Mel Frequency Cepstral Coefficients for Speaker Recognition Using Gaussian Mixture Model-Artificial Neural Network Model

Cheang Soo Yee[1] and Abdul Manan Ahmad[2]
Faculty of Computer Science and Information System,
University of Technology Malaysia
Syee85@gmail.com

ABSTRACT

Speaker Recognition (SP) is a topic of great significance in areas of intelligent and security. In Biometric SP using automated method of verifying or recognizing the identity of the person on the basis of some application, such as a finger print or face pattern and human voice. Many method have been proposed in the literature are focusing on front end processing such as PLP and LPC. In this paper, we study the applicability of Artificial Neural Network (ANNs) as core classifiers and Gaussian Mixture Mode (GMMs) for Mel Frequency Cepstral Coefficients (MFCC). Two different approaches have been compared. The GMMs commonly used in many application domains firstly review. We also applied a sampled method for speaker recognition that is based on ANNs. The experiment result shows that the Gaussian Mixture Model achieved highest accuracy than ANN model. However, GMM despite certain disadvantages they present mainly at the training stage, the Artificial Neural Network show better performance for speech and need less training data than the GMM-based ones [1]. It is assumed that hybrid of both models will perform better and merit for further development.

Keywords: *Speaker Recognition (SP), Gaussian Mixture Model (GMM), Artificial Neural Network (ANN), Mel Frequency Cepstral Coefficients (MFCC).*

1. INTRODUCTION

Human voice conveys information about the language being spoken and the emotion, gender and, generally, the identity of the speaker. Speaker recognition is a process where a person is recognized on the basis of his voice signals [2, 3]. The Objective of speaker recognition is to determine which speaker is present based on the individual's utterance. This is in contrast with speaker verification, where the objective is to verify the person's claimed identity based on his or her utterance. Speaker identification and speaker verification fall under the general category of speaker recognition [4, 5**].** A generic speaker recognition system is shown in Fig. 1**.** In Fig. 1, the desired features are first extracted from the speech signal. The extracted features are then used as input to a classifier, which makes the final decision regarding verification or identification.

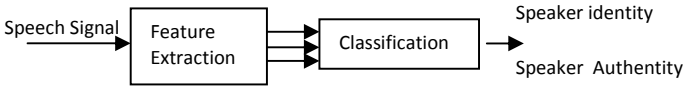

Fig.1. Speaker Recognition system

Speaker identification systems can be feature of speaker recognition systems is whether they are text dependent or text independent. Text-dependent speaker recognition systems require that the speaker utter a specific phrase or a given password. Text-independent speaker identification systems identify the speaker regardless of his utterance.

In the 1990s, a number of innovations took place in the field of pattern recognition. The idea of single-state HMM, which is now called Gaussian mixture model (GMM) was investigated to solve text-independent speaker verification problems [6]. The adapted GMM approach also leads to a fast-scoring technique. Computing the log LR requires computing

the likelihood for the speaker and background model for each feature vector.

Another way to solve the classification problem for speaker verification systems is to use discrimination-based learning procedures such as artificial neural networks (ANN) [7, 8]. As explained in [9, 10], the main advantages of ANN include their discriminate-training power, a flexible architecture that permits easy use of contextual information, and weaker hypothesis about the statistical distributions. They can be used as binary classifiers for speaker verification systems to separate the speaker and the non-speaker classes as well as multi-category classifiers for speaker identification purposes. ANN has been used for speaker verification [11, 12, 13]. Among the different ANN architectures, multilayer perceptrons (MLP) are often used [14].

2. FRONT-END PROCESSING/FEATURE EXTRACTION

Speech front-end processing consists of transforming the speech signal to a set of feature vectors. The aims of this process are to obtain a new representation which is more compact, less redundant, and more suitable for statistical modeling. Feature extraction is the key to front-end process; it mainly consists in a coding phase. The attributes of features that are desirable for speaker verification systems are [15]:

- Easy to extract, easy to measure, occur frequently and naturally in speech
- Not affected by speaker physical state
- Not change over time and utterance variations (fast talking vs. slow talking rates)
- Not affected by ambient noise
- Not subject to mimicry

In this paper, we are focusing in Mel Frequency Cepstral coefficients (MFCC). Mel Frequency Cepstral coefficients (MFCC) (Davis and Mermelstein, 1980) [16] are the most popular acoustic features used in speech recognition. Often it depends on the task; this method leads a better performance. Due to the high performance of MFCC, this technique has been chosen as front-end processing for this research. MFCC are based on the known variation of the human ear's critical bandwidths with frequency; filters spaced linearly at low frequencies and logarithmically at high frequencies have been used to capture the phonetically important characteristics of speech. Several step of MFCC are described in these following phases show in fig.2.

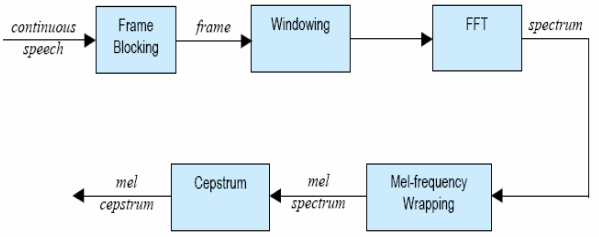

Fig.2. MFCC processing

A. Frame Blocking

Framing is the first applied to the speech signal of the speaker. The signal is partitioned or blocked into N segments (frames).

B. Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame.

C. Fast Fourier Transform

Next step is the Fast Fourier Transform which converts each frame of N samples in time domain to frequency domain.

D. Mel-Frequency Wrapping

The spectrum obtained from the above step is Mel Frequency Wrapped; the major work done in this process is to convert the frequency spectrum to Mel spectrum.

E. Cepstrum

In this final step, we convert the log Mel spectrum back to time. The result is called the Mel frequency Cepstrum coefficients (MFCC).

3. BACK-END PROCESSING/ PATTERN MATCHING

The problem of speaker recognition belongs to a much broader topic in scientific and engineering so called pattern matching. The goal of pattern matching is to classify objects of interest into one of a number of categories or classes. The objects of interest are generically called patterns and in our case are sequences of acoustic vectors that are extracted from an input speech using the techniques described in the previous section. The classes here refer to individual speakers. Since the classification procedure in our case is applied on extracted features, it can be also referred to as feature matching.

Many forms of pattern matching and corresponding models are possible. Pattern-matching methods include dynamic time warping (DTW), the hidden Markov model (HMM), artificial neural networks (ANN), and Gaussian Mixture Models (GMM). Template models are used in DTW whereas statistical models are used in HMM. In this paper, we are focusing and discussing in GMM and ANN model.

3.1 GAUSSIAN MIXTURE MODEL APPROACH

This section describes the form of the Gaussian mixture model (GMM) and motivates its use as a representation of speaker identity for speaker recognition. The speech analysis for extracting the MFCC feature representation used in this work is presented first. Next, the Gaussian mixture speaker model and its parameterization are described. The Gaussian mixture model (GMM) is a density estimator and is one of the most commonly used types of classifier. The implementation of tthe maximum-likelihood parameter estimation and speaker identification procedures is described. The classification stage uses the Gaussian Mixture Model (GMM) shown in Fig. 3.

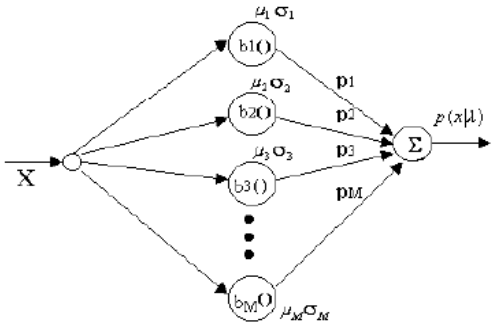

Fig.3. a Gaussian mixture density is a weighted sum of Gaussian densities, where *p,, i* = 1, . . . , *M,* are the mixture weights and *b,(), i* = I , . . . , *M,* are the component Gaussians.

Model Description

A Gaussian mixture density is a weighted sum of *M* component densities, as depicted in Fig. 3 and given by the equation

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} p_i b_i(\mathbf{x}), \text{ with } \sum_{i=1}^{M} p_i = 1 \qquad (1)$$

where **x** is a random vector of D-dimension, λ is the speaker model, pi are the mixture weights, bi(**x**) are the density components, that is formed by the mean μ and covariance matrix σi to i = 1,2,3,….M., and each density component is a D-Variate- Gaussian distribution of the form

$$b_i(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\boldsymbol{\sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)' \boldsymbol{\sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right\} \qquad (2)$$

The mean vector, μi, variance matrix, σi, and mixture weights pi of all the density components, determines the complete Gaussian Mixture Density

$$\lambda = \{\mu, \sigma, p\}, \qquad (3)$$

, used to represent the speaker model. To obtain an optimum model representing each speaker we need to calculate a good estimation of the GMM parameters. To do that, a very efficient method is the Maximum-Likelihood Estimation (ML) approach. For speaker

identification, each speaker is represented by a GMM and is referred to by his/her model λ.

Maximum Likelihood Parameter Estimation

Given training speech from a speaker, the goal of speaker model training is to estimate the parameters of the GMM, x, which in some sense best matches the distribution of the training feature vectors. There are several techniques available for estimating the parameters of a GMM [17]. By far the most popular and well-established method is maximum likelihood (ML) estimation.

The aim of ML estimation is to find the model parameters which maximize the likelihood of the GMM, given the training data. For a sequence of *T* training vectors $X = \{X_1 \ldots X_T\}$, the GMM likelihood can be w

$$p(X \mid \lambda) = \prod_{t=1}^{T} p(\vec{x}_t \mid \lambda). \tag{4}$$

ML parameter estimates can be obtained iteratively using a special case of the expectation-maximization (EM) algorithm [18]. The basic idea of the EM algorithm is, beginning with an initial model, x, to estimate a new model 1, such that $p(X \mid \lambda) \geq p(X \mid \lambda)$. The new model then becomes the initial model for the next iteration and the process is repeated until some convergence threshold is reached. This is the same basic technique used for estimating HMM parameters via the Baurn-Welch re-estimation algorithm. On each EM iteration, the following re-estimation formulas are used which guarantee a monotonic increase in the model's likelihood value:

Mixture Weights:

$$\bar{p}_i = \frac{1}{T} \sum_{t=1}^{T} p(i \mid \vec{x}_t, \lambda) \tag{5}$$

Means:

$$\vec{\bar{\mu}}_i = \frac{\sum_{t=1}^{T} p(i \mid \vec{x}_t, \lambda) \vec{x}_t}{\sum_{t=1}^{T} p(i \mid \vec{x}_t, \lambda)} \tag{6}$$

Variances:

$$\bar{\sigma}_i^2 = \frac{\sum_{t=1}^{T} p(i \mid \vec{x}_t, \lambda) x_t^2}{\sum_{t=1}^{T} p(i \mid \vec{x}_t, \lambda)} - \bar{\mu}_i^2 \tag{7}$$

Where σ_i^2, X_T and μ_i, refer to arbitrary elements of the vectors σ_i^2, X_T and μ_i, respectively.

The *a posteriori* probability for acoustic class *i* is given by

$$p(i \mid \vec{x}_t, \lambda) = \frac{p_i b_i(\vec{x}_t)}{\sum_{k=1}^{M} p_k b_k(\vec{x}_t)}. \tag{8}$$

Two critical factors in training a Gaussian mixture speaker model are selecting the order M of the mixture and initializing the model parameters prior to the EM algorithm. An experimental examination of these factors on speaker ID performance is discussed in Section 4.

3.2 ARTIFICIAL NEURAL NETWORK (ANN) APPROACH

An artificial neural network (ANN), often just called a neural network (NN), is an interconnected group of artificial neurons that uses a mathematical model or computational model for information processing based on a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network.

The particular model used in this technique can have many forms, such as multi-layer perceptions or radial basis functions. The **MLP** is a type of neural network that has grown popular over the past several years. A **MLP** with one input layer, one hidden layer, and one output layer is shown in Fig.4.
MLP's are usually trained with an iterative gradient algorithm known as back propagation [18].

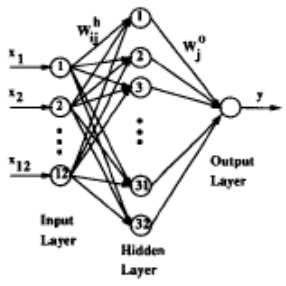

Fig.4. Multilevel Perceptron

The MLP is convenient to use for problems with limited information regarding characteristics of the input. However, the optimal MLP architecture (number of nodes, hidden layers, etc.) to solve a particular problem must be selected by trial and error,

which is a drawback. In addition, the training time required to solve large problems can be excessive, and the algorithm is vulnerable to converging to a local minima instead of the global optimum.

The MLP can be applied to speaker recognition [19] as follows. First, the feature vectors are gathered for all speakers in the population. The feature vectors for one speaker are labeled as "one" and the feature vectors for the remaining speakers are labeled as "zero." An MLP is then trained for that speaker using these feature vectors. The MLP's for all speakers in the population are trained using this method.

In this paper, we have chosen to use a back propagation neural network [20, 21,22] since it has been successfully applied to many pattern classification problems including speaker recognition [23] and our problem has been considered to be suitable with the supervised rule.

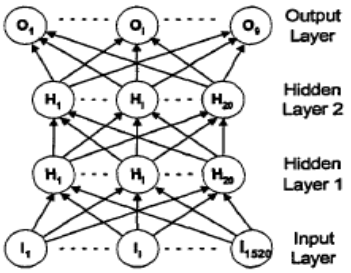

Fig.5. Back propagation neural network

MLP neural network we used consists of four layers; one input layer, two hidden layers and one output layer. The structure of the back propagation neural network is shown in Figure 5. The first layer has 1,520 input neurons (152 frames x 10 LPC-orders) which are fully connected to the first hidden layer. The two next hidden layers consist of 20 neurons per layer. The last layer is the output layer consisting of 9 neurons which one output neuron represented one speaker. All four layers are fully feed forwarded.

4. EXPERIMENT

The experimental results for the classification of pathological voice are shown in this section. And the purpose of our experiment is to classify the mixed voice data set into normal and pathological voice and to compare the classification performance to the results of artificial neural network. So the same pre-condition was configured, such as the same data set, the same characteristic parameters and the same training times.

Since the total number of data was small, we tried to train and test the ANN and GMM model by splitting total data sets into two parts. Two thirds of the data were used for training and the remaining one third of data was used for test. In each training stage, the artificial neural network and Gaussian mixture model were trained and tested And each stage for training and test was performed 5 times.

The GMM based classifier was constructed to classify the normal voices and pathological voices after the computation of the 6 parameters. Total data combination sets were performed 5 times. Different Gaussian mixtures (3, 4 and 5) are set to compare which Gaussian mixture model can attain a high classification performance. Table 1 shows the classification rate from the GMM training and testing. The "1st Run" means to run the first set of data and the others own the same meaning. From the Table 2, the highest classification is 5 Gaussian mixtures model. The average classification rate for training data is 98.4% and for test data is 95.2%.

GMM Mixtures	Performance Times	Training Data%	Test Data%
3mixtures	1st Run	98.0	96.0
	2nd Run	99.0	92.0
	3rd Run	98.0	92.0
	4th Run	90.0	95.0
	5th Run	95.0	94.0
4Mixtures	1st Run	97.0	94.0
	2nd Run	98.0	84.0
	3rd Run	96.0	88.0
	4th Run	90.9	94.0
	5th Run	97.0	92.0
5 Mixtures	1st Run	98.0	96.0
	2nd Run	99.0	96.0
	3rd Run	100.0	94.0
	4th Run	97.0	96.0
	5th Run	98.0	94.0

Table 1 Classification rate in % by GMM

Table 2 shows the experiment using artificial neural network. It shows that In ANN for each hidden layers, the average classification rate for training data and test data was calculated as shown in Fig.6 and table 2.

Hidden layer	Running time	Training Data	Test Data
3 layer	1st run speaker A	97.0%	91.0%
	2nd run speaker B	98.0%	93.0%
	3rd run speaker C	96.0%	90.0%
9 layer	1st run speaker A	97.0%	91.0%
	2nd run speaker B	96.0%	94.0%
	3rd run speaker C	97.0%	93.0%
12 layer	1st run speaker A	96.0%	97.5%
	2nd run speaker B	95.0%	97.1%
	3rd run speaker C	98.0%	98.4%

Table 2: Testing data And Training Data by Experiment Multilayer

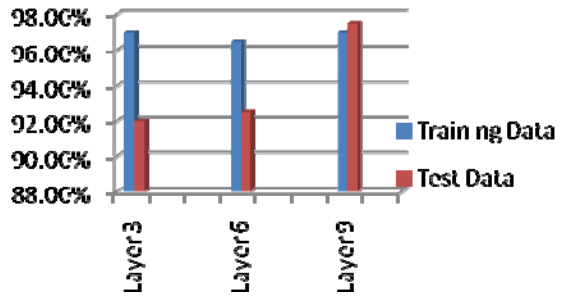

Fig.6.The average classification rate (%) in ANN

5. DISCUSSION

In this section, we discuss some experimental results obtained from the proposed analysis methods.

Table 3 shown the results obtained from the experiment on both model using same dataset. Accuracy correctly predicted for GMM and ANN is 95.2% and 94.2% respectively. Although the accuracy achieved is good but we observed that this 2 model is only for 120 speakers. If this model is used for huge dataset, the accuracy prediction might also be decreased. The result might be different if the datasets extends to 650 speakers from TIMIT datasets.

Table 3: Experiment result using ANN and GMM on 120 speaker

Model	Error	Correct	Accuracy Predicted%
GMM	2	118	95.2
ANN	4	116	94.2

In my experiment, we also need to know which number of mixture can give us the optimal classification rate. So the different mixture number (3 to 15) of GMM has been trained to find the optimal number. In Fig. 7, it shows the classification rate for different mixtures (3 to 15). From the Fig. 7, the best mixture for test data is 5 and train data is 11. Due to the limited number of data, each time we used the different data set for train and test. In Fig. 7, the average classification rate for each mixture was calculated.

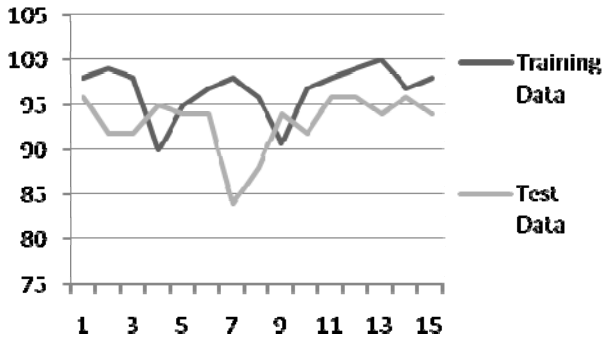

Fig.7.The classification rate for different mixtures.

In GMM for each number of mixtures, the average classification rate for training data and test data was obtained as shown in Fig. 7. In ANN for each hidden layers, the average classification rate for training data and test data was calculated as shown in Fig. 6. Comparing the results between Fig. 6 and Fig. 7, the best GMM configuration showed slightly better classification compared to the best performance of ANN (0.4% in training data, 1% in test data). Although the absolute amount of the rate difference is small, it can present that GMM can be used as more

robust classifier in pathological voice classification giving us a comparable classification rate to ANN.

6. CONCLUSIONS

This paper has performed a classification of voice using GMM method as one of the trial to obtain a better classification performance than ANN method, which was previously performed. The classification method based on GMM shows 0.4 % improvement with training data and 1% improvement with test data on the average. Although the amount of the difference is small, it is proved that GMM can be used effectively for the classification method of the pathological voice giving us more robustness in practical applications.

In the future work, to improve the performance with real data, more investigations are required on the proper number of mixtures on Gaussian model and on the proper parameter sets. Hybrid method of this both models will be experimented in future for more extensively speaker recognition.

REFERENCES

[1] C.E. Vivaracho, J. Ortega-Garcia, L. Alonso, Q.I. Moro, "A Comparative Study of MLP-based Artificial Neural Networks in Text-Independent Speaker Verification against GMM-based Systems", *EUROSPEECH 2001-SCANDINAVIA*, Aalborg Denmark, Volume 3, pp. 1753-1756, September 2001

[2] Campbell J.P. and Jr. "Speaker recognition: A Tutorial" Proceeding of the IEEE. Vol 85, 1437-1462 1997.

[3] S.Furui. "Fifty years of progress in speech and speaker recognition," Proc. 148th ASA Meeting, 2004.

[4] A. Rosenberg, "Automatic speaker recognition: A review," *Proc. IEEE,* vol. *64,* pp. 475487, Apr. 1976.

[5] G. Doddington, "Speaker recognition-Identifying people by their voices," *Proc. IEEE,* vol. 73, pp. 1651-1664, 1985

[6] Douglas A. Reynolds, *Member, IEEE,* and Richard C. Rose, *Member, IEEE*, "Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models", TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, 1995

[7] J. Hertz, A. Krogh, and R. J. Palmer, *Introduction to the Theory of Neural Computation*, Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley, Reading, Mass, USA. 1991.

[8] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan, New York, NY, USA, 1994.

[9] H. Bourlard and C. J. Wellekens, "Links between Markov models and multilayer perceptrons," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1167– 1178, 1990.

[10] M. D. Richard and R. P. Lippmann, "Neural network classifiers estimate Bayesian a posteriori probabilities," *Neural Computation*, vol. 3, no. 4, pp. 461–483, 1991.

[11] J. Oglesby and J. S. Mason, "Optimization of neural models for speaker identification," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP '90)*, vol. 1, pp. 261–264, Albuquerque, NM, USA, April 1990.

[12] Y. Bennani and P. Gallinari, "Connectionist approaches for automatic speaker recognition,"

in *Proc. 1st ESCA Workshop on Automatic Speaker Recognition, Identification and Verification*, pp. 95 102, Martigny, Switzerland, April 1994.

[13] K. R. Farrell, R.Mammone, and K. Assaleh, "Speaker recognition using neural networks and conventional classifiers," *IEEE Trans. Speech, and Audio Processing*, vol. 2, no. 1, pp. 194–205, 1994.

[14] J. M. Naik and D. Lubenskt, "A hybrid HMM-MLP speaker verification algorithm for telephone speech," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP '94)*, vol. 1, pp. 153–156, Adelaide, Australia, April 1994.

[15] Reynolds, D., and Heck, L.P., "Automatic Speaker Recognition", AAAS 2000 Meeting, Humans, Computers and Speech Symposium, 2000.

[16] Davis, S. B. and Mermelstein, P. "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. on Acoustic, Speech and Signal Processing, ASSP-28, No. 4, 1980.

[17] G. McLachlan, *Mixture Models.* New York: Marcel Dekker, 1988.

[18] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.,* vol. 39, pp. 1-38, 1977.

[19] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing.* Cambridge, MA: MIT Press, 1986.

[20] J. Oglesby and J. *S.* Mason, "Optimization of neural models for speaker identification," in *Proc. ICASSP,* 1990, pp. 261-264.

[21] *L* Fausette, "Fundamentals of Neural Networks-Architecture, Algorithm, and Applications", Prentice Hall, 1994.

[22] SNNS (Stuttgart Neural Network Simulator) User Manual, Version 4.1, University of Stuttgart, Institute for Parallel and Distributed High Performance Systems (IPVR), Report No. 6/95 W. Sintupinyo, P. D~bey, S. Sa-tang, V. Thailand, p. 238-246, March-April 1999.

[23] Y.-Yan, M. Fanty, and R. Cole, "Speech Recognition Using Neural Networks with Forward-backward Probability *Generated Targets", Proceedings of International Conference on Acoustics, Speech, and Signal Processing,* Munich, April 1997.