MMI-based Training for a Probabilistic Neural Network

Nan Bu and Toshio Tsuji
Department of the Artificial Complex Systems Engineering
Hiroshima University
Higashi-Hiroshima, 739-8527 JAPAN
Email: bu@bsys.hiroshima-u.ac.jp

Osamu Fukuda
National Institute of Advanced
Industrial Science and Technology
Tsukuba, 305-8564 JAPAN
Email: fukuda.o@aist.go.jp

***Abstract*— Probabilistic neural networks (PNNs) that incorporate the Bayesian decision rule and statistical models have been widely used for pattern classification. Efficient estimation of the PNN's weights, however, is still a major problem. In this paper, we propose a new training scheme based on a discriminative criterion, *maximum mutual information* (MMI), and apply this method to the log-linearized Gaussian mixture network (LLGMN) which is one of the PNNs. The MMI training achieves a consistent estimator of network weights, and includes the conventional maximum likelihood (ML) algorithm as a special case. Also, the dynamics of terminal attractor (TA) is introduced for iteration control of the MMI training. Finally, the classification ability of the proposed method is examined with a pattern classification problem of the electromyogram (EMG) signals, and found that the MMI training results in better classification than the conventional ML algorithm.**

I. Introduction

Recently, neural networks have been increasingly popular in a pattern classification field because of their outstanding performance in approximating the desired functional mapping between input and output patterns. However, several factors have hindered the development of NN classifiers such as the slow learning convergence, need for a large amount of training data, and local minima. To tackle these problems, numerous attempts have been made to integrate specific knowledge into the NN architecture. In one approach, the NN called the probabilistic neural network (PNN), is trained to estimate the probability density function (pdf) of the pattern in order to improve the classification ability [1]-[5].

In [4], Tsuji *et al.* have proposed a feedforward probabilistic NN, a log-linearized Gaussian mixture network (LLGMN), which is based on a log-linear model and a Gaussian mixture model (GMM). LLGMN is a three-layer NN with the semiparametric model of the pdf incorporated, the outputs of which are expected to approximate the posterior probabilities of the patterns to be discriminated. The units in the hidden layer can be interpreted as mixture components in GMM, and the weight coefficients correspond to the statistical parameters in GMM, such as the mixture coefficients, mean values, and standard deviation. As an application example, LLGMN has been successfully used for the electromyogram (EMG) pattern classification problem [5].

A maximum likelihood (ML) criterion is employed to estimate the weights of LLGMN with a backpropagation-based training algorithm. The ML criterion has been extensively used to estimate unknown parameters of hypothesized pdf models. Unbiased estimators can be obtained, if 1) the true distribution of the sampled data is included in the space of the distribution of the hypothesized models, and 2) sufficient training data is available [6]. Yet if the assumptions are not satisfied, the ML criterion may no longer provide a reliable estimator, thus causing misclassification, and high classification performance may not be expected. Although LLGMN has a reasonable structure as a NN classifier, it suffers the drawbacks of the ML criterion as well. Therefore, a better discriminative training criterion is needed in order to reach the full potential of LLGMN.

In the field of speech recognition, a maximum mutual information (MMI) criterion has been used to train classifier based on hidden Markov model (HMM), and it has been shown that a training algorithm based on the MMI criterion can provide more consistent classification than ML [6]-[10]. The MMI estimation was popularized by L.R. Bahl [7] who applied it to HMM parameter estimation in 1986. Since then, many researchers have reported the promising discriminative training ability of MMI [6][8]-[10]. Valtchev *et al.* implemented the MMI estimation for optimizing the structure and parameters of a continuous density HMM-based vocabulary recognition system [8], and Woodland and Povey obtained better recognition accuracy than the corresponding ML estimation [9]. Also, in [10], Schlüter *et al.* used the MMI criterion for continuous speech recognition, and a better word recognition rate was achieved than the ML criterion.

The principal idea of MMI, when applied to estimate HMMs, is maximizing the ratio of the correct model to all other models, which can be simply expressed in the form:

$$\theta_{m^*} = \arg\max_{\theta} \frac{P(O|m^*)}{\sum_m P(m)P(O|m)},$$

where O is a time sequence of acoustic data, and m is a model or a class that categorizes the data; θ is the parameter of HMM model; m^* is the correct model for the given data, and θ_{m^*} is the estimated parameter for model m^*. $P(m)$ and $P(O|m)$ are the prior probability for model m and the posterior probability for O. In contrast, the ML estimation just maximizes the likelihood, namely, $P(O|m^*)$. This is an important benefit of MMI, since not only the correct model, but all possible models of the training data are considered. Furthermore, the MMI criterion attempts to maximize the discrimination between the correct model and the incorrect models, and to maximize the model separability rather than a likelihood function [6]. From

0-7803-7898-9/03/$17.00 ©2003 IEEE

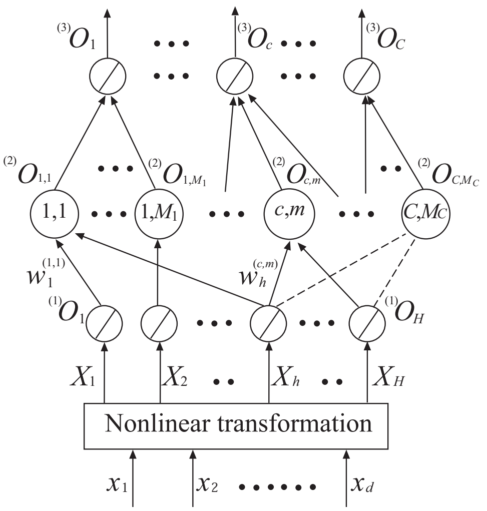

Fig. 1. The structure of LLGMN.

the viewpoint of discrimination ability, the MMI criterion is suit for training probabilistic NN, such as LLGMN.

Inspired by the above, the present paper proposes a MMI-based training scheme for LLGMN instead of the conventional ML method to improve pattern discrimination. A gradient optimization method is used according to the backpropagation rule, and the concept of a terminal attractor (TA) is introduced into the training scheme for iteration control, considering that standard gradient methods are often slow to converge. MMI is a consistent estimator, and it is expected that better discrimination can be realized with the proposed training scheme.

This paper is organized as follows. Section II briefly introduces LLGMN and the ML training algorithm. The MMI-based training algorithm and the comparison between MMI and ML criterion are described in Section III. Section IV proposes the extended training scheme including the dynamics of TA. The results of pattern classification experiments of the electromyogram (EMG) are presented in Section V. The final section concludes the paper.

II. A Probabilistic Neural Network - LLGMN

The LLGMN is based on the Gaussian mixture model (GMM) and the log-linear model of pdf. By applying the log-linear model to a product of the mixture coefficient and the mixture component of GMM, the semiparametric model of pdf is incorporated into a three-layer feedforward NN as shown in Fig. 1. For training, a simple algorithm based on the ML criterion and the backpropagation rule is employed [4].

A. Structure of LLGMN

First, in the pre-process, the input vector $\mathbf{x} \in \Re^d$ is converted into the modified vector $\mathbf{X} \in \Re^H$ as follows:

$$\begin{aligned}\mathbf{X} \;\; = \;\; & (1, \mathbf{x}^{\mathrm{T}}, {x_1}^2, x_1x_2, \cdots, x_1x_d, {x_2}^2, x_2x_3, \cdots, \\ & x_2x_d, \cdots, {x_d}^2)^{\mathrm{T}} \end{aligned} \tag{1}$$

where $x_i, i = 1, 2, \cdots, d$, are the elements of $\mathbf{x}$ and $H = 1 + d(d+3)/2$. The first layer consists of H units corresponding to the dimension of $\mathbf{X}$ and the identity function is used for activation of each unit. ${}^{(1)}O_h$ $(h = 1, \cdots, H)$ in Fig. 1 denotes the output of the hth unit in the first layer.

In the second layer, each unit receives the output of the first layer weighted by the weight $w_h^{(c,m)}$ and outputs the posterior probability of each component. The relationships between the input of unit $\{c, m\}$ in the second layer (${}^{(2)}I_{c,m}$) and the output (${}^{(2)}O_{c,m}$) are defined as

$$^{(2)}I_{c,m} = \sum_{h=1}^{H} {}^{(1)}O_h w_h^{(c,m)} \tag{2}$$

$$^{(2)}O_{c,m} = \frac{\exp[^{(2)}I_{c,m}]}{\sum_{c'=1}^{C}\sum_{m'=1}^{M_C} \exp[^{(2)}I_{c',m'}]} \tag{3}$$

where $w_h^{(C,M_C)} = 0$ $(h = 1, \cdots, H)$.

The third layer consists of C units corresponding to the number of classes. The unit c $(c = 1, \cdots, C)$ sums up the outputs of M_c units $\{c, m\}$ $(m = 1, \cdots, M_c)$ in the second layer. The function between the input and the output is described as

$$^{(3)}O_c = {}^{(3)}I_c = \sum_{m=1}^{M_c} {}^{(2)}O_{c,m} \tag{4}$$

where the output ${}^{(3)}O_c$ corresponds to the posterior probability of class c.

By optimizing the weight $w_h^{(c,m)}$, LLGMN is expected to approximate the posterior probability $P(c|\mathbf{x})$ $(c = 1, \cdots, C)$, when an input vector $\mathbf{x}$ is provided to it. The next subsection briefly describes the supervised training algorithm based on ML criterion.

B. Training Scheme Based on an ML Criterion

In the training procedure, a set of vector $\tilde{\mathbf{X}}$= $(\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)})$ and the corresponding teacher vector $\mathbf{T}^{(n)} = (T_1^{(n)}, \cdots, T_c^{(n)}, \cdots, T_C^{(n)})$ $(n = 1, \cdots, N)$ are used. The teacher vector provides perfect classification, that is, $T_{\hat{c}}^{(n)} = 1$ for the particular class $\hat{c}$ and $T_c^{(n)} = 0$ for the other classes. The network is assumed to acquire the probability distribution of the training data if for all $\mathbf{x}^{(n)}$ the output ${}^{(3)}\mathbf{O}^{(n)}$ is close enough to the teacher vector $\mathbf{T}^{(n)}$. The ML-training estimates the weight W as

$$W_{ML} = \arg\max_{W} \prod_{n=1}^{N}\prod_{c=1}^{C} P(c|\mathbf{x}^{(n)})^{T_c^{(n)}}. \tag{5}$$

Taking the logarithm of the right side of (5), the following log-likelihood function can be derived:

$$L = \sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)} \log P(c|\mathbf{x}^{(n)}) = \sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)} \log {}^{(3)}O_c^{(n)} \tag{6}$$

where ${}^{(3)}O_c^{(n)}$ corresponds to $P(c|\mathbf{x}^{(n)})$. The objective function J is defined as

$$J_{ML} = \sum_{n=1}^{N} J_n = -\sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)} \log {}^{(3)}O_c^{(n)} \tag{7}$$

and training process is to minimize J_{ML}, that is, to maximize the likelihood. The weight modification $\Delta w_h^{(c,m)}$ is given as follows:

$$\Delta w_h^{(c,m)} = -\eta \sum_{n=1}^{N} \frac{\partial J_n}{\partial w_h^{(c,m)}} \tag{8}$$

$$\begin{aligned}
\frac{\partial J_n}{\partial w_h^{(c,m)}} &= \frac{\partial}{\partial w_h^{(c,m)}}\left(-\sum_{c=1}^{C} T_c^{(n)} \log {}^{(3)}O_c^{(n)}\right) \\
&= -\sum_{c'=1}^{C} \frac{\partial T_{c'}^{(n)} \log {}^{(3)}O_{c'}^{(n)}}{\partial {}^{(3)}O_{c'}^{(n)}} \\
&\quad \times \sum_{m'=1}^{M_{c'}} \frac{\partial {}^{(3)}O_{c'}^{(n)}}{\partial {}^{(2)}O_{c',m'}^{(n)}} \frac{\partial {}^{(2)}O_{c',m'}^{(n)}}{\partial {}^{(2)}I_{c,m}^{(n)}} \frac{\partial {}^{(2)}I_{c,m}^{(n)}}{\partial w_h^{(c,m)}} \\
&= \left({}^{(3)}O_c^{(n)} - T_c^{(n)}\right) \frac{\partial {}^{(2)}O_{c,m}^{(n)}}{\partial {}^{(3)}O_c^{(n)}} X_h^{(n)}
\end{aligned} \tag{9}$$

where $\eta > 0$ is the learning rate.

III. MMI-based Training

A. MMI Training Criterion

The MMI criterion used in the present paper is based on Shannon's information theory [11]. Consider two random variables C and X, the mutual information (MI), $I(C;X)$, is defined as

$$I(C;X) = H(C) - H(C|X) \tag{10}$$

where $H(C)$ is the entropy of C and $H(C|X)$ is the conditional entropy of C given X. They can be expressed in the form:

$$H(C) = -\sum_{c} P(c) \log P(c) \tag{11}$$

$$H(C|X) = -\sum_{c}\sum_{x} P(c,x) \log P(c|x) \tag{12}$$

where $P(c)$, $P(c,x)$, and $P(c|x)$ stand for prior probability of c, joint probability of c and x, and posterior probability of c, respectively. Because entropy is also interpreted as a measure of uncertainty of a random variable, it is reasonable to take the MI between the class (C) and the input data (X) as the reduction of uncertainty of classes given the observed data generated from these classes. Suppose that the distribution of prior probability of C is known, i.e. $H(C)$ is constant, maximizing $I(C;X)$ means minimizing the conditional entropy $H(C|X)$, intuitively, the uncertainty of the distribution of $P(c|x)$ is reduced. The parameters in PNN, e.g. LLGMN, are thus expected to be estimated according to the MMI criterion.

Extending (10) with (11) and (12), $I(C;X)$ can be rewritten as follows

$$I(C;X) = \sum_{c}\sum_{x} P(c|x)P(x) \log \frac{P(c|x)}{P(c)}. \tag{13}$$

Thus, the MI can be expressed as a function of the weights in PNN, that is, $I(C;X) = f(P(c|x)) = f(g(W))$, and by the MMI criterion, the weight W is estimated as

$$\begin{aligned}
W_{MMI} &= \arg\max_{W} I(C;X) \\
&= \arg\max_{W} \sum_{c}\sum_{x} P(c|x)P(x) \log \frac{P(c|x)}{P(c)}.
\end{aligned} \tag{14}$$

According to the definition in II, the objective function for LLGMN training based on MMI, J_{MMI}, is now defined as

$$\begin{aligned}
J_{MMI} &= -I(C;\tilde{\mathbf{X}}) \\
&= -\sum_{n=1}^{N}\sum_{c=1}^{C} P(c|\mathbf{x}^{(n)})P(\mathbf{x}^{(n)}) \log \frac{P(c|\mathbf{x}^{(n)})}{P(c)} \\
&= -\sum_{n=1}^{N}\sum_{c=1}^{C} {}^{(3)}O_c^{(n)} P(\mathbf{x}^{(n)}) \log \frac{{}^{(3)}O_c^{(n)}}{P(c)}.
\end{aligned} \tag{15}$$

The MI can be maximized by minimizing the objective function J_{MMI}.

B. Comparing MMI with ML

Rewriting the objective function (15), we get

$$\begin{aligned}
&J_{MMI} \\
&= \sum_{n=1}^{N}\sum_{c=1}^{C} P(c|\mathbf{x}^{(n)})P(\mathbf{x}^{(n)})(\log P(c) - \log P(c|\mathbf{x}^{(n)})) \\
&= \sum_{n=1}^{N}\sum_{c=1}^{C} P(c,\mathbf{x}^{(n)}) \log P(c) - P(c,\mathbf{x}^{(n)}) \log P(c|\mathbf{x}^{(n)}).
\end{aligned} \tag{16}$$

Assuming that the prior probability $P(c)$ and the joint probability $P(c,\mathbf{x}^{(n)})$ are 0 or 1, $P(c,\mathbf{x}^{(n)}) \log P(c)$ in (16) equals 0 because $P(c,\mathbf{x}^{(n)}) \log P(c) = P(\mathbf{x}^{(n)}|c)P(c) \log P(c)$. Furthermore, $P(c,\mathbf{x}^{(n)})$ can be interpreted as the teaching vector $\mathbf{T}^{(n)}$, because $P(c,\mathbf{x}^{(n)})$ equals 1 when $\mathbf{x}^{(n)}$ is generated from class c and 0 for the other cases. Consequently, (16) can be simplified as

$$\begin{aligned}
J_{MMI} &= \sum_{n=1}^{N}\sum_{c=1}^{C} -P(c,\mathbf{x}^{(n)}) \log P(c|\mathbf{x}^{(n)}) \\
&= \sum_{n=1}^{N}\sum_{c=1}^{C} -T_c^{(n)} \log P(c|\mathbf{x}^{(n)}) \\
&= -\sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)} \log {}^{(3)}O_c^{(n)} = J_{ML}.
\end{aligned} \tag{17}$$

This means that J_{ML} is included in J_{MMI}, and is a special case of J_{MMI}.

On the other hand, by symmetry, $I(C;X)$ also follows that

$$I(C;X) = H(X) - H(X|C). \tag{18}$$

As the distribution of $P(x)$ is not affected by the weight of LLGMN, $H(X)$ can be considered to be constant. Similar to the explanation for (10)-(12), maximizing the MI would reduce the uncertainty of the distribution of $P(x|c)$. As a result, both the posterior probability distribution, $p(c|x)$, and the emission probability distribution, $p(x|c)$, can be estimated simultaneously when maximizing the MI. Unlike MMI, the ML criterion can estimate just one probability distribution. If both the ML estimation and the MMI estimation reach global minima in the training process of LLGMN, the resultant weights should be the same. However, when the global minima are not reachable or the training iteration is stopped at some local minima, training with the MMI criterion would provide better discrimination.

IV. MMI Training Algorithm with Terminal Attractor

This section introduces the details of the proposed training scheme. The modified objective function J_M is defined on the basis of the MMI objective function (15),

$$\begin{aligned} J_M &= I' - I(C;\tilde{\mathbf{X}}) \\ &= I' - \sum_{n=1}^{N}\sum_{c=1}^{C} {}^{(3)}O_c^{(n)} P(\mathbf{x}^{(n)}) \log \frac{{}^{(3)}O_c^{(n)}}{P(c)} \end{aligned} \quad (19)$$

where I' is constant and equals the maximum of $I(C;\tilde{\mathbf{X}})$, so that J_M has the minimum as 0. However, it is difficult to calculate I' with no desired NN outputs value, e.g. the teacher vector. Furthermore, without teacher vector, the NN would be trained like an self-organizing maps (SOM), and training of the SOM with a gradient method must be tough.

In this paper, for simplification, we applied the teacher vector defined in II-B to (19), and defined the following objective function:

$$J \equiv I' - \sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)}\,{}^{(3)}O_c^{(n)} P(\mathbf{x}^{(n)}) \log \frac{{}^{(3)}O_c^{(n)}}{P(c)} \quad (20)$$

where $P(c) = \sum_{n=1}^{N} P(c|\mathbf{x}^{(n)})P(\mathbf{x}^{(n)})$ $(c = 1, \cdots, C)$, and it is assumed $P(\mathbf{x}^{(n)}) = 1/N$ $(n = 1, \cdots, N)$. It is considered that $I(C;\tilde{\mathbf{X}})$ becomes maximum when ${}^{(3)}O_c^{(n)} = T_c^{(n)}$, which suggests that, given the input $\mathbf{x}^{(n)}$, the desired probabilities are obtained; thus we get $I' = \log C$.

For standard gradient method, the weight modification $\Delta w_h^{(c,m)}$ is derived as

$$\Delta w_h^{(c,m)} = -\eta \frac{\partial J}{\partial w_h^{(c,m)}} \quad (21)$$

with a fixed $\eta > 0$ as the learning rate. $\frac{\partial J}{\partial w_h^{(c,m)}}$ can be derived as follows:

$$\begin{aligned} &\frac{\partial J}{\partial w_h^{(c,m)}} \\ &= \frac{\partial}{\partial w_h^{(c,m)}}\Big(-\sum_{n=1}^{N}\sum_{c=1}^{C} T_c^{(n)}\,{}^{(3)}O_c^{(n)} P(\mathbf{x}^{(n)}) \log \frac{{}^{(3)}O_c^{(n)}}{P(c)}\Big) \\ &= -\sum_{n=1}^{N}\sum_{c'=1}^{C} \frac{\partial}{\partial\,{}^{(3)}O_{c'}^{(n)}}\Big(T_{c'}^{(n)}\,{}^{(3)}O_{c'}^{(n)} P(\mathbf{x}^{(n)}) \log \frac{{}^{(3)}O_{c'}^{(n)}}{P(c')}\Big) \\ &\quad \times \sum_{m'=1}^{M_{c'}} \frac{\partial\,{}^{(3)}O_{c'}^{(n)}}{\partial\,{}^{(2)}O_{c',m'}^{(n)}} \frac{\partial\,{}^{(2)}O_{c',m'}^{(n)}}{\partial\,{}^{(2)}I_{c,m}^{(n)}} \frac{\partial\,{}^{(2)}I_{c,m}^{(n)}}{\partial w_h^{(c,m)}} \\ &= -\sum_{n=1}^{N}\sum_{c'=1}^{C} T_{c'}^{(n)} P(\mathbf{x}^{(n)})[\log \frac{{}^{(3)}O_{c'}^{(n)}}{P(c')} - \frac{{}^{(3)}O_{c'}^{(n)} P(\mathbf{x}^{(n)})}{P(c')} \\ &\quad +1] \times (\delta_{c',c} - {}^{(3)}O_{c'}^{(n)})\,{}^{(2)}O_{c,m}^{(n)} X_h^{(n)} \end{aligned} \quad (22)$$

where $\delta_{c',c}$ equals 1 when $c' = c$, and 0 otherwise.

Because the standard gradient method is always criticized for its slow convergence, this paper incorporates the dynamics of the terminal attractor (TA) [12] into the training scheme in order to regulate the convergence time. The differential equation of TA is defined as

$$\dot{u} = -u^{\beta}. \quad (23)$$

When the parameter β is determined as $0 < \beta < 1$, u is a monotonically non-increasing function, and always converges stably to the equilibrium point in a finite time, since the Lipschitz conditions are violated at $u = 0$:

$$\left.\frac{d\dot{u}}{du}\right|_{u=0} = -\beta u^{\beta-1}\big|_{u=0} = -\infty. \quad (24)$$

The convergence time is fixed depending on the initial condition $u = u_0$:

$$t_f = \int_0^{t_f} dt = \int_{u_0}^{u\to 0} \frac{du}{\dot{u}} = \frac{u_0^{1-\beta}}{(1-\beta)} < \infty \quad (25)$$

where β determines how the dynamics converges, such as smoothly or sharply.

Incorporating TA into the objective function (20) of LLGMN can regulate the convergence time of the training, and the network training converges to the global minimum or one of the local minima in a finite specified time [13]. The weights of LLGMN are considered as the time dependent continuous variables, and time derivative of $w_h^{c,m}$ is defined as:

$$\dot{w}_h^{c,m} = -\eta_{ta}\gamma \frac{\partial J}{\partial w_h^{c,m}} \quad (26)$$

$$\gamma = \frac{J^{\beta}}{\displaystyle\sum_{c=1}^{C}\sum_{m=1}^{M_c}\sum_{h=1}^{H}\left(\frac{\partial J}{\partial w_h^{c,m}}\right)^2} \quad (27)$$

where $\eta_{ta} > 0$ is positive, and γ is calculated using the constant β. The time derivative of the energy function J can be calculated as:

$$\begin{aligned} \dot{J} &= \sum_{c=1}^{C}\sum_{m=1}^{M_c}\sum_{h=1}^{H}\left(\frac{\partial J}{\partial w_h^{c,m}}\dot{w}_h^{c,m}\right) \\ &= -\eta_{ta}J^{\beta} \leq 0. \end{aligned} \quad (28)$$

According to the dynamics of TA (23)-(25), the convergence time can be given as

$$\begin{aligned} t_f &= \int_0^{t_f} dt = \int_{J_0}^{J_f} \frac{dJ}{\dot{J}} = \frac{J_0^{1-\beta} - J_f^{1-\beta}}{\eta_{ta}(1-\beta)} \\ &\leq \frac{J_0^{1-\beta}}{\eta_{ta}(1-\beta)} \end{aligned} \quad (29)$$

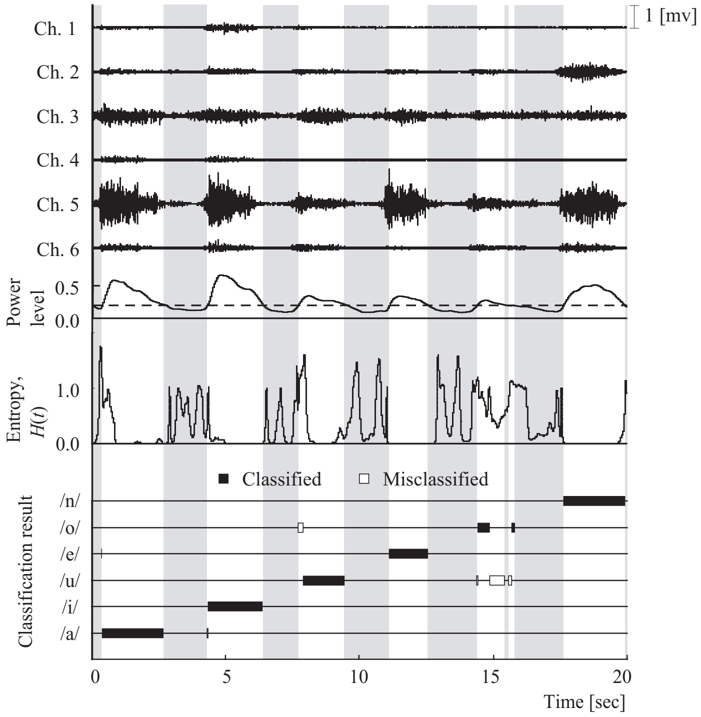

Fig. 2. An example of the classification result based on the proposed MMI training algorithm.

where J_0 is an initial value of the energy function J calculated using initial weights, and J_f is the final value of J at the equilibrium point. For $J_f = 0$, the equal sign of (29) is held. Thus, the convergence time can be specified by learning rate η_{ta}. In contrast, for $J_f \neq 0$, the convergence time is always less than the upper limit of (29).

V. Experiments

Pattern classification experiments for the EMG signals were conducted using the training methods based on MMI and ML criteria. The EMG signal used were six-channel data ($d = 6, H = 28$) corresponding to six classes ($C = 6$), which were measured from six mimetic and cervical muscles of a patient with cervical spine injury, and six classes are corresponding to six vocable sounds, i.e. /a/, /i/, /u/, /e/, /o/, and /n/. In the experiments, the patient was asked to utter the six vowels in the order. The EMG signals were measured with a sampling frequency $f_d = 1000$ [Hz], then rectified and filtered by a Butterworth filter (cutoff frequency: 1 [Hz]). Each sampled data was normalized to make the sum of six channels equal 1, and the feature vectors $\mathbf{x}^{(\mathbf{n})} = [x_1^{(n)}, \cdots, x_6^{(n)}]$ were inputted into the LLGMN. A power level was estimated from the EMG signals, and was compared with a prefixed threshold to determine whether the patient uttered or not. The LLGMN includes 28 units in the first layer, 18 units in the second layer corresponding to the total number of components ($M_c = 3, (c = 1, \ldots, 6)$), and six units in the third layer. In training, the objective functions J (20) and J_{ML} (7) were used. The dynamics of TA was used in both the MMI and the ML training algorithm, and set with $\beta = 0.8$, $t_f = 1$ sec and $\Delta t = 0.00025$ sec, that results 4000 iterations.

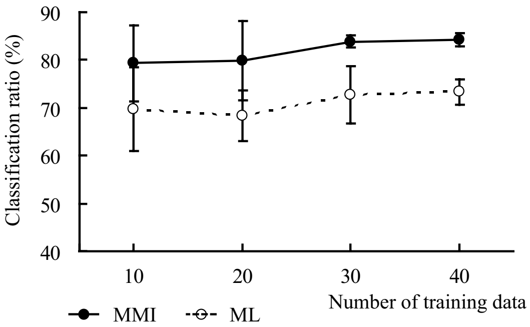

Fig. 3. Classification ratios for various training data number with decision rule 1.

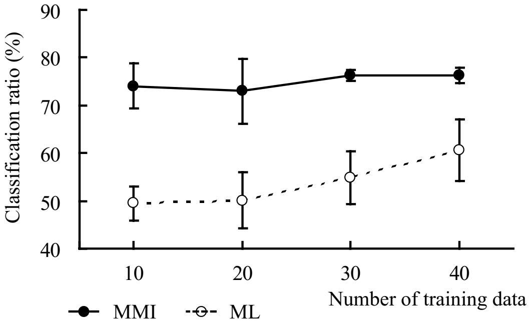

Fig. 4. Classification ratios for various training data number with decision rule 2.

An example of the classification result based on the proposed MMI training algorithm is shown in Fig. 2. In this figure, six channels of EMG signals, the power level, the entropy $H(t)$ calculated from the output probability of LLGMN, and the classification results are plotted. The gray areas indicate no utterance. The classification rate was about 90.9% in this experiment. Although misclassifications were made for /o/, it should be noticed that EMG patterns for utterance of /u/ and /o/ are similar, furthermore, the power level of /o/ is relatively lower than the others.

To verify the discrimination performance of the proposed training algorithm, comparison experiments with ML were conducted. Five sets of training data were used for both methods to train LLGMN, and in each set the numbers of training data for each class are changed from 10 to 40. After training, 1000 data patterns for each class that were not used in training were prepared for classification. The class for the input pattern is decided with two decision rules:

1) The class with the largest posterior probability.
2) The class whose posterior probability is larger than 0.8. If none is larger than 0.8, the determination is suspended.

The experimental results were given in terms of the ratio of correct classification, which is the average of the classification ratios of six classes. Fig. 3 plots the mean values and the standard deviations of the ratios of correct classification for various numbers of training data, with the decision rule 1).

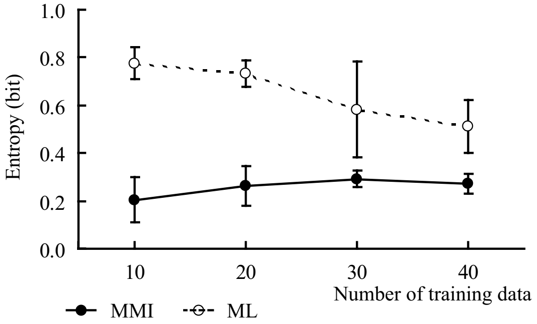

Fig. 5. Effect of the training data number on the entropy of LLGMN's outputs.

From this figure, we can see that the MMI method gives better classification than ML for all training data numbers. The classification ratios according to decision rule 2) are shown in Fig. 4. When a strict decision rule is used, the result of ML method degrades much more than MMI. Given the input data, LLGMN trained with MMI criterion generally outputted higher posterior probabilities for the corresponding *correct* classes, and this demonstrates the superior estimation ability and the training efficiency of the MMI method. The MMI training method realizes a better estimation of the parameters in LLGMN in the same training period, so that LLGMN approximates more accurate posterior probability than the ML method.

On the other hand, when the number of training data decreases, the results based on the MMI method keep in a high level in Figs. 3 and 4, while classification rate of the ML-trained LLGMN tends to decrease in Fig. 4. The ML method cannot achieve a consistent estimation when the training data is not sufficient. This can also be illustrated by examining the entropy of the NN's outputs, H (see Fig. 5), which is defined as

$$H = \sum_{c=1}^{C} {}^{(3)}O_c \log_2 {}^{(3)}O_c. \tag{30}$$

Higher entropy means more uncertainty of the classification result. In contrast to the ML training method, we find the MMI training provides a reliable estimate of each weight in LLGMN even with a small training data size.

VI. Conclusions

In the present paper, an MMI based training algorithm has been proposed for a probabilistic NN, LLGMN. The algorithm is derived from the formulation of mutual information. The uncertainty of the probability distribution that is the function of NN's weights is reduced when maximizing the mutual information between the input and output of LLGMN, and thus, the network weights can be optimized according to the MMI criterion. Comparing with the conventional ML training criterion, the ML criterion can be interpreted as a special case of MMI; furthermore, MMI training can theoretically optimize the posterior probability and the emission probability simultaneously. For simplification, a supervised training method has been proposed in this paper, and the dynamics of TA has been introduced into the training algorithm to regulate the convergence time.

To examine the performance of the proposed training algorithm, pattern classification experiments were conducted on the EMG signal. The results showed that the MMI training method achieves more efficient estimate of the NN weights, and performs better classification than ML. With respect to the entropy of the NN's outputs, it is clear that the MMI method provides more reliable estimates of the NN's weights.

Further research is needed to make a detailed investigation into the probabilistic interpretation on the MMI and the relationship between MMI and ML. Also, it will be interesting to apply the proposed method to other probabilistic NNs [14].

Acknowledgment

The authors would like to thank New Energy and Industrial Technology Development Organization (NEDO) of Japan for the financial support.

References

[1] D.F. Specht, "Probabilistic neural networks," *Neural Networks*, Vol. 3, No. 1, pp. 109-118, 1990.

[2] J.S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," in *Neurocomputing: Algorithms, Architectures, and Applications*, S.F. Fogelman and J. Hault, Eds. New York: Springer-Verlag, 1989, pp.227-236.

[3] H.G.C. Tråvén, "A neural network approach to statistical pattern classification by "semiparametric" estimation of probability density functions," *IEEE Trans. Neural Networks*, Vol. 2, No. 3, pp. 366-377, 1991.

[4] T. Tsuji, O. Fukuda, H. Ichinobu, and M. Kaneko, "A Log-Linearized Gaussian Mixture Netwrok and Its Application to EEG Pattern Classification," *IEEE Trans. on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol. 29, No. 1, pp. 60-72, 1999.

[5] O. Fukuda, T. Tsuji, A. Ohtsuka, and M. Kaneko, "EMG-based human-robot interface for rehabilitation aid," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3492-3497, Leuven, 1998.

[6] L.R. Bahl, M. Padmanabhan, D. Nahamoo, and P.S. Gopalakrishnan, "Discriminative Training Of Gaussian Mixture Models For Large Vocabulary Speech Recognition Systems," *Proceeding of International Conf. on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 613-616, Atlanta, 1996.

[7] L.R. Bahl, P.F. Brown, P.V. de Souza, and R.L. Mercer, "Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition," *Proceeding of International Conf. on Acoustics, Speech and Signal Processing*, pp. 49-52, Tokyo, 1986.

[8] V. Valtchev, J.J. Odell, P.C. Woodland, S.J. Young, "MMIE Training of Large Vocabulary Recognition Systems," *Speech Communication*, Vol. 22, No. 4, pp. 303-314, 1997.

[9] P.C. Woodland and D. Povey, "Large Scale Discriminative Training for Speech Recognition," *Proc. of the Workshop on Automatic Speech Recognition*, Paris, France, 2000.

[10] R. Schlüter, W. Macherey, B. Müller, H. Ney, "Comparison of Discriminative Training Criteria and Optimization Methods for Speech Recognition," *Speech Communication*, Vol. 34, No. 3, pp. 287-310, 2001.

[11] T.M. Cover, J.A. Thomas, "Elements of Information Theory," Wiley, New York, 1991.

[12] M. Zak, "Terminal attractors for addressable memory in neural networks," *Physics Letters A*, Vol. 133, No. 1,2, pp. 18-22, 1988.

[13] O. Fukuda, T. Tsuji, and M. Kaneko, "Pattern classification of EEG signals using a log-linearized Gaussian mixture neural networks," *Proc. IEEE Int. Conf. Neural Networks*, Vol. V, pp. 2479-2484, 1995.

[14] T. Tsuji, N. Bu, O. Fukuda, and M. Kaneko, "A recurrent log-linearized Gaussian mixture network," *IEEE Trans. Neural Networks*, Vol. 14, No. 2, pp. 304- 316, 2003.