lexicon or dictionary. The main role of acoustic model is the mapping from each sub word unit to acoustic observations. Popular subword units being used are context independent phones, syllables and triphones. In language model rules are introduced to follow the linguistic restrictions present in the language and to allow redemption of possible invalid phoneme sequences [4].

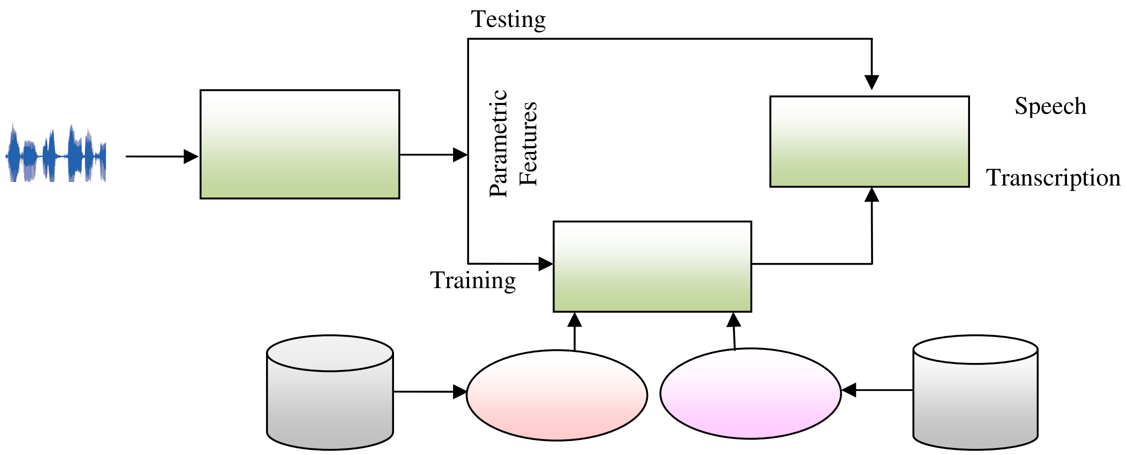


**Figure 1.** Components of ASR

## III. Feature Extraction

The parameters representations of speech signals may be divided into two groups: those based on linear prediction spectrum and based on Fourier spectrum. In second category filter bank energies have been used for spectral analysis in ASR by accumulating the energy in each band over short segments of time. Filter bank energies are determined by applying filters directly on the DFT-derived power spectrum of the signal. The power in each band is calculated as the weighted sum of adjacent power values. The filter bank representation allows incorporation of perceptually based frequency scales such as Mel-warped filter bank and Bark-warped filter bank [5]. The popular techniques based on filter bank approach are MFCC, PLP, and MF-PLP.

It is common to append an energy coefficient to the cepstrum feature vector. The energy is computed as the logarithm of the accumulated frame energy:

$$E_{t_0} = log\,(\textstyle\sum_{n=1}^{N} x^2(n,t_0\,)). \qquad (1)$$

Where $x\,(n)$ is the input speech signal and $N$ corresponds to size of the Hamming window. Energy is useful since differences in energy are seen among different phonemes.

MFCC proposed by Davis and Mermelstein 1980 [6] includes computing the cosine transform of the real logarithm of the short-time power spectrum on a Mel warped frequency scale. Here a speech spectrum passes through a filter bank of Mel-spaced triangular filters, and the filter output energies are log-compressed and transformed to the cepstral domain by DCT. Normally first 13 coefficients are enough for the representation of the signal. This cepstral as such along with their first and second order derivatives are used as features for recognition. In PLP, the spectrum is multiplied by a mathematical curve modeling the ear's behavior in judging loudness as a function of frequency. The output is then raised to the power 0.33 to simulate the power law of hearing [7]. In MF-PLP (PLP Derived from Mel scale Filter Bank), the MFCC and PLP techniques are merged into one algorithm [8]. The first steps until generating the output of the Mel scale triangular filter bank are taken from the MFCC algorithm. The only difference here is that the filter bank is applied to the power spectrum instead of the magnitude spectrum. The last steps generating the cepstrum coefficients are taken from the PLP algorithm. The steps followed for MF-PLP extraction are given below and shown in Figure 2.

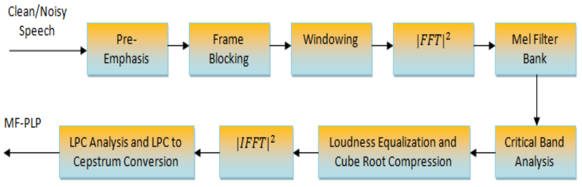


**Figure 2:** MF-PLP Extraction Modeling

- Compute power spectrum of windowed speech.
- Perform grouping to 21 critical bands in Bark scale especially for 16 kHz sampling frequency.
- Perform loudness equalization and cube root compression to simulate the power law of hearing.
- Perform Inverse Fast Fourier Transform (IFFT).
- Derive LP coefficients by Levinson-Durbin procedure [9] and convert them into cepstral coefficients.

## IV. Feature Reduction Techniques

An important task for any pattern recognition problem is to find a good feature space, which should be both compact and contain the richest possible discriminant information. Feature dimensions which contain less discriminant information should be discarded because their existence not only slows down the classification process but also degrades the performance in many situations. This step is aimed at incorporating the techniques which project the features into low dimensional subspace, while preserving discriminative information. To derive such a good feature subspace is through finding a linear transformation$\boldsymbol{\theta}_T$from the original feature space $\Re^N$ to a new low dimensional one $\Re^M$such that:

$$F_M(T) = \theta_T F_N(T) \qquad (2)$$

where $F_M(T)$is the feature vector in the transformed feature space and$F_N(T)$ is the feature vector in original feature space and $M$ is the target feature space's dimensionality

The techniques mainly used for feature decorrelation and dimensionality reduction are principal component analysis (PCA), linear discriminant analysis (LDA), Heteroscedastic linear discriminant analysis (HLDA) [10]

### *A. Principal Component Analysis*

PCA [11] defines the orthogonal linear transforms using a matrix $P$, where target feature vector$\bar{F}$is obtained by:

$$\bar{F} = P.F \tag{3}$$

Transform matrix $P$ corresponds to the Eigen vectors of covariance matrix of the original feature space arranged on the basis of the Eigen values. The first row of matrix $P$ (called first base vector, $p_1$), shows the direction of the largest variability in $n$-dimensional space of feature vectors. The second row shows a direction perpendicular to direction given by the first row with the second largest variability and so on.

From the feature space, covariance matrix $cov$ is derived as:

$$cov = \frac{1}{N}\sum_{i=1.......N}(x_1 - \bar{x})(x_1 - \bar{x}) \tag{4}$$

where $N$is the number of training feature vectors, $x_1$is the $i^{th}$ training feature vector, $\bar{x}$is the mean vector calculated as:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{5}$$

Since the covariance matrix $cov$ is a square, one can calculate the eigenvectors and eigen values for this matrix. The $i^{th}$ base vector ( $i^{th}$ row of matrix $P$) of PCA transformation $p_i$, is given by the eigen vector corresponding to $i^{th}$ largest eigen value. In order to reduce the dimension, only first $M$ base vectors such that $M < N$ which preserves the most variability of feature space are selected.

### *B. Linear discriminant analysis*

The other dimensionality reduction method widely used in pattern recognition is linear discriminant analysis (LDA) [12], where the optimization criterion is to maximize the Fisher ratio value in the transformed feature space. In LDA the objective is to increase the ratio of the between class variance to the average within class variance for each dimension. It assumes that features belonging to each particular class obey Gaussian distribution with the same covariance matrix for all classes. Base vectors of LDA transformation matrix are calculated by the eigenvectors of the product of across-class covariance matrix and inverse of within-class covariance matrix i.e., $cov_{AC} \times cov_{WC}^{-1}$. Across-class covariance matrix $cov_{AC}$ represents the wanted variability in data and computed as:

$$cov_{AC} = \frac{1}{N}\sum_{k=1}^{K} N_k.(\bar{x}^k - \bar{x})\,(x^k - \bar{x})^T \tag{6}$$

where $K$ is the number of classes, $N_k$is number of training vectors belonging to class and $\bar{x}^k$is the mean vector for class $k$ defined as

$$\bar{x}^k = \frac{1}{N}\sum_{i=1}^{K} x_i^k \tag{7}$$

Within-class covariance matrix $cov_{WC}$ represents the unwanted variability in data and is given as weighted average of covariance matrix of all classes.

$$cov_{WC} = \frac{1}{N}\sum_{k=1}^{K} N_k.cov^k \tag{8}$$

where $cov^k$ is covariance matrix for class $k$. In order to achieve feature reduction, LDA chooses first $M$ base vectors that retain the maximum amount of class discrimination information.

### *C. Heteroscedastic Linear Discriminant Analysis*

HLDA, a generalization of LDA and first proposed by N. Kumar [13], assumes that original *N* dimensional feature space can be split into two statistically independent subspaces, one containing the necessary information and another with nuisance information. Here classes obey Gaussian distribution with different covariance for each class and maximizes the likelihood of all the training data in the transformed space. Each training data contributes equally in the transformed space as well as to the objective functions. The main difference between LDA and HLDA is that full covariance matrix statistics for each component are required to estimate an HLDA transform, whereas only the average within and between class covariance matrices are required for LDA. Thus HLDA, a refinement of LDA, uses the actual class covariance matrices rather than using the averages.

## V. ACOUSTIC MODELING TECHNIQUES

Among the various acoustic models, HMM is so far the most widely used technique due to its efficient algorithm for training and recognition. It is a statistical model for an ordered sequence of symbols, acting as a stochastic finite state machine which is assumed to be built up from a finite set of possible states [14]. An HMM is the same as a Markov chain, except for one important difference: each state emits output symbol based on some emission probability. Instead of associating a single output symbol per state in an HMM, all symbols are possible at each state, each with its own probability.

In ASR, only forward transition of states is allowed in a left to right way as shown in Figure 3 [15]. The Markov chain is specified in terms of an initial state distribution vector $\pi = \{\pi_1, \pi_2, \ldots, \pi_n\}$ and a state transition matrix $A = [a_{ij}]\ 1 \le i, j \le k$. Here $\pi_i$ is the probability of $s_i$ to be the initial state and $a_{ij}$ has the property of

$$a_{ij} = 0\ where\ j < i \tag{9}$$

Consider a first order $k$ state ( $s_1, s_2, \ldots, s_k$) Markov chain the initial state probability are defined as

$$\pi_i = \begin{cases} 0 & i \neq 1 \\ 1 & i = 1 \end{cases} \tag{10}$$

The Basic HMM theory was published in series of classical papers by Baum and his colleague [16, 17] and details of HMM for ASR can be found in many references [18, 4].

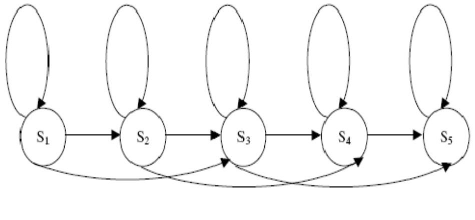


**Fig.3.** 5-state hidden Markov model

In GMM each of the HMM states are associated with a multivariate Gaussian probability distribution, defined as:

$$N(x; \boldsymbol{\mu}, \textstyle\sum) = \frac{1}{(2\pi)^{n/2}|\sum|^{1/2}} exp\left[-\frac{1}{2}(x - \boldsymbol{\mu})^t \textstyle\sum^{-\mathbf{1}}(x - \boldsymbol{\mu})\right] \tag{11}$$

Where $\boldsymbol{\mu}$ is the $n$ dimensional mean vector, $\sum$ is the $n \times n$ covariance matrix, and $|\sum|$ is the determinant of covariance matrix $\sum$. In the mixtures of Gaussian density, the probability density for observable data *x* is the weighted sum of each Gaussian component:

$$p(x|\lambda) = \sum_{m=1}^{M} c_m N_m(x|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \tag{12}$$

Where $c_m$, the mixture weight associated with $m^{th}$ Gaussian component is subject to the following constraint $c_m \geq 0$ and $\sum_{m=1}^{M} c_m = 1$, and $p(x|\lambda)$ is an utterance level score of $x$ given the model $\boldsymbol{\lambda}$ [14].

The complete Gaussian mixture model is parameterized by the mean vectors, covariance matrices and mixture weights from all component densities.

## VI. Experimental Results

The speech signal is sampled at 16 kHz using 16 bits quantization and pre emphasized using a first order filter with a coefficient of 0.97. The samples are blocked into the overlapping frames of 30 ms in duration and updated at 10 ms intervals. These frames are processed by filter bank approach for feature extraction. The outputs of the filter bank are then transformed to cepstral coefficients, where only the first 12 coefficients are retained as a part of feature vector [19]. The complete feature vector consist of 52 values including the 12 cepstral coefficients with one energy, 13 delta coefficients, 13 delta delta coefficients and 13 triple delta coefficients. The 52 dimensional feature vector is reduced to 39 dimension by using any of the projection schemes (reduction techniques) as shown in Figure 4.

Many public domain software tools are available for the research work in the field of ASR such as Sphinx from Carnegie Mellon University [20], hidden Markov model toolkit (HTK) from Cambridge University [21] and LVCSR engine Julius from Japan [22]. We have used HTK-3.4.1 in LINUX environment for our experimental work. Further the experiment consists of an evaluation of the system using the room condition and standard speech capturing hardware such as sound card and a head set microphone.

3-states along with dummy (non-emitting) initial and final nodes were used for each phonetic transcription in HMM topology. The choice of the number of model states is suggested by the observation that each phoneme instance could be divided into three quasi-stationary parts: an initial part, a central part and a final part.

The experiment was performed on a set of speech data consisting of four hundred words of Hindi language recording by 10 male 10 female speakers. Testing of randomly chosen 50 sentences spoken by different speakers is made and recognition rate (i.e. accuracy) is calculated as Accuracy (%) = 100-WER(%).

Word error rate runs three types of errors: insertion, deletion and substitution errors. If there are N words in the reference transcript, and the ASR output has S substitutions, D deletions and I insertions, then

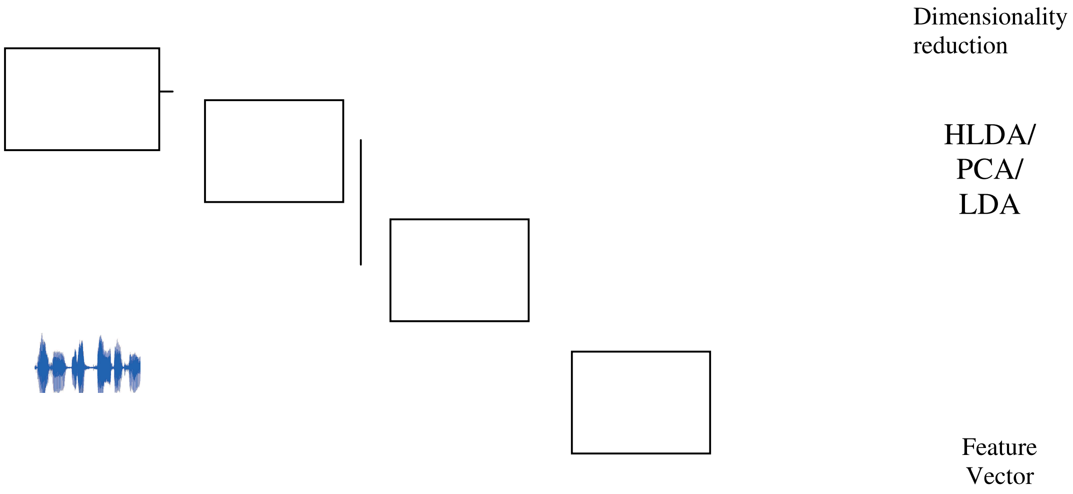
$$\mathrm{WER} = 100 * \frac{\mathrm{S+D+I}}{\mathrm{N}} \% \qquad (13)$$

### A. *Experiments with Projection Schemes*

In this experiment at front-end, the feature vectors of MF-PLP were appended up to triple deltas (13 static + 13Δ + 13ΔΔ + 13ΔΔΔ) and these 52 dimensional feature vectors were reduced to the standard 39 values, using the projection schemes. At back-end, classical HMM was used with two modeling units, whole word and triphone. A triphone HMM is applied to model the acoustic characteristic of a phoneme in the context of a specified preceding and a specified succeeding phoneme. For example a triphone HMM "h-I-s" may be generated for the vowel "i" in the context of a preceding "h" and a succeeding "s".

Results were compared in case of three projection schemes HLDA, LDA and PCA as shown in Figure 5. HLDA based reduction shows maximum accuracy in this scenario. The difference between the performance of HLDA and LDA is less. We can also use LDA if less computation is required to make the application fit in real time environment or in embedded system.

**Figure 4**. Extended Feature Vector Generation



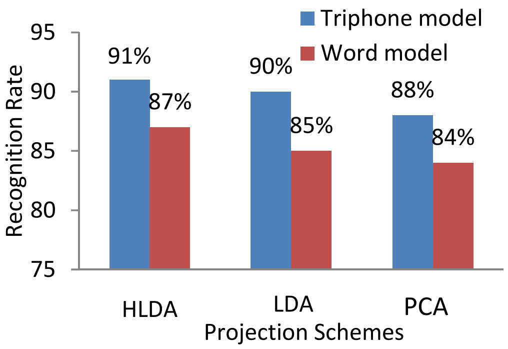


**Figure 5.** Accuracy versus Projection Schemes

### *B. Experiment with Gaussian Mixtures*

Thirteen extra triple delta features are added in standard 39 MFCC features forming a feature vector of 52 values. These 52 values are then reduced to 39 by applying projection schemes. At back-end, classical HMM was used with different Gaussian mixtures, 1 mix, 4 mix, 8 mix, and 16 mix. Maximum accuracy was observed for HLDA, when 8 Gaussian mixtures were used for experiments as shown in Table 1.

**Table 1.** Accuracy for Mixtures and Projection Schemes

| Projected Features | Accuracy for different mixtures | | | |
|---|---|---|---|---|
| | 1 mix | 4 mix | 8 mix | 16 mix |
| PCA | 76 | 85 | 88 | 86 |
| LDA | 77 | 87 | 90 | 89 |
| HLDA | 78 | 87 | 91 | 90 |
| Std. Feature | 73 | 82 | 86 | 84 |

## VII. Conclusion

Recognizing and understanding of speech is the basic for facing a broad class of challenging problems related with natural language conversational interface. For the design and development of efficient and accurate ASR, discriminant and compact features play an important role. In the paper we reviewed the projection schemes and applied them for feature reduction purpose. Experimental results showed that HLDA performs best in comparison to others. Further the extended features (includes triple deltas) showed 2 to 5% more accuracy in comparison to standard features.

## VIII. References

[1] Douglas O'Shaughnessy, "Interacting with Computers by Voice-Automatic Speech Recognitions and Synthesis," Proceedings of the IEEE, vol. 91, no. 9, pp. 1272-1305, 2003.

[2] R. K. Aggarwal and M. Dave, "Implementing a Speech Recognition System Interface for Indian Languages," Proc. IJCNLP, Workshop on NLP for Less Privileged Languages, IIIT Hyderabad, 2008, pp. 105-112.

[3] M. A. Anusuya and S. K. Katti, "Front End Analysis of Speech Recognition: A Review"**,** International Journal of Speech Technology, vol. 14(2), pp. 99-145, 2011.

.

[4] Frederick Jelinek, Statistical Methods for Speech Recognition, MIT press, 1997

[5] L.R. Rabiner and R.W. Schafer, "Introduction to Digital Speech Processing," Foundations and Trends in Signal Processing, vol. 1, Issue 1-2, 2007.

[6] S. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," IEEE Transactions on Acoustics, Speech and Signal Processing, 28, pp. 357-366, 1980.

[7] H. Hermansky, "Perceptually predictive (PLP) Analysis of Speech," Journal of Acoustic Society of America, 87, 1738-1752, 1990.

[8] P. Woodland, M. Gales, D. Pye and S. Young, "Broadcast News Transcription using HTK," Proceeding ICASSP, vol. 2, 719-722, 1997.

[9] J.D. Markel and A.H. Gray, "Linear Prediction of Speech," Springer-Verlag, 1976.

[10] Mark Gales and Steve Young, "The Application of Hidden Markov Model in Speech Recognition," Foundations and Trends in Signal Processing, vol. 1, Issue 3, 2007.

[11] O. Duda and P.E. Hart, Pattern Classification and Scene Analysis. Wiley, New York, 1973.

[12] R. Haeb-Umbach and H. Ney, "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition," Proc. of ICASSP, pp13-16, 1992.

[13] N. Kumar and A. G. Andreou "Heteroscedastic Discriminant Analysis and Reduced Rank HMMs for Improved Speech Recognition," Speech Communication, Vol.26, pp. 283-297, 1998.

[14] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. of the IEEE,* 77(2), pp. 257- 286*,* 1989.

[15] X.D. Huang, Y. Ariki, and M.A. Jack, Hidden Markov Models for Speech Recognition. Edinburg University Press, 1990.

[16] L.E. Baum, and J. A Eagon, "An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology," Bulletin of American Mathematical Society, 73, 360-363, 1967.

[17] L. R. Welch, "HMMs and the Baum-Welch algorithms," IEEE Information Theory Society Newsletter, 53(4), 10-13, 2003.

[18] X. Huang, A. Acero and H. W. Hon, Spoken Language Processing: A Guide to Theory Algorithm and System Development. Prentice Hall-PTR, New Jersy, 2001.

[19] Claudio Becchetti and Klucio Prina Ricotti, Speech Recognition Theory and C++ Implementation. Wiley Publisher, 2004.

[20] SPHINX: An open source at CMU: http://cmusphinx.sourceforge.net/html/cmusphinx.php

[21] Hidden Markov Model Toolkit (HTK-3.4.1): http://htk.eng.cam.ac.uk.

[22] Julius: An open source for LVCSR engine: http://julius.sourceforge.jp