Variational Bayes for Continuous Hidden Markov Models and Its Application to Active Learning

Shihao Ji, Balaji Krishnapuram, and Lawrence Carin, *Fellow, IEEE*

Abstract

In this paper we present a *variational Bayes* (VB) framework for learning continuous hidden Markov models (CHMMs), and we examine the VB framework within active learning. Unlike a maximum likelihood or maximum *a posteriori* training procedure, which yield a point estimate of the CHMM parameters, VB-based training yields an estimate of the full posterior of the model parameters. This is particularly important for small training sets, since it gives a measure of confidence in the accuracy of the learned model. This is utilized within the context of active learning, for which we acquire labels for those feature vectors for which knowledge of the associated label would be most informative for reducing model-parameter uncertainty. Three active learning algorithms are considered in this paper: (i) query by committee (QBC), with the goal of selecting data for labeling that minimize the classification variance; (ii) a maximum expected information gain method that seeks to label data with the goal of reducing the entropy of the model parameters; (iii) an error-reduction-based procedure that attempts to minimize classification error over the test data. The experimental results are presented for synthetic and measured data. We demonstrate that all of these active learning methods can significantly reduce the amount of required labeling, compared to random selection of samples for labeling.

Index Terms

Variational Bayes (VB), continuous hidden Markov models (CHMMs), active learning (AL), query by committee (QBC), maximum expected information gain (MEIG), error-reduction-based active learning.

S. Ji, B. Krishnapuram and L. Carin are with the Department of Electrical and Computer Engineering, Duke University, Box 90291, Durham, NC 27708-0291.

I. Introduction

There has recently been an increasing interest in the area of variational Bayes (VB) learning [1]–[6]. Compared to standard maximum likelihood (ML) or maximum *a posteriori* (MAP) learning, VB does not yield a single point estimate of the model parameters. Rather, an ensemble of models are learned, with the goal of estimating the posterior density function on the model parameters, given a prescribed set of training data. This framework has often proven to be less sensitive to overfitting, and since the full posterior of the model parameters is available, it is well suited for active learning.

After the VB algorithm is trained, testing (classification) is performed by integrating out, in a Bayesian sense, the model parameters. Consequently, classification is not performed based on a single (point) estimate of parameters, but on a weighted sum over the ensemble. In many applications this framework has been found to be less sensitive to overfitting, *vis-à-vis* a ML/MAP training procedure. The VB procedure is a practical implementation of Bayesian learning for the true posterior probabilities of model parameters. Instead of computing the true posterior probabilities of the model directly, the VB approximates the true posterior to a variational one by maximizing a negative free energy. The resulting algorithm is closely related to the EM algorithm, and each iteration guarantees to monotonically increase the negative free energy or leave it unchanged, until converges is achieved at a local maximum. MacKay presents in [4] VB learning for a discrete hidden Markov model (HMM). We here present VB learning for a continuous HMM. Besides providing a new learning algorithm, we focus on active learning for HMMs, exploiting the VB machinery. We demonstrate that with the posterior probability of the model parameters estimated via VB, active learning can be solved in a convenient manner. Two previous active-learning criterions, minimizing the classification variance and minimizing the model's variance, can be implemented easily, while previously they were either computationally inefficient or intractable for continuous HMMs (CHMMs) [7]–[9].

The remainder of the paper is organized as follows. In Sec. II we present variational Bayes learning of continuous HMMs. The application of variational Bayes to active learning is illustrated in Sec. III, by extending the query by committee (QBC) and the maximum expected information gain (MEIG) active learning algorithms. We also consider an active-learning algorithm based on minimizing classification error. The relationship among these three active-learning

algorithms is also addressed in detail. In Sec. IV we present experimental results on synthetic and measured data. The conclusions and future work are addressed in Sec. V.

II. Variational Bayes Learning of Continuous Hidden Markov Models

Parameter estimation is a fundamental problem in system identification, pattern classification, and signal processing. We assume a (typically small) set of data is available from the system of interest. The objective is to fit a model to the data, to best describe the system. There are two broad treatments of this problem: 1) The model parameters are treated as fixed but unknown, and 2) the model parameters are treated as random variables [10]. The former treatment results in maximum likelihood (ML) or maximum *a posteriori* (MAP) estimation. In a Bayesian analysis, of interest for (2), before observing any data one assigns a prior distribution over the model parameters; after observing the data, Bayes' rule is used to infer their posterior distributions. In this way, the *marginal likelihood* or "*evidence*" can be obtained by integrating out the model parameters:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|\Phi)p(\Phi)d\Phi \tag{1}$$

where $\boldsymbol{x}$ represents the observed data and Φ denotes the model parameters. The *marginal likelihood* or "*evidence*" does not fit any single model to the data, but regards all model parameters as possible with different probabilities, defined via $p(\Phi)$. However, the Bayesian integration is typically computationally intractable, even in very simple cases. Most existing methods, such as Markov Chain Monte Carlo (MCMC) [11] and the Laplace approximation [6] either require vast computational resources to get accurate results or crudely approximate all the posteriors via a normal distribution. Between these two extremes, the VB method attempts to approximate the integration as accurately as possible while remaining computationally tractable [6].

We are interested in estimating the parameters of a continuous hidden Markov model (CHMM), in which a Gaussian mixture model is used for the state-dependent density function. A traditional method for estimating the parameters of a CHMM employs an ML estimation based on the expectation-maximization (EM) algorithm [12], [13]. In the EM algorithm the unobserved state sequence is treated as hidden. The algorithm starts from an initial guess of the model parameters, and iteratively updates them via the E and M steps. In the E step, the model parameters are fixed

and the probability of the hidden variables are estimated based on the current model. In the M step, the probability of the hidden variables resulting from the E step is fixed and the model parameters are updated by maximizing the expected complete-data likelihood over the hidden variables. It is proved in [13] that the EM algorithm is guaranteed to increase the likelihood or leave it unchanged until converges to a local maximum. In Neal and Hinton's paper [1], the EM algorithm is interpreted from a variational perspective. The M step remains unchanged. However, the E step is replaced by optimizing a variational posterior over the hidden variables, to maximize the "free energy". Similarly, in the Bayesian learning framework, since the model parameters are also treated as random or hidden variables, the varitional approximation can be employed on both the hidden variables and model parameters. From this perspective, variational Bayes is also an iterative technique that is similar to the EM algorithm and whose convergence is guaranteed. In the following, we derive the variational Bayes implementation of the CHMM, in a manner similar to MacKay [4], who considered a discrete hidden Markov model (DHMM).

Consider an N-state HMM, with the state-dependent observation defined by a Gaussian mixture. For notational convenience, it is assumed that all states have the same number of mixture components, K, and the dimensionality of the feature vectors is d. Then an HMM can be modeled as $\Phi = \{\pi^N, A^{N\times N}, C^{N\times K}, \Theta^{N\times K}\}$, where π is the initial-state probability vector, A is the transition matrix, C is the mixture-coefficient matrix, and Θ is the parameter matrix composed of the Gaussian parameters $\theta_{ik} = \{\mu_{ik}, R_{ik}\}$ for the kth mixture component of the ith state, with mean μ_{ik} and precision matrix R_{ik}, which is the inverse of the covariance matrix.

For an observation sequence $X = (x_1, x_2, \cdots, x_T)$, the associated complete-data is $Y = (X, S, L)$, where $S = (s_1, s_2, \cdots, s_T)$ is the unobserved state sequence, and $L = (l_1, l_2, \cdots, l_T)$ is the indicator sequence, which indicates which mixture component generates the observation. Thus, $s_t \in [1, N]$ and $l_t \in [1, K]$. For given model parameters Φ, the probability of the complete data can be expressed as

$$p(X, S, L|\Phi) = \pi_{s_1} \cdot \prod_{t=1}^{T-1} a_{s_t s_{t+1}} \cdot \prod_{t=1}^{T} c_{s_t l_t} f(x_t|\theta_{s_t l_t}) \tag{2}$$

and the likelihood of the model parameters Φ given the data X is

$$p(X|\Phi) = \sum_{S,L} \pi_{s_1} \cdot \prod_{t=1}^{T-1} a_{s_t s_{t+1}} \cdot \prod_{t=1}^{T} c_{s_t l_t} f(x_t|\theta_{s_t l_t}) \tag{3}$$

By Bayes' rule, the posterior density for the model parameters can be expressed as

$$p(\Phi|X) = \frac{p(X|\Phi)p(\Phi)}{\int p(X|\Phi)p(\Phi)d\Phi} \tag{4}$$

where in the denominator of (4) we must integrate (sum) over all parameters, covering the complete range of each parameter. The computational cost required for the denominator is what has motivated previous ML/MAP solutions (which simply maximize the numerator in (4)). The VB algorithm represents an approximate and computationally tractable means of computing (4), and approximates this integration by maximizing a lower bound [1], [6], which can be derived from a fundamental relationship between the log-likelihood, negative free energy and the Kullback-Leibler (KL) divergence. The marginal likelihood can be expressed as

$$p(X) = \frac{p(X, S, L, \Phi)}{p(S, L, \Phi|X)} \tag{5}$$

In this case, both the hidden variables and model parameters are all treated as random variables. Taking the logarithm and then the expectation with respect to the distribution $q(S, L, \Phi)$ on both sides, we obtain

$$\log p(X) = \int q(S, L, \Phi) \log p(X, S, L, \Phi) dS dL d\Phi - \int q(S, L, \Phi) \log p(S, L, \Phi|X) dS dL d\Phi \tag{6}$$

where the distribution $q(S, L, \Phi)$ is called the approximate or variational posterior of the model parameters and hidden variables. By re-arranging (6), we obtain

$$\log p(X) = F(q) + KL(q||p) \tag{7}$$

where

$$F(q) = \int q(S, L, \Phi) \log \frac{p(X, S, L, \Phi)}{q(S, L, \Phi)} dS dL d\Phi \tag{8}$$

$$KL(q||p) = \int q(S, L, \Phi) \log \frac{q(S, L, \Phi)}{p(S, L, \Phi|X)} dS dL d\Phi \tag{9}$$

The term $F(q)$ is known as the negative free energy used in statistical physics and $KL(q||p)$ is the Kullback-Leibler (KL) divergence between the approximate and true posterior. Since the KL divergence is non-negative and is zero for identical distributions, this indicates that $F(q)$ is a strict lower bound on $\log p(X)$,

$$\log p(X) \geq F(q) \tag{10}$$

with equality if the approximate posterior density equals the true posterior density, i.e., $q(S, L, \Phi) = p(S, L, \Phi|X)$. The aim of VB is to maximize this lower bound by tuning the variational posterior $q(S, L, \Phi)$ such that as the variational posterior approaches the true posterior, the bound becomes tight, thus the marginal likelihood can be approximated efficiently.

For the computation of the negative free energy, two key issues remain to be addressed: the choice of the form of the variational density and the prior distribution of model parameters. We need to choose a density form that is tractable and meanwhile can make a good approximation to the true posterior. One choice is a factorized form

$$q(S, L, \Phi) = q(S)q(L)q(\pi)q(A)q(C)q(\Theta) \tag{11}$$

which has been successfully applied in many applications of the variational method [2], [4], [5]. A natural choice for the prior over π, the rows of A and the rows of C is the Dirichlet distribution, since the Dirichlet distribution is the conjugate prior over the multinomial distribution [4]. Similarly, we choose the Normal-Wishart distribution as the prior over the Gaussian distribution [6], [14]. Thus, the prior on the model parameters can be expressed as

$$p(\Phi) = p(\pi)p(A)p(C)p(\Theta) \tag{12}$$

where

$$p(\pi) = Dir(\pi_1, \cdots, \pi_N|u_1^\pi, \cdots, u_N^\pi) \tag{13}$$

$$p(A) = \prod_{i=1}^{N} Dir(a_{i1}, \cdots, a_{iN}|u_{i1}^A, \cdots, u_{iN}^A) \tag{14}$$

$$p(C) = \prod_{i=1}^{N} Dir(c_{i1}, \cdots, c_{iK}|u_{i1}^C, \cdots, u_{iK}^C) \tag{15}$$

$$p(\Phi) = \prod_{i=1}^{N}\prod_{k=1}^{K} NW(\mu_{ik}, R_{ik}|a_{ik}, b_{ik}, \lambda_{ik}, m_{ik}) \tag{16}$$

The form of the Dirichlet distribution and the Normal-Wishart distribution are discussed in the Appendix.

M Step: *With the variational posterior on hidden variables fixed at $q(S, L)$, update the variational posterior on model parameters $q(\Phi)$ to maximize $F(q)$.*

We can substitute (2) and (11)-(16) into (8) to yield

$$\begin{aligned} F(q) &= \int q(S)q(L)q(\pi)q(A)q(C)q(\Phi)[\log \pi_{s_1} + \sum_{t=1}^{T-1} \log a_{s_t s_{t+1}} + \sum_{t=1}^{T} \log c_{s_t l_t} \\ &\quad + \sum_{t=1}^{T} \log f(x_t|\theta_{s_t l_t}) + \log p(\pi) + \log p(A) + \log p(C) + \log p(\Theta) \\ &\quad - \log q(S) - \log q(L) - \log q(\pi) - \log q(A) - \log q(C) - \log q(\Phi)]dSdLd\Phi \\ &= F(q(\pi)) + F(q(A)) + F(q(C)) + F(q(\Phi)) + H(q(S,L)) \end{aligned} \tag{17}$$

In the equation above, the last term is constant since $q(S, L)$ is fixed for the M step and is ignored in the subsequent optimization steps. The independence among the functions over $q(\pi)$, $q(A)$, and $q(\Phi)$ enables us to optimize them separately.

1. Optimization of $q(A)$, $q(\pi)$ and $q(C)$

By collecting all the quantities related to together, we obtain the expression

$$F(q(A)) = \int q(A) \sum_{S} q(S) \sum_{t=1}^{T-1} \log a_{s_t s_{t+1}} dA + \int q(A) \log p(A) dA - \int q(A) \log q(A) dA \tag{18}$$

Further, we define a quantity

$$w_{ij}^{t} = \sum_{S} q(S)\delta(s_t = i, s_{t+1} = j) = q(s_t = i, s_{t+1} = j) \tag{19}$$

which is similar to the quantity $\xi_t(i, j)$ defined in [12] for the probability of being in state i at time t, and state j at time $t + 1$. Then, we have

$$F(q(A)) = -\int q(A) \log \left(\frac{q(A)}{\prod_{i,j=1}^{N} a_{ij}^{W_{ij}^{A}-1}} \right) dA \tag{20}$$

where

$$W_{ij}^{A} = \sum_{t=1}^{T-1} w_{ij}^{t} + u_{ij}^{A} \tag{21}$$

By Gibbs inequality, $F(q(A))$ is maximized with respect to $q(A)$ by

$$q(A) = \prod_{i=1}^{N} Dir(a_{i1}, \cdots, a_{iN}|W_{i1}^{A}, \cdots, W_{iN}^{A}) \tag{22}$$

which is a product of Dirichlet distributions with the hyperparameters W_{ij}^{A}.

Similarly, we can optimize $F(q)$ with respect to $q(\pi)$, $q(C)$ separately by using a similar procedure and obtain the optimized $q(\pi)$, $q(C)$ expressed as

$$q(\pi) = Dir(\pi_1, \cdots, \pi_N | W_1^\pi, \cdots, W_N^\pi) \tag{23}$$

$$q(C) = \prod_{i=1}^{N} Dir(c_{i1}, \cdots, c_{iK} | W_{i1}^C, \cdots, W_{iK}^C) \tag{24}$$

where

$$W_i^\pi = w_i^\pi + u_i^\pi \tag{25}$$

$$W_{ik}^C = \sum_{t=1}^{T} w_{ik}^t + u_{ik}^C \tag{26}$$

$$w_i^\pi = \sum_{S} q(S)\delta(s_1 = i) = q(s_1 = i) \tag{27}$$

$$w_{ik}^t = \sum_{S,L} q(S)q(L)\delta(s_t = i, l_t = k) = q(s_t = i, l_t = k) \tag{28}$$

For the similar definitions to the conventional EM training, the quantities w_i^π, w_{ij}^t and w_{ik}^t can all be calculated using the forward-backward algorithm [12].

2. Optimization of $q(\Phi)$

By collecting all the quantities related to $q(\Phi)$ together, we obtain the expression

$$F(q(\Theta)) = \int q(\Theta) \sum_{S,L} q(S)q(L) \sum_{t=1}^{T} \log f(x_t|\theta_{s_t l_t}) d\Theta + \int q(\Theta) \log p(\Theta) d\Theta - \int q(\Theta) \log q(\Theta) d\Theta \tag{29}$$

With w_{ik}^t defined in (28), we obtain

$$F(q(\Theta)) = -\int q(\Theta) \log \left(\frac{q(\Theta)}{\prod_{i=1}^{N} \prod_{k=1}^{K} \prod_{t=1}^{T} f^{w_{ik}^t}(x_t|\theta_{ik}) \times p(\theta_{ik})} \right) d\Theta \tag{30}$$

The optimized $q(\theta_{ik})$ becomes

$$\begin{aligned} q(\theta_{ik}) &= \prod_{t=1}^{T} f^{w_{ik}^t}(x_t|\theta_{ik}) \times p(\theta_{ik}) = \prod_{t=1}^{T} f^{w_{ik}^t}(x_t|\mu_{ik}, R_{ik}) \times p(\mu_{ik}, R_{ik}|a_{ik}, b_{ik}, \lambda_{ik}, m_{ik}) \\ &= \frac{(\lambda_{ik}/2\pi)^{d/2}}{Z(a_{ik}, b_{ik}) \times (2\pi)^{dw_{ik}/2}} |R_{ik}|^{\frac{a_{ik}+w_{ik}-d}{2}} \exp\left[-\frac{\lambda'_{ik}}{2} (\mu_{ik} - m'_{ik})^T R_{ik} (\mu_{ik} - m'_{ik}) \right] \\ &\quad \times \exp\left[-\frac{1}{2} Tr(b'_{ik} R_{ik}) \right] \end{aligned} \tag{31}$$

where

$$w_{ik} = \sum_{t=1}^{T} w_{ik}^t \tag{32}$$

$$\bar{x}_{ik} = \sum_{t=1}^{T} w_{ik}^t x_t / w_{ik} \tag{33}$$

$$S_{ik} = \sum_{t=1}^{T} w_{ik}^t (x_t - \bar{x}_{ik})(x_t - \bar{x}_{ik})^T \tag{34}$$

$$a'_{ik} = a_{ik} + w_{ik} \tag{35}$$

$$b'_{ik} = b_{ik} + S_{ik} + \frac{\lambda_{ik} w_{ik}}{\lambda_{ik} + w_{ik}} (m_{ik} - \bar{x}_{ik})(m_{ik} - \bar{x}_{ik})^T \tag{36}$$

$$\lambda'_{ik} = \lambda_{ik} + w_{ik} \tag{37}$$

$$m'_{ik} = \frac{\lambda_{ik} m_{ik} + w_{ik} \bar{x}_{ik}}{\lambda_{ik} + w_{ik}} \tag{38}$$

E Step: *With the variational posterior on model parameters $q(\Phi)$ fixed, update the variational posterior on hidden variables $q(S, L)$ to maximize $F(q)$.*

By substituting (2) and (11)-(16) into (8), and re-arranging, the negative free energy function can be expressed as:

$$F(q) = F(q(S, L)) - KL(q(\Phi)||p(\Phi)) \tag{39}$$

where

$$\begin{aligned} F(q(S, L)) = & \sum_S q(S) \int q(\pi) \log \pi_{s_1} d\pi + \sum_S q(S) \int q(A) \sum_{t=1}^{T-1} \log a_{s_t s_{t+1}} dA \\ & + \sum_{S,L} q(S, L) \int q(C) \sum_{t=1}^{T} \log c_{s_t l_t} dC + \sum_{S,L} q(S, L) \int q(\Theta) \sum_{t=1}^{T} \log f(x_t | \theta_{s_t l_t}) d\Theta \\ & - \sum_{S,L} q(S, L) \log q(S, L) \end{aligned} \tag{40}$$

Since $q(\Phi)$ is fixed, the second term in (39) is constant. We only need to optimize the first term.

We start by defining

$$\log \pi^*_{s_1} = \int q(\pi) \log \pi_{s_1} d\pi = \psi(W^\pi_{s_1}) - \psi(W^\pi_0) \quad (41)$$

$$\log a^*_{s_t s_{t+1}} = \int q(A) \log a_{s_t s_{t+1}} dA = \psi(W^A_{s_t s_{t+1}}) - \psi(W^A_{s_t 0}) \quad (42)$$

$$\log c^*_{s_t l_t} = \int q(C) \log c_{s_t l_t} dC = \psi(W^C_{s_t l_t}) - \psi(W^C_{s_t 0}) \quad (43)$$

$$\begin{aligned} \log f^*(x_t|\theta_{s_t l_t}) &= \int q(\Theta) \log f(x_t|\theta_{s_t l_t}) d\Theta \\ &= -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\frac{b_{s_t l_t}}{2}| + \frac{1}{2}\sum_{i=1}^{d}\psi(\frac{a_{s_t l_t}+1-i}{2}) \\ &\quad - \frac{1}{2}a_{s_t l_t}(x_t - m_{s_t l_t})^T b^{-1}_{s_t l_t}(x_t - m_{s_t l_t}) - \frac{d}{2\lambda_{s_t l_t}} \end{aligned} \quad (44)$$

where $\psi(\cdot)$ is the digamma function defined as $\psi(x) = \frac{\partial}{\partial x}\log\Gamma(x)$; W^π_0, $W^A_{s_t 0}$ and $W^C_{s_t 0}$ are strength of their associated Dirichlet distributions.

Then substituting (41)-(44) into (40), we obtain

$$F(q(S,L)) = -\sum_{S,L} q(S,L) \log\left(\frac{q(S,L)}{\pi^*_{s_1}\cdot\prod_{t=1}^{T-1} a^*_{s_t s_{t+1}}\cdot\prod_{t=1}^{T} c^*_{s_t l_t} f^*(x_t|\theta_{s_t l_t})}\right) \quad (45)$$

The optimized $q(S,L)$ becomes

$$q(S,L) = \frac{1}{Z}\cdot\pi^*_{s_1}\cdot\prod_{t=1}^{T-1} a^*_{s_t s_{t+1}}\cdot\prod_{t=1}^{T} c^*_{s_t l_t} f^*(x_t|\theta_{s_t l_t}) \quad (46)$$

with the normalizing constant yielding a probability density. Comparing with (2), we notice that

$$Z = q(X|\Phi^*) \quad (47)$$

is the approximate likelihood of the optimized model Φ^*, which can be computed efficiently by the forward-backward algorithm [12].

Convergence

The variational Bayes approach is a generalization of the conventional EM algorithm [12], [13]. Each iteration guarantees to increase the negative free energy or leave it unchanged, until it converges to a local maximum. The negative free energy is an important quantity to approximate the marginal likelihood, with this critical in model selection and density estimation [6]. We

terminate the algorithm when the change in the negative free energy is negligibly small, and this quantity can be calculated by substituting (46)-(48) into (39)

$$\begin{aligned} F(q) &= F(q(S,L)) - KL(q(\Phi)||p(\Phi)) \\ &= \log q(X|\Phi^*) - KL_{Dir}(q(\pi)||p(\pi)) - KL_{Dir}(q(A)||p(A)) \\ &\quad - KL_{Dir}(q(C)||p(C)) - KL_{NW}(q(\Theta)||p(\Theta)) \end{aligned} \tag{48}$$

where the KL divergence is between the variational posterior and the prior distribution. The KL divergences of Dirichlet and Normal-Wishart distribution are discussed in the Appendix.

Computation of the predictive likelihood

For the classification task, the ultimate goal of Bayesian learning is to compute the predictive likelihood. In the Bayesian framework, the predictive likelihood of a test sequence $\boldsymbol{x} = (x_1, \cdots, x_T)$, given a set of training data D_l, is obtained by averaging over all models and weighting each model by its posterior:

$$p(\boldsymbol{x}|D_l) = \int p(\boldsymbol{x}|\Phi)p(\Phi|D_l)d\Phi \tag{49}$$

The true posterior is unknown. However, we may approximate it with a variational posterior resulting from the VB. The approximate predictive likelihood can therefore be expressed as

$$\begin{aligned} p(\boldsymbol{x}|D_l) &\approx \int p(\boldsymbol{x}|\Phi)q(\Phi)d\Phi \\ &= \int \sum_{S,L} \pi_{s_1} \prod_{t=1}^{T-1} a_{s_t s_{t+1}} \prod_{t=1}^{T} c_{s_t s_{t+1}} f(x_t|\theta_{s_t l_t}) \cdot q(\pi)q(A)q(C)q(\Theta)d\pi dA dC d\Theta \\ &= \sum_{S,L} \left[\int \pi_{s_1} q(\pi)d\pi \cdot \int \prod_{t=1}^{T-1} a_{s_t s_{t+1}} q(A)dA \cdot \int \prod_{t=1}^{T} c_{s_t l_t} q(C)dC \cdot \int \prod_{t=1}^{T} f(x_t|\theta_{s_t l_t}) q(\Theta)d\Theta \right] \\ &= \sum_{S,L} \left[E(\pi_{s_t}) \cdot E\left(\prod_{t=1}^{T-1} a_{s_t s_{t+1}} \right) \cdot E\left(\prod_{t=1}^{T} c_{s_t l_t} \right) \cdot E\left(\prod_{t=1}^{T} f(x_t|\theta_{s_t l_t}) \right) \right] \end{aligned} \tag{50}$$

Although it can be expressed analytically, this quantity is still intractable since the states, mixture component indicators and model parameters are coupled together. An approximation to this quantity is to assume that the states, indicators and model parameters are independent of each other,

$$p(\boldsymbol{x}|D_l) \approx \sum_{S,L} \left[E(\pi_{s_1}) \cdot \prod_{t=1}^{T-1} E(a_{s_t s_{t+1}}) \cdot \prod_{t=1}^{T} E(c_{s_t l_t}) \cdot E\left(f(x_t|\theta_{s_t l_t}) \right) \right] \tag{51}$$

where

$$E(\pi_{s_1}) = \pi_{s_1}/\pi_0, \quad E(a_{s_t s_{t+1}}) = a_{s_t s_{t+1}}/a_{s_t 0}, \quad E(c_{s_t l_t}) = c_{s_t l_t}/c_{s_t 0}$$
$$E(f(x_t|\theta_{s_t l_t})) = \left(\frac{\lambda_{s_t l_t}}{\pi(\lambda_{s_t l_t}+1)}\right)^{d/2} \cdot \frac{|b_{s_t l_t}|^{a_{s_t l_t}/2}}{|b_{s_t l_t}+\Delta b_{s_t l_t}|^{(a_{s_t l_t}+1)/2}} \cdot \frac{\Gamma((a_{s_t l_t}+1)/2)}{\Gamma((a_{s_t l_t}+1-d)/2)} \quad (52)$$

with

$$\Delta b_{s_t l_t} = \frac{\lambda_{s_t l_t}}{\lambda_{s_t l_t}+1}(m_{s_t l_t}-x_t)(m_{s_t l_t}-x_t)^T \quad (53)$$

The independence assumption resumes the first-order Markovian property. Thus, the (52) can be evaluated efficiently by the forward-backward algorithm [12]. We notice that the approximation in (52) results in evaluating the integrand still at a single value of . However, this point estimation is neither the maximum likelihood nor the maximum *a posteriori* estimation.

III. Active Learning with Continuous Hidden Markov Models

Learning may be more effective if the learner can actively participate in the learning process (i.e., in selection of the labeled data). Compared to conventional supervised learning, in which the learner "passively" receives the labeled data and generates a learned model, in active learning we start with a small set of labeled data, and identify those unlabeled examples that would be most informative if the associated label were available; the labels redeemed to be informative are subsequently queried (acquired). Such a setting is critical in machine learning tasks for which acquiring labels is expensive or time consuming, and therefore we prioritize those items for labeling that are most informative.

Active learning has been a focus of significant research for many years. It has demonstrated success in a wide range of learning models, such as: naive Bayes [7], [15], the SVM [16], and in neural networks [8]. Depending on the data source, the active learning settings can be classified in two broad categories: *pool-based active learning* [7], [15], [17]–[19] and *membership queries* [8], [9]. In *pool-based active learning*, the learner is provided with a fixed pool of unlabeled data and the learner is only allowed to choose data from the pool, and request the label. In *membership queries*, the learner has the control to construct the data in the data space and request the label. In the task of sequential data classification, such as the HMMs, a large pool of unlabeled sequential data is often available. Thus, we only focus on pool-based active learning. To our knowledge, there are very few previous studies on active learning focused on HMMs.

As indicated above, the general idea in active learning is to choose that unlabeled sample that would be most informative if the associated label were made available for training. In terms of measuring informativeness, the algorithms can be classified as an *implicit measure* and an *explicit measure* [19]. In the context of *explicit measure*, Cohn [9] states that if the learner is unbiased, the informativeness of an example can be assessed by the expected decrease in the overall variance of the model's prediction. Similarly, within a Bayesian framework, MacKay [8] attempts to measure the information that can be gained about the unknown target hypothesis using new labeled data. An explicit method requires a closed form calculation on the learner's variance on the target hypothesis, which is only available for simple learning schemes, such as locally weighted regression [9], or based on various approximations which may undermine the precision of this method [8]. The alternative is to measure the informativeness implicitly by computing the model's variance on classifying the data considered. The query by committee (QBC) method [7], [17]–[19] falls in the framework of the *implicit measure*. In this method the classification variance is estimated by computing the classification uncertainty with respect to the entire space of possible models consistent with the training data. The three algorithms we present in this paper are QBC, the maximum expected information gain method (MEIG) and an error-reduction-based method. We show that with the posterior density of the model parameters obtained via VB, both the implicit measure and explicit measure of informativeness can be calculated efficiently. New aspects of this work include consideration of sequential data, modeled via an HMM. Further, the variational form of HMM training plays a key role in implementing the active-learning algorithms.

A. Query By Committee

The QBC algorithm is formulated and analyzed in [17], [18]. This algorithm is based on a theoretical result stating that by halving the version space after each query, the generalization error decreases exponentially. The version space is a subset of hypotheses that is consistent with the labeled training data. In a binary case, this method randomly samples the version space and induces an even number of classifiers (committee). The label of an unlabeled data is requested whenever a voting between the classifiers on the unlabeled data results in a tie. This algorithm, originally designed for the binary case, has been extended to the probabilistic model in [7], [19], which inspires the VB implementation of the QBC presented here.

In the framework of QBC, the informativeness of an example is measured by computing the classification variance with respect to the entire space of possible models consistent with the training data thus far. However, estimation with respect to the entire model space requires vast computation resource. Thus, the QBC algorithm approximates the entire space by randomly sampling the model-parameter distribution that resulted from the training data. These randomly selected models serve as a "committee" of classifiers to classify each unlabeled example. The classification variance is measured by computing the disagreement over their classifications. The data with the strongest disagreement among the committee are selected for labeling. In [7], the degree of disagreement is measured via the KL divergence, measuring the average distance of the class posterior density resulting from each committee member to the their mean value.

An obvious method to generate the committee members is by exploiting the local-maxima property of the conventional HMM EM training algorithm [12]. That is, by starting the EM algorithm with different initial guesses, ML estimation can converge to an ensemble of different local maximas, forming a committee. However, this method has several disadvantages. First, it needs to learn the model multiple times to form the committee. Second, some ML estimations may converge to the same or similar local maximum, undermining committee diversity. However, with the posterior density of the model parameters obtained via the VB, this problem can be solved simply by random sampling from the posterior density of the model parameters obtained from VB learning (discussed in Sec. II).

Let $\boldsymbol{x}^*$ be an unlabeled data sequence whose informativeness we want to evaluate, with its unknown class label $y^* \in \{1, \cdots, C\}$. With VB learning, the posterior density of all model parameters $\boldsymbol{\lambda} = \{\Phi^1, \cdots, \Phi^C\}$ can be induced from the labeled data $D_l = \{D_l^1, \cdots, D_l^C\}$, where $\boldsymbol{\lambda}$ consists of model parameters of each class, $\forall i \in \{1, \cdots, C\}$, $D_l^i \rightarrow \Phi^i$. In other words, $p(\boldsymbol{\lambda}|D_l)$ represents the posterior probability of $\boldsymbol{\lambda}$ given the training data D_l. We can then randomly sample $p(\boldsymbol{\lambda}|D_l)$ M times to generate a committee of classifiers with M members: $\hat{\boldsymbol{\lambda}}_1, \cdots, \hat{\boldsymbol{\lambda}}_M$. The degree of disagreement with regard to an unlabeled data $\boldsymbol{x}^*$ can be evaluated by the KL divergence [7],

$$score(\boldsymbol{x}^*) = \frac{1}{M} \sum_{m=1}^{M} KL\left(p(y^*|\boldsymbol{x}^*, \hat{\boldsymbol{\lambda}}_m)||p_{avg}(y^*|\boldsymbol{x}^*)\right) \tag{54}$$

where $p(y^*|\boldsymbol{x}^*, \hat{\boldsymbol{\lambda}}_m)$ is the class posterior of unlabeled data $\boldsymbol{x}^*$ with regard to the mth committee member, and $p_{avg}(y^*|\boldsymbol{x}^*) = \frac{1}{M}\sum_{m-1}^{M} p(y^*|\boldsymbol{x}^*, \hat{\boldsymbol{\lambda}}_m)$. By Bayes' rule, $p(y^*|\boldsymbol{x}^*, \hat{\boldsymbol{\lambda}}_m)$ can be

calculated as:

$$p(y^*|\boldsymbol{x}^*, \hat{\boldsymbol{\lambda}}_m) = \frac{p(\boldsymbol{x}^*|\hat{\boldsymbol{\lambda}}_m^{y^*})p(y^*)}{\sum_{y^*=1}^{C} p(\boldsymbol{x}^*|\hat{\boldsymbol{\lambda}}_m^{y^*})p(y^*)} \tag{55}$$

where $p(\boldsymbol{x}^*|\hat{\boldsymbol{\lambda}}_m^{y^*})$ can be calculated via the forward-backward algorithm [12], and $p(y^*)$ is the class prior, which can be estimated directly from the labeled data (or we can just assume a non-informative prior).

B. Maximum Expected Information Gain (MEIG)

Within a Bayesian framework, MacKay [8] attempts to measure the information that can be gained about the unknown target hypothesis using a new labeled data. Thus, the informativeness of a new labeled data can be accessed analytically. In pool-based active learning, we may view this active learning setting as an information extraction process: we select the data that gives us maximum information about the pool. As we only select one most-informative data each time, the maximum expected information gain (MEIG) approach becomes a greedy (myopic) algorithm.

Let $\boldsymbol{x}^*$ be an unlabeled data sequence, its class label $y^* \in \{1, \cdots, C\}$. With VB learning the posterior density of all parameters $\boldsymbol{\lambda} = \{\Phi^1, \cdots, \Phi^C\}$ is estimated from the labeled data $D_l = \{D_l^1, \cdots, D_l^C\}$, where $\boldsymbol{\lambda}$ consists of model parameters of each class, $\forall i \in \{1, \cdots, C\}$, $D_l^i \rightarrow \Phi^i$. In other words, $p(\boldsymbol{\lambda}|D_l)$ dictates the posterior probability of $\boldsymbol{\lambda}$ given the training data D_l. Then, information gain after augmenting an unlabeled data into the training set can be expressed in the context of information theory: how much information about $\boldsymbol{\lambda}$ can be obtained if we add an unlabeled data into the training set? Since the class label y^* of $\boldsymbol{x}^*$ is unknown, it is treated as a random variable whose probability can be estimated from the training data. The information gain of an unlabeled data can be expressed as the mutual information (MI) between the random variable $\boldsymbol{\lambda}$ and y^*, and consequently we call this method the maximum mutual information (MMI).

$$\begin{aligned} G(\boldsymbol{x}^*) &= \boldsymbol{I}(\boldsymbol{\lambda}; y^*) = H(\boldsymbol{\lambda}|D_l) - \sum_{y=1}^{C} H(\boldsymbol{\lambda}|D_l, \boldsymbol{x}^*, y^*)p(y^*|\boldsymbol{x}^*, D_l) \\ &= \sum_{y^*=1}^{C} \left[H(\Theta^{y^*}|D_l^{y^*}) - H(\Theta^{y^*}|D_l^{y^*}, \boldsymbol{x}^*, y^*) \right] p(y^*|\boldsymbol{x}^*, D_l) \end{aligned} \tag{56}$$

where $p(y^*|\boldsymbol{x}^*, D_l)$ is the class posterior of $\boldsymbol{x}^*$ given the training data D_l, and the expression represents $H(\cdot)$ Shannon entropy [20]. By Bayes' rule,

$$p(y^*|\boldsymbol{x}^*, D_l) = \frac{p(\boldsymbol{x}^*|D_l^{y^*})p(y^*)}{\sum_{y^*=1}^{C} p(\boldsymbol{x}^*|D_l^{y^*})p(y^*)} \tag{57}$$

where $p(\boldsymbol{x}^*|, D_l^{y^*})$ is the predictive probability of $\boldsymbol{x}^*$ given the training data $D_l^{y^*}$, which can be calculated from (52), and $p(y^*)$ is the class prior that can be estimated directly from the labeled data (or we can just assume a non-informative prior).

In addition to the mutual information, another information measure we consider is the KL divergence between the posterior density of model parameter $\boldsymbol{\lambda}$ obtained after augmenting an unlabeled data into the training set and before the augmentation:

$$G'(\boldsymbol{x}^*) = \sum_{y^*=1}^{C} KL\left[p(\Phi^{y^*}|D_l^{y^*}, \boldsymbol{x}^*, y^*)||p(\Phi^{y^*}|D_l^{y^*})\right] p(y^*|\boldsymbol{x}^*, D_l) \tag{58}$$

When we use the KL divergence as the measure of information gain, we call this method the maximum KL divergence (MKL). The MI measure only seeks the labels to most shrink the model variance while the KL measure seeks labels that can most shrink or expand (i.e., change) the model variance. Thus, the information gain of the KL measure is defined in terms of the possible change in the model variance, which may be more appropriate for active learning; since the previous estimation of the model may be biased, only minimizing the variance is not correct. In Sec. IV we validate this idea using synthetic and measured data. The equations for the entropy and KL divergence of Dirichlet and Normal-Wishart distributions can be obtained from the Appendix. We note that this active-learning methodology attempts to reduce uncertainty on the model parameters, which does not necessarily translate to classification performance.

C. Error-Reduction-Based Active Learning

The ultimate goal of active learning is to achieve the lowest expected error on future test data, with the fewest possible labeling queries. Toward this criterion, the active learner should select the data sample that once incorporated into training will result in the lowest expected error on the set of testing samples. The method follows the general bias and variance decomposition of prediction error [9], [21].

Let $p(\boldsymbol{x}, y)$ be the unknown joint distribution over input $\boldsymbol{x}$ and label y, and $p(\boldsymbol{x})$ be the (known, at least approximately) input distribution of $\boldsymbol{x}$. The goal of the learner is to estimate

$p(y|\boldsymbol{x})$ from a labeled training set $D_l = \{(\boldsymbol{x}_1, y_1), \cdots, \boldsymbol{x}_l, y_l)\}$, $y_i \in [1, \cdots, C]$. We denote the learner's prediction on an unlabeled data $\boldsymbol{x}$ given training set D_l as $\hat{y}(\boldsymbol{x}; D_l)$, which is a random variable due to the randomness of y dictated by $p(y|\boldsymbol{x})$ and randomness of learning algorithm on D_l dictated by $p(\hat{y}|\boldsymbol{x}, D_l)$. By using $p(y|\boldsymbol{x})$ we are allowing possible label noise in the data. The error of the learner over the input distribution can be expressed as

$$Error = \int E_T \left[\hat{y}(\boldsymbol{x}; D_l) - y(\boldsymbol{x})\right]^2 p(\boldsymbol{x}) d\boldsymbol{x} \tag{59}$$

where $E_T[\cdot]$ denotes expectation over $p(y|\boldsymbol{x})$ and over $p(\hat{y}|\boldsymbol{x}, D_l)$. The expectation inside the integration may be decomposed as [9], [21]

$$\begin{aligned} E_T[\hat{y}(\boldsymbol{x}; D_l) - y(\boldsymbol{x})]^2 &= E[y(\boldsymbol{x}) - E(y|\boldsymbol{x})]^2 + [E_{D_l}(\hat{y}(\boldsymbol{x}; D_l)) - E(y|\boldsymbol{x})]^2 \\ &\quad + E_{D_l}\left[\hat{y}(\boldsymbol{x}; D_l) - E_{D_l}(\hat{y}(\boldsymbol{x}; D_l))\right]^2 \end{aligned} \tag{60}$$

The first term in (62) is the noise in the distribution, which does not depend on the learner or on the training data, and represents the minimal error of an ideal learner can achieve. The second term is the learner's squared bias, and the third is the learner's variance; these last two terms comprise the mean squared error of the learner. If we assume that the data set is noiseless and the learner is unbiased, then the first and second terms in (62) vanish and the error only depends on the learner's variance,

$$Error \approx \int E_{D_l}\left[\hat{y}(\boldsymbol{x}; D_l) - E_{D_l}(\hat{y}(\boldsymbol{x}; D_l))\right]^2 p(\boldsymbol{x}) d\boldsymbol{x} \tag{61}$$

This equation motivates the use of a new function

$$Error \approx \int H(\hat{y}|\boldsymbol{x}, D_l) p(\boldsymbol{x}) d\boldsymbol{x} \tag{62}$$

where $H(\hat{y}|\boldsymbol{x}, D_l)$ is the uncertainty (entropy) in the classifier given labeled data D_l and sample $\boldsymbol{x}$. We then obtain a similar expression as in [15] if the entropy is substituted by the log loss function. However, we should point out a significant difference between them: in [15] $p(\hat{y}|\boldsymbol{x}, D_l)$ is approximated by the prediction of a single classifier induced from D_l, while in a rigorous sense, this quantity should be an averaged value on D_l which may be calculated by averaging all the predictions of an ensemble of classifiers induced from D_l. In the framework of VB learning, this quantity can be calculated by (59), i.e., the VB algorithm yields an ensemble of classifiers $\hat{y}(\boldsymbol{x}; D_l)$, allowing computation of the entropy $H(\hat{y}|\boldsymbol{x}, D_l)$.

To actively select the data, we may need to calculate the *expected* error of the learner after adding an unlabeled data $\boldsymbol{x}^2$ into D_l, and select the one that has the minimal expected error to query the label:

$$E(\boldsymbol{x}^*) = \int \sum_{y^*=1}^{C} H(\hat{y}|\boldsymbol{x}, D_l, \boldsymbol{x}^*, y^*) p(y^*|\boldsymbol{x}^*, D_l) p(\boldsymbol{x}) d\boldsymbol{x} \tag{63}$$

The expectation is over the predicted label y^* since the true label of the unlabeled data is unknown.

D. Interpretation and Connections

The error-reduction-based active learning selects the unlabeled data that has the minimal expected error for querying (defined in terms of the expected entropy in the classifier output $\hat{y}(\boldsymbol{x}; D_l)$). This is equivalent to choosing the unlabeled data that gives the maximum information about the labels of the testing set. Expressed in terms of information theory, the information gain is the mutual information between all the predicted labels $\hat{Y}$ of the testing data and the predicted label y^* of unlabeled data $\boldsymbol{x}^*$:

$$\begin{aligned} \boldsymbol{I}(\hat{Y}; y^*) &= \int \left(H(\hat{y}|\boldsymbol{x}, D_l) - \sum_{y^*=1}^{C} H(\hat{y}|\boldsymbol{x}, D_l, \boldsymbol{x}^*, y^*) p(y^*|\boldsymbol{x}^*, D_l) \right) \cdot p(\boldsymbol{x}) d\boldsymbol{x} \\ &= \int \boldsymbol{I}(\hat{y}|\boldsymbol{x}; y^*|\boldsymbol{x}^*) \cdot p(\boldsymbol{x}) d\boldsymbol{x} \end{aligned} \tag{64}$$

Similar to the KL-based measure discussed above, we can also evaluate the information gain by using the KL divergence instead of using the mutual information. This is expressed as

$$\boldsymbol{I}'(\hat{Y}; y^*) = \int \left(\sum_{y^*=1}^{C} KL(p(\hat{y}|\boldsymbol{x}, D_l, \boldsymbol{x}^*, y^*) || p(\hat{y}|\boldsymbol{x}, D_l)) \cdot p(y^*|\boldsymbol{x}^*, D_l) \right) \cdot p(\boldsymbol{x}) d\boldsymbol{x} \tag{65}$$

Now consider the active learning procedure. First, we select an unlabeled data $\boldsymbol{x}^*$ and acquire its label y^*; then, $(\boldsymbol{x}^*, y^*)$ is added into D_l to induce the refined model parameter $\boldsymbol{\lambda}$ of the classifier; finally, this model is applied to predict the class labels $\hat{Y}$ of all the testing data. This procedure forms a first order Markov chain with $y^* \rightarrow \boldsymbol{\lambda} \rightarrow \hat{Y}$. By the Data Processing Inequality [20], we observe that

$$\boldsymbol{I}(\boldsymbol{\lambda}; y^*) \geq \boldsymbol{I}(\hat{Y}; y^*) \tag{66}$$

This inequality shows that $\boldsymbol{I}(\boldsymbol{\lambda}; y^*)$ is an upper bound on $\boldsymbol{I}(\hat{Y}; y^*)$, and maximizing $\boldsymbol{I}(\boldsymbol{\lambda}; y^*)$ doesn't necessarily increase $\boldsymbol{I}(\hat{Y}; y^*)$. Thus, toward the criterion of minimizing the expected error

on the test data, the MEIG, which maximizes $\boldsymbol{I}(\boldsymbol{\lambda}; y^*)$, is less desirable compared with active learning that directly maximizes $\boldsymbol{I}(\hat{Y}; y^*)$. However, as we can see from the implementation of these two algorithms, computation of $\boldsymbol{I}(\hat{Y}; y^*)$ may be intractable since it requires re-training the classifier for each unlabeled data once, and for each unlabeled data it requires to re-test on all the remaining unlabeled data. While the MEIG algorithm only requires re-training on each unlabeled data once, the estimation of the prediction error is alternatively replaced by calculating the variance of the model parameters. After practical implementation, we notice that even for the MEIG, its computational burden may be still high. In this case, the QBC algorithm appears as a simplified algorithm that measures the informativeness of an unlabeled data by calculating its classification variance among a set of classifiers. Neither re-training nor re-testing is required in the QBC.

IV. Experimental Results

We demonstrate the VB HMM and its extension to active learning, considering synthetic and measured data. For the synthetic data, the number of classes is $C = 5$ and the data of each class are generated by a 3-state HMM, with each state-dependent observation density generated by two-dimensional single Gaussian distribution. A set of sequential data are generated per class, with the sequence length of each data . This data set can be found at web site `http://www.ee.duke.edu/~lcarin/synthetic_data.zip`.

For the first experiment, we compare the classification performances of the ML and VB HMMs. We randomly select $N_l = 5$ data sequences per class as the initial labeled data set. We then sequentially select a random data sequence from the unlabeled data, acquire the associated label, and then augment the labeled data. After each augment of the training data, the ML and VB algorithms are used to retrain the HMMs, and the testing is applied on the remaining unlabeled data. In this manner we compare ML and VB training as a function of the size of the labeled data set. The average correct classification rates are calculated by averaging the correct classification rates of the five classes. The experiment is repeated 50 times and the averaged results and the standard deviations are shown in Figure 1. The results show that the VB consistently outperforms the ML, especially for small sets of labeled data. With the initial training set $(N_l = 5)$, the ML learning apparently overfits to the data and the classification performance is rather poor, while the VB obtains greater than 15% improvement. As the training data set increases, the classification

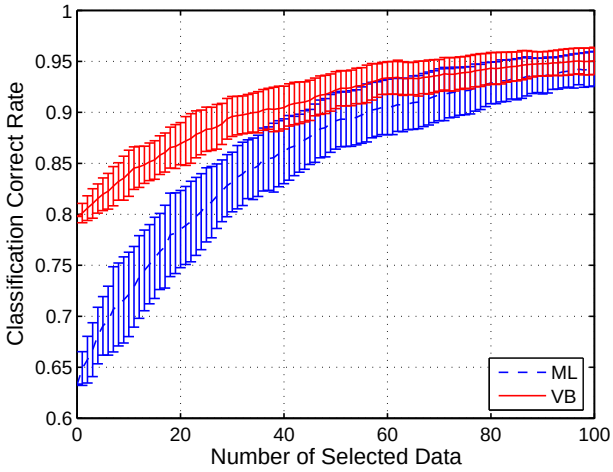

Fig. 1. Comparison of VB and ML learning. The horizontal axis indicates the number of additional (randomly selected) labeled data added to the training set. The mean (lines) results are presented as well as a standard deviation (error bars), based on 50 runs of the random data selection.

results of both methods become closer. This is not surprising since as the size of training data increases, the posterior density of model parameters becomes more sharply peaked around the ML estimate. As a detail of the experiment implementation, the parameters of VB learning are initialized by the ML point estimation. First, the ML learning is run to convergence, and then the VB learning runs from that point in parameter space to convergence. In Figure 2, an example learning curve of the VB is presented.

In the second experiment, we compare the classification performance of the active learning algorithms. Three active learning methods are considered in the experiment: the QBC with KL divergence (Sec. III.1), the MKL/MMI (Sec. III.2) and the error-reduction-based active learning (Sec. III.3). In addition, random selection of data for labeling is also included for comparison. We randomly select $N_l = 5$ data sequences per class as the initial training data set and incrementally actively select the other 100 data sequences sequentially (as in Figure 1, but now the additional labeled data are selected actively). The results on active learning are shown in Figure 3 in which some curves are the average of the multiple realizations to address the randomness of the algorithms. For example, the "random selection" results are averaged over 50 trials; "QBC (KL)" results are averaged over 20 trials. The other curves are based on one realization. For the purpose of comparison, one standard deviation of the "QBC (KL)" is

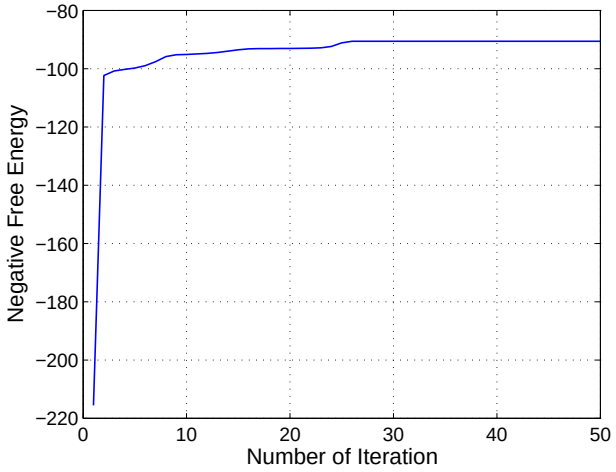

Fig. 2. An example learning curve of the VB algorithm, for the results in Figure 1. The parameters of VB learning are initialized by the ML point estimation.

also shown. All active-learning methods consistently outperform random selection. To achieve the same correct classification rate, the active learning methods need much less labeled data compared to the random selection. The MMI outperforms the QBC at the initial part of the learning process (early queries), but underperforms the QBC at the later stages. This may be because the MMI seeks the data sequence to shrink the model posterior, but discards the data sequences that may expand the model posterior. However, this effect has been considered by the MKL for the non-negative information measure of the KL divergence. We notice that the classification performance of the MKL indeed outperforms that of the MMI. This may suggest that the KL divergence is more appropriate for active learning compared to the MI measure. Moreover, the MKL approaches the upper bound of one standard deviation of the QBC (KL). Another notable comparison of the MKL to the error-reduction-based active learning is also shown in the figure. Both of their classification performances are very similar. This may suggest that without the model bias, maximizing $\boldsymbol{I}(\boldsymbol{\lambda}; y^*)$ is similar to maximizing $\boldsymbol{I}(\hat{Y}; y^*)$. In Figure 4, the maximum expected information gain of each query is plotted. The information extracted at each query generally decreases exponentially. This characteristic may be useful to design the stopping criterion. As the expected information gain approaches zero, we may stop the active learning and declare that all the information in the data set has been absorbed; no additional

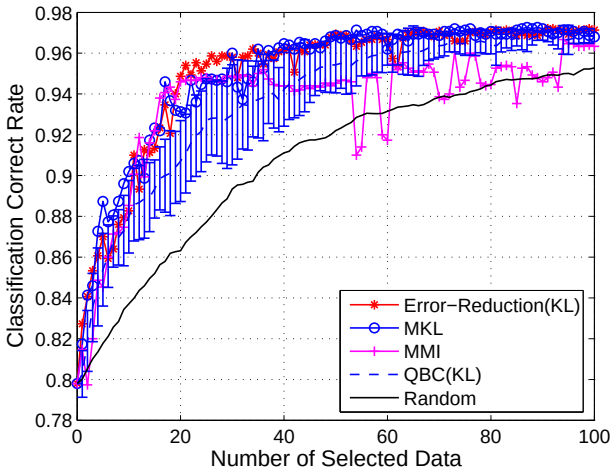

Fig. 3. Comparison of active learning on synthetic data. The horizontal axis indicates the number of actively selected labeled data added to the training set. The averaged results of the QBC are presented as well as a standard deviation (error bars), based on 20 runs of the QBC. The averaged results of the random selection via the VB are also presented for comparison.

data sequences are deemed informative for subsequent labeling.

As the final experiment, we apply the active-learning algorithms to measured acoustic-scattering data. In particular, we apply the HMMs to multi-aspect target classification. For the general theory on multi-aspect target classification with HMMs, and a description of the data and targets, interested readers should see [22]. The targets are five rotationally symmetric underwater scatters, and therefore the scattering data is collected over 360° in a plane bisecting the target axis of rotation. The data are sampled in 1° increments, in the far zone of the target (at radial distance large with respect to the target). The features of the data are extracted using matching pursuits [22] with feature-vector dimensionality 8. We generate the data sequence by sampling the target every 5° with sequence length 5. The active data selection starts after we assume access to five labeled data sequences for each target. Therefore, we have 5×5 data sequences as the initial training data set and 355×5unlabeled data sequence to which the active learning algorithms are applied. We assume a 5-state continuous HMM with each observation density generated by a mixture of two Gaussians. The results in Figure 5 are similar to that of the synthetic data, except that in this real data, the MMI outperforms the MKL. This may due to the bias of the model, since the model we selected to fit the real data may deviate from the true model. Again, the error-reduction-based active learning approaches the one-standard deviation upper bound of

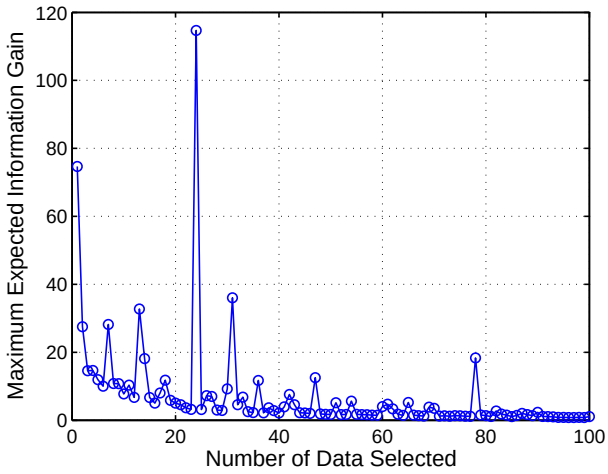

Fig. 4. The maximum expected information gain of every data query, computed for MKL (corresponding to the MKL results shown in Figure 3). The horizontal axis indicates the number of actively selected labeled data added to the training set, and the vertical axis shows the maximum information that can be extracted by each query.

the QBC. The MMI is close to the error-based active learning at the first half of the learning process (early queries), but deteriorates subsequently (when later samples are queried).

V. Conclusion

We have presented a *variational Bayes* (VB) learning algorithm for continuous HMMs, and demonstrate that the VB has the advantage of not overfitting small sets of labeled data, which often happens in v*maximum likelihood* (ML) learning. More significantly, with the posterior density of the model parameters approximated via the VB, the problem of active learning can be solved in an effective manner. The query by committee (QBC) algorithm can be implemented by directly sampling the posterior density of model parameters, to form a committee of classifiers, while previously QBC typically required multiple ML re-trainings to form the committee. The maximum expected information gain (MMI/MKL) algorithm has not been applied previously to HMMs, and has been facilitated here by minimizing the posterior density of model parameters obtained by the VB. Finally, active learning based on reducing expected classification error has been implemented in a rigorous sense via VB learning. We have also interpreted the relationships among these three algorithms in an information-theoretic context. The experiments on synthetic and measured data demonstrate the significant improvement of the active learning compared

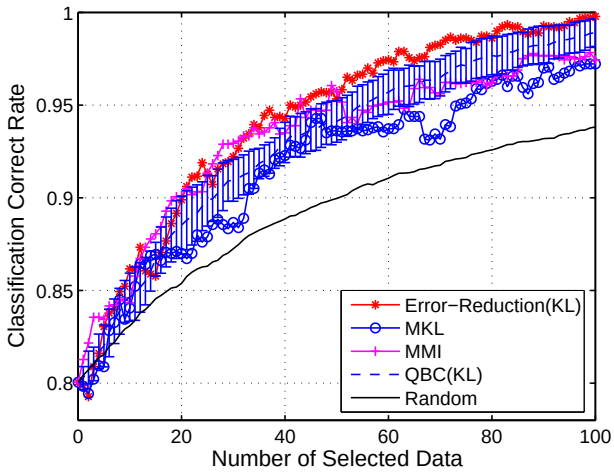

Fig. 5. Comparison of active learning on measured data. The horizontal axis indicates the number of actively selected labeled data added to the training set. The averaged results of the QBC are presented as well as a standard deviation (error bars), based on 20 runs of the QBC. The averaged results of the random selection are also presented for comparison.

to random selection of labeled data. Moreover, the MMI/MKL outperforms the QBC, and the MKL approaches one standard deviation of the upper bound of the QBC. Overall, the results of the error-reduction-based active learning were the best considered. However, the computation requirements of this approach may be infeasible compared to that of the MMI/MKL, which can be computed with much less computational resources and yield results of only slightly less quality. The future research on the active learning may focus on fast implementation of the error-reduction-based active learning, and in approximating it by the MMI/MKL with a tighter bound as expressed in (68).

APPENDIX

1. Dirichlet distribution

$$Dir(p_1, \cdots, p_N | u_1, \cdots, u_N) = \frac{\Gamma(u_0)}{\prod_{i=1}^{N} \Gamma(u_i)} \prod_{i=1}^{N} p_i^{u_i - 1} \tag{67}$$

where $\sum_{i=1}^{N} p_i = 1$, $u_i \geq 0$, and $u_0 = \sum_{i=1}^{N} u_i$ is the strength of the Dirichlet distribution.

1.1 KL divergence of Dirichlet distribution

For two Dirichlet distributions $q(p_1, \cdots, p_N) = Dir(p_1, \cdots, p_N | u_1, \cdots, u_N)$ and $p(p_1, \cdots, p_N) = Dir(p_1, \cdots, p_N | u'_1, \cdots, u'_N)$,

$$KL_{Dir}(q||p) = \log \frac{\Gamma(u_0)}{\Gamma(u'_0)} + \sum_{i=1}^{N} \log \frac{\Gamma(u'_i)}{\Gamma(u_i)} + \sum_{i=1}^{N} (u_i - u'_i)(\psi(u_i) - \psi(u_0)) \tag{68}$$

2. Wishart distribution

$$p(R|a, b) = \frac{1}{Z(a, b)} |R|^{(a-d-1)/2} \exp\left(-\frac{1}{2} Tr(bR)\right) \tag{69}$$

where

$$Z(a, b) = \pi^{d(d-1)/4} |b/2|^{-a/2} \prod_{i=1}^{d} \Gamma\left(\frac{a + 1 - i}{2}\right) \tag{70}$$

2.1 Moments of Wishart distribution

$$E(R) = ab^{-1} \tag{71}$$

$$E(\log |R|) = -\log |b/2| + \sum_{i=1}^{d} \psi\left(\frac{a + 1 - i}{2}\right) \tag{72}$$

2.2 KL divergence of Wishart distribution

For two Wishart distributions $q(R|a, b)$ and $p(R|a', b')$

$$KL_{wishart}(q||p) = \frac{a - a'}{2} E(\log |R|) - \frac{ad}{2} + \frac{a}{2} Tr(b'b^{-1}) + \log \frac{Z(a', b')}{Z(a, b)} \tag{73}$$

3. Normal-Wishart distribution

$$\begin{aligned} p(\mu, R|a, b, \boldsymbol{\lambda}, m) &= \mathcal{W}(R|a, b) \cdot \mathcal{N}(\mu|m, \boldsymbol{\lambda} R) \\ &= \frac{1}{Z(a, b)} \left(\frac{\boldsymbol{\lambda}}{2\pi}\right)^{d/2} |R|^{(a-d)/2} \exp\left(-\frac{\boldsymbol{\lambda}}{2} (\mu - m)^T R (\mu - m)\right) \\ &\quad \times \exp\left(-\frac{1}{2} Tr(bR)\right) \end{aligned} \tag{74}$$

where $\mathcal{W}(R|a, b)$ is the Wishart distribution with the degree of freedom a and the covariance matrix b; $\mathcal{N}(\mu|m, \boldsymbol{\lambda} R)$ is the Normal distribution with the mean vector m and the precision matrix $\boldsymbol{\lambda} R$.

3.1 Moments of Normal-Wishart distribution

$$E\left((x_t-\mu)^T R(x_t-\mu)\right) = a(x_t-m)^T b^{-1}(x_t-m) + d/\boldsymbol{\lambda} \tag{75}$$

3.2 KL divergence of Normal-Wishart distribution

For two normal-wishart distributions $q(\mu, R|a, b, \boldsymbol{\lambda}, m)$ and $p(\mu, R|a', b', \boldsymbol{\lambda}', m')$

$$\begin{aligned} KL_{NW}(q||p) &= \int q(\mu,R)\log\frac{q(\mu,R)}{p(\mu,R)}d\mu dR \\ &= \int q(R|a,b)\log\frac{q(R|a,b)}{p(R|a',b')}dR + \int q(R|a,b)q(\mu|m,\boldsymbol{\lambda}R)\log\frac{q(\mu|m,\boldsymbol{\lambda}R)}{p(\mu|m',\boldsymbol{\lambda}'R)}d\mu dR \\ &= KL_{wishart}(q||p) + \frac{1}{2}(d\log\frac{\boldsymbol{\lambda}}{\boldsymbol{\lambda}'} + d\frac{\boldsymbol{\lambda}'}{\boldsymbol{\lambda}} - d + \boldsymbol{\lambda}'(m-m')^T a b^{-1}(m-m')) \end{aligned} \tag{76}$$

3.3 Entropy of Normal-Wishart distribution

$$\begin{aligned} H &= -\int p(\mu,R)\log p(\mu,R)d\mu dR \\ &= \log Z(a,b) - \frac{d}{2}\log\frac{\boldsymbol{\lambda}}{2\pi} - \frac{a-d}{2}E(\log|R|) + \frac{\boldsymbol{\lambda}}{2}E\left((\mu-m)^T R(\mu-m)\right) + \frac{1}{2}Tr(bE(R)) \\ &= \frac{d(d+1)}{4}\log 4\pi + \frac{d}{2}(a+1) + \sum_{i=1}^{d}\log\Gamma\left(\frac{a+1-i}{2}\right) - \frac{d}{2}\log\boldsymbol{\lambda} - \frac{d}{2}\log|b| \\ &\quad - \frac{a-d}{2}\sum_{i=1}^{d}\psi\left(\frac{a+1-i}{2}\right) \end{aligned} \tag{77}$$

REFERENCES

[1] R. M. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed. Kluwer Academic Press, 1998.

[2] C. M. Bishop and M. E. Tipping, "Variational relevance vector machines," in *Proc. of the 16th Conference on Uncertainty in Artificial Intelligence*, 2000, pp. 46–53.

[3] T. Jaakkola and M. I. Jordan, "Bayesian parameter estimation via variational methods," *Statistics and Computing*, no. 10, pp. 25–37, 2000.

[4] D. MacKay, "Ensemble learning for hidden Markov models," Department of Physics, University of Cambridge, Tech. Rep., 1997.

[5] H. Attias, "A variational Bayesian framework for graphical models," in *Proc. Advances in Neural Information Processing Systems 12*, MIT Press, Cambridge, MA, 2000.

[6] T. P. Minka, "Using lower bounds to approximate integrals," 2001. [Online]. Available: http://www.stat.cmu.edu/~minka/papers/rem.html

[7] A. McCallum and K. Nigam, "Employing EM and pool-based active learning for text classification," in *Proc. of 15th International Conference on Machine Learning*, 1998, pp. 350–358.

[8] D. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, no. 4, pp. 590–604, 1992.

[9] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research*, vol. 4, pp. 129–145, 1996.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*, 2nd ed. Wiley Interscience, 2001.

[11] J. C. Spall, "Estimation via Markov chain Monte Carlo," *IEEE Control System Magazine*, Apr. 2003.

[12] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[13] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[14] J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Trans. Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[15] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. 18th International Conference on Machine Learning*, 2001, pp. 441–448.

[16] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2001.

[17] H. S. Seung, M. Opper, and H. Smopolinsky, "Query by committee," in *Proc. of the Fifth Annual ACM Workshop on Computational Learning Theory*, 1992, pp. 287–294.

[18] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine learning*, vol. 28, pp. 133–168, 1997.

[19] S. A. Engelson and I. Dagan, "Committee-based sample selection for probabilistic classifiers," *Journal of Artificial Intelligence Research*, pp. 335–360, 1999.

[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: Wiley, 1991.

[21] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Computation*, vol. 4, pp. 1–58, 1992.

[22] P. R. Runkle, P. K. Bharadwaj, L. Couchman, and L. Carin, "Hidden Markov models for multiaspect target classification," *IEEE Trans. Signal Proc.*, vol. 47, pp. 2035–2040, Jul. 1999.