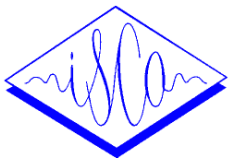

3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

MINIMUM-ERROR-RATE TRAINING OF PREDICTIVE NEURAL NETWORK MODELS

KyungMin NA, JaeYeol RHEEM and SouGuil ANN

Dept. of Electronics Engineering, Seoul National University
San 56-1, Shinlim-dong, Kwanak-gu, Seoul 151-742, KOREA.

ABSTRACT

Recently, predictive neural network models (PNNM) have proven successful in various speech recognition tasks. But, they suffer from poor discrimination for acoustically similar speech signals. In this paper, a new discriminative training algorithm based on the minimum-error-rate decision rule is proposed. Experiments on the Korean digits recognition have shown 37.5 % reduction of the number of recognition errors.

I. INTRODUCTION

Recently, several predictive neural network models (PNNM) and their effective training algorithms have been proposed for various speech recognition tasks [1-3]. In PNNM, a multilayer perceptron (MLP) is used as a nonlinear predictor of speech, and time alignment algorithm such as the dynamic programming and Viterbi algorithm is jointly used for an optimal segmentation. A single word is represented as an ordered sequence of such MLP predictors. Each MLP predictor emits prediction residual corresponding to current input, and a prediction residual matrix is formed from these residuals. Switchings between MLP's are determined along the optimal path over the prediction residual matrix. In training phase, each MLP on the optimal path is trained by the error backpropagation algorithm using corresponding input as a teaching signal. A word model that scores a minimum accumulated prediction residual is determined as a recognition result.

PNNM is classified into several categories such as neural prediction model (NPM) proposed by K. Iso and T. Watanabe [1], linked predictive neural network (LPNN) by J. Tebelskis *et al.* [2], and hidden control neural network (HCNN) by E. Levin [3]. PNNM is superior to the other speech recognition models including their neural rivals in (1) that temporal correlations between adjacent speech frames are used as recognition cues, (2) that temporal distortions of speech are efficiently normalized by time alignment algorithms, (3) that the required amount of training data is relatively small, (4) that PNNM is easily applicable to the continuous speech recognition tasks, and (5) that it is easy to add new classes.

Despite such remarkable merits, PNNM suffers from poor discrimination for acoustically confusable speech signals. J. Tebelskis *et al.* pointed out this problem, and emphasized the necessity of some type of discriminatory training technique in [2]. E. Levin also referred to the Bayseian approach instead of the standard maximum likelihood (ML) method in [3]. It is because PNNM is usually trained independently so as to optimize its parameters based upon a criterion like ML criterion on each class. That is, for acoustically similar classes A and B, the minimum accumulated prediction residual from the model of incorrect class B can be small enough to make correct class A and incorrect class B confusable when an applied input is included in class A and PNNM is trained by the conventional training algorithm.

We have already derived new discriminative training formulas based on the generalized probabilistic descent (GPD) method with the minimum classification error formulation [4]. But the GPD approach requires many learning parameters which must be chosen carefully. In this paper, we propose a new discriminative training algorithm for PNNM based on the minimum-error-rate decision rule. The proposed algorithm is simpler than the GPD-based algorithm with at least same performance. According to the Bayes decision theory, the decision rule for the minimum-error-rate classifier is to select the class that maximizes the a posteriori probability for given input [5]. We view PNNM as the a posteriori probability estimator by introducing a "softmax" function [6]. Instead of training each model to minimize the average accumulated prediction residual of corresponding class [1-3], or to directly minimize the number of recognition error [4], the proposed training algorithm trains each model to maximize the a posteriori probability. So, we call the proposed algorithm *minimum-error-rate (MER) training algorithm* or *maximum a posteriori (MAP) training algorithm*.

Experimental results on Korean digits have shown totally 37.5 % reduction of the number of recognition errors that was occurred when the conventional error backpropagation algorithm was used for training. Among several predictive neural network models, NPM has been chosen for our experiments, but it makes no difference because the proposed algorithm can be considered as a modified error backpropagation algorithm, and can be easily applied to the other models.

II. NEURAL PREDICTION MODEL AND ITS CONVENTIONAL TRAINING ALGORITHM

2.1 Neural prediction model

Neural Prediction Model (NPM) [1] uses a sequence of MLP's as a separate nonlinear predictor for each class model. Temporal correlations between the successive speech vectors are modeled by the MLP approximators, and temporal distortions of speech signals are effectively normalized by the dynamic programming technique.

Fig. 1 represents the structure of the MLP predictor.

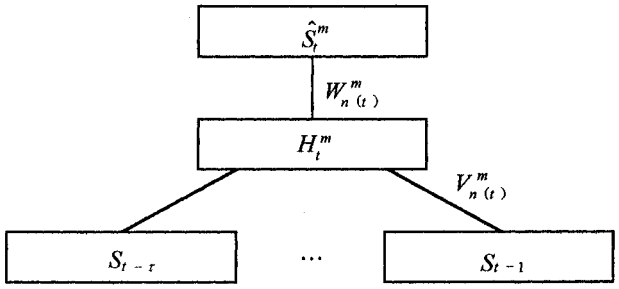

Fig 1. MLP predictor.

The MLP predictor emits a predicted speech feature vector $\hat{S}_t^m = (\hat{s}_{1,t}^m, \hat{s}_{2,t}^m, \cdots, \hat{s}_{K,t}^m)$ using the τ-preceding speech vectors $S_{t-\tau}, S_{t-\tau-1}, \cdots, S_{t-1}$, where $S_t = (s_{1,t}, s_{2,t}, \cdots, s_{K,t})$, if a word belongs to C^m among M classes $C^{m'}, m' = 1,2,\cdots,M$. Let $V_{n(t)}^m = (v_{ij,n(t)}^m)$ and $W_{n(t)}^m = (w_{jk,n(t)}^m)$ be a weight matrix between the input layer and the hidden layer and a weight matrix between the hidden layer and the output layer of $n(t)$-th predictor for class m, respectively. Let $H_t^m = (h_{j,t}^m)$ be an output from a hidden unit of class m at time t, and $f(A)$ be a sigmoid function which operates on each element of a matrix A. Given an optimal path $(t, n(t))$ and an input vector $\bar{S}_t = (s_{1,t-\tau}, \cdots, s_{K,t-\tau}, \cdots, s_{K,t-1})$, the input-output relation for the MLP predictor is as follows:

$$H_t^m = f(V_{n(t)}^m \cdot \bar{S}_t), \tag{1}$$

$$\hat{S}_t^m = W_{n(t)}^m \cdot H_t^m. \tag{2}$$

A prediction residual, $\|\hat{S}_t^m - \bar{S}_t\|^2$, is calculated from the predicted speech feature vector $\hat{S}_t^m$, and a prediction residual matrix is created in the end.

A single word model is composed of a sequence of such MLP predictors. Fig. 2 illustrates such a model for class m, where each circle denotes a corresponding MLP predictor and N_m is its total number.

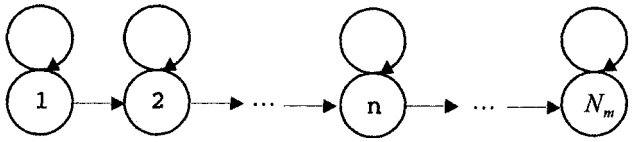

Fig. 2. NPM for word in C^m.

In training phase, the optimal segmentation of input speech feature vectors is carried out by the dynamic programming technique to minimize the accumulated prediction residual $D(m)$.

$$D(m) = \min_{n(t)} \sum_{t=1}^{T} \left\| \hat{S}_t^m(t, n(t)) - \bar{S}_t \right\|^2 \tag{3}$$

By the backtracking technique, the optimal path for segmentation is determined. Along the optimal path $(t, n(t))$, the conventional error backpropagation algorithm is carried out.

2.2 The conventional training algorithm for NPM

The conventional algorithm trains NPM independently each other so as to optimize it based on a criterion like ML criterion on each class model. If a training word is included in C^m, the conventional error backpropagation algorithm formulas are as follows:

$$(w_{jk,n(t)}^m)_{q+1} = (w_{jk,n(t)}^m)_q + \eta(s_{k,t} - \hat{s}_{k,t}^m)h_{j,t}^m, \tag{4.a}$$

$$(v_{ij,n(t)}^m)_{q+1} = (v_{ij,n(t)}^m)_q + \eta\delta_{j,t}^m h_{j,t}^m(1 - h_{j,t}^m)\bar{s}_{i,t}, \tag{4.b}$$

where $\hat{s}_{k,t}^m = \sum_{j=1}^{J} h_{j,t}^m \cdot w_{jk,n(t)}^m$, $h_{j,t}^m = f(\sum_{i=1}^{I} \bar{s}_i \cdot v_{ij,n(t)}^m)$, and $\delta_{j,t}^m = \sum_{k=1}^{K} (s_{k,t} - \hat{s}_{k,t}^m) \cdot w_{jk,n(t)}^m$. η is a learning coefficient, and $\bar{S}_t = (s_{1,t-\tau}, \cdots, s_{K,t-\tau}, \cdots, s_{K,t-1}) = (\bar{s}_{1,t}, \cdots, \bar{s}_{i,t}, \cdots, \bar{s}_{I,t})$ is an input vector. I, J, and K are the numbers of input, hidden, and output units, respectively.

III. MINIMUM-ERROR-RATE TRAINING ALGORITHM

For a derivation of maximum a posteriori (MAP) criterion for minimum-error-rate classifier, several formulas should be defined and analyzed ahead. All notations for formulas are the same with the above.

The accumulated prediction residual $g_m(x, V, W)$ of the m-th class model for a given speech input $\bar{S}_t$ is defined as

$$g_m(x,V,W)=\ln\left\{\sum_{\theta=1}^{\Theta} e^{-\left[\sum_{t=1}^{T} D_m^{\theta}(t,n(t))\right]\rho}\right\}^{-\frac{1}{\rho}}. \qquad (5.a)$$

$\sum_{t=1}^{T} D_m^{\theta}(t,n(t))$ is an accumulated prediction residual along the θ-th best path among all the possible Θ paths. If $\rho\to\infty$, then (5.a) becomes the minimum accumulated prediction residual along the best optimal path θ^*.

$$g_m(x,V,W)=\min_{n(t)}\sum_{t=1}^{T} D_m^{\theta^*}(t,n(t)), \qquad (5.b)$$

$$\text{where } D_m^{\theta^*}(t,n(t))=\frac{1}{2}\sum_{k=1}^{K}\left(s_{k,t}-\hat{s}_{k,t}^m\right)^2.$$

(5.b) is adopted in this paper for convenience. (5) are called "discriminant functions" in the area of the traditional pattern classification [5]. PNNM computes the minimum accumulated prediction residuals from the observed data x and the models of each class C^i, $i=1,2,\cdots,M$, and selects the class C^m corresponding to

$$g_m(x,V,W)=\min_i g_i(x,V,W). \qquad (6)$$

Consequently, the whole training space is divided into M subspaces by these M-discriminant functions.

From this viewpoint, we can consider PNNM as a posteriori probability estimator and define an estimate of the a posteriori probability $P_\theta(C^m|x)$ of model for class C^m given a training sample as a function of the discriminant functions where θ represents weight parameter set (V^m,W^m) of class C^m. Such estimate $f_m(x,\theta)$ of the a posteriori probability of class model m given a training sample x is defined as

$$f_m(x,\theta)=\frac{e^{-\alpha\cdot g_m(x,V,W)}}{\sum_{m'=1}^{M} e^{-\alpha\cdot g_{m'}(x,V,W)}}, \qquad (7)$$

where α is a positive constant.

(7) is a "softmax" function and has nice relationships with Gibbs distribution. In comparison with a loss function and a misclassification measure defined in [4], (7) has the opposite physical meaning to the misclassification measure, and the same property with the loss function at the same time. A larger the a posteriori probability implies that the input x is classified better, while a larger misclassification measure implies that the input is miclassified more definitely. The a posteriori probability plays a role of a cost function, while a smoothed function like a sigmoid function is introduced as a cost function in [4]. Consequently, (7) can be considered as a mingled form of those two measures in that it provides a suitable measure on the states for classification and an adequate cost function at the same time.

From (7), we can define a global a posteriori criterion $L(\theta)$ as an objective criterion, and can apply the probabilistic ascent method over this criterion.

$$L(\theta)=\prod_{l=1}^{M}\prod_{x\in C^l} f_l(x,\theta) \qquad (8)$$

Many cases of estimation theory prefer the natural logarithm of the a posteriori probability to the a posteriori probability because the logarithm is a monotonic increasing function and frequently more convenient. So, we substitute the natural logarithm of (8) for (8) itself, and define a new objective criterion as

$$-\ln L(\theta)=-\sum_{l=1}^{M}\sum_{x\in C^l}\ln f_l(x,\theta). \qquad (9)$$

The final goal is to derive new adaptation formulas such that $L(\theta)$ should be minimized over the whole training space. By the gradient descent search method and combining (5.b), (7) and (9), new discriminative training algorithm formulas will be derived.

For $S\in C^m$,

$$\frac{\partial \ln L(\theta)}{\partial w_{jk,n(t)}}=-\alpha\sum_{x\in C^m}\left(1-f_m(x,\theta)\right)\cdot\frac{\partial g_m(x,\theta)}{\partial w_{jk,n(t)}^m}$$

$$+\alpha\sum_{l,l\neq m}\sum_{x\in C^l} f_l(x,\theta)\cdot\frac{\partial g_m(x,\theta)}{\partial w_{jk,n(t)}^m}, \qquad (10.a)$$

$$\frac{\partial \ln L(\theta)}{\partial v_{jk,n(t)}}=-\alpha\sum_{x\in C^m}\left(1-f_m(x,\theta)\right)\cdot\frac{\partial g_m(x,\theta)}{\partial v_{ij,n(t)}^m}$$

$$+\alpha\sum_{l,l\neq m}\sum_{x\in C^l} f_l(x,\theta)\cdot\frac{\partial g_m(x,\theta)}{\partial v_{ij,n(t)}^m}. \qquad (10.b)$$

From (10), we can obtain new training formulas.

for $S\in C^m$,

$$\delta w_{jk,n(t)}^m=\eta\alpha\left(1-f_m(x,\theta)\right)\left(s_{k,t}-\hat{s}_{k,t}^m\right)h_{j,t}^m, \qquad (11.a)$$

$$\delta v_{ij,n(t)}^m=\eta\alpha\left(1-f_m(x,\theta)\right)\delta_{j,t}^m h_{j,t}^m\left(1-h_{j,t}^m\right)\bar{s}_{i,t}, \qquad (11.b)$$

for all $l\neq m$,

$$\delta w_{jk,n^*(t)}^l=-\eta\alpha f_l(x,\theta)\left(s_{k,t}-\hat{s}_{k,t}^l\right)h_{j,t}^l, \qquad (11.c)$$

$$\delta v_{ij,n^*(t)}^l=-\eta\alpha f_l(x,\theta)\delta_{j,t}^l h_{j,t}^l\left(1-h_{j,t}^l\right)\bar{s}_{i,t}. \qquad (11.d)$$

(11.a) and (11.b) represent the gradient descent method with step size $\eta\alpha(1-f_m(x,\theta))$, and (11.c) and (11.d) represent the gradient ascent method with step size $\eta\alpha f_l(x,\theta)$, both along their optimal paths. The step sizes(or costs) change with $f_m(x,\theta)$ and $f_l(x,\theta)$. The lower the a posteriori probability of it is, the more the proposed training algorithm optimizes the parameters of the correct class C^m along its optimal path. The maximum a posteriori estimate is obtained when (10) become zero, and this means that $f_m(x,\theta)=1$, and $f_l(x,\theta)=0$ at the same time.

IV. EXPERIMENTAL RESULTS

We have evaluated the proposed minimum-error-rate training algorithm on a data base of ten isolated Korean digits with each digit pronounced once by 20 male speakers. Only 80 data from 8 speakers have been used for training, and other 120 data for test. The speech data were sampled at 10kHz and analyzed by 25.6ms frame periods with preemphasis and Hamming window. As an input vector for each frame, 12-LPC cepstral coefficients (excluding 0-th order coefficient) were derived. We have used NPM among the several PNNM's, but the proposed algorithm can be easily applied to the other models without loss of generality. The learning coefficient η was 0.01, and the number of iterations were 20 in our experiments.

The recognition rate of the conventional training algorithm have scored 93.3 % (112/120), while that of the proposed training algorithm have scored 95.8 % (115/120). Totally 37.5 % reduction of the number of recognition errors has been achieved. Table 1 shows the results. Fig. 1 and 2 shows the learning curve and error-count curve, respectively.

Table 1. Recognition results

Conventional Algorithm	Proposed Algorithm
93.3 %	95.8 %

V. CONCLUSION

A new discriminative training algorithm, minimum-error-rate training (or maximum a posteriori training) algorithm for PNNM, were derived in this paper. The proposed algorithm is based on the Bayes decision theory. PNNM is considered as the a posteriori probability estimator introducing a "softmax" function. The derived training formulas have reasonable meaning, and they are simpler than the formulas from the GPD algorithm in [4]. Experimental results on ten Korean digits recognition have shown totally 37.5 % reduction of recognition errors.

The proposed algorithm can be extended to various classifiers both for static patterns and dynamic patterns.

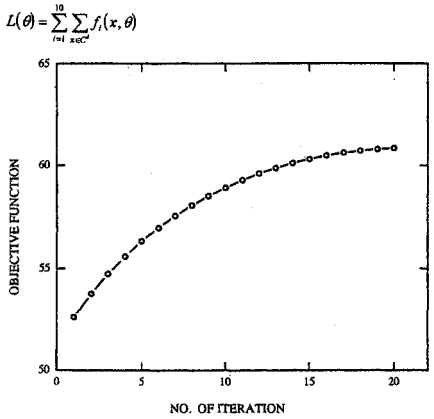

Fig.1. Maximization process of objective function.

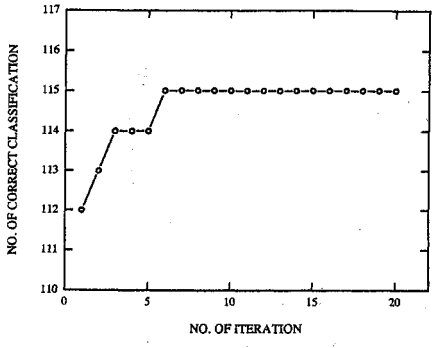

Fig. 2. Reduction of the number of errors.

Further studies on those application and other normalizing method in lieu of a "softmax" function will be continued.

REFERENCES

[1] K. Iso and T. Watanabe, "Large vocabulary speech recognition using neural prediction model," *Proc. ICASSP-91*, pp. 57-60, 1991.

[2] J. Tebelskis, A. Waibel, B. Petek and O. Schmidbauer, "Continuous speech recognition using linked predictive neural network," *Proc. ICASSP-91*, pp. 61-64, 1991.

[3] E. Levin, "Hidden control neural architecture modeling of nonlinear time varying systems and its applications," *IEEE Trans. Neural networks*, vol. 4, no. 1, pp. 109-116, 1993.

[4] K. M. Na, J. Y. Rheem and S. G. Ann, "A discriminative training algorithm for predictive neural network models," *Proc. ISCAS-94*, vol. 6, pp. 431-434, 1994.

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

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