A Sequential Pattern Classifier Based on Hidden Markov Kernel Machine and Its Application to Phoneme Classification

Yotaro Kubo, *Member, IEEE*, Shinji Watanabe, *Member, IEEE*, Atsushi Nakamura, *Senior Member, IEEE*, Erik McDermott, and Tetsunori Kobayashi, *Member, IEEE*

***Abstract*—This paper describes a novel classifier for sequential data based on nonlinear classification derived from kernel methods. In the proposed method, kernel methods are used for enhancing the emission probability density functions (pdfs) of hidden Markov models (HMMs). Because the emission pdfs enhanced by kernel methods have sufficient nonlinear classification performance, mixture models such as Gaussian mixture models (GMMs), which might cause problems of overfitting and local optima, are not necessary in the proposed method. Unlike the methods used in earlier studies on sequential pattern classification using kernel methods, our method can be regarded as an extension of conventional HMMs, and therefore, it can completely model the transition of hidden states with the observed vectors. Therefore, our method can be applied to many applications developed with conventional HMMs, especially for speech recognition. In this paper, we carried out an isolated phoneme classification as a preliminary experiment in order to evaluate the efficiency of the proposed sequential pattern classifier. We confirmed that the proposed method achieved steady improvements as compared to conventional HMMs with Gaussian-mixture emission pdfs trained by the maximum likelihood and the maximum mutual information procedures.**

***Index Terms*—Discriminative training, hidden Markov models (HMMs), kernel methods, sequential pattern classifiers.**

I. Introduction

HIDDEN Markov models (HMMs) have been widely used in classification problems of sequential data, such as speech recognition, speaker recognition, handwriting recognition, and gesture recognition because of their extensibility. In such classification problems, nonlinear classification techniques are essential because feature vectors are not linearly separable in the natural feature space. To deal with such nonseparable sequences, kernel-based nonlinear classification techniques have been especially developed based on support vector machines (SVMs).

Several approaches can be used for carrying out kernel-based classification of sequential data based on SVMs [1]–[3], such as an approach that involves the use of SVMs with a kernel function that directly handles sequential data (sequential kernel) [3]–[5] and SVM/HMM hybrid approaches [6], [7] that involve the use of SVMs as static classifiers for the fixed alignment segments determined by HMMs. However, these approaches cannot explicitly hold the HMM representation, which makes it difficult to integrate them to large-scale systems straightforwardly. Because of the above-mentioned lack of the conventional SVM-based approaches, HMM-based approaches are still used in some sequential pattern classification problems, especially in speech recognition. In order to carry out a nonlinear classification, the current state-of-the-art HMM-based sequential classifiers introduce several discriminative training methods to HMMs [8]–[11] and use Gaussian mixture models (GMMs) with a large number of mixture components as emission probability density functions (pdfs) of HMMs. Since GMMs are capable of representing arbitrary pdfs, increasing the number of mixture components in GMMs could, in principle, lead to optimal nonlinear classification. However, the risk of local optima and overfitting also arises with an increase in the number of mixture components. The objective of this study is to prevent these risks by enhancing the emission pdfs of HMMs based on kernel methods.

In this paper, we propose a novel kernel machine for the classification of sequential data. Since the proposed method is formulated as a natural extension for conventional HMMs, our method can explicitly model the transition of hidden states behind the observed vectors. Therefore, the proposed method can be applied to many applications developed with conventional HMMs straightforwardly, especially for speech recognition. In addition, the proposed method can avoid the overfitting and local optima problems by using kernel-based nonlinear classification instead of mixture models.

We performed preliminary experiments that involved a phoneme classification task of speech data to show the effectiveness of our proposed method. It should be noted that kernel-based methods for classification require, in principle,

Manuscript received November 21, 2009; accepted February 19, 2010. Date of publication September 13, 2010; date of current version November 17, 2010. This work was supported in part by a Grant-in-Aid for JSPS Fellows (21·04190) from the Ministry of Education, Culture, Sports, Science, and Technology, Japan. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaodong He.

Y. Kubo was with the Department of Computer Science, Waseda University, Tokyo 169-8050, Japan. He is now with NTT Communication Science Laboratory, Kyoto 619-0237, Japan (e-mail: yotaro@ieee.org).

S. Watanabe and A. Nakamura are with NTT Communication Science Laboratories, NTT Corporation, Kyoto 619-0237, Japan (e-mail: watanabe@cslab.kecl.ntt.co.jp; ats@cslab.kecl.ntt.co.jp).

E. McDermott was with NTT Communication Science Laboratories, NTT Corporation, Kyoto 619-0237, Japan. He is now with Google, Inc., Mountain View, CA 94043 (e-mail: erikmcd@google.com).

T. Kobayashi is with the Department of Computer Science, Waseda University, Tokyo 169-8555, Japan (e-mail: koba@waseda.jp).

Digital Object Identifier 10.1109/JSTSP.2010.2076030

a computational cost of $O(l^3)$ for training and $O(lm)$ for evaluation, where l and m denote the numbers of frames in the training dataset and test dataset, respectively. Hence, evaluations on current standard corpora are prohibitive without any approximation, even if a small-sized corpus (e.g., TIMIT) is used for training and evaluation. In this paper, as an initial attempt, we focus on the exact performance of the proposed kernel machines by using a subset of the standard corpus (TIMIT). Since many approximation techniques aimed at the acceleration of kernel-based methods have been developed in the machine learning community [12], we consider that the proposed method can be scalably applied to large-scale corpora by applying appropriate approximation techniques. Therefore, we choose a phoneme classification problem in our evaluations as a normal sequential pattern classification problem.

The remainder of this paper is organized as follows. Section II briefly describes how kernel methods achieve nonlinear classification without mixture models. Section III defines the models used in the proposed method and a discriminant function that is the foundation for the proposed method. Section IV describes a parameter estimation method and a method for introducing kernel techniques into the estimation process of the model parameters. In Section V, the preliminary results of isolated phoneme classification experiments are presented and discussed.

II. Reproducing Kernel Hilbert Spaces

In this section, a method used to carry out nonlinear classification without using mixture models is described conceptually. The detailed formulations required for the application of this method to HMMs are described in Sections III and IV. Table I lists the mathematical notations used in this study.

Conventional classifiers based on probabilistic models use pdfs $P(\boldsymbol{x})$ in the input feature space $\boldsymbol{x} \in \mathcal{X}$ as models of feature vectors. Although the classification boundaries constructed by Gaussian pdfs are a quadratic surface in an input feature space $\mathcal{X}$, boundaries with higher order nonlinearities are required in most applications. Therefore, to enhance the representation of emission pdfs, mixtures of Gaussian pdfs are often used to construct accurate boundaries. However, the use of mixtures might introduce the risk of local optima and overfitting.

The objective of this study is to construct classifiers in a higher dimensional space $\mathcal{K} \stackrel{\text{def}}{=} \{\boldsymbol{\phi}(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\}$. It is well known that if an appropriate nonlinear warping function $\boldsymbol{\phi}$ is given, the optimal classification boundary can be represented as a linear function in a higher dimensional space $\mathcal{K}$. Therefore, simple pdfs obtained without using mixture models (e.g., exponential distributions) can be used as models for warped feature vectors, as shown in Fig. 1.

By using an appropriate kernel function $K : (\mathcal{X}, \mathcal{X}) \rightarrow \mathcal{R}$, there exists $\boldsymbol{\phi}$, which satisfies $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^{\top}\boldsymbol{\phi}(\boldsymbol{y})$. Therefore, $\boldsymbol{\phi}$ is not defined explicitly in general. If all operations in the higher dimensional space $\mathcal{K}$ can be written by using inner products in $\mathcal{K}$, the explicit representation and computation of $\boldsymbol{\phi}$ can be omitted by substituting $\boldsymbol{\phi}(.)^{\top}\boldsymbol{\phi}(.)$ with $K(.,.)$. The higher dimensional space $\mathcal{K}$ defined by the kernel function K is called the reproducing kernel Hilbert space (RKHS). SVMs, which are formulated as linear classifiers, achieve nonlinear classification by considering linear classification in an RKHS.

As an example, the average of warped feature vectors $\boldsymbol{\phi}(\boldsymbol{x}^i)$ in a dataset $\{\boldsymbol{\phi}(\boldsymbol{x}^i) | i \in [1 \ldots N]\}$ is discussed. The computation of the inner product between the average of warped feature vectors

TABLE I
Mathematical Notations

Symbol	Description
	Constants, Data and Hyper-parameters
D	Number of dimensions of feature vectors
S	Number of states in all HMMs
$\mathbf{I}$	D-dimensional identity matrix ($I \in \mathbb{R}^{D \times D}$)
$\mathcal{X}$	Feature space ($\mathcal{X} = \mathbb{R}^D$)
$\mathcal{K}$	Reproducing kernel Hilbert space
$\mathcal{W}$	Set of all possible word sequences
T^i	Length of i^{th} feature sequence in training dataset
X^i	i^{th} feature sequence in training dataset $X^i \stackrel{\text{def}}{=} \{\boldsymbol{x}^i(1), \boldsymbol{x}^i(2), \cdots, \boldsymbol{x}^i(T^i)\}$
$\boldsymbol{x}^i(t)$	t^{th} vector (frame) in feature sequence. $\boldsymbol{x}^i(t) \in \mathcal{X}$
W^i	i^{th} label sequence in training dataset
c	Hyper-parameter for prior pdf of slack variables $P^0(\xi)$
	Variables
i, i'	Index for training examples
t, t'	Index for frame numbers
s, s'	Index for states
X	Variable for feature sequences
$\boldsymbol{x}$	Variable for feature vectors ($x \in \mathcal{X}$)
W, W'	Variable for label sequences
q	Variable for state sequences
Λ	Parameter set of emission pdfs ($\Lambda = \{\boldsymbol{\lambda}_s | s \in [1..S]\}$)
$\boldsymbol{\lambda}_s$	Parameter vector of emission pdf at s^{th} state ($\boldsymbol{\lambda}_s \in \mathcal{K}$)
α	Set of Lagrange multipliers $\alpha = \{\alpha^i_W | \forall i, \forall W \neq W^i\}$
ξ	Set of slack variables $\xi = \{\xi^i_W | \forall i, \forall W \neq W^i\}$
	Functions and Functionals
$\hat{q}^i$	Viterbi state sequence obtained from X^i and W^i $\hat{q}^i \stackrel{\text{def}}{=} \{\hat{q}(1; X^i, W^i), \cdots, \hat{q}(t; X^i, W^i), \cdots\}$
$\hat{q}^i_W$	Viterbi state sequence obtained from X^i and a word sequence W $\hat{q}^i \stackrel{\text{def}}{=} \{\hat{q}(1; X^i, W), \cdots, \hat{q}(t; X^i, W), \cdots\}$
Ψ_s	Viterbi delta occupancy function
$\boldsymbol{\phi}$	Feature warping function
$K(\boldsymbol{x}, \boldsymbol{y})$	Kernel function ($K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^{\top}\boldsymbol{\phi}(\boldsymbol{y})$)
$\mathbb{1}(a, b)$	Indicator function that returns 1 when $a = b$ and 0 otherwise
$\mathcal{D}(X^i, W; \Lambda)$	Discriminant function
$\hat{\mathcal{D}}(X^i, W; \Lambda)$	Viterbi discriminant function
$J(\alpha)$	Dual-optimization function
$\mathrm{KL}[f(.) \| \| g(.)]$	Kullback-Leibler divergence of g from f
$\langle b(.) \rangle_{a(.)}$	Expectation of $b(.)$ over a pdf $a(.)$

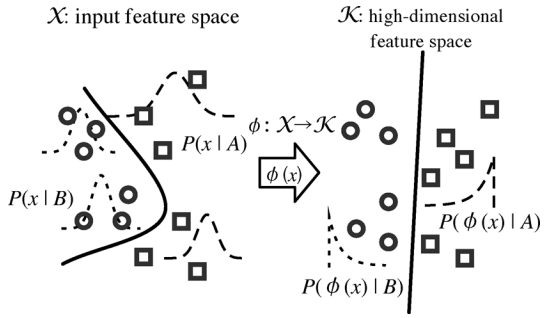


Fig. 1. Basic concept of using probability density functions in reproducing kernel Hilbert space.

and an input vector $\boldsymbol{\phi}(\boldsymbol{x})$ can be expressed by using the kernel function K as follows:

$$\boldsymbol{\phi}(\boldsymbol{x})^{\top}\left(\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\phi}(\boldsymbol{x}^i)\right)=\frac{1}{N}\sum_{i=1}^{N}K(\boldsymbol{x},\boldsymbol{x}^i). \tag{1}$$

Here, because of the summation $(\sum_{i=1}^{N})$ over K, the loop computation and storage attributed to all vectors $\boldsymbol{x}^i$ in the dataset are essential for kernel methods. This is the main cause of the computational complexity in kernel-based methods. However, several methods to eliminate this loop computation can be identified if an appropriate kernel function K is chosen [12], [13].

III. Hidden Markov Models With Log-Linear Emission PDFs

In this section, we formulate a classifier by introducing HMMs as generative models and a discriminant function that indicates the classification performance of the models. In this study, although we propose a general-purpose sequential classifier, we describe the proposed method in terms of speech recognition. Thus, "phoneme" refers to the elements in label sequences, and "language model" refers to the models of label sequences.

Model formulation described in this section includes explicit representation of feature warping function $\boldsymbol{\phi}$. Therefore, the straightforward implementation of the models described in this section might be impossible because the number of dimensions of $\boldsymbol{\phi}(\boldsymbol{x})$ might be infinite in general RKHSs. This problem is resolved by introducing a training method that can avoid the use of explicit representation of $\boldsymbol{\phi}(\boldsymbol{x})$; this method is described in Section IV.

A. Definition of Discriminant Function

Let $\{(X^i,W^i)|i\in[1\dots N]\}$ be a set of training data, where N is the number of examples in the training dataset. X^i is a sequence of D-dimensional feature vectors with length T^i, i.e., $X^i=\{\boldsymbol{x}^i(1),\cdots,\boldsymbol{x}^i(t),\cdots,\boldsymbol{x}^i(T^i)|\boldsymbol{x}^i(t)\in\mathcal{X}\}$, and W^i is the corresponding label (phoneme or word) sequence (classifier outputs), i.e., $W^i=\{w^i(1),\cdots,w^i(m),\cdots\}$.

Conventional HMM-based sequential pattern classifiers can be used to obtain a classification result $\hat{W}$ of an input feature sequence X by solving the following search problem:

$$\hat{W}=\arg\max_{W}\log P(W|X,\Lambda,\Pi,\Theta) \tag{2}$$

where Θ is a parameter of language models, Π a set of the transition probability matrices of the HMMs and Λ a set of parameters of emission pdfs. Since the objective of this study is to enhance the emission pdfs by kernel methods, the estimation of Θ and Π is not discussed here. Therefore, we have omitted Θ and Π for the sake of readability in the remainder of this paper.

First, we introduce a parametric discriminant function that indicates the performance of the model parameter Λ. By considering $W\neq W^i$ as a possible sequence of labels that is different from the correct label sequence W^i, we use the following discriminant function $\mathcal{D}(X^i,W;\Lambda)$:

$$\begin{aligned}\mathcal{D}(X^i,W;\Lambda)&\triangleq\log\frac{P(X^i,W^i|\Lambda)}{P(X^i,W|\Lambda)},\\&\triangleq\log\frac{P(X^i|W^i,\Lambda)P(W^i)}{P(X^i|W,\Lambda)P(W)}.\end{aligned} \tag{3}$$

Here, we assumed that the word sequences W,W^i are independent of the parameters of emission pdfs Λ.

In this discriminant function, when X^i is misclassified into an incorrect word sequence $W\neq W^i$, it is found that $\mathcal{D}(X^i,W;\Lambda)$ is less than 0 (i.e., the denominator is greater than the numerator in (3)). Therefore, in order to eliminate misclassifications, Λ should be estimated such that $\mathcal{D}(X^i,W;\Lambda)>0$ for all possible $W\neq W^i$. It should be noted that it is also possible to provide an alternative definition of the discriminant function $\mathcal{D}$ (e.g., *maximum mutual information estimation* (MMIE) criterion [14] and *minimum phone error* (MPE) criterion [10]). In this paper, a discriminant function that is similar to the one used in the *minimum classification error* (MCE) training of HMMs [9] is used because the MCE-type discriminant function yields large-margin criterion when combined with the model training method described in Section IV.

B. Hidden Markov Models With Log-Linear Emission PDFs

As in the case of conventional HMMs, we assume that the tth vector in an observed sequence $\boldsymbol{x}(t)$ depends on the tth HMM state $q(t)$ in a state sequence $q=\{q(1),\cdots,q(t),\cdots|q(t)\in[1\dots S]\}$, and q depends on a given word sequence W, where S is the number of HMM states. We also assume that the parameter Λ can be decomposed into parameters associated with a specific HMM state, i.e., $\Lambda=\{\boldsymbol{\lambda}_1,\cdots,\boldsymbol{\lambda}_s,\cdots\boldsymbol{\lambda}_S\}$, where s is an HMM state index. Then, (3) is expressed as follows:

$$\begin{aligned}\mathcal{D}(X^i,W;\Lambda)=\log\frac{\sum_{all\ q}\prod_t P\left(\boldsymbol{x}^i(t)|\boldsymbol{\lambda}_{q(t)}\right)P(q|W^i,X^i)}{\sum_{all\ q}\prod_t P\left(\boldsymbol{x}^i(t)|\boldsymbol{\lambda}_{q(t)}\right)P(q|W,X^i)}\\+\log\frac{P(W^i)}{P(W)}.\end{aligned} \tag{4}$$

Most applications of HMMs approximate the sum of probabilities over every possible state sequence $(\sum_{all\ q})$ by a probability calculated from a single Viterbi (maximum likelihood) path. Therefore, we used the following Viterbi discriminant function $\tilde{\mathcal{D}}(X^i,W;\Lambda)$ instead of (4):

$$\begin{aligned}\tilde{\mathcal{D}}(X^i,W;\Lambda)\stackrel{\text{def}}{=}\sum_t\log\frac{P\left(\boldsymbol{x}^i(t)|\boldsymbol{\lambda}_{\hat{q}(t;X^i,W^i)}\right)}{P\left(\boldsymbol{x}^i(t)|\boldsymbol{\lambda}_{\hat{q}(t;X^i,W)}\right)}\\+\log\frac{P(\hat{q}^i)P(W^i)}{P\left(\hat{q}^i_W\right)P(W)}\end{aligned} \tag{5}$$

where $\hat{q}^i$ denotes a Viterbi path for the correct word sequence W^i and $\hat{q}^i_W$ denotes a Viterbi path for an incorrect word sequence W. Further, $\hat{q}(t;X^i,W^i)$ and $\hat{q}(t;X^i,W)$ are tth elements in $\hat{q}^i$ and $\hat{q}^i_W$, respectively. $\hat{q}^i$ and $\hat{q}^i_W$ are expressed as follows:

$$\begin{aligned}\hat{q}^i_W&\stackrel{\text{def}}{=}\left\{\hat{q}(1;X^i,W),\cdots,\hat{q}(t;X^i,W),\cdots\right\},\\&\stackrel{\text{def}}{=}\arg\max_q P(q,X^i|W,\Lambda)\\\hat{q}^i&\stackrel{\text{def}}{=}\left\{\hat{q}(1;X^i,W^i),\cdots,\hat{q}(t;X^i,W^i),\cdots\right\}\\&\stackrel{\text{def}}{=}\arg\max_q P(q,X^i|W^i,\Lambda).\end{aligned} \tag{6}$$

It should be noted that the Viterbi paths depend on Λ.

As an emission pdf, we consider a model for a vector $\boldsymbol{x}^i(t)$ in a sequence as a log-linear model in an RKHS, as follows:

$$P\left(\boldsymbol{x}^i(t)|\boldsymbol{\lambda}_s\right) = \frac{1}{Z_{\boldsymbol{\phi}}(\boldsymbol{\lambda}_s)} \exp\left\{\boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}\left(\boldsymbol{x}^i(t)\right)\right\},$$
$$Z_{\boldsymbol{\phi}}(\boldsymbol{\lambda}_s) = \int_x \exp\left\{\boldsymbol{\lambda}_s^{\mathsf{T}} \boldsymbol{\phi}(\boldsymbol{x})\right\} dx. \quad (7)$$

Here, $\boldsymbol{\lambda}_s$ is a weight vector in a log-linear model, $\boldsymbol{\phi}$ a feature warping function (as described in Section II), and $Z_{\boldsymbol{\phi}}$ a partition function obtained by marginalizing out a vector $x \in \mathcal{X}$. The likelihood evaluation form of the proposed HMMs is very similar to that of HCRFs [15], [16]. It should be noted that although kernel machines based on HCRFs are not realized in [15] and [16], the proposed extensions can also be applied to HCRFs. In this paper, we focused on HMM-based kernel machines.

In general cases, the integral in $Z_{\boldsymbol{\phi}}$ is intractable. Here, we omit the calculation of $Z_{\boldsymbol{\phi}}$ and assume it to be constant, as in [11]. By substituting (7) into (5) and by omitting the $Z_{\boldsymbol{\phi}}$, we obtain the discriminant function $\tilde{\mathcal{D}}$ as follows:

$$\tilde{\mathcal{D}}(X^i, W; \Lambda) = \sum_t \left(\boldsymbol{\lambda}_{\hat{q}(t;X^i,W^i)} - \boldsymbol{\lambda}_{\hat{q}(t;X^i,W)}\right)^{\mathsf{T}} \boldsymbol{\phi}\left(\boldsymbol{x}^i(t)\right) + \log \frac{P(\hat{q}^i)P(W^i)}{P\left(\hat{q}^i_W\right)P(W)} + \text{const.} \quad (8)$$

Because of the omission of the normalization term $Z_{\boldsymbol{\phi}}$ in (8), the non-normalized log-likelihood $\boldsymbol{\lambda}_s^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{x})$ is used to compute the emission probability at the state s in our methods. Therefore, hereinafter, we refer to this quantity (non-normalized log-likelihood) as a score. While conventional methods use GMMs to model $P(X^i(t)|s, \Lambda)$, the proposed method uses the simple pseudo-probabilistic distribution $\exp(\boldsymbol{\lambda}_s^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{x}))$ to apply kernel methods.

IV. Minimum Relative Entropy Discrimination Training

In this section, we describe an estimation method for parameters of emission pdfs $\boldsymbol{\lambda}_s$ used in the discriminant function [(8)].

Although several frameworks have been identified for estimating the parameters, we propose a training method derived from the minimum relative entropy discrimination (MRED) framework,[1] which enables the use of prior distributions and hidden variables, as proposed by Jebara *et al.* [17]. As a result of using the MRED framework, several extensions for MRED can be used in future works. For example, multistream speech recognition [18] can also be integrated into this framework by employing dynamic kernel combination methods for MRED [19].

The remainder of this section is organized as follows. First, in Section IV-A, the formulation of MRED, a general solution of posterior pdf, and a general form of the objective function are presented and described. Then, in Section IV-B, an analytical posterior pdf and an analytical objective function are derived by introducing conjugate prior pdfs. Finally, in Section IV-C, a method for avoiding the explicit representation of the feature warping function $\boldsymbol{\phi}(\boldsymbol{x})$ by plugging in a kernel function $K(\boldsymbol{x}, \boldsymbol{y})$ into the objective function and the emission pdfs is described.

A. Minimum Relative Entropy Discrimination Framework

In MRED, the training of classifiers is formulated as a convex optimization problem, where MRED treats all variables in convex optimization (both the parameters Λ and the slack variables ξ) as random variables. By representing these random variables as distributions, regularization can be performed by minimizing the Kullback–Leibler divergence (KL divergence) of the prior distribution $P^0(\Lambda, \xi)$ from the posterior distribution $P(\Lambda, \xi)$ under the discriminative constraints. We emphasize that MRED can estimate a model parameter even if the model is not a probabilistic model. Therefore, the omission of the normalization term $Z_{\boldsymbol{\phi}}$ in the discriminant function $\tilde{\mathcal{D}}$ [(8)] is not crucial in the MRED training process.

The primary problem of this optimization is expressed as follows:

$$\underset{P(\Lambda,\xi)}{\text{minimize}}\ \text{KL}\left[P(\Lambda,\xi)\|P^0(\Lambda,\xi)\right],$$
$$\text{subject to}\ \left\langle \tilde{\mathcal{D}}(X^i, W; \Lambda) - \xi^i_W \right\rangle_{P(\Lambda,\xi)} \geq 0,$$
$$\forall i, \forall W \neq W^i. \quad (9)$$

Here, $\langle f(x)\rangle_{g(x)}$ is the expectation of $f(x)$ over the distribution $g(x)$, that is, $\langle f(x)\rangle_{g(x)} \stackrel{\text{def}}{=} \int_x g(x)f(x)dx$. $\text{KL}[f(x)\|g(x)]$ is the KL divergence of $g(x)$ from $f(x)$, and it is defined as follows:

$$\text{KL}\left[f(x)\|g(x)\right] \stackrel{\Delta}{=} \left\langle \log f(x) - \log g(x)\right\rangle_{f(x)}. \quad (10)$$

$\xi = \{\xi^i_W | \forall i, \forall W \neq W^i\}$ is a set of slack variables. Each slack variable corresponds to each constraint (i.e., each i and $W \neq W^i$) in the optimization. By decreasing the slack variable ξ^i_W, the area of the feasible region of the constraint can be increased. However, in general, decrease in slack variables is penalized by introducing a prior pdf $P^0(\xi^i_W)$ that favors larger slack variables.

By considering the Lagrange functional of the above optimization problem and from the Karush–Kuhn–Tucker conditions (KKT conditions), we can obtain the following posterior distribution by using the variational method:

$$P(\Lambda,\xi) \propto P^0(\Lambda,\xi) \exp\left[\sum_{i,W\neq W^i} \alpha^i_W \left(\tilde{\mathcal{D}}(X^i, W; \Lambda) - \xi^i_W\right)\right], \quad \alpha^i_W \geq 0. \quad (11)$$

Here, $\alpha \stackrel{\text{def}}{=} \{\alpha^i_W \geq 0 | \forall i, \forall W \neq W^i\}$ is the set of Lagrange multipliers of this optimization problem [(9)]. Similar to slack variables, the Lagrange multipliers are also introduced for each constraint in the optimization.

Then, the primary problem [(9)] with the $P(\Lambda, \xi)$ optimization is replaced with the following dual problem with α optimization as follows:

$$\underset{\alpha}{\text{maximize}}\ J(\alpha),$$
$$\text{subject to}\ \alpha^i_W \geq 0, \qquad \forall i, \forall W \neq W^i$$

[1] Conventionally, this framework is also known as "maximum entropy discrimination." However, we use a more specific notation, i.e., "minimum relative entropy discrimination," because the maximum entropy property can only be acquired when specific prior pdfs are used. The author of [17] also uses this specific notation in several papers.

where

$$J(\alpha) = -\log Z(\alpha),$$
$$Z(\alpha) = \left\langle \exp \sum_{i,W} \alpha_W^i \left(\tilde{\mathcal{D}}(X^i, W; \Lambda) - \xi_W^i \right) \right\rangle_{P^0(\Lambda,\xi)}. \quad (12)$$

The detailed derivations of the dual problem are described in Appendix A.

B. Definitions of Prior PDFs and Derivations of Closed-Form J

In this section, we derive the closed-form expressions of the posterior pdf $P(\Lambda)$ and the objective function $J(\alpha)$ by introducing conjugate prior pdfs into (11) and (12), respectively. Here, we assume that the prior pdf $P^0(\Lambda, \xi)$ can be decomposed into the product of the prior pdf of the parameter of each HMM state $P^0(\boldsymbol{\lambda}_s)$ and that of the slack variable of each constraint $P^0(\xi_W^i)$ as follows:

$$P^0(\Lambda, \xi) \stackrel{\text{def}}{=} \prod_s P^0(\boldsymbol{\lambda}_s) \prod_{i, W \neq W^i} P^0\left(\xi_W^i\right). \quad (13)$$

As in the case of large-margin methods including soft-margin SVMs, we perform regularization by minimizing the L2-norm of weight vectors $\|\boldsymbol{\lambda}_s\|^2$ and the L1-norm of slack variables $\|\xi_W^i\|^1$. Since KL-divergence is defined as the expectation of the difference between log-likelihoods of two distributions [(10)], the functional forms of regularization terms used in this method are identical to the logarithm of prior pdfs. Therefore, these regularization criteria are realized by employing Gaussian distributions as the priors of the parameter $\boldsymbol{\lambda}_s$ and exponential distributions as the priors of slack variables ξ_W^i. The prior pdfs are expressed as follows:

$$P^0(\boldsymbol{\lambda}_s) \stackrel{\text{def}}{=} \mathcal{N}(\boldsymbol{\lambda}_s | 0, \mathbf{I})$$
$$P^0\left(\xi_W^i\right) \stackrel{\text{def}}{=} \begin{cases} \frac{1}{c} \exp\left\{-c\left|\delta(W^i, W) - \xi_W^i\right|\right\}, & \xi_W^i < \delta(W^i, W) \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

where $\delta(W^i, W)$ is the label similarity between W and W^i. The determinant of the covariance matrix of $P^0(\boldsymbol{\lambda}_s)$ and the hyperparameter c correspond to the weight variable in soft-margin SVM, which control the tradeoff between empirical error minimization and margin maximization. We simplify the prior distribution by setting the covariance matrix in $P^0(\boldsymbol{\lambda}_s)$ as $\mathbf{I}$, without any loss of generality. c is scaled appropriately. These prior settings lead to analytical and explicit solutions of the posterior pdfs ($P(\boldsymbol{\lambda}_s)$ and $P(\xi_W^i)$), because the prior pdfs given in (14) can be assumed to be conjugate prior pdfs.

In discriminative training methods, it is important to determine which measurement of error should be minimized. For example, MCE [9] attempts to minimize the sequence-level error that is "0" when all elements in a hypothesis sequence W are correct, and "1" otherwise. Because this measurement is coarse and is difficult to minimize, recent approaches measure the impact of an error hypothesis by introducing a fine error measurement. For example, MPE [10] uses the approximated phoneme-level edit distance of label sequences, and LM-HMM [11] uses Hamming distance (frame-level error measurement) between the Viterbi sequence of the correct label sequence and that of a hypothesis sequence W. In the proposed method, several error measurements can be incorporated by designing label similarity function $\delta(W^i, W)$ in the prior pdf of slack variables ξ_W^i [(14)]. Because the optimization attempts to ensure that the value of the discriminant function $\tilde{\mathcal{D}}(.)$ is higher than that of ξ_W^i [(9)], setting of $\delta(W^i, W)$, which is equivalent to the mode value of the prior pdf $P^0(\xi_W^i)$, is equivalent to designing the error measurement that need to be minimized. The definition of the label similarity function is provided in the experimental sections.

We now obtain the following posterior distribution of parameters $P(\Lambda|\alpha)$[2] by substituting the prior distributions $P^0(\Lambda, \xi)$ [(13) and (14)) into the posterior pdf ((11)), as follows:

$$\begin{aligned} P(\Lambda|\alpha) &= \prod_t P(\boldsymbol{\lambda}_s|\alpha) \\ P(\boldsymbol{\lambda}_s|\alpha) &\propto \underbrace{\mathcal{N}(\boldsymbol{\lambda}_s|0, I)}_{\text{Prior pdf}} \\ &\quad \times \exp\left[\sum_{i, W \neq W^i} \alpha_W^i \left(\tilde{\mathcal{D}}(X^i; W, \Lambda) - \xi_W^i \right) \right] \\ &\propto \exp\left\{ -\frac{1}{2} \|\boldsymbol{\lambda}_s\|^2 \right\} \\ &\quad \times \exp\underbrace{\left[\sum_{i, W \neq W^i} \alpha_W^i \sum_t \Psi_s(t; i, W) \boldsymbol{\phi}\left(\boldsymbol{x}^i(t)\right)^\top \boldsymbol{\lambda}_s \right]}_{\hat{\boldsymbol{\lambda}}_s^\top \lambda_s} \\ &\propto \underbrace{\mathcal{N}\left(\boldsymbol{\lambda}_s | \hat{\boldsymbol{\lambda}}_s(\alpha), I\right)}_{\text{Posterior pdf}} \end{aligned} \quad (15)$$

where

$$\hat{\boldsymbol{\lambda}}_s(\alpha) \stackrel{\text{def}}{=} \sum_{i, W \neq W^i} \alpha_W^i \sum_t \Psi_s(t; i, W) \boldsymbol{\phi}\left(\boldsymbol{x}^i(t)\right)$$
$$\Psi_s(t; i, W) \stackrel{\text{def}}{=} \mathbb{1}\left(\hat{q}(t; X^i, W^i), s\right) - \mathbb{1}\left(\hat{q}(t; X^i, W), s\right). \quad (16)$$

Here, $\mathbb{1}(x, y)$ is an indicator function that returns 1 when $x = y$ and 0 otherwise, $\Psi_s(t; i, W)$ denotes the difference between the occupation probability of the tth frame in the ith feature sequence of the correct Viterbi path $\hat{q}^i$ and that of the incorrect Viterbi path $\hat{q}_W^i$.

Then, we focused on the objective function $J(\alpha)$ in (12). By solving the integral in the objective function with given priors [(14)], we analytically obtain a closed-form expression for $J(\alpha)$ that can be factorized into a sum of parameter terms $J_{\boldsymbol{\lambda}_s}$, slack variable terms $J_{\xi_W^i}$, and hidden variable terms $J_{q_W^i}$, as follows:

$$J(\alpha) = \sum_s J_{\boldsymbol{\lambda}_s}(\alpha) + \sum_{i, W \neq W^i} \left(J_{\xi_W^i}(\alpha) + J_{q_W^i}(\alpha) \right). \quad (17)$$

Here, the parameter term can be written as follows:

$$J_{\boldsymbol{\lambda}_s}(\alpha) = -\left\| \hat{\boldsymbol{\lambda}}_s(\alpha) \right\|^2. \quad (18)$$

[2]Since the posterior pdf $P(\Lambda)$ depends on the Lagrange multiplier α, as shown in (11), we used the notation "$P(\Lambda|\alpha)$" in the rest of this paper.

The term $J_{\boldsymbol{\lambda}_s}$ involves the L2-regularization criterion of the parameter vector $\boldsymbol{\lambda}_s$.

The other terms represent the loss function used in MRED that causes an increase in the Lagrange multipliers α_W^i such that the discriminative constraints are satisfied, as follows:

$$J_{\xi_W^i}(\alpha) = \delta(W, W^i)\alpha_W^i + \log\left(c - \alpha_W^i\right)$$
$$J_{q_W^i}(\alpha) = -\alpha_W^i \left(\log \frac{P(\hat{q}^i)P(W^i)}{P\left(\hat{q}_W^i\right)P(W)}\right). \quad (19)$$

Thus, the α optimization can be solved by maximizing (18) and (19). Further, the optimal posterior pdf can be obtained as $P(\Lambda|\hat{\alpha})$, where $\hat{\alpha} = \arg\max_\alpha J(\alpha)$.

C. Kernel-Based Representations of $J_{\boldsymbol{\lambda}_s}$ and $P(\boldsymbol{x}|\hat{\boldsymbol{\lambda}}_s)$

As a result of deriving the dual problem, the parameter term of the objective function [$J_{\boldsymbol{\lambda}_s}(\alpha)$ in (18)] can be rewritten as the weighted sum of the inner product between $\boldsymbol{\phi}(\boldsymbol{x})$'s. As discussed in Section II, since the objective function can be expressed by using the inner product of warped features $\boldsymbol{\phi}(\boldsymbol{x})^\top\boldsymbol{\phi}(\boldsymbol{y})$, it is not necessary to explicitly represent the warped features $\boldsymbol{\phi}(\boldsymbol{x})$. The parameter term of the objective function can be rewritten by using the kernel function $K(\boldsymbol{x},\boldsymbol{y}) \stackrel{\text{def}}{=} \boldsymbol{\phi}(\boldsymbol{x})^\top\boldsymbol{\phi}(\boldsymbol{y})$ as follows:

$$J_{\boldsymbol{\lambda}_s}(\alpha) = -\sum_{i',W'\neq W^{i'}} \sum_{i,W\neq W^i} \sum_{t,t'} \alpha_W^i \alpha_{W'}^{i'} \Psi_s(t;i,W) \times \Psi_s(t';i',W') K\left(\boldsymbol{x}^i(t), \boldsymbol{x}^{i'}(t')\right). \quad (20)$$

When this representation of the parameter term $J_{\boldsymbol{\lambda}_s}(\alpha)$ is used, it is found that the explicit representation of $\boldsymbol{\phi}$ is removed from all the terms in the objective function [(18) and (19)], and the kernel-based representation can be used to compute the objective function. The practical solver for α optimization is described in Appendix B.

In the evaluation phase (including Viterbi path computation), the score of an unknown input vector $\boldsymbol{x}$ can be evaluated by marginalizing out parameter $\boldsymbol{\lambda}_s$ from the posterior pdf $P(\boldsymbol{\lambda}_s|\hat{\alpha})$ as follows:

$$\langle \log P(\boldsymbol{x}|\boldsymbol{\lambda}_s)\rangle_{P(\boldsymbol{\lambda}_s|\alpha)} = \left\langle \boldsymbol{\phi}(\boldsymbol{x})^\top\boldsymbol{\lambda}_s \right\rangle_{\mathcal{N}(\lambda_s|\hat{\lambda}_s(\hat{\alpha}))} = \boldsymbol{\phi}(\boldsymbol{x})^\top\hat{\boldsymbol{\lambda}}_s(\hat{\alpha}). \quad (21)$$

As in the case of the objective function, the kernel-based representation of the score can be obtained by substituting (21) into (16) as follows:

$$\langle \log P(\boldsymbol{x}|\boldsymbol{\lambda}_s)\rangle_{P(\boldsymbol{\lambda}_s|\alpha)} = \boldsymbol{\phi}(\boldsymbol{x})^\top \sum_{i,W\neq W^i} \alpha_W^i \sum_t \Psi_s(t;i,W)\boldsymbol{\phi}\left(\boldsymbol{x}^i(t)\right) = \sum_{i,W\neq W^i} \alpha_W^i \sum_t \Psi_s(t;i,W) K\left(\boldsymbol{x}^i(t), \boldsymbol{x}\right). \quad (22)$$

Similar to the objective function, the explicit representation of $\boldsymbol{\phi}$ is not necessary in the score evaluation procedure.

Thus, we obtained the kernel machines that could handle RKHS via kernel function K. We termed the models specified by $(\hat{\alpha}, \Psi, K)$ as "hidden Markov kernel machines (HMKMs)." As described in the previous section, although HMKM is a kernel machine, the model formulation of HMKM can be treated as that of standard HMMs. Therefore, the scheme of proposed method is similar to that of conventional HMMs trained by discriminative training methods.

V. Phoneme Classification Experiments

In order to evaluate the performance of the proposed method as a sequential classifier, we performed isolated phoneme classification experiments to compare the proposed method with the conventional HMMs that use GMMs as emission pdfs (continuous density HMMs; CD-HMMs).

The objective of the experiments in this section is to evaluate the exact performance of the proposed method, and therefore, approximation techniques of kernel machines are not applied. Since the training session of kernel machines requires enormous computational resources, it is unrealistic to evaluate the exact performance of the proposed method using a large-scale dataset, as discussed in Section I. Therefore, we restricted the amount of training datasets used in the experiments.

A. Experimental Setup

We compared our method with conventional GMM-based CD-HMMs using two training methods, i.e., maximum-likelihood estimation (MLE) and maximum mutual information estimation (MMIE) [8]. The extended Baum-Welch (EBW) algorithm is used to implement the optimization of MMIE. Although MLE is the most widely used estimation method for CD-HMMs, MLE procedures are not designed for minimizing classification error. Therefore, we also compared our method with the most popular discriminative training method MMIE.[3]

We prepared training datasets of three sizes (*small*, *medium* and *large*) and one test dataset for isolated phoneme classification experiments by segmenting the TIMIT dataset according to the label information. There is no overlap between the speakers of the test dataset and those of the training dataset. Table II summarizes the details of the datasets. All acoustical models in these experiments were constructed as gender-independent models. All feature vectors in the training and test data were whitened by using statistics (covariance matrix and average vector) obtained from the training dataset; the whitening operation is commonly used for the training of discriminative models. In the experiments, both the conventional CD-HMMs and the proposed method have left-to-right 3 states for each 39 phoneme categories defined in [21]. Configurations for acoustical analysis are summarized in Table III.

The following Gaussian kernel was used in the experiments:

$$K(\boldsymbol{x},\boldsymbol{y}) = \exp\left\{-\gamma\|\boldsymbol{x}-\boldsymbol{y}\|^2\right\} \quad (23)$$

where γ denotes a hyper-parameter. The Gaussian kernel is widely used in kernel machines because the number of dimen-

[3]It is reported that the performance of MMIE is similar to that of other discriminative training methods, such as MCE [20].

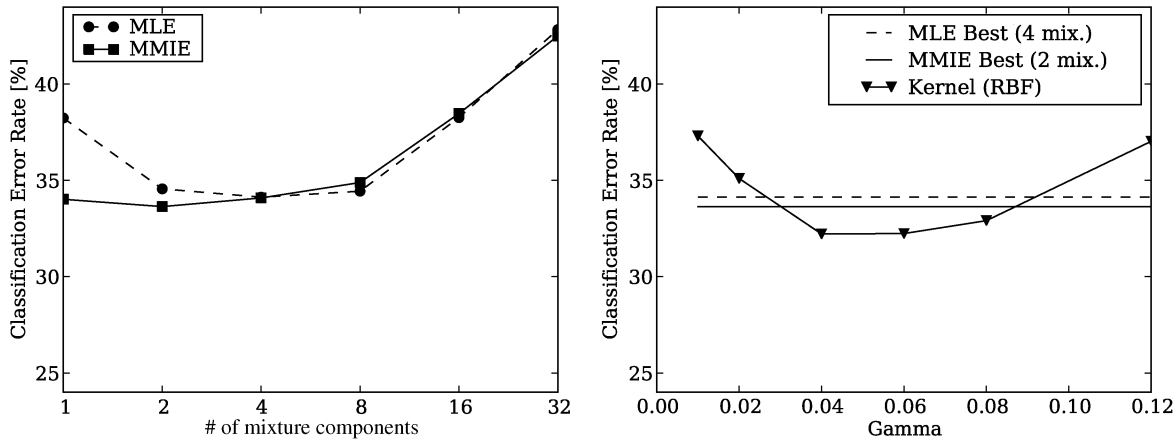


Fig. 2. Left: classification error rates of CD-HMMs trained by maximum-likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: classification error rates of hidden Markov kernel machines and CD-HMMs. (*small* dataset).

TABLE II
DATASET DESCRIPTION

# categories	39 (defined in [21])		
Training set	*Small*	*Medium*	*Large*
# segments	3,089	9,275	26,208
# frames	25,390	77,463	219,675
Test set			
# segments		4,243	
# frames		36,790	

TABLE III
ACOUSTICAL ANALYSIS CONFIGURATION; Δ DENOTES TIME-DOMAIN DERIVATIVE OF FEATURE SEQUENCE

Sampling rate	16 kHz
Quantization	16 bits
Feature vector	MFCC (12 dims.), Energy, Δ MFCC, Δ Energy, $\Delta\Delta$ MFCC, $\Delta\Delta$ Energy. (Total: 39 dims.)
Window len./ shift	25 ms / 10 ms

sions of $\boldsymbol{\phi}(\boldsymbol{x})$, which satisfies $K(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^{\top}\boldsymbol{\phi}(\boldsymbol{y})$, is infinite when Gaussian kernels are used as K [22].

We defined the Hamming distance between the Viterbi sequences computed from the given word sequences as a label similarity $\delta(W^i, W)$ in (14), as follows:

$$\delta(W^i, W) = \sum_{t=1}^{T^i} \left(1 - \mathbb{1}\left(\hat{q}^i(t), \hat{q}^i_W(t)\right)\right). \tag{24}$$

The Hamming-distance-based measurement of label similarity is widely used in the discriminative training methods (e.g., LM-HMMs [11] use this measurement); therefore, we also used this label similarity function in our experiments. The Hamming distance between two phonemes is identical to the number of frames in the sequence ($\delta(W^i, W) = T^i$) because the experiments in this section are isolated phoneme classification experiments (i.e., $\hat{q}^i(t)$ and $\hat{q}^i_W(t)$ are always different for all possible $W \neq W^i$ and t). The hyper-parameter c was set to five empirically, and the hyper-parameter γ was varied to examine the behavior of the proposed method.

B. Discussions on Classification Performance

Figs. 2–4 show the comparisons between the phoneme classification error rate of the proposed model and the conventional CD-HMMs trained by *small*, *medium*, and *large* datasets, respectively. The numbers of mixture components in the conventional CD-HMMs are varied, as shown in these figures.

From the experimental results, we confirmed that our kernel-based models steadily reduce the classification errors. In comparison with CD-HMMs trained by the standard MLE procedure, the proposed kernel machines (HMKMs) reduced the errors by 5.6%, 6.1%, and 10.3% relatively over *small*, *medium*, and *large* datasets, respectively, under the best condition of each method. In comparison with CD-HMMs with a discriminative training procedure (MMIE), HMKMs reduced the errors by 4.2%, 5.4%, and 5.2% relatively over *small*, *medium*, and *large* datasets, respectively, under the best condition of each method. Therefore, it was concluded that the proposed method achieved improvements in terms of reducing the errors in comparison with conventional CD-HMMs, with the best setting of the number of mixture components for all training datasets.

From Fig. 2 (*small* dataset), we confirmed that the performances of CD-HMMs are saturated by increasing the number of mixture components. In particular, we found that the performance of the discriminative training method (MMIE) degraded for the models with a large number of mixture components. We consider that these results are attributed to overfitting problems. However, the proposed method achieved lower error rates even under such conditions. We consider that this advantage results from the L2-regularization introduced to $\boldsymbol{\lambda}_s$. As in the case of SVMs, the L2-regularization introduced by Gaussian prior [(14)] yields large-margin classifiers that have advantages in generalization ability.

As shown in Fig. 4 (*large* dataset), although the overfitting problems might be avoided due to sufficient amounts of data, the relative advantages of the proposed method are confirmed. We consider that this relative advantage probably results from the prevention of problems arising from local optima. Because our method can prevent the risk of local optima by avoiding mixture models, the problems arising from local optima might be avoided as compared to those occurring in conventional CD-HMMs with a large number of mixture components.

Further, we observed that setting of the hyper-parameter γ was not so sensitive to classification performance in the pro-

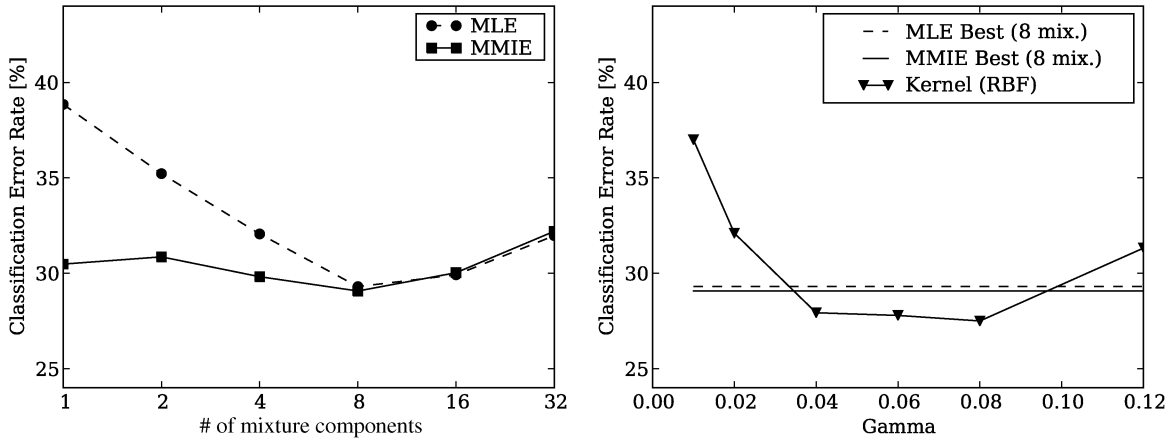


Fig. 3. Left: classification error rates of CD-HMMs trained by maximum-likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: classification error rates of hidden Markov kernel machines and CD-HMMs. (*medium* dataset).

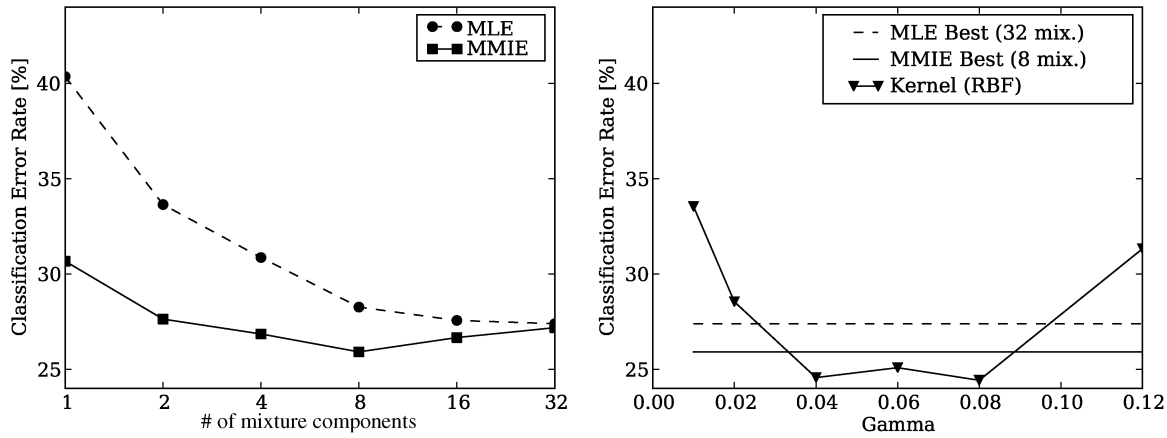


Fig. 4. Left: classification error rates of CD-HMMs trained by maximum-likelihood estimation (MLE) and maximum mutual information estimation (MMIE). Right: classification error rates of hidden Markov kernel machines and CD-HMMs. (*large* dataset).

posed method as compared to the setting of the number of mixture components in the GMM methods. For example, in the case of conventional CD-HMMs, it is observed that the number of mixture components that achieved the best performance in the evaluation of the *small* dataset yielded poor performance in the evaluation of the *large* dataset. On the other hand, the hyper-parameter γ that achieved the best performance in the evaluation of the *small* dataset ($\gamma = 0.04$) also caused performance improvements as compared with CD-HMMs in the evaluation of the *large* dataset. In the experiments, improvements were confirmed in the range $0.4 \leq \gamma \leq 0.8$, even when the amount of training datasets was varied. This property is important for a practical situation since we do not need to tune γ for new datasets, unlike the number of mixture components used in conventional CD-HMMs.

Therefore, we confirmed that the problems associated with CD-HMMs with GMM-type emission pdfs, i.e., overfitting and local optima, are avoided in HMKMs. Further, we confirmed that HMKMs with Gaussian kernel offer an advantage in terms of tuning parameters.

C. Discussions on Sparseness

Here, we examined the number of nonzero Lagrange multipliers obtained in the above experimental results, which is used to evaluate the generalization ability of SVMs. As shown in (15), the sequence X^i, corresponding to $\alpha^i_W = 0$, does not disturb the estimated posterior pdf $P(\Lambda|\alpha)$ even if it is removed from the training set. In addition, because the zero Lagrange multiplier ($\alpha^i_W = 0$) indicates that the inequality constraint attributed to ith training sequence and an incorrect label W is satisfied, the sequence is certainly not misclassified into the incorrect label W by using the estimated posterior pdf $P(\Lambda)$ when $\alpha^i_W = 0$. These two properties of Lagrange multipliers indicate that X^i with $\alpha^i_W = 0$ is not misclassified into the incorrect label W by using a posterior pdf estimated from the remaining training data. Therefore, a decrease in the number of nonzero α^i_W leads to a better performance in leave-one-out cross-validation (LOO-CV), which is commonly used to estimate the generalization performance of SVMs.

In the isolated phoneme classification experiments described in this section, the number of the Lagrange multipliers M corresponds to the product of the number of sequences in the training dataset N and the number of error hypothesis, i.e., $M \stackrel{\text{def}}{=} |\{(i,W)|i \in [1,N], W \neq W^i\}|$. Table IV lists the number of nonzero multipliers $M_+ \stackrel{\text{def}}{=} |\{(i,W)|\alpha^i_W \neq 0, i \in [1,N], W \neq W^i\}|$ and the ratio of M_+ to M. From the table, we confirmed that the proposed method also leads to sparse solutions. The ratios of nonzero multipliers in the experiments

TABLE IV
NUMBER OF NONZERO LAGRANGE MULTIPLIERS M^+ IN ESTIMATED MODELS THAT ACHIEVED THE BEST PERFORMANCE ON EACH DATASET, AND THE RATIO OF M^+ TO THE NUMBER OF LAGRANGE MULTIPLIERS M

Dataset	*Small*	*Medium*	*Large*
The best setting of γ	0.04	0.08	0.08
M^+	9843	35771	37518
M^+/M	8.4%	10.1%	3.8%

were less than or around 10%, as shown in Table IV (8.4%, 10.1%, and 3.8%, respectively), and therefore, we can say that the proposed method achieved good generalization ability.

Further, a sparse solution is also important for reducing the computational complexity. As mentioned in Section II, loop computation over the training data is essential in kernel-based methods. However, if α_W^i is 0, the computation due to vectors $\boldsymbol{x}^i(t)$, which are related to α_W^i, can be omitted. Thus, the computational cost required for evaluating HMKMs can be reduced to M^+/M. Although training and evaluation still require a considerably high computational cost, the proposed method is effective in comparison with kernel-based methods, which yield dense solutions.

VI. CONCLUSION

In this paper, a method for sequential pattern classification derived from kernel methods was proposed; this method is called the HMKM. In the proposed method, vectors in the input sequences are warped to a high-dimensional feature space (reproducing kernel Hilbert space; RKHS) defined by a kernel function and then modeled (HMMs with log-linear emission pdfs. Nonlinear classification is achieved without using mixture models by using emission pdfs in RKHS.

The efficiency of the proposed method is confirmed by isolated phoneme classification experiments. The experimental results show that the proposed method outperforms conventional hidden Markov models that use Gaussian mixture models as emission pdfs.

In future, we intend to reduce the computational costs of training and evaluation by using approximation techniques, aiming for acceleration of kernel-based methods developed in the machine learning community [12], [13]. Then, we also intend to apply our method to large-scale problems, e.g., large-vocabulary continuous speech recognition.

APPENDIX A
DERIVATIONS OF DUAL PROBLEMS

To derive the dual problem, we first introduce the Lagrange functional of the primary problem [(9)], as follows:

$$\begin{aligned} &L(\alpha,\beta)\left[f(\Lambda,\xi)\right] \\ &\quad=\left\langle \log f(\Lambda,\xi)-\log P^0(\Lambda,\xi)\right\rangle_{f(\Lambda,\xi)} \\ &\qquad-\sum_{i,W\neq W^i}\alpha_W^i\left\langle \tilde{\mathcal{D}}(X^i,W;\Lambda)-\xi_W^i\right\rangle_{f(\Lambda,\xi)} \\ &\qquad-\beta\left(\int_\Lambda\int_\xi f(\Lambda,\xi)d\xi d\Lambda-1\right). \end{aligned} \tag{25}$$

Here, α and β are Lagrange multipliers; $f(\Lambda,\xi)$, an argument function that represents a posterior pdf. α_W^i must remain nonnegative.

From the KKT conditions, it is found that the solution of the primary problem $P(\Lambda,\xi)$ is located on the saddle point of the Lagrange functional. By applying the variational method to the Lagrange functional, we obtain the following relational expression:

$$\begin{aligned} \frac{\delta}{\delta f(\Lambda,\xi)}L(\alpha,\beta)\left[f(\Lambda,\xi)\right] &= 1+\log f(\Lambda,\xi)-\log P^0(\Lambda,\xi) \\ &\quad-\sum_{i,W\neq W^i}\alpha_W^i\left(\tilde{\mathcal{D}}(X^i,W;\Lambda)-\xi_W^i\right)-\beta=0. \end{aligned} \tag{26}$$

Using this equation, we obtain the optimal posterior $P(\Lambda,\xi)$ as follows:

$$\begin{aligned} P(\Lambda,\xi) &= e^{\beta-1}P^0(\Lambda,\xi) \\ &\quad\times\exp\left[\sum_{i,W\neq W^i}\alpha_W^i\left(\tilde{\mathcal{D}}(X^i,W;\Lambda)-\xi_W^i\right)\right]. \end{aligned} \tag{27}$$

Since $P(\Lambda,\xi)$ is a pdf and it must be normalized (i.e., $\int_{\Lambda,\xi}P(\Lambda,\xi)d(\Lambda,\xi)=1$), the multiplier β depends on α, and it can be rewritten as follows:

$$\begin{aligned} e^{\beta-1} &= \left\langle\exp\left[\sum_{i,W\neq W^i}\alpha_W^i\left(\tilde{\mathcal{D}}(X^i,W;\Lambda)-\xi_W^i\right)\right]\right\rangle_{P^0(\Lambda,\xi)} \\ &\stackrel{\text{def}}{=}\frac{1}{Z(\alpha)}. \end{aligned} \tag{28}$$

Because of the convexity of the problem, the saddle point is located at the minimum point obtained by varying $f(\Lambda,\xi)$ and the maximum point obtained by varying α. The dual problem is defined by substituting $f(\Lambda,\xi)$ in the Lagrange functional L [(25)] by $P(\Lambda,\xi)$ [(27)] and by considering the maximization problem with respect to α, as follows:

$$L(\alpha,\beta)\left[P(\Lambda,\xi)\right]=\underbrace{-\log Z(\alpha)}_{J(\alpha)}+\text{const.} \tag{29}$$

Thus, we obtained the dual objective function used in (12).

APPENDIX B
SOLVER

It is inefficient to carry out optimization by a naive implementation for convex programming because the number of possible W is large. In order to handle a large number of possible W, we employed a method used in structured SVMs [1]. Because the Viterbi alignment computations are required in the proposed

method, some modifications to the structured SVM are required. The modified algorithm is described in Algorithm 1.

Algorithm 1 Modified cutting plane algorithm

1: $\hat{\Lambda}(\alpha) \stackrel{\text{def}}{=} \{\hat{\boldsymbol{\lambda}}_1(\alpha), \cdots, \hat{\boldsymbol{\lambda}}_s(\alpha), \cdots, \hat{\boldsymbol{\lambda}}_s(\alpha)\}$ [(16)]
2: $M(W;\Lambda) \stackrel{\text{def}}{=} \tilde{\mathcal{D}}(X^i, W;\Lambda) - \delta(W^i, W)$
3: $\alpha_W^i \leftarrow 0$ for all i and $W \neq W^i$
4: $\mathcal{C}_i \leftarrow \boldsymbol{\phi}$ for all i
5: **loop**
6: $i \leftarrow$ choose one training example
7: **if** $\hat{\boldsymbol{\lambda}}_s(\alpha) \neq 0$ for all s **then**
8: $\hat{W} \leftarrow \arg\min_{W \neq W^i} M(W; \hat{\Lambda}(\alpha))$ /* Performed by conventional decoding algorithms.*/
9: **else**
10: $\hat{W} \leftarrow$ choose one possible incorrect label sequence randomly
11: **end if**
12: **if** $\hat{\boldsymbol{\lambda}}_s(\alpha) = 0\ \exists s$, or $(\min_{W \in \mathcal{C}_i} M(W; \hat{\Lambda}(\alpha))) > \min\{0, M(\hat{W}; \hat{\Lambda}(\alpha))\} + \epsilon$ **then**
13: $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup \{\hat{W}\}$
14: **end if**
15: compute $\hat{q}^i$ and $\hat{q}_W^i$ $(\forall W \in \mathcal{C}_i)$ by using Viterbi algorithms with current parameters $\hat{\Lambda}(\alpha)$
16: **while** α_W^i converges for all W **do**
17: $W \leftarrow$ choose random W from $\mathcal{C}_i$
18: optimize α_W^i with given $\hat{q}^i$ and $\hat{q}_W^i$
19: **end while**
20: **end loop**

In Algorithm 1, the set $\mathcal{C}_i$ stores the working sets of label sequences (called "cutting planes" in [1]) associated with the ith training data. The label sequence $\hat{W}$ with the smallest expected margin $M(\hat{W};\Lambda)$ is selected and incrementally added to the set $\mathcal{C}_i$ if the expected margin $M(\hat{W};\Lambda)$ (defined in Line 2) is smaller than the smallest expected margin among the label sequences in the current working set $\mathcal{C}_i$.

The expected margin $M(W;\Lambda)$ for a given label sequence W is defined as the difference between the current discriminant function $\tilde{\mathcal{D}}(X^i, W;\Lambda)$ and the label similarity function $\delta(W, W^i) = \arg\max_{\xi_W^i} P^0(\xi_W^i)$, as follows:

$$\begin{aligned} M(W;\Lambda) &\stackrel{\text{def}}{=} \tilde{\mathcal{D}}(X^i, W;\Lambda) - \arg\max_{\xi_W^i} P^0\left(\xi_W^i\right) \\ &= \tilde{\mathcal{D}}(X^i, W;\Lambda) - \delta(W, W^i). \end{aligned} \tag{30}$$

Similar to the axis-parallel optimization described in [17], the proposed algorithm only considers updating a single Lagrange multiplier α_W^i at each iteration (Line 18), where i and W are randomly selected in Lines 6 and 17, respectively. Because the maximization of the objective function in the direction of a single multiplier can be solved analytically, the optimization is typically very fast in comparison to gradient-based methods. Further, because the hidden state sequences, $\hat{q}_W^i$ and $\hat{q}^i$, used in the optimization (Line 18) may be obtained as interim results of the decoding process carried out in Line 8, optimization is carried out efficiently by using conventional decoding algorithms.

The working set selection algorithm is similar to the conventional N-best approach [9], [23]. In our approach, the competitor W, which is considered to be important for optimization, is selected and incrementally added to working set $\mathcal{C}_i$. Thus, it is ensured that the proposed solver converges to the explicit solution by adding all possible W to the working set. It should be noted that optimization over all possible W is not necessary in common cases because most competitors are redundant, and most α_W^i remain 0.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions improving the quality of this manuscript. The authors also would like to thank Dr. J. Suzuki at NTT Communication Science Laboratories for valuable discussions.

REFERENCES

[1] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *J. Mach. Learn. Res.*, vol. 6, pp. 1453–1484, Oct. 2005.
[2] A. Ganapathiraju, J. Hamaker, and J. Picone, "Applications of support vector machines to speech recognition," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2348–2355, Aug. 2004.
[3] C. Joder, S. Essid, and G. Richard, "Alignment kernels for audio classification with application to music instrument recognition," in *Proc. EUSIPCO'08*, Lausanne, Switzerland, Sep. 2008.
[4] H. Shimodaira, K. Noma, M. Nakai, and S. Sagayama, "Dynamic time-alignment kernel in support vector machine," *Adv. Neural Inf. Process. Syst.*, vol. 2, pp. 921–928, 2002.
[5] M. Cuturi, J. Vert, O. Birkenes, and T. Matsui, "A kernel for time series based on global alignments," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'07)*, Apr. 2007, vol. 2, pp. II-413–II-416.
[6] B. Q. Huang, C. J. Du, Y. B. Zhang, and M.-T. Kechadi, "A hybrid HMM-SVM method for online handwriting symbol recognition," in *Proc. Intell. Syst. Design Applicat. (ISDA'06)*, Shaodong, China, Nov. 2006, vol. 1, pp. 887–891.
[7] A. Ganapathiraju, J. Hamaker, and J. Picone, "Hybrid SVM/HMM architectures for speech recognition," in *Proc. 6th Int. Conf. Spoken Lang. Process. (ICSLP'00)*, Beijing, China, Nov. 2000, vol. 4, pp. 504–507.
[8] P. C. Woodland, "Large scale discriminative training of hidden Markov models for speech recognition," *Comput. Speech Lang.*, vol. 16, no. 1, pp. 25–47, Feb. 2002.
[9] E. McDermott and S. Katagiri, "String-level MCE for continuous phoneme recognition," in *Proc. 5th Eur. Conf. Speech Commun. Technol. (Eurospeech-97)*, Rhodes, Greece, Sep. 1997, pp. 123–126.
[10] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '02)*, Orlando, FL, May 2002, pp. I-105–I-108.
[11] F. Sha and L. K. Saul, "Large margin hidden markov models for automatic speech recognition," *Adv. Neural Inf. Process. Syst.*, pp. 1249–1256, Feb. 2007.
[12] H. Kashima, T. Ide, T. Kato, and M. Sugiyama, "Recent advances and trends in large-scale kernel methods," *IEICE Trans. Inf. Syst.*, vol. E92-D, no. 7, pp. 1338–1353, 2009.
[13] N. D. Freitas, Y. Wang, M. Mahdaviani, and D. Lang, "Fast krylov methods for n-body learning," *Adv. Neural Inf. Process. Syst.*, vol. 18, pp. 251–268, 2006.
[14] L. Bahl, P. Brown, P. de Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'86)*, Apr. 1986, vol. 11, pp. 49–52.

[15] A. Gunawardana, M. Mahajan, and A. Acero, "Hidden conditional random fields for phone classification," in *Proc. European Conf. Speech Communication and Technology (Interspeech'05)*, Lisbon, Portugal, Oct. 2005, pp. 1117–1120.
[16] S. Reiter, B. Schuller, and G. Rigoll, "Hidden conditional random fields for meeting segmentation," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME-2007)*, Jul. 2007, pp. 639–642.
[17] T. Jebara, "Discriminative, generative and imitative learning," Ph.D. dissertation, Columbia Univ., New York, Feb. 2001.
[18] A. Janin, D. P. W. Ellis, and N. Morgan, "Multi-stream speech recognition: Ready for prime time?," in *Proc. 6th Eur. Conf. Speech Commun. Technol.*, Budapest, Hungary, Sep. 1999, pp. 591–594.
[19] D. P. Lewis, "Combining kernels for classification," Ph.D. dissertation, Columnbia Univ., New York, Mar. 2008.
[20] R. Schlüter, W. Macherey, B. Müller, and H. Ney, "Comparison of discriminative training criteria and optimization methods for speech recognition," *Speech Commun.*, vol. 34, pp. 287–310, 2001.
[21] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 11, pp. 1641–1648, Nov. 1989.
[22] B. Schölkopf and A. J. Smola, *Learning With Kernels*. Cambridge, MA: MIT Press, 2002.
[23] J.-K. Chen and F. K. Soong, "An N-best candidates-based discriminative training for speech recognition applications," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 1, pp. 206–216, Jan. 1994.

Yotaro Kubo (M'10) received the B.E., M.E., and Dr.Eng. degrees from Waseda University, Tokyo, Japan, in 2007, 2008, and 2010, respectively.

He is currently with NTT Communication Science Laboratory, Kyoto, Japan

Dr. Kubo received the Awaya Award from the ASJ in 2010. His research interests include machine learning and signal processing. He is a member of the International Speech Communication Association (ISCA), the Acoustical Society of Japan (ASJ), the Institute of Electronics, Information, and Communication Engineers (IEICE), and the Information Processing Society of Japan (IPSJ).

Shinji Watanabe (M'03) received the B.S., M.S., and Dr.Eng. degrees from Waseda University, Tokyo, Japan, in 1999, 2001, and 2006, respectively.

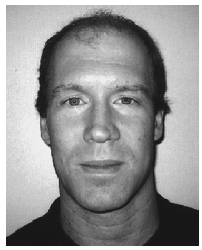
In 2001, he joined Nippon Telegraph and Telephone Corporation (NTT) and has since been working at NTT Communication Science Laboratories, Kyoto, Japan. From January to March in 2009, he was a Visiting Scholar at the Georgia Institute of Technology, Atlanta, at Fred Juang's laboratory. His research interests include Bayesian learning, pattern recognition, and speech and spoken language processing.

Dr. Watanabe is a member of the Acoustical Society of Japan (ASJ) and the Institute of Electronics, Information, and Communications Engineers (IEICE). He received the Awaya Award from the ASJ in 2003, the Paper Award from the IEICE in 2004, the Itakura Award from ASJ in 2006, and the TELECOM System Technology Award from the Telecommunications Advancement Foundation in 2006.

Atsushi Nakamura (M'03–SM'07) received the B.E., M.E., and Dr.Eng. degrees from Kyushu University, Fukuoka, Japan, in 1985, 1987 and 2001, respectively.

In 1987, he joined Nippon Telegraph and Telephone Corporation (NTT), where he engaged in the research and development of network service platforms, including studies on application of speech processing technologies into network services, at Musashino Electrical Communication Laboratories, Tokyo, Japan. From 1994 to 2000, he was with Advanced Telecommunications Research (ATR) Institute, Kyoto, Japan, as a Senior Researcher, working on the research of spontaneous speech recognition, construction of spoken language database, and development of speech translation systems. Since April 2000, he has been with NTT Communication Science Laboratories, Kyoto, Japan. His research interests include acoustic modeling of speech, speech recognition and synthesis, spoken language processing systems, speech production and perception, computational phonetics and phonology, and application of learning theories to signal analysis and modeling.

Dr. Nakamura is a member of the Machine Learning for Signal Processing (MLSP) Technical Committee, as well as served as a Vice Chair of the Signal Processing Society Kansai Chapter. He is also a member of the Institute of Electronics, Information, and Communication Engineers (IEICE) and the Acoustical Society of Japan (ASJ). He received the IEICE Paper Award in 2004, and received twice the Telecom-Technology Award of The Telecommunications Advancement Foundation, in 2006 and 2009.



Erik McDermott received the B.S. degree from the Symbolic Systems Program, Stanford University, Stanford, CA, in 1987 and the Ph.D. degree from the School of Engineering, Waseda University, Tokyo, Japan, in 1997.

He is currently with Google, Inc., Mountain View, CA. In December 2009, he joined Nippon Telegraph and Telephone Corporation (NTT) Communication Science Laboratories, Kyoto, Japan, as a Research Specialist. From 1987 to 1991, he worked at ATR Auditory and Visual Perception Research Labs, Kyoto, and at ATR Human Information Processing Labs from 1991 to 1999. He spent six months in 2000 in the SRI Speech Technology Laboratory, and one month in 2003 in the MIT Spoken Language Systems Group. His research has primarily been in speech recognition, with a focus on acoustic modeling paradigms for hidden Markov model (HMM)-based speech recognition. His recent work has focused on improving both theoretical and practical aspects of discriminative training applied to speech recognition system design.

Tetsunori Kobayashi (M'85) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Waseda University, Tokyo, Japan, in 1980, 1982, and 1985, respectively.

He was a Lecturer (1985–1987) and an Associate Professor (1987–1991) at Hosei University, Tokyo, Japan. In 1991, he joined Waseda University as an Associate Professor and since 1997 has been a Professor. He was a Visiting Scientist at Spoken Language System Group, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge (1994–1995). His research interests include perceptual computing and intelligent robotics. He is a member of Information Processing Society of Japan (IPSJ), the Acoustical Society of Japan (ASJ), the Institute of Electronics, Information, and Communication Engineers (IEICE), and the Institute of Electrical and Electronics Engineers (IEEE).