Correspondence

Voiced-Unvoiced-Silence Classifications of Speech Using Hybrid Features and a Network Classifier

Yingyong Qi and Bobby R. Hunt

***Abstract*— Voiced–unvoiced-silence classification of speech was made using a multilayer feedforward network. The network was evaluated and compared to a maximum-likelihood classifier. Results indicated that the network performance was not significantly affected by the size of training set and a classification rate as high as 96% was obtained.**

I. Introduction

The classification of the speech signal into voiced, unvoiced, and silence (V/U/S) provides a preliminary acoustic segmentation of speech, which is important for speech analysis. The nature of the classification is to determine whether a speech signal is present and, if so, whether the production of speech involves the vibration of the vocal folds. The vibration of vocal folds produces periodic or quasi-periodic excitations to the vocal tract for voiced speech whereas pure transient and/or turbulent noises are aperiodic excitations to the vocal tract for unvoiced speech. When both quasi-periodic and noisy excitations are present simultaneously (mixed excitations), the speech is classified here as voiced because the vibration of vocal folds is part of the speech act. The mixed excitation, however, could also be treated as an independent category.

The V/U/S classification could be made using a single parameter derived from the speech signal such as rms energy or zero-crossing rate. Such a method can only achieve limited accuracy because the value of any single parameter usually overlaps between categories, particularly when the speech is not recorded in a high fidelity environment. The V/U/S classification is also traditionally tied to the determination of periodicity (pitch determination) of speech [1]. However, because the vibration of the vocal folds may not necessarily produce a periodic signal, a failure in detecting periodicity for voiced speech would result in an error of V/U/S classification. Atal and Rabiner [2] proposed a pattern recognition approach that used multiple features of speech for V/U/S classification. The classification was independent of pitch determination and was basically a Bayesian decision process in which the assumption about the unknown statistical distribution of the features and the estimation of parameters for the distribution were essential. Large sets of training data are typically needed to reliably estimate the statistical parameters before a decision rule can be synthesized. In Atal and Rabiner's work, the distribution for the features was assumed to be multidimensional Gaussian.

To avoid making simplified assumptions about the unknown statistical distribution of features, Siegel [3] suggested an alternative approach for making voiced and unvoiced classification. The method followed the general procedure of pattern recognition using a linear discrimination function [4]. The discrimination function was a weight matrix which linearly mapped each feature vector into one side of a multidimensional pattern space partitioned by a hyperplane. The discrimination function and the hyperplane were determined by minimizing an error function from training patterns. This approach is a non-parametric treatment of the classification problem and the classification results are comparable to those obtained using the statistical parametric method. The training procedure, however, was rather complicated partly because the discontinuity of the discrimination function prevented a straightforward analytical derivation of a training algorithm. Siegel and Bessey [5] later included mixed excitation as a third category in their classification using this non-parametric approach.

The feature vector in both the parametric and non-parametric methods consisted of selected acoustic parameters whose values had a certain degree of separability between sound categories. For example, the zero-crossing rate is one of the typical parameters in the feature vector which is small for voiced speech and large for unvoiced speech because of the noise nature of unvoiced speech. The feature vector as a whole, however, was assembled somewhat artificially. Some features in the feature vector were even well correlated [2]. Improvements of classification rate would be difficult using a feature vector so defined because any modification of the feature vector has to be done on a trial by error basis. Unsatisfied with the classification rate in earlier work, Rabiner and Samuer [6] used spectral distances for making the classification. The V/U/S classification was made based on spectral proximity between the input and the class template. By using spectral distance, all spectral information was included in the decision making. A spectrum is, indeed, an independent set of features for speech signals. Although significant improvements of classification rate were obtained, the classification procedure was again a Bayesian decision process. A large set of training samples was required for building a reliable classifier. As pointed out by the authors, "The main disadvantage of the method is the need for training the algorithm to obtain the average spectral representation for the three signal class."

The lack of an effective training method is, in fact, a drawback for all classification algorithms discussed above. Applications of these methods are, therefore, limited because adaptive modifications of the classifier are often necessary in practical situations. The adaptive formation of a discrimination function for pattern classification, however, can be easily achieved using a multilayer feedforward network (MFN) due to developments in connectionist network theories [7]. In this study, a procedure is developed for making the V/U/S classification using an MFN. The feature vector for the classification is a combination of cepstral coefficients and waveform features. The cepstral coefficients are an equivalent representation of log linear predictive (LP) spectrum of speech and provide the necessary spectral information for the classification. Additional waveform features are included to enhance the separation in pattern space when spectral information alone is not sufficient for making the classification. The underlying assumption of using a feedforward network for the V/U/S classification is that temporal or contextual information of speech can be neglected in making the classification. Such an assumption can be justified by the fact that the modulation of the speech signal is largely accomplished by the continuous variation of the vocal tract and that phonetic contexts have relatively insignificant effect on the acoustic characteristics of the sound source [8].

Manuscript received January 14, 1991; revised February 24, 1992. The associate editor coordinating the review of this paper and approving it for publication was Dr. Mark A. Clements.

The authors are with the Department of Speech and Hearing Sciences, University of Arizona, Tucson, AZ 85721.

IEEE Log Number 9206397.

1063-6676/93$03.00 © 1993 IEEE

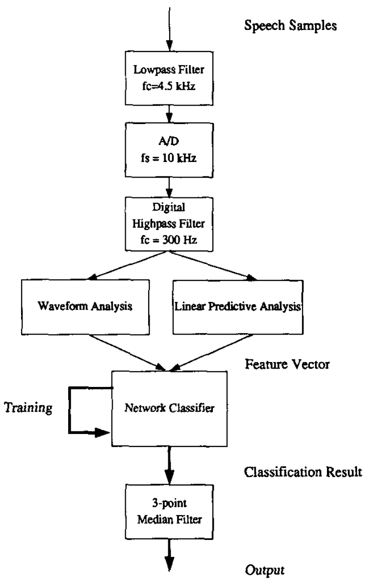


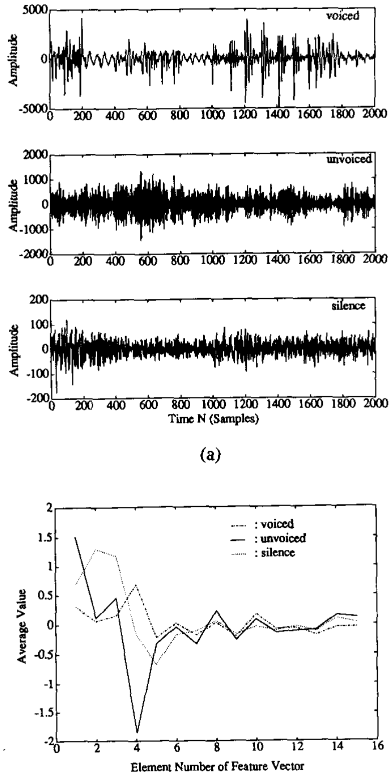
Fig. 1. Flow chart of network training and classification processes.

An MFN will, in principle, function more similarly to the non-parametric method than to the parametric method because its basic function is to partition the feature space using hyperplanes and perform pattern classification accordingly [9]. The unique advantage of an MFN is that the decision rule can be much more easily synthesized than both parametric and non-parametric methods. The network implementation of the classifier also promotes the perspective of building V/U/S classification hardware with adaptive training mechanism.

Finally, we note that a recent discovery has illuminated the relationship between the distribution free techniques of an MFN and the parametric distribution assumptions of an optimum Bayesian classifier. A recent article by Ruck *et al.* [10] has shown that the outputs of the MFN approximate the *a posteriori* probability density functions of the class being trained. The proof of this behavior is independent of any particular network architecture, i.e., is valid for any number of layers, processing nodes and connection geometry. Thus we are justified in using the training algorithm of an MFN, which is convenient and routine in application, without sacrificing the desirable properties associated with a Bayesian decision process.

II. Network Training and Classification

A block diagram for the network training and classification process is illustrated in Fig. 1. Speech signals were low-pass filtered at 4.5 kHz, sampled at 10 kHz, and quantized with 16-bit accuracy. The digitized signals were further high-pass filtered at 300 Hz by a fourth-order Butterworth digital filter to eliminate low-frequency hum or noise. A feature vector was obtained for each 20 ms segment of speech. The feature vector was a combination of 13 cepstral coefficients and two waveform parameters, the zero-crossing rate and a nonlinear function of rms energy. The cepstral coefficients were derived from 12 LP coefficients and the energy of prediction error [11]. The autocorrelation method, Hamming window, and pre-emphasis (0.98) were used in calculating the LP coefficients. An inverse squareroot function was applied to the rms energy to limit its numerical range. An example set of training samples and the average feature vectors are shown in Fig. 2. The V/U/S classification was made for each input feature vector after training was completed. The classification output was further decoded and passed through a three-point median filter to eliminate isolated "impulse" noise.



(a)

(b)

Fig. 2. (a) Example training samples (each sample has 200 points) for the 3 sound categories. (b) Example average feature vectors (element 1: zero-crossing rate, element 2: distorted rms energy, and element 3-15: cepstral coefficients).

The network was trained using the generalized delta rule for back propagation of error with a learning rate of $\alpha = 0.9$. A momentum term was added in updating the weights ($\beta = 0.6$) [12]. The training loop would not terminate until the total error was less than 10^{-4} and the error difference between consecutive training iterations was less than 5×10^{-9} or a total of 5×10^{4} training iterations had been exhausted.

The input and ouput layers of the network had fixed number of PE's. There were 15 PE's in the input layer that matched the dimension of the feature vector (13 cepstral coefficients and 2 waveform parameters). There were 3 PE's in the output layer. The output vector was coded as [100] for voiced sound, [010] for unvoiced sound, and [001] for silence. This coding was selected to maximize the code differences between categories. Because of the minimum and maximum of the activation function could only be reached at infinity, 0 and 1 were replaced by 0.1 and 0.9, respectively, in practical calculations. The overall architecture of the network, i.e., the number of hidden layer and the number of nodes per hidden layer, was a parameter to be determined in the experimental evaluation of the network. The network performance as a function of the size of training set and signal to noise ratio was also evaluated and compared to a Bayesian, maximum-likelihood (ML) classifier.

III. Network Performance

A. Data Base

Six speakers (3 men and 3 women) provided speech samples for evaluating the performance of the network. The speech samples included 10 three-digit numbers and the rainbow paragraph which begins with "*when the sunlight strikes raindrops in the air, it acts like a prism and forms a rainbow* . . . ." Recordings were made in a quiet office environment. The speech recordings were pre-processed (see Fig. 1) and were interactively labeled for membership in the three sound categories using waveform and spectrographic displays and audio output as feedback. The membership assignment was made largely based on the acoustic features of the signal. The phonetic content of a sound was taken only as a reference. For example, when part of a voiced fricative such as /z/ is devoiced based on its acoustic features (reduced periodicity and increased high frequency noisy), the devoiced part will be labeled as unvoiced. The network classification rate was obtained using a three-step procedure: (1) a set of training samples of a given size was randomly selected from the database, (2) all data samples (excluding the training samples) were classified once training was completed, and (3) an error was counted whenever the network classification differed from manual classification.

B. Network Architecture

Because a method for optimal selection of network architecture has not been well established, the objective here was to empirically select a network that had a simple architecture and reasonably high classification performance. An extensive search for an optimal network was not undertaken. Based on previous works, the starting number for the hidden node was set to 15 and was increased from 15 to 40 with an incremental step of 5. Classification rates were obtained for these single hidden layer networks as well as for a double hidden layer network. Each network was trained by a set of 150 randomly selected training samples (50 from each sound category). The classification rate as a function of network architecture is illustrated in Fig. 3(a).

As shown in the figure, the network with the architecture of 15-20-3 (a single hidden layer with 20 nodes) was a preferable choice in terms of the network simplicity and classification rate. In fact, the classification rate was not significantly altered when the number of hidden node or the number of hidden layer was increased. Because the classification rates were relatively high for all the networks, a substantial increase of classification rate due to the change of network architecture was not expected. This 15-20-3 network was used for comparing the performance of the network classifier and a ML classifier.

C. Comparison of Network and ML Classifier

The primary objectives of this comparison was to determine how the performance of each classifier would be affected by the size of training set and by noise corruption. As stated earlier, a large training set is typically required for building a reliable Bayesian classifier. Such a requirement, however, is not a mandate for training a network. Thus it was hypothesized that the performance of the network would not critically depend on the size of training set whereas a ML classifier would.

The ML classifier here was a strict software implementation of the ML algorithm. No additional decision logic was added. The training of the ML classifier involved the computation of the mean and inverse covariance matrix of the training vectors. Classifications were made based on the likelihood ratios. The procedure for network training and classification was the same as described above except that the size of the training set was manipulated. The same set of training samples of a given size were used to train both the network and the ML classifier. The classification rate as a function of training size is shown in Fig. 3(b).

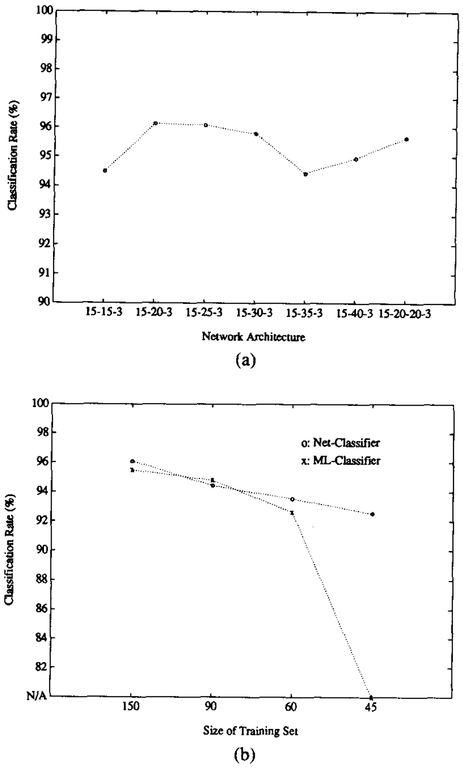


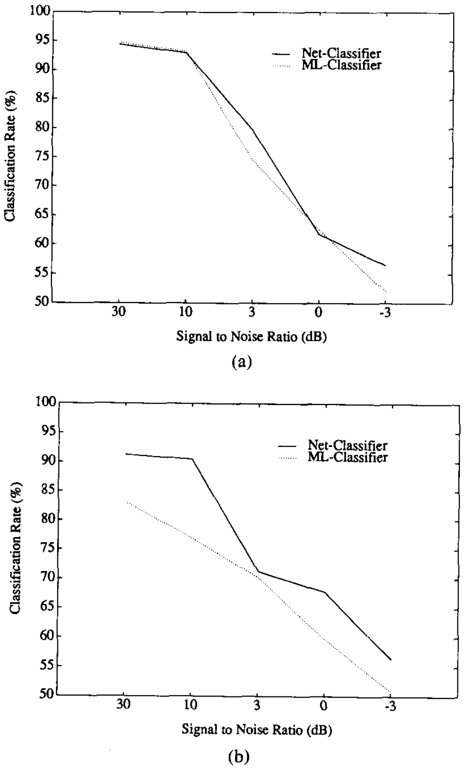
Fig. 3. (a) Network classification rate as a function of network architecture (training size = 150). (b) Classification rate as a function of training size for the network (15-20-3) and the ML classifier.

The results indicated that the performance of both classifiers as a function of training size were similar when the number of training samples was relatively large. When the size of training samples for each category was less the the dimension of the training vector, however, the inverse covariance matrix became ill-conditioned and, thus, subsequent classifications could not be computed for the ML classifier. In contrast, a reasonably high classification rate was achieved even when the size of training set was less than the dimension of the feature vector. The insensitivity of classification rate to the size of training set was apparently a significant advantage of the network classifier [13], [14].

As known, the V/U/S classification is susceptible to noise corruption because the unvoiced speech itself is a noise and the corruptive noise will significantly obscure the distinction between silence and unvoiced speech. In would be interesting, however, to compare how the network and the ML classifier would stand for noise corruption. The noise added was a Gaussian random noise whose variance was manipulated to control the signal-to-noise ratio. 90 training samples (30 from each sound category) were randomly selected after the noise of an appropriate level (depended on the signal level of each speaker) was added. The network training and classification processes remained the same as above. The classification results as a function of signal-to-noise ratio is shown in Fig. 4(a). As can be seen, both classifiers were degraded in a comparable rate when the signal to noise ratio was reduced. Both classifiers had practically failed when

TABLE I
WEIGHTS CORRELATION MATRIX AND TRAINING ITERATIONS

Layer	Speaker	F1	F2	F3	M1	M2	M3
1st	F1	1.00000	0.7452	0.8632	0.7457	0.8711	0.7124
	F2		1.00000	0.8415	0.7482	0.8381	0.7212
	F3			1.00000	0.7528	0.8407	0.7804
	M1				1.00000	0.8311	0.8940
	M2					1.00000	0.7076
	M3						1.00000
2nd	F1	1.00000	0.1709	0.4060	0.5185	0.1166	−0.0348
	F2		1.00000	0.4145	0.4061	0.5812	0.2907
	F3			1.00000	0.3579	0.1435	0.1698
	M1				1.00000	0.4417	0.1519
	M2					1.00000	0.2089
	M3						1.00000
Total	Iterations	8218	5701	2348	3524	1367	504



(a)

(b)

Fig. 4. (a) Classification rate as a function of signal to noise ratio. (b) Classification rate as a function of signal to noise ratio when only the cepstral coefficients are used as feature vector.

TABLE II
ERROR AND RATE FOR SPEAKER-DEPENDENT CLASSIFICATION

		Classification			Summary	
Speaker	Class	VD	UV	SL	Error	Decision
F1	VD	0	52	7	180 (4.0%)	4506
	UV	18	0	23		
	SL	37	43	0		
F2	VD	0	127	0	223 (3.2%)	6979
	UV	6	0	0		
	SL	32	58	0		
F3	VD	0	30	11	248 (3.8%)	6533
	UV	69	0	10		
	SL	76	52	0		
M1	VD	0	73	0	217 (3.4%)	6476
	UV	34	0	8		
	SL	35	67	0		
M2	VD	0	87	26	176 (3.2%)	5497
	UV	13	0	0		
	SL	8	42	0		
M3	VD	0	53	5	183 (3.1%)	5861
	UV	23	0	9		
	SL	18	75	0		
Total Error and Classification					1227 (3.4%)	35852

the signal to noise ratio was reduced to −3 dB. To demonstrate the advantages of using hybrid features, the classification rate as a function of signal-to-noise ratio was also computed when only the cepstral coefficients were used as the feature vector. The results are presented in Fig. 4(b).

D. Speaker-Dependent and Speaker-Independent Classification

Finally, the performance of the network was evaluated for both speaker-dependent and speaker-independent classifications. For speaker-dependent classification, the network was trained by samples from one speaker and subsequent classification was made for the same speaker. Training samples were 10 segments of speech from each sound category and were excluded from the classification. It was noted that the duration of network training was a function of both training sample and speaker. The more typical (far away from class boundaries) the samples were, the shorter the training time needed. The number of iterations also differed significantly from one speaker to another. But, the final network weights between the input layer and the hidden layer were surprisingly similar among speakers although the similarity was not found for weights between the hidden layer and the output layer. The correlation matrices for weights in each layer are shown in Table I together with the number of training iterations needed to meet the error criteria for each speaker. The classification error matrix and rate for speaker-dependent classification are tabulated in Table II. An overall classification rate of 96% ±2% was achieved for the speaker-dependent classification. Sample classification results are shown in Fig. 5.

For speaker-independent classification, the network was trained by samples from two speakers and subsequent classification was made for all speakers. One male (M1) and one female (F2) speaker were randomly selected to provide the training samples. Training samples were again 10 segments of speech from each sound category. The classification was made for all speech recordings except for the training samples. An overall classification rate of 94% ± 3% was obtained.

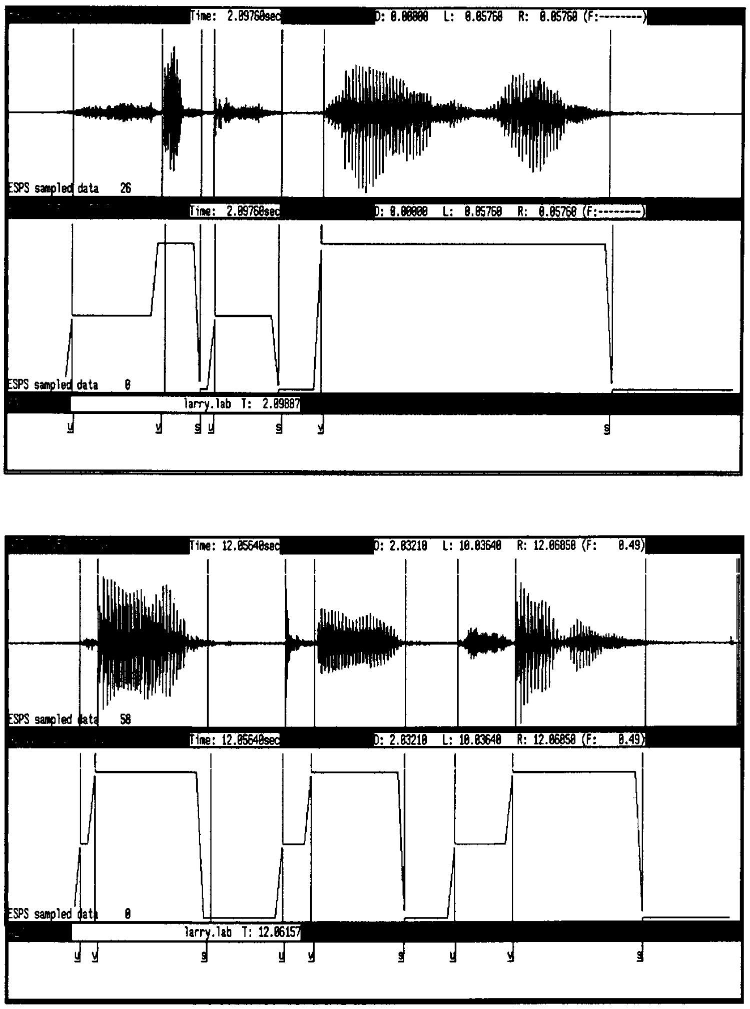


Fig. 5. Speech waveform (top window), network classification (middle window), and manual classification (bottom window) for the numbers "691" and "427" spoken by one male speaker (M1).

IV. Discussion and Conclusiosn

The results of the study clearly demonstrate that the voiced, unvoiced, and silence classification can be effectively accomplished using a multilayer feedforward network and hybrid features. Unlike the methods previously reported, the network can be effectively trained for the task. Reasonably high classification rates have been achieved.

The most significant advantage of the network classifier is that it can be trained by a few training samples and yet achieve reasonably high classification rate. In contrast, a large training set is a prerequisite for building a workable ML classifier. When the size of training set is limited by applications, the network classifier is apparently a preferable alternative.

The network classification is also computationally much simpler than an ML classifier. Only one-pass computation is needed for a network classification whereas a classification can not be made until cross-comparisons with all templates have been completed for an ML classifier. The network training, however, may take much longer than the calculation of means and covariance matrices for the ML classifier. Such a tradeoff should be recognized.

The ML classifier in this study is a straightforward implementation of the ML algorithm. The classification rate for the ML classifier could be higher than demonstrated if additional decision logic were introduced. Such a work is not intended because the ML classifier is primarily used as a comparative baseline and a more complicated implementation of the ML classifier can be found in the literature [6]. It is also worth to mention that the network classifier could be easily converted for making the voiced and unvoiced (V/U) classification only. Informal results indicate that the U/V classifier is much more robust to noise corruption than the V/U/S classifier.

Our observations indicate that the network training time is closely related to the selection of training samples. A much longer training time was noted when the training set includes samples that were close to the boundary between categories than when the training set only consisted of obvious samples from each sound category. The performance of the network for the two circumstances, however, were found to be comparable. Thus using "typical" samples for training and letting the network make the decision for ambiguous cases is probably more efficient than trying to let the network accept ambiguous cases as a prototypes for classification. The use of "typical" samples for training, however, is not a common approach for building a statistical discrimination function. A method of including ambiguous samples for network training is currently under investigation [15].

In conclusion, a procedure was developed for making voiced, unvoiced, and silence classifications of speech using an MFN. The network V/U/S classifier is expected to provide a useful tool for speech analysis and may also have applications in speech-data mixed communication systems.

References

[1] B. Atal and S. Hanuer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, pp. 637–655, Aug. 1971.

[2] B. Atal and L. Rabiner, "A pattern recognition approach to Voiced-Unvoiced-Silence classification with applications to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 201–212, June 1976.

[3] L. Siegel, "A procedure for using pattern classification techniques to obtain a Voiced/Unvoiced classifier," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 83–88, Feb. 1979.

[4] R. Duda and P. Hart, *Pattern Classification and Scene Analysis.* New York: Wiley, 1973.

[5] L. Siegel and A. Bessey, "Voiced/Unvoiced/Mixed excitation classification of speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 451–460, June 1982.

[6] L. Rabiner and M. Sambur, "Application of an LPC distance measure to the Voiced-Unvoiced-Silence detection problem," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 338–343, Aug. 1977.

[7] D. Rumelhart, G. Hinton, and R. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructures of Cognition* D. Rumelhart and J. McClelland, Eds., vol. 1,\quad Cambridge, MA: MIT Press, 1986, pp. 318–362.

[8] G. Fant, "The source filter concept in voice production," *QPSR—Speech Transmission Laboratory*, vol. 1, pp. 21–37, 1981.

[9] R. Lippman, "An introduction to computing with neural nets," *IEEE ASSP Mag.*, vol. 1, pp. 4–22, 1987.

[10] D. Ruck, S. Rogers, M. Kabrisky, M. Oxley, and B. Suter, "The multilayer perceptron as an approximation to a Bayes optimal discriminant function," *IEEE. Trans. Neural Networks*, vol. pp. 296–268, Dec. 1990.

[11] A. Gray and J. Markel, "Distance measures for speech processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 381–391, Oct. 1976.

[12] Chang and Fallside, "An adaptive training algorithm for bp networks," *Computer Speech and Language*, pp. 205–218, 1987.

[13] L. Niles, H. Silverman, G. Tajchman, and M. Bush, "How limited training data can allow a neural network to outperform an optimal statistical classifier," in *Proc. ICASSP89*, vol. 1, pp. 17–20, 1989.

[14] L. Niles, H. Silverman, G. Tajchman, and M. Bush, "The effects of training set size on relative performance of neural network and other pattern classifiers," *Tech. Rep. LEMS-51*, Brown University, Providence, RI, 1989.

[15] B. Hunt, Y. Qi, and D. Dekruger, "Fuzzy classification using set membership functions in the back propagation algorithm," *Heuristics, J. Knowledge Eng.*, vo. 5, no. 2, pp. 62–74, 1992.

On the Locality of the Forward–Backward Algorithm

Bernard Merialdo

Abstract—**In this paper, we present a theorem which shows that the local maximum found by the Forward–Backward algorithm in the case of discrete hidden Markov models is really "local." By this we mean that this local maximum is restricted to lie in the same connected component of the set $\{x : P(x) \geq P(x_0)\}$ as the initial point x_0 (where $P(x)$ is the polynomial being maximized). This theoretical result suggests that, in practice, the choice of the initial point is important for the quality of the maximum obtained by the algorithm.**

I. Introduction

Hidden Markov models are increasingly being used in various domains and, in particular, in speech recognition [1], [7]–[9]. Their popularity comes from the existence of an efficient training procedure, which, given an observed output string, allows the values of their parameters (transition and emission probabilities) to be estimated. This procedure is known as the *Baum–Welch algorithm* or the *Forward–Backward algorithm.* It is an iterative algorithm which starts from an initial point (a set of parameter values) and builds a sequence of reestimates which improve the likelihood of the training data. This sequence converges to a local maximum of the likelihood function.

A detailed presentation of the theory and practice of hidden Markov models can be found in [11]. Nadas [10] discusses the use of the Baum–Welch algorithm and makes some remarks on the choice of the initial point.

II. The Baum–Welch Algorithm

In the discrete case (i.e., when the output symbols belong to a finite alphabet), the convergence of this algorithm comes from the following theorem:

Theorem A [3], [4]: Let $p\ (X) = p\ (\{X_{ij}\})$ be a polynomial with positive coefficients, homogeneous of degree d in its variables X_{ij}.

Let $x = \{x_{ij}\}$ be any point of the domain:

$$D : x_{ij} \geq 0,\ \sum_{j=1}^{q_i} xij = 1, \qquad j = 1, \cdots, q_i$$

such that,

$$\sum_{j=1}^{q_i} x_{ij} \frac{\partial P}{\partial X_{ij}}(x) \neq 0, \qquad \text{for all } i.$$

Let $y = T_p(x)$ denote the point defined by

$$y_{ij} = \left(x_{ij} \frac{\partial P}{\partial X_{ij}}(x) \right) \Big/ \left(\sum_{j=1}^{q_i} x_{ij} \frac{\partial P}{\partial X_{ij}}(x) \right).$$

Then,

$$P(T_p(x)) > P(x) \ \text{ unless } T_P(x) = x.$$

From Theorem A we can see that when we choose an initial point x_0 and build the sequence of iterates:

$$x_{i+1} = T_P(x_i)$$

Manuscript received June 6, 1991; revised July 6, 1992. The associate editor coordinating the review of this paper and approving it for publication is Dr. Brian A. Hanson.

The author is with IBM France Scientific Center, 75001 Paris, France.

IEEE Log Number 9206396.

1063–6676/93$03.00 © 1993 IEEE