Tradeoffs between generative and discriminative hidden Markov models

David Haussler, Tommi Jaakkola, Stephen Winters-Hilt
Computer Science Department
University of California
Santa Cruz, CA 95064

Abstract

We look at generalizations of hidden Markov models in which discriminative methods such as logistic regression are embedded within a generative framework. The discriminative methods are used to model the conditional distribution over the next state given the previous state and a context that includes sequence observations both to the left and right of the current state. These models are related to hybrid neural net/hidden Markov model methods used in speech recognition and other fields. We examine some of the statistical and computational aspects of these models. Then we report on a series of experiments we did applying these models to the problem of prediction genes in DNA.

1 Introduction

Hidden Markov models (HMMs) have proven very useful in speech recognition, biosequence analysis and other application areas [7, 12, 5]. In speech recognition and bioinformatics research, there has been considerable interest lately in more general models that combine aspects of hidden Markov models with discrimination models used in classification, such as artificial neural networks [4, 3, 2, 10]. There is evidence that richer models constructed in this manner can capture more of the subtleties of the signals being analyzed. It is also felt that better classification performance on test data will be obtained by estimating the parameters of the models to maximize classification performance on the training data, rather than maximizing the likelihood of the training data [4, 10, 6].

The straightforward statistical interpretations of standard HMMs are lost in much of the literature on hybrids of neural networks and hidden Markov models. In particular, often it is no longer the case that these models are associated with a well-defined joint distribution on the sequences of observations and hidden states. However, in a recent review of Smyth, Heckerman and Jordan, it is demonstrated that a variety of generalizations of HMMs can be defined within the framework of graphical models (also called probabilistic independence networks)[14]. Graphical models include as special cases Markov random fields (the undirected graph version) and Bayesian inference networks (the directed version). In this paper we look at a family of generalizations of HMMs that can be defined in this framework. We begin with a simple, direct derivation of this model family, without using the full machinery of graphical models, and after this we show how this family fits into the graphical model framework. We are interested in how components of these models can be viewed as doing discriminative classification, rather than generative modeling. Along the

way we look briefly at parameter representation and estimation issues, and relate these to properties of homogeneity and stationarity of the process being modeled.

2 Kinds of Hidden Markov Models

2.1 Standard HMMs

Assume that at time t an observation X_t is made from a finite set $\mathcal{A}$ of possible observations. Let $X = X_1, \ldots, X_n$ be a sequence of such observations. Our goal is to model the probability distribution over X. At each time t we assume that the system generating X is in a particular *state* Q_t, chosen from a finite set $\mathcal{S}$ of possible states. The Q_t are latent, or "hidden" variables. Let $Q = Q_0, \ldots, Q_n$. Then

$$P(X) = \sum_Q P(X, Q) \tag{1}$$

We must find a tractable parametric form for the joint distribution $P(X, Q)$. The hidden Markov model approach is to observe that

$$P(X, Q) = P(Q_0) \prod_{t=1}^{n} P(X_t, Q_t | X_1, \ldots, X_{t-1}, Q_0, \ldots, Q_{t-1}) \tag{2}$$

and then to approximate by

$$\begin{aligned} P(X_t, Q_t | X_1, \ldots, X_{t-1}, Q_0, \ldots, Q_{t-1}) &\approx P(X_t, Q_t | X_{t-1}, Q_{t-1}) && (3) \\ &= P(Q_t | X_{t-1}, Q_{t-1}) P(X_t | Q_t, X_{t-1}, Q_{t-1}) \\ &\approx P(Q_t | Q_{t-1}) P(X_t | Q_t) && (4) \end{aligned}$$

We refer to $P(Q_t|Q_{t-1})$ as the *transition probability distribution* and $P(X_t|Q_t)$ as the *output probability distribution.*

At this point we make the critical assumption that the process is homogeneous, i.e. that the transition and output distributions do not depend on the time t. Let us define the nonnegative parameter π_s denoting $P(Q_0 = s)$ for each state $s \in \mathcal{S}$ in such a way that $\sum_{s \in \mathcal{S}} \pi_s = 1$. Also define the nonnegative parameter $\theta_{s,q}$ for each pair of states s, q denoting the probability of making a transition from state s to state q, and the nonnegative parameter $\phi_q(x)$ for each state q and possible observation $x \in \mathcal{A}$, denoting the probability of "outputing" observation x in state q, in such a way that $\sum_{q \in \mathcal{S}} \theta_{s,q} = 1$ for all s and $\sum_{x \in \mathcal{A}} \phi_q(x) = 1$ for all q. Let us denote by Θ the set of all parameters π_s, $\theta_{s,q}$ and $\phi_q(x)$. For all t, set $P(Q_t = q | Q_{t-1} = s, \Theta) = \theta_{s,q}$ and $P(X_t = x | Q_t = q, \Theta) = \phi_q(x)$. Putting this together with the above equations, this gives the HMM model

$$P(X|\Theta) = \sum_Q P(X, Q|\Theta) = \sum_Q P(Q_0) \prod_{t=1}^{n} P(Q_t|Q_{t-1}, \Theta) P(X_t|Q_t, \Theta) = \sum_Q \pi_{Q_0} \prod_{t=1}^{n} \theta_{Q_{t-1}, Q_t} \phi_{Q_t}(X_t) \tag{5}$$

What makes this approximation useful is that the dependence of the current state Q_t on the previous states is reduced to a simple Markov dependence on Q_{t-1}. This makes it possible to consider all possible reconstructions of the full state path hidden variable Q in an efficient manner, using recursive equations that are implemented by dynamic programming or message passing methods. This is discussed in detail in [12, 14].

2.2 Discriminative versus generative models

Often a hidden Markov model is used to predict something about the hidden state sequence Q. For example, the Viterbi dynamic programming algorithm can be used to efficiently compute

$$Q^* = \text{argmax}_Q P(X, Q|\theta) = \text{argmax}_Q P(Q|X, \Theta),$$

the most likely reconstruction of the hidden state sequence given the observed sequence X and the parameters Θ. HMMs are *globally generative* models in this context, because they specify a joint distribution on X and Q, and condition on this distribution to make predictions about Q. This contrasts with methods discussed in [3, 10], which directly define a conditional distribution $P(Q|X)$ and reason from this. The latter methods can be called *globally discriminative*. They only provide the conditional distribution, and in fact cannot be used to define a joint distribution on X and Q.

HMMs are also *locally generative* models, in that for every t, the parameters of an HMM specify $P(X_t|Q_t)$ and a conditional "prior" $P(Q_t|Q_{t-1})$, rather than attempting to model $P(Q_t|Q_{t-1}, \text{context}(X_t))$, where $\text{context}(X_t)$ includes X_t and surrounding observations to the left and the right. A model of the latter type might be called *locally discriminative*. See [3, 2, 10] for examples and discussion. There is evidence that locally discriminative models can sometimes be quite effective. Here we examine the probabilistic foundations for such models, and look at their strengths and weaknesses.

2.3 Including a left context

To move toward locally discriminative hidden Markov models, first let us begin with the often noted observation that it is relatively straightforward to make somewhat less severe approximations in (3) and (4) regarding the dependence on previous observations. In particular, without substantially changing the dynamic programming evaluation methods, for any finite *left window size* l, we can instead use the approximations

$$\begin{aligned}
&P(X_t, Q_t|X_1, \ldots, X_{t-1}, Q_0, \ldots, Q_{t-1}) \\
&\approx P(X_t, Q_t|X_{t-l}, \ldots, X_{t-1}, Q_{t-1}) \qquad (6) \\
&= P(Q_t|X_{t-l}, \ldots, X_{t-1}, Q_{t-1}) P(X_t|Q_t, X_{t-l}, \ldots, X_{t-1}, Q_{t-1}) \\
&\approx P(Q_t|X_{t-l}, \ldots, X_{t-1}, Q_{t-1}) P(X_t|Q_t, X_{t-l}, \ldots, X_{t-1}) \qquad (7)
\end{aligned}$$

To handle the beginning of X where there is insufficient left context without special cases, it is convenient to re-index the sequence X so that now $X = X_{1-l}, \ldots, X_n$, but keep $Q = Q_0, \ldots, Q_n$. We assume that the parameter vector Θ_l (discussed further below) includes a specification for the joint distribution $P(X_{1-l}, \ldots, X_0, Q_0)$. Then Equation (7) gives the model

$$\begin{aligned}
&P(X|\Theta_l) = \\
&\sum_Q P(X_{1-l}, \ldots, X_0, Q_0|\Theta_l) P(X_1, \ldots, X_n, Q_1, \ldots, Q_n|X_{1-l}, \ldots, X_0, Q_0, \Theta_l) \approx
\end{aligned}$$

$$\sum_{Q} P(X_{1-l}, \ldots, X_0, Q_0|\Theta_l) \prod_{t=1}^{n} P(Q_t|X_{t-l}, \ldots, X_{t-1}, Q_{t-1}, \Theta_l) P(X_t|Q_t, X_{t-l}, \ldots, X_{t-1}, \Theta_l) \quad (8)$$

The cost of this more general model is in the increased complexity of the parametric sub-models used for the transition and output probability distributions. We must define a set of parameters Θ_l that specifies

$$P(Q_t = q|X_{t-l} = x_{t-l}, \ldots, X_{t-1} = x_{t-1}, Q_{t-1} = s, \Theta_l)$$

for all possible $s, q, x_{t-l}, \ldots, x_{t-1}$ and

$$P(X_t = x|Q_t = q, X_{t-l} = x_{t-l}, \ldots, X_{t-1} = x_{t-1}, \Theta_l)$$

for all possible $x, q, x_{t-l}, \ldots, x_{t-1}$. If Θ_l is specified using tables of parameters, as in a simple lth order Markov process, then these tables become unmanageably large for large l, and, more importantly, they contain more parameters than can be accurately estimated from a reasonable amount of training data. However, we can instead define these distributions using a flexible class of parametric models such as standard logistic regression models, kernel regression models [9], multi-layer neural network models, or with two stage methods such as (real) AdaBoost [13] or LogitBoost [8].

If we choose standard logistic regression models, we would first choose a set of real-valued functions to extract features from the context $X_{t-l}, \ldots, X_{t-1}$. These can be anything from simple functions indicating the presence or absence of particular kinds of letters in particular positions in this context window, to more complex sequence pattern measurements. The features chosen can be different for each state s. Let us denote the feature functions associated with state s by $f_{s,1}, \ldots, f_{s,m}$. Using these features, to model the transition probabilities we would define

$$P(Q_t = q|X_{t-l} = x_{t-l}, \ldots, X_{t-1} = x_{t-1}, Q_{t-1} = s, \Theta_l) = \frac{\exp\left(\theta_{s,q,0} + \sum_{k=1}^{m} \theta_{s,q,k} f_{s,k}(x_{t-l}, \ldots, x_{t-1})\right)}{Z}$$

where

$$Z = \sum_{q'} \exp\left(\theta_{s,q',0} + \sum_{k=1}^{m} \theta_{s,q',k} f_{s,k}(x_{t-l}, \ldots, x_{t-1})\right)$$

for some real-valued parameters $\theta = \{\theta_{s,q,k}\}$.

$$P(X_t = x|Q_t = q, X_{t-l} = x_{t-l}, \ldots, X_{t-1} = x_{t-1}, \Theta_l)$$

can be defined in an analogous manner using parameters $\phi = \{\phi_{q,k}(x)\}$. A series of logistic regression models on successively shorter contexts can be used to define the joint distribution $P(X_{1-l}, \ldots, X_0, Q_0|\Theta_l)$. Let π is the collection of all parameters from these models. Then $\Theta_l = (\theta, \phi, \pi)$.

The parameters of Θ_l can be estimated from "labeled" training data consisting of several pairs (X, Q) generated from $P(X, Q)$. In fact, θ, ϕ and π can be estimated entirely independently from such data, say by Maximum Likelihood (ML) or Maximum A Posteriori (MAP) methods. If we only have unlabeled training data consisting of sequences generated from

$P(X)$, then we can use the Expectation Maximization (EM) algorithm, or a generalization of this method [12, 1].

Finally, note that at the expense of introducing a larger parameter set, we might avoid the last approximation (7) in the derivation above, where the dependence on Q_{t-1} is dropped. This just expands Θ to $\{\theta_{s,s',q,k}\}$, which might be tolerable if the set of states is small. However, it slows down the dynamic programming methods used to evaluate likelihoods and conditional probabilities, as now they must keep track of all possible pairs of values for Q_{t-1} and Q_t at each time step t. If we go further and allow dependence on $Q_{t-l}, \ldots, Q_{t-1}$ for large l, then we would have to keep track of all possible combinations of values for these l variables for each t, which would be intolerable unless the set of possible state transitions is severely restricted due to some special structure of the Markov process.

2.4 Two-sided context: DHMMs

We now consider what happens if we allow a still more general transition probability distribution that depends not only on a left context window $X_{t-l}, \ldots, X_{t-1}$, but also on a *right context window* $X_t, \ldots, X_{t+r-1}$ for some fixed $r \geq 0$. We denote the entire context window by

$$W_t = X_{t-l}, \ldots, X_{t+r-1}.$$

We re-index the sequence X so that now $X = X_{1-l}, \ldots, X_{n+r}$, and assume that the parameter vector $\Theta_{l,r}$ includes a specification for the joint distribution $P(X_{1-l}, \ldots, X_r, Q_0)$. Then we have the approximation

$$\begin{aligned} &P(X|\Theta_{l,r}) = \\ &\sum_Q P(W_1, Q_0|\Theta_{l,r}) P(X_{r+1}, \ldots, X_{n+r}, Q_1, \ldots, Q_n|W_1, Q_0, \Theta_{l,r}) \end{aligned} \tag{9}$$

and

$$\begin{aligned} &P(X_{r+1}, \ldots, X_{n+r}, Q_1, \ldots, Q_n|W_1, Q_0, \Theta_{l,r}) \\ &= \prod_{t=1}^{n} P(X_{t+r}, Q_t|X_{1-l}, \ldots, X_{t+r-1}, Q_0, \ldots, Q_{t-1}, \Theta_{l,r}) \\ &\approx \prod_{t=1}^{n} P(X_{t+r}, Q_t|W_t, Q_{t-1}, \Theta_{l,r}) \qquad (10) \\ &= \prod_{t=1}^{n} P(Q_t|W_t, Q_{t-1}, \Theta_{l,r}) P(X_{t+r}|W_t, Q_t, Q_{t-1}, \Theta_{l,r}) \\ &\approx \prod_{t=1}^{n} P(Q_t|W_t, Q_{t-1}, \Theta_{l,r}) P(X_{t+r}|W_t, Q_t, \Theta_{l,r}) \qquad (11) \end{aligned}$$

Putting this altogether, we get the following class of models, indexed by l and r:

$$P(X|\Theta_{l,r}) = \sum_Q P(W_1, Q_0, \Theta_{l,r}) \prod_{t=1}^{n} P(Q_t|W_t, Q_{t-1}, \Theta_{l,r}) P(X_{t+r}|W_t, Q_t, \Theta_{l,r}) \tag{12}$$

For $l = r = 0$, we get the standard HMM model (5) as a special case. As above, we may use any parametric models, such as logistic regression models, to define the initial, transition and output probability distributions. We will call this general class of models *discriminative hidden Markov models (DHMMs)*.

In any particular application, we may assume that there is a true joint distribution $P(X, Q)$, and that $P(X) = \sum_Q P(X, Q)$. For each l and r, (12) can be used to get a tractable approximation to this true distribution. It is not clear, however, which choice of l and r will give the best results. The term $P(Q_t|W_t, Q_{t-1}, \Theta_{l,r})$ constitutes a locally discriminative transition probability distribution. If we use a logistic regression model, then to represent this distribution, we would extract features from the window W_t, which includes the context both before and after the current state Q_t, and use these features, along with the previous state Q_{t-1}, to predict this current state. Presumably, this prediction will become more accurate as we increase the "lookahead" r, although beyond a certain limit, with limited training data we may begin to suffer from overfitting in our estimation of $\Theta_{l,r}$ even with a carefully designed parametric model. We might call the term $P(X_{t+r}|W_t, Q_t, \Theta_{l,r})$ an *offset* output probability distribution, since it specifies the probability of outputting the observation at a position that is r symbols to the right to the current state Q_t. Presumably this term will become less accurate as r increases, since the output symbol you are predicting becomes more distant from the position where you have state information. Thus, there appears to be a tradeoff in the choice of r. This trade off can be viewed as the tradeoff between emphasizing the locally discriminative model (large r) and emphasizing the locally generative model (small r). On the other hand, it appears that it is always desirable to make l as large as possible, subject to the problem of overfitting.

3 Interpretation of DHMMs as graphical models

DHMMs can easily be represented as graphical models. In such models, there is a node for every random variable, e.g. in our case there is a node for every X_t and a node for every Q_t. An edge between a pair of nodes represents a dependence between the corresponding random variables, and lack of an edge represents a kind of conditional independence relation. A self contained review of this methodology is given in [14]. For simplicity, in the following we will identify a node with the random variable it represents.

The definition we have given for DHMMs translates immediately into a graphical model defined by a directed acyclic graph. These models are sometimes called Bayesian (belief or inference) nets [11]. In this representation, each random variable has an incoming edge from each other variable that it directly depends on. Thus, the equation

$$P(X, Q|\Theta_{l,r}) = P(W_1, Q_0, \Theta_{l,r}) \prod_{t=1}^{n} P(Q_t|W_t, Q_{t-1}, \Theta_{l,r}) P(X_{t+r}|W_t, Q_t, \Theta_{l,r})$$

specifies a directed graph in which, for $1 \leq t \leq n$, each Q_t has an incoming edge from Q_{t-1} and from each node in W_t, and each X_{t+r} has an incoming edge from Q_t and from each node in W_t. The equation does not specify the incoming edges for Q_0 or for the nodes in W_1; these can be specified in an arbitrary manner. For example, if we decompose the joint

distribution on these random variables as

$$P(W_1, Q_0) = P(X_{1-l}) \left(\prod_{i=1-l+1}^{0} P(X_i | X_{1-l}, \ldots, X_{i-1}) \right) P(Q_0 | W_1),$$

then each of these terms specifies an explicit set of incoming edges for the variable for which it defines a conditional distribution.

In Bayesian belief nets, a node Y with incoming edges from $V_1, \ldots, V_k$ has associated with it a representation of the conditional distribution $P(Y|V_1, \ldots, V_k)$. When the variables are discrete, as they are here, and k is small, this conditional distribution is often specified by an explicit table of conditional probabilities for every combination of values of $V_1, \ldots, V_k$. However, any (parametric or non parametric) model can be used to specify this conditional distribution. In applications of DHMMs to biosequence analysis, speech recognition and other time series analysis, it is impractical to have separate models for the conditional distributions for X_{t+r} and Q_t for each time t, since the total length n varies widely from instance to instance, and in some cases is extremely large. Instead we have been assuming that the underlying process is homogeneous in t, so that these conditional distributions are the same for all t. As shown in [14], the primary parameter estimation methods for graphical models are easily modified to handle this constraint, and generalize the methods used for standard HMMs.

A more symmetric model is obtained by turning to undirected graphical models, which are equivalent to Markov random fields. Let us return to Equation (10)[1], and rewrite each term as

$$P(X_{t+r}, Q_t | W_t, Q_{t-1}) = \frac{P(W_t, X_{t+r}, Q_{t-1}, Q_t)}{P(W_t, Q_{t-1})}.$$

Let

$$C_t = W_t, X_{t+r}, Q_{t-1}, Q_t = X_{t-l}, \ldots, X_{t+r}, Q_{t-1}, Q_t.$$

Then for a DHMM we have

$$P(X, Q) = \frac{\prod_{t=1}^{n} C_t}{\prod_{t=1}^{n-1} C_t \cap C_{t+1}} \tag{13}$$

This expansion shows that DHMMs are decomposable models [11]. If we connect every pair of nodes in every set C_t with a undirected edge, then the sets C_t form maximal complete subgraphs (called *cliques*). The resulting (chordal) graph represents the Markov random field, and (13) gives the factorization of the joint distribution into a product of clique potential functions associated with this Markov random field.

To specify a parametric representation for this model, we must define a joint distribution $P(C_t|\Theta_{l,r})$ for every clique C_t. As above, it is impractical to define a separate joint distribution for every t. However, assuming that these distributions are the same for all t is not equivalent to the assumption we have been making, namely that the process is homogeneous in t. Since $P(C_t) = P(W_t, Q_{t-1})P(X_{t+r}, Q_t | W_t, Q_{t-1})$ this is equivalent to the stronger assumption that the process is homogeneous, i.e. that the conditional $P(X_{t+r}, Q_t | W_t, Q_{t-1})$ does not depend on t, and *stationary*, i.e. that the marginal $P(W_t, Q_{t-1})$ does not depend

[1]This is equivalent to using the slightly more general version of DHMMs in which the term $P(X_{t+r}|W_t, Q_t, \Theta_{l,r})$ is replaced by $P(X_{t+r}|W_t, Q_{t-1}, Q_t, \Theta_{l,r})$ in (12).

on t. At first blush this seems like too strong an assumption to make in practice. For example, if one were modeling speech utterances, each of which was a complete sentence, then surely the marginal distributions over the hidden states and observations at the beginning of the sentence are different from those at the end of the sentence. In contrast, when we are modeling continuous speech, with random choice of where we start and stop recording observations, then it is entirely reasonable to assume a stationary model. In that case, due to the invariance imposed by randomizing the choice the subsequence X from within a much longer sequence, we have insured that the marginal distributions of any piece of X and Q do not depend on t.

Even if your ultimate goal is only to model individual sentences, you can easily create an artificial stationary process that embeds your nonstationary model. First you add a special "end of sentence" symbol to $\mathcal{A}$, appending this to the end of every observed sentence, and add a corresponding special end state in $\mathcal{S}$ with outgoing Markov transitions that reinitialize the system to the initial state distribution π from the old, non stationary process. Whenever the "end of sentence" symbol is encountered, a transition is made to the special end state with probability 1. Reasonable care is taken to insure that the resulting system is aperiodic, and thus has a well defined ergodic limit. A new initial distribution is defined that is equal to this ergodic limit. The resulting model is a stationary model for random samples excised from a simulated continuous speech process in which sentences are concatenated together endlessly, separated by "end of sentence" symbols. No information is lost in this exercise, and modeling with a stationary model becomes possible. Thus the assumption of stationarity is not so onerous as it may first appear.

Making the additional assumption of stationarity gives us more options in choosing a model and a parameter estimation method. The central issue is choosing how to represent the joint distribution $P(C_t)$, as the distribution for $P(C_t \cap C_{t+1})$ in (13) can be obtained as a marginal of $P(C_t)$. One option is to use a locally generative approach, decomposing as $P(C_t) = P(W_t, X_{t+r}|Q_{t-1}, Q_t)P(Q_{t-1}, Q_t)$. Since the distributions do not dependent on t, we can model $P(Q_{t-1}, Q_t)$ by $|\mathcal{A}|^2 - 1$ explicit parameters, and use as sophisticated a generative model as we like to model $P(W_t, X_{t+r}|Q_{t-1}, Q_t)$. Even though this is a fully generative framework, it still allows us to model context effects easier than the standard HMM model, in which the equivalent term would represent $P(X_t|Q_t)$. It might be argued that such context effects can be incorporated into the definition of the state, and thereby a DHMM would be reduced to a standard HMM. However, this might not lead to a finite state set. For example, consider the case when we have a mixture model

$$P(W_t, X_{t+r}|Q_{t-1}, Q_t) = \int_0^1 P(W_t, X_{t+r}|\alpha, Q_{t-1}, Q_t)p(\alpha)d\alpha,$$

where α might represent some local "noise level" or other locally varying aspect of the signal. In such a case there may be no direct translation into a standard HMM, only a series of approximations using larger and larger state sets, which end up being impractical.

If one's desire is only to predict Q from X, then it might be better to try to use locally discriminative models instead, decomposing by $P(C_t) = P(Q_{t-1}, Q_t|W_t, X_{t+r})P(W_t, X_{t+r})$, and representing $P(Q_{t-1}, Q_t|W_t, X_{t+r})$ by a logistic regression model, or a more complex model, as discussed above. Note however, that we still need to model $P(W_t, X_{t+r})$. We can

do this with a separate generative model, giving an over all model of the form

$$P(C_t) = P(Q_{t-1}, Q_t | W_t, X_{t+r}, \phi) \sum_{q,r \in \mathcal{S}} P(W_t, X_{t+r} | Q_{t-1} = q, Q_t = r, \theta) P(Q_{t-1} = q, Q_t = r | \pi),$$

where ϕ is the parameter vector for the discriminative model, and θ and π are the parameters for the generative model. This may seem like a senselessly redundant model, however, it may have some advantages if the discrimative model $P(Q_{t-1}, Q_t | W_t, X_{t+r}, \phi)$ can capture some aspects of the joint distribution that the generative model is unable to represent.

Because they define a full joint probability distribution on X and Q, the methods we have been using are easily extended to the case when there is missing data in the training examples. A common situation is that most or all of the training examples are unlabeled, consisting of X and not Q. It is often cheaper to get unlabeled data, resulting in a training set with a few labeled examples and a massive number of unlabeled examples. Another common situation is partially labeled examples. MAP of ML parameter estimation in these situations can be handled by (generalized) expectation maximization methods, as described in [14].

An alternate approach is to define a globally discriminative model $P(Q|X, \theta)$, and estimate θ directly from labeled examples, by maximizing $P(Q|X, \theta)$ over θ, sometimes called *conditional maximum likelihood*, and equivalent to what is called the *MMI method* in speech reognition [3, 10]. In this approach, there is no principled way to use unlabeled and partially labeled training examples, since no joint distribution on X and Q is defined by the model. However, if prediction of Q from X is what you are after, one can often obtain better performance by directly optimizing the model for this task. One general model here is

$$P(Q|X, \theta) = \frac{\exp\left(\sum_{t=1}^{n} f(W_t, X_{t+r}, Q_{t-1}, Q_t, \theta)\right)}{Z(X, \theta)},$$

where

$$Z(X, \theta) = \sum_{Q'} \exp\left(\sum_{t=1}^{n} f(W_t, X_{t+r}, Q'_{t-1}, Q'_t, \theta)\right),$$

and f is any function. One can attempt to optimize the parameters θ by gradient descent, or by more sophisticated methods, as described in [10], using dynamic programming methods to avoid doing an explict sum over possible state paths Q'. In particular, it is possible to set $f(W_t, X_{t+r}, Q_{t-1}, Q_t, \theta) = \log P(Q_t, X_{t+r} | W_t, Q_{t-1})$ and get the DHMM model essentially as a special case of this approach, in which we work directly with the conditional distribution $P(Q|X, \theta)$, but the underlying model supports a full joint distribution on X and Q [10]. The key difference is then in the method used to estimate the parameters.

4 Experiments

???

5 Conclusion

We have explored a family of generalizations of hidden Markov models that allows the system designer to emphasize either the discriminative or the generative aspects of the model by choosing different parameterizations of the local transition and output probability distributions. Alternate methods for parameter estimation can also be applied to these models, optimizing either a globally generative or a globally discriminative criterion. It now remains to compare the different choices experimentally on real datasets, to see which method is the most effective. Experiments could also indicate how large the left and right context windows should be to optimize performance on typical datasets. Further theoretical work is needed to generalize these methods to use tree-structured models, e.g. for applications in image analysis.

References

[1] P. Baldi and Y. Chauvin. Smooth on-line learning algorithms for hidden Markov models. *Neural Computation*, 6(2):305–316, 1994.

[2] Y. Bengio. *Neural Networks for Speech and Sequence Recognition.* International Thomson Computer Press, 1996.

[3] H. Bourlard and N. Morgan. *Connectionist Speech Recognition, a hybrid approach.* Kluwer, 1993.

[4] H. Bourlard, N. Morgan, and S. Renals. Neural nets and hidden markov models - review and generalizations. *Speech Communication*, 11(2-3):237–246, 1992.

[5] S. Eddy. Hidden Markov models. *Curr. Opin. Struct. Biol.*, 6(3):361–365, 1996.

[6] S. Eddy, G. Mitchison, and R. Durbin. Maximum discrimination hidden Markov models of sequence consensus. *J. Comput. Biol.*, 2:9–23, 1995.

[7] R. Elliott, L. Aggoun, and J. Moore. *Hidden Markov Models, Estimation and Control.* Springer Verlag, 1995.

[8] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting, 1998. unpublished manuscript.

[9] T. Jaakkola and D. Haussler. Probabilistic kernel regression models, 1998. unpublished manuscript.

[10] A. Krogh and S. Riis. Hidden neural networks. *Neural Computation*, 1998. too appear.

[11] J. Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, 1988.

[12] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, Feb. 1989.

[13] R. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. In *Proc. 11th conference on computational learning theory*, 1997.

[14] P. Smyth, D. Heckerman, and M. Jordan. Probabilistic independence networks for hidden markov probability models. *Neural Computation*, 9:227–269, Feb. 1997.