Artificial Intelligence 174 (2010) 215–243

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Hidden semi-Markov models

Shun-Zheng Yu

Department of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou 510275, PR China

ARTICLE INFO

Article history:
Received 14 April 2009
Available online 17 November 2009

Keywords:
Hidden Markov model (HMM)
Hidden semi-Markov model (HSMM)
Explicit duration HMM
Variable duration HMM
Forward–backward (FB) algorithm
Viterbi algorithm

ABSTRACT

As an extension to the popular hidden Markov model (HMM), a hidden semi-Markov model (HSMM) allows the underlying stochastic process to be a semi-Markov chain. Each state has variable duration and a number of observations being produced while in the state. This makes it suitable for use in a wider range of applications. Its forward–backward algorithms can be used to estimate/update the model parameters, determine the predicted, filtered and smoothed probabilities, evaluate goodness of an observation sequence fitting to the model, and find the best state sequence of the underlying stochastic process. Since the HSMM was initially introduced in 1980 for machine recognition of speech, it has been applied in thirty scientific and engineering areas, such as speech recognition/synthesis, human activity recognition/prediction, handwriting recognition, functional MRI brain mapping, and network anomaly detection. There are about three hundred papers published in the literature. An overview of HSMMs is presented in this paper, including modelling, inference, estimation, implementation and applications. It first provides a unified description of various HSMMs and discusses the general issues behind them. The boundary conditions of HSMM are extended. Then the conventional models, including the explicit duration, variable transition, and residential time of HSMM, are discussed. Various duration distributions and observation models are presented. Finally, the paper draws an outline of the applications.

1. Introduction (History)

A hidden Markov model (HMM) is defined as a doubly stochastic process. The underlying stochastic process is a discrete-time finite-state homogeneous Markov chain. The state sequence is not observable and so is called hidden. It influences another stochastic process that produces a sequence of observations. An excellent tutorial of HMMs can be found in Rabiner [150], a theoretic overview of HMMs can be found in Ephraim and Merhav [57] and a discussion on learning and inference in HMMs in understanding of Bayesian networks is presented in Ghahramani [66]. The HMMs are an important class of models that are successful in many application areas. However, due to the non-zero probability of self-transition of a non-absorbing state, the state duration of an HMM is implicitly a geometric distribution. This makes the HMM has limitations in some applications.

As an extension of the HMM, a hidden semi-Markov model (HSMM) is traditionally defined by allowing the underlying process to be a semi-Markov chain. Each state has a variable duration, which is associated with the number of observations produced while in the state. The HSMM is also called “explicit duration HMM” [60,150], “variable-duration HMM” [107, 155,150], “HMM with explicit duration” [124], “hidden semi-Markov model” [126], generalized HMM [94], segmental HMM [157] and segment model [135,136] in the literature, depending on their assumptions and their application areas.

E-mail address: syu@mail.sysu.edu.cn.

0004-3702/$ – see front matter
doi:10.1016/j.artint.2009.11.011

The first approach to hidden semi-Markov model was proposed by Ferguson [60], which is partially included in the survey paper by Rabiner [150]. This approach is called the explicit duration HMM in contrast to the implicit duration of the HMM. It assumes that the state duration is generally distributed depending on the current state of the underlying semi-Markov process. It also assumes the "conditional independence" of outputs. Levinson [107] replaced the probability mass functions of duration with continuous probability density functions to form a continuously variable duration HMM. As Ferguson [60] pointed out, an HSMM can be realized in the HMM framework in which both the state and its sojourn time since entering the state are taken as a complex HMM state. This idea was exploited in 1991 by a 2-vector HMM [93] and a duration-dependent state transition model [179]. Since then, similar approaches were proposed in many applications. They are called in different names such as inhomogeneous HMM [151], non-stationary HMM [164], and recently triplet Markov chains [144]. These approaches, however, have the common problem of computational complexity in some applications. A more efficient algorithm was proposed in 2003 by Yu and Kobayashi [199], in which the forward–backward variables are defined using the notion of a state together with its remaining sojourn (or residual life) time. This makes the algorithm practical in many applications.

The HSMM has been successfully applied in many areas. The most successful application is in speech recognition. The first application of HSMM in this area was made by Ferguson [60]. Since then, there have been more than one hundred such papers published in the literature. It is the application of HSMM in speech recognition that enriches the theory of HSMM and develops many algorithms for HSMM.

Since the beginning of 1990's, the HSMM started being applied in many other areas such as electrocardiograph (ECG) [174], printed text recognition [4] or handwritten word recognition [95], recognition of human genes in DNA [94], language identification [118], ground target tracking [88], document image comparison and classification at the spatial layout level [81], etc.

In recent years from 2000 to present, the HSMM has been obtained more and more attentions from vast application areas such as change-point/end-point detection for semi-conductor manufacturing [64], protein structure prediction [162], mobility tracking in cellular networks [197], analysis of branching and flowering patterns in plants [69], rain events time series model [159], brain functional MRI sequence analysis [58], satellite propagation channel modelling [112], Internet traffic modelling [198], event recognition in videos [79], speech synthesis [204,125], image segmentation [98], semantic learning for a mobile robot [167], anomaly detection for network security [201], symbolic plan recognition [54], terrain modelling [185], adaptive cumulative sum test for change detection in non-invasive mean blood pressure trend [193], equipment prognosis [14], financial time series modelling [22], remote sensing [147], classification of music [113], and prediction of particulate matter in the air [52], etc.

The rest of the paper is organized as follows: Section 2 is the major part of this paper that defines a unified HSMM and addresses important issues related to inference, estimation and implementation. Section 3 then presents three conventional HSMMs that are applied vastly in practice. Section 4 discusses the specific modelling issues, regarding duration distributions, observation distributions, variants of HSMMs, and the relationship to the conventional HMM. Finally, Section 5 highlights major applications of HSMMs and concludes the paper in Section 6.

2. Hidden semi-Markov model

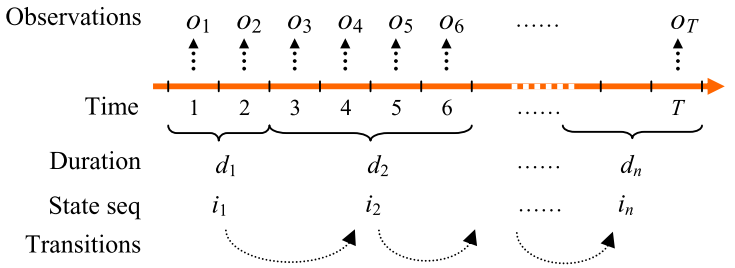
This section provides a unified description of HSMMs. A general HSMM is defined without specific assumptions on the state transitions, duration distributions and observation distributions. Then the important issues related to inference, estimation and implementation of the HSMM are discussed. A general expression of the explicit-duration HMMs and segment HMMs can be found in Murphy [126], and a unified view of the segment HMMs can be found in Ostendorf et al. [136]. Detailed review for the conventional HMM can be found in the tutorial by Rabiner [150], the overview by Ephraim and Merhav [57], the Bayesian networks-based discussion by Ghahramani [66], and the book by Cappe et al. [29].

2.1. General model

A hidden semi-Markov model (HSMM) is an extension of HMM by allowing the underlying process to be a semi-Markov chain with a variable *duration* or *sojourn time* for each state. Therefore, in addition to the notation defined for the HMM, the duration d of a given state is explicitly defined for the HSMM. State duration is a random variable and assumes an integer value in the set $\mathcal{D} = \{1, 2, \ldots, D\}$. The important difference between HMM and HSMM is that one observation per state is assumed in HMM while in HSMM each state can emit a sequence of observations. The number of observations produced while in state i is determined by the length of time spent in state i, i.e., the duration d. Now we provide a unified description of HSMMs.

Assume a discrete-time Markov chain with the set of (hidden) states $\mathcal{S} = \{1, \ldots, M\}$. The state sequence is denoted by $S_{1:T} \triangleq S_1, \ldots, S_T$, where $S_t \in \mathcal{S}$ is the state at time t. A realization of $S_{1:T}$ is denoted as $s_{1:T}$. For simplicity of notation in the following sections, we denote:

- $S_{t_1:t_2} = i$ – state i that the system stays in during the period from t_1 to t_2. In other words, it means $S_{t_1} = i$, $S_{t_1+1} = i, \ldots,$ and $S_{t_2} = i$. Note that the previous state S_{t_1-1} and the next state S_{t_2+1} may or may not be i.



The first state i_1 and its duration d_1 are selected according to the transition probability $a_{(i_0,d_0)(i_1,d_1)}$. (i_0, d_0) is the initial state and duration. i_1 lasts for $d_1 \geq 2$ time units in this instance. It produces two observations (o_1, o_2) according to the emission probability $b_{i_1,d_1}(o_{1:2})$. It transits, according to the transition probability $a_{(i_1,d_1)(i_2,d_2)}$, to state i_2. i_2 lasts for $d_2 = 4$ time units, which produces four observations (o_3, o_4, o_5, o_6) according to the emission probability $b_{i_2,d_2}(o_{3:6})$. (i_2, d_2) then transits to (i_3, d_3), ..., (i_n, d_n) until the final observation o_T is produced. The last state i_n may last beyond time T.

Fig. 1. General HSMM.

- $S_{[t_1:t_2]} = i$ – state i which starts at time t_1 and ends at t_2 with duration $d = t_2 - t_1 + 1$. This implies that the previous state S_{t_1-1} and the next state S_{t_2+1} must not be i.
- $S_{[t_1:t_2} = i$ – state i that starts at time t_1 and lasts till t_2, with $S_{[t_1} = i$, $S_{t_1+1} = i, \ldots, S_{t_2} = i$, where $S_{[t_1} = i$ means that at t_1 the system switched from some other state to i, i.e., the previous state s_{t_1-1} must not be i. The next state S_{t_2+1} may or may not be i.
- $S_{t_1:t_2]} = i$ – state i that lasts from t_1 to t_2 and ends at t_2 with $S_{t_1} = i$, $S_{t_1+1} = i, \ldots, S_{t_2]} = i$, where $S_{t_2]} = i$ means that at time t_2 the state will end and transit to some other state, i.e., the next state S_{t_2+1} must not be i. The previous state S_{t_1-1} may or may not be i.

Based on these definitions, $S_{[t]} = i$ means state i starting and ending at t with duration 1, $S_{[t} = i$ means state i starting at t, $S_{t]} = i$ means state i ending at t, and $S_t = i$ means the state at t being state i.

Denote the observation sequence by $O_{1:T} \triangleq O_1, \ldots, O_T$, where $O_t \in \mathcal{V}$ is the observable at time t and $\mathcal{V} = \{v_1, v_2, \ldots, v_K\}$ is the set of observable values. For observation sequence $o_{1:T}$, the underlying state sequence is $S_{1:d_1]} = i_1$, $S_{[d_1+1:d_1+d_2]} = i_2, \ldots, S_{[d_1+\cdots+d_{n-1}+1:d_1+\cdots+d_n} = i_n$, and the state transitions are $(i_m, d_m) \to (i_{m+1}, d_{m+1})$, for $m = 1, \ldots, n-1$, where $\sum_{m=1}^{n} d_m = T$, $i_1, \ldots, i_n \in \mathcal{S}$, and $d_1, \ldots, d_n \in \mathcal{D}$. Note that the first state i_1 is not necessary starting at time 1 associated with the first observation o_1 and the last state i_n is not necessary ending at time T associated with the last observation o_T. Detailed discussion about the censoring issues can be found in Section 2.2.1. Define the state transition probability from $(i, d') \to (j, d)$ for $i \neq j$ by

$$a_{(i,d')(j,d)} \triangleq P[S_{[t+1:t+d]} = j | S_{[t-d'+1:t]} = i],$$

subject to $\sum_{j \in \mathcal{S} \setminus \{i\}} \sum_{d \in \mathcal{D}} a_{(i,d')(j,d)} = 1$ with zero self-transition probabilities $a_{(i,d')(i,d)} = 0$, where $i, j \in \mathcal{S}$ and $d, d' \in \mathcal{D}$. From the definition we can see that the previous state i started at $t - d' + 1$ and ended at t, with duration d'. Then it transits to state j having duration d, according to the state transition probability $a_{(i,d')(j,d)}$. State j will start at $t + 1$ and end at $t + d$. This means both the state and the duration are dependent on both the previous state and its duration. While in state j, there will be d observations $o_{t+1:t+d}$ being emitted. Denote this emission probability by

$$b_{j,d}(o_{t+1:t+d}) \triangleq P[o_{t+1:t+d} | S_{[t+1:t+d]} = j]$$

which is assumed to be independent to time t. Let the initial distribution of the state be

$$\pi_{j,d} \triangleq P[S_{[t-d+1:t]} = j], \quad t \leqslant 0, \ d \in \mathcal{D}.$$

It represents the probability of the initial state and its duration before time $t = 1$ or before the first observation o_1 obtained. Then the set of the model parameters for the HSMM is defined by

$$\lambda \triangleq \left\{a_{(i,d')(j,d)}, b_{j,d}(v_{k_1:k_d}), \pi_{i,d}\right\},$$

where $i, j \in \mathcal{S}$, $d, d' \in \mathcal{D}$, and $v_{k_1:k_d}$ represents $v_{k_1} \ldots v_{k_d} \in \mathcal{V} \times \cdots \times \mathcal{V}$. This general HSMM is shown in Fig. 1.

The general HSMM is reduced to specific models of HSMM depending on the assumptions they made. For instance:

- If the state duration is assumed to be independent to the previous state, then the state transition probability can be further specified as $a_{(i,d')(j,d)} = a_{(i,d')j} p_j(d)$, where

$$a_{(i,d')j} \triangleq P[S_{[t+1} = j | S_{[t-d'+1:t]} = i] \tag{1}$$

is the transition probability from state i that has stayed for duration d' to state j that will start at $t+1$, and

$$p_j(d) \triangleq P[S_{t+1:t+d} = j | S_{[t+1} = j] \tag{2}$$

is the probability of duration d that state j takes. This is the model proposed by Marhasev et al. [119].

- If a state transition is assumed to be independent to the duration of the previous state, then the state transition probability becomes $a_{(i,d')(j,d)} = a_{i(j,d)}$, where

$$a_{i(j,d)} \triangleq P[S_{[t+1:t+d]} = j | S_t] = i] \tag{3}$$

is the transition probability that state i ended at t and transits to state j having duration d. This is the residential time HMM (see Section 3.3 for details). In this model, a state transition for $i \neq j$ is $(i,1) \to (j,\tau)$ and a self-transition is assumed to be $(i,\tau) \to (i,\tau-1)$ for $\tau > 1$, where τ represents the residential time of the state.

- If a self-transition is allowed and is assumed to be independent to the previous state, then the state transition probability becomes

$$a_{(i,d')(j,d)} = a_{(i,d')j} \prod_{\tau=1}^{d-1} a_{jj}(\tau)\big[1 - a_{jj}(d)\big],$$

where $a_{jj}(d) \triangleq P[S_{t+d+1} = j | S_{[t-d'+1:t]} = i, S_{[t+1:t+d} = j] = P[S_{t+d+1} = j | S_{[t+1:t+d} = j]$ is the self-transition probability when state j has stayed for d time units, and $1 - a_{jj}(d) = P[S_{t+d]} = j | S_{[t+1:t+d} = j]$ is the probability state j ends with duration d. This is the variable transition HMM (see Section 3.2 for details). In this model, a state transition is either $(i,d) \to (j,1)$ for $i \neq j$ or $(i,d) \to (i,d+1)$ for a self-transition.

- If a transition to the current state is independent to the duration of the previous state and the duration is only conditioned on the current state, then $a_{(i,d')(j,d)} = a_{ij} p_j(d)$, where $a_{ij} \triangleq P[S_{[t+1} = j | S_{t]} = i]$ is the transition probability from state i to j, with the self-transition probability $a_{ii} = 0$. This is the explicit duration HMM (see Section 3.1 for details).

Besides, the state duration distributions, $p_j(d)$, can be parametric or non-parametric. The detailed discussion on various duration distributions can be found in Section 4.1. Similarly, the observation distributions $b_{j,d}(v_{k_1:k_d})$ can be parametric or non-parametric, discrete or continuous, and dependent or independent on the state durations. It can also be a mixture of distributions. The detailed discussion on various observation distributions can be found in Section 4.2.

2.2. Inference

In this subsection we discuss the issues related to inference, including the forward–backward algorithm, calculation of probabilities and expectations, maximum a posteriori (MAP) estimate of states, maximum likelihood estimate (MLE) of state sequence, and constrained estimate of states.

2.2.1. The forward–backward algorithm

We define the forward variables for HSMM by:

$$\alpha_t(j,d) \triangleq P[S_{[t-d+1:t]} = j, o_{1:t} | \lambda]$$

and the backward variables by

$$\beta_t(j,d) \triangleq P[o_{t+1:T} | S_{[t-d+1:t]} = j, \lambda].$$

Similar to deriving the formulas for the HMM (see e.g., Rabiner [150], Ephraim and Merhav [57]), it is easy to obtain the forward–backward algorithm for a general HSMM:

$$\alpha_t(j,d) = \sum_{i \in \mathcal{S} \setminus \{j\}} \sum_{d' \in \mathcal{D}} \alpha_{t-d}\big(i,d'\big) \cdot a_{(i,d')(j,d)} \cdot b_{j,d}(o_{t-d+1:t}), \tag{4}$$

for $t > 0, d \in \mathcal{D}, j \in \mathcal{S}$, and

$$\beta_t(j,d) = \sum_{i \in \mathcal{S} \setminus \{j\}} \sum_{d' \in \mathcal{D}} a_{(j,d)(i,d')} \cdot b_{i,d'}(o_{t+1:t+d'}) \cdot \beta_{t+d'}\big(i,d'\big), \tag{5}$$

for $t < T$.

The initial conditions generally can have two different assumptions:

- *The general assumption*: assumes that the first state begins at or before observation o_1 and the last state ends at or after observation o_T. In this case, we can assume that the process starts at $-\infty$ and terminates at $+\infty$. The observations out of the sampling period $[1,T]$ can be any possible values, i.e., $b_{j,d}(\cdot) = 1$ for any $j \in \mathcal{S}, d \in \mathcal{D}$. Therefore, in the forward

formula (4) $b_{j,d}(o_{t-d+1:t})$ is replaced with the marginal distribution $b_{j,d}(o_{1:t})$ if $t-d+1 \leqslant 1$ and $t \geqslant 1$, and in the backward formula (5) $b_{i,d'}(o_{t+1:t+d'})$ is replaced with $b_{i,d'}(o_{t+1:T})$ if $t+1 \leqslant T$ and $t+d' \geqslant T$. We then have the initial conditions for the forward recursion formula given by (4) as follows:

$$\alpha_\tau(j,d) = P[S_{[\tau-d+1:\tau]} = j|\lambda] = \pi_{j,d}, \quad \tau \leqslant 0,\ d \in \mathcal{D},$$

where $\{\pi_{j,d}\}$ can be the equilibrium distribution of the underlying semi-Markov process. Because, for $t+d' \geqslant T$,

$$P[S_{[t+1:t+d']} = i, o_{t+1:T}|S_{[t-d+1:t]} = j, \lambda] = a_{(j,d)(i,d')} b_{i,d'}\left(o_{t+1}^T\right)$$

then from the backward recursion formula (5) we can see that $\beta_{t+d'}(i,d') = 1$, for $t+d' \geqslant T$. Therefore, the initial conditions for the backward recursion formula given by (5) are as follows:

$$\beta_\tau(i,d) = 1, \quad \tau \geqslant T,\ d \in \mathcal{D}.$$

If the model assumes that the first state begins at $t=1$ and the last state ends at or after observation o_T, it is a right-censored HSMM introduced by Guedon [70]. Because this is desirable for many applications, it is taken as a basis for an R package for analyzing HSMMs [23].

- *The simplifying assumption*: assumes that the first state begins at time 1 and the last state ends at time T. This is the most popular assumption one can find in the literature. In this case, the initial conditions for the forward recursion formula given by (4) are:

$$\alpha_0(j,d) = \pi_{j,d}, \quad d \in \mathcal{D},$$
$$\alpha_\tau(j,d) = 0, \quad \tau < 0,\ d \in \mathcal{D},$$

and the initial conditions for the backward recursion formula given by (5) are:

$$\beta_T(i,d) = 1, \quad d \in \mathcal{D},$$
$$\beta_\tau(i,d) = 0, \quad \tau > T,\ d \in \mathcal{D}.$$

Note that the initial distribution of states can be assumed as $\pi'_{j,d} \triangleq P[S_{[1:d]} = j|\lambda]$, which obviously equals to $\sum_{i,d'} \pi_{i,d'} a_{(i,d')(j,d)}$. Therefore, the initial conditions for the forward recursion formula can also be $\alpha_d(j,d) = \pi'_{j,d} b_{j,d}(o_{1:d})$, for $d \in \mathcal{D}$.

2.2.2. Probabilities and expectations

After the forward variables $\{\alpha_t(j,d)\}$ and the backward variables $\{\beta_t(j,d)\}$ are determined, all other probabilities of interest can be computed. For instance, the filtered probability that state j started at $t-d+1$ and ends at t, with duration d, given partial observed sequence $o_{1:t}$ can be determined by

$$P[S_{[t-d+1:t]} = j|o_{1:t}, \lambda] = \frac{\alpha_t(j,d)}{\sum_{j,d} \alpha_t(j,d)}$$

and the predicted probability that state j will start at $t+1$ and end at $t+d$, with duration d, given partial observed sequence $o_{1:t}$ by

$$P[S_{[t+1:t+d]} = j|o_{1:t}, \lambda] = \frac{\sum_{i \neq j, d'} \alpha_t(i,d') a_{(i,d')(j,d)}}{\sum_{i,d'} \alpha_t(i,d')}.$$

These readily yield the filtered probability of state j ending at t, $P[S_t = j|o_{1:t}, \lambda] = \sum_d P[S_{[t-d+1:t]} = j|o_{1:t}, \lambda]$, and the predicted probability of state j starting at $t+1$, $P[S_{[t+1} = j|o_{1:t}, \lambda] = \sum_d P[S_{[t+1:t+d]} = j|o_{1:t}, \lambda]$.

The posterior probabilities $P[S_t = j|o_{1:T}, \lambda]$, $P[S_t = i, S_{t+1} = j|o_{1:T}, \lambda]$ and $P[S_{[t-d+1:t]} = j|o_{1:T}, \lambda]$ for given entire observation sequence $o_{1:T}$ can be determined by the following equations

$$\eta_t(j,d) \triangleq P[S_{[t-d+1:t]} = j, o_{1:T}|\lambda] = \alpha_t(j,d)\beta_t(j,d), \tag{6}$$

$$\xi_t\left(i,d';j,d\right) \triangleq P[S_{[t-d'+1:t]} = i, S_{[t+1:t+d]} = j, o_{1:T}|\lambda] = \alpha_t\left(i,d'\right) a_{(i,d')(j,d)} b_{j,d}(o_{t+1:t+d}) \beta_{t+d}(j,d),$$

$$\xi_t(i,j) \triangleq P[S_t = i, S_{[t+1} = j, o_{1:T}|\lambda] = \sum_{d' \in \mathcal{D}} \sum_{d \in \mathcal{D}} \xi_t\left(i,d';j,d\right), \tag{7}$$

$$\gamma_t(j) \triangleq P[S_t = j, o_{1:T}|\lambda] = \sum_{\tau \geqslant t} \sum_{d=\tau-t+1}^{D} \eta_\tau(j,d) \tag{8}$$

and

$$P[o_{1:T}|\lambda] = \sum_{j\in\mathcal{S}} P[S_t = j, o_{1:T}|\lambda] = \sum_{j\in\mathcal{S}} \gamma_t(j),$$

for $d, d' \in \mathcal{D}$, $j \in \mathcal{S}$, $i \in \mathcal{S}\backslash\{j\}$ and $t = 1, \ldots, T$, where $\eta_t(j,d)/P[o_{1:T}|\lambda]$ represents the probability of being in state j having duration d by time t given the model and the observation sequence; $\xi_t(i,d';j,d)/P[o_{1:T}|\lambda]$ the probability of transition at time t from state i occurred with duration d' to state j having duration d given the model and the observation sequence; $\xi_t(i,j)/P[o_{1:T}|\lambda]$ the probability of transition at time t from state i to state j given the model and the observation sequence; $\gamma_t(j)/P[o_{1:T}|\lambda]$ the probability of state j at time t given the model and the observation sequence; and $P[o_{1:T}|\lambda]$ the probability that the observed sequence $o_{1:T}$ is generated by the model λ. Obviously, the conditional factor $P[o_{1:T}|\lambda]$ is common for all the posterior probabilities, which will be eliminated when the posterior probabilities are used in parameter estimation. Therefore, it is often omitted for simplicity in the literature. Similarly, in the rest of this paper, we sometimes will not explicitly mention this conditional factor in calculating the posterior probabilities by $\eta_t(j,d)$, $\xi_t(i,d';j,d)$, $\xi_t(i,j)$, and $\gamma_t(j)$.

In considering the following identity

$$P[S_{t:t+1} = j, o_{1:T}|\lambda] = P[S_t = j, o_{1:T}|\lambda] - P[S_{t]} = j, o_{1:T}|\lambda],$$
$$P[S_{t:t+1} = j, o_{1:T}|\lambda] = P[S_{t+1} = j, o_{1:T}|\lambda] - P[S_{[t+1} = j, o_{1:T}|\lambda]$$

we have a recursive formula for calculating $\gamma_t(j)$:

$$\gamma_t(j) = \gamma_{t+1}(j) + P[S_{t]} = j, o_{1:T}|\lambda] - P[S_{[t+1} = j, o_{1:T}|\lambda] = \gamma_{t+1}(j) + \sum_{i\in\mathcal{S}\backslash\{j\}} \big[\xi_t(j,i) - \xi_t(i,j)\big]. \tag{9}$$

Denote $P[o_{1:T}|\lambda]$ by L in the following expressions. Then using the forward and backward variables, one can compute various expectations [60]:

(a) The expected number of times state i ends before t: $\frac{1}{L}\sum_{t'\leqslant t}\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_{t'}(i,j)$; The expected number of times state i starts at t or before: $\frac{1}{L}\sum_{t'\leqslant t-1}\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_{t'}(j,i)$.
(b) Expected total duration spent in state i: $\frac{1}{L}\sum_t \gamma_t(i)$.
(c) Expected number of times that state i occurred with observation $o_t = v_k$: $\frac{1}{L}\sum_t \gamma_t(i)\mathbb{I}(o_t = v_k)$, where the indicator function $\mathbb{I}(x) = 1$ if x is true and zero otherwise.
(d) Estimated average observable values of state i: $\frac{\sum_t \gamma_t(i)o_t}{\sum_t \gamma_t(i)}$.
(e) Probability that state i was the first state: $\frac{1}{L}\gamma_1(i)$.
(f) Expected total number of times state i commenced: $\frac{1}{L}\sum_t\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_t(j,i)$ or terminated: $\frac{1}{L}\sum_t\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_t(i,j)$. For the simplifying assumption for the boundary conditions described in the last subsection, we have $\sum_{t=0}^{T-1}\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_t(j,i) = \sum_{t=1}^{T}\sum_{j\in\mathcal{S}\backslash\{i\}}\xi_t(i,j)$.
(g) Estimated average duration of state i: $\frac{\sum_t\sum_d \eta_t(i,d)d}{\sum_t\sum_d \eta_t(i,d)}$.

2.2.3. MAP and MLE estimate of states

The maximum a posteriori (MAP) estimate of state S_t given a specific observation sequence $o_{1:T}$ can be obtained [60] by maximizing $\gamma_t(j)$ given by (8), i.e.,

$$\hat{s}_t = \arg\max_{i\in\mathcal{S}}\big\{\gamma_t(i)\big\}.$$

If we choose $\eta_t(i,d)$ of (6), instead of $\gamma_t(i)$, as the MAP criterion, we obtain the joint MAP estimate of the state that ends at time t and the duration of this state, when a specific sequence $o_{1:T}$ is observed:

$$(\hat{s}_t, \hat{d}_t) = \arg\max_{(i,d)} \eta_t(i,d). \tag{10}$$

Viterbi algorithms are the most popular dynamic programming algorithms for the maximum likelihood estimate (MLE) of state sequence of HMMs. There exist the similar algorithms for the HSMM [115,151,35,26]. Define the forward variable for the extended Viterbi algorithm by

$$\delta_t(j,d) \triangleq \max_{s_{1:t-d}} P[s_{1:t-d}, S_{[t-d+1:t]} = j, o_{1:t}|\lambda] = \max_{i\in\mathcal{S}\backslash\{j\},\ d'\in\mathcal{D}}\big\{\delta_{t-d}\big(i,d'\big)a_{(i,d')(j,d)}b_{j,d}(o_{t-d+1:t})\big\}, \tag{11}$$

for $1 \leqslant t \leqslant T$, $j \in \mathcal{S}$, $d \in \mathcal{D}$. $\delta_t(j,d)$ represents the maximum likelihood that the partial state sequence ends at t in state j of duration d. Record the previous state that $\delta_t(j,d)$ selects by $\Psi(t,j,d) \triangleq (t-d, i^*, d^*)$, where i^* is the previous state survived, d^* its duration, and $(t-d)$ its ending time. $\Psi(t,j,d)$ is determined by letting

$$(i^*, d^*) = \arg \max_{i \in \mathcal{S} \setminus \{j\},\ d' \in \mathcal{D}} \{\delta_{t-d}(i, d') a_{(i,d')(j,d)} b_{j,d}(o_{t-d+1:t})\}.$$

Now the maximum likelihood state sequence can be determined by finding the last state that maximizes the likelihood. For the general assumption of the boundary conditions on page 5, the last ML state is

$$(t_1, j_1^*, d_1^*) = \arg \max_{\substack{t \geqslant T \\ i \in \mathcal{S} \\ d \geqslant t-T+1,\ d \in \mathcal{D}}} \delta_t(i, d),$$

or, for the simplifying assumption of the boundary conditions, $t_1 = T$ and

$$(j_1^*, d_1^*) = \arg \max_{\substack{i \in \mathcal{S} \\ d \in \mathcal{D}}} \delta_T(i, d).$$

Trace back the state sequence by letting

$$(t_2, j_2^*, d_2^*) = \Psi(t_1, j_1^*, d_1^*),$$
$$\dots$$
$$(t_n, j_n^*, d_n^*) = \Psi(t_{n-1}, j_{n-1}^*, d_{n-1}^*),$$

until the first state S_1 is determined, where $S_1 = j_n^*$ and $(j_n^*, d_n^*), \dots, (j_1^*, d_1^*)$ is the maximum likelihood state sequence.

If the state duration density function is log-convex parametric, which is fulfilled by the commonly used parametric functions, Bonafonte et al. [17] empirically showed that the computational complexity can be reduced to about 3.2 times of the conventional HMM. If the model is a left–right HSMM or the particular state sequence, $i_1, \dots, i_n$, is given, then only the optimal segmentation of state durations needs to be determined. This is accomplished by simply rewriting (11) as [109,110]

$$\delta_t(i_m, d) = \max_{d' \in \mathcal{D}} \{\delta_{t-d}(i_{m-1}, d') a_{(i_{m-1},d')(i_m,d)} b_{i_m,d}(o_{t-d+1:t})\},$$

for $1 \leqslant m \leqslant n$, $1 \leqslant t \leqslant T$, $d \in \mathcal{D}$.

2.2.4. Constrained estimate of states

As discussed in the previous subsections, the posterior probabilities $P[S_t = j|o_{1:T}, \lambda]$, $P[S_t = i, S_{t+1} = j|o_{1:T}, \lambda]$ and $P[S_{[t-d+1:t]} = j|o_{1:T}, \lambda]$ for given entire observation sequence $o_{1:T}$ are determined by the forward–backward algorithm, and are used for the computation of various expectations and the MAP estimation of states. These posterior probabilities can be interpreted as the probabilities that the path taken (a random variable) passes through the constraint states for the given observation sequence $o_{1:T}$. For example, $P[S_{[t-d+1:t]} = j, o_{1:T}|\lambda] = \alpha_t(j, d)\beta_t(j, d)$ counts for all the paths that pass through the constraint state j during the constraint period of $t-d+1$ to t, where $\alpha_t(j, d)$ is given by (4) and $\beta_t(j, d)$ by (5). This is useful for the confidence calculation in the state estimation. The confidence can be simply defined as

$$\max_{j,d} \alpha_t(j, d)\beta_t(j, d) \Big/ \sum_{j', d'} \alpha_t(j', d')\beta_t(j', d'),$$

where $\arg\max_{j,d} \alpha_t(j, d)\beta_t(j, d)$ is used for the MAP estimate of the state as given by (10). Calculating the confidence over every t, one can find out in practice when the errors are most likely to occur in the state estimation.

Now we compute the posterior probability $P[s_{t+1:t+k}|o_{1:T}, \lambda]$ corresponding to a segment of k observations. It is expected useful in some applications. For example, this probability can be used to estimate the confidence of an individual field of words for information extraction [41]. It can also be used for computing some expectations, which can be used for estimating the states corresponding to the segment of k observations.

If one assumes that the first state of the subsequence $s_{t+1:t+k}$ starts at $t+1$ and the last state ends at $t+k$, i.e., $s_{t+1:t+k} = (j_1, d_1) \dots (j_n, d_n)$, s.t. $d_1 + \cdots + d_n = k$, it is easy to compute the posterior probability by

$$P[s_{t+1:t+k}, o_{1:T}|\lambda] = \alpha_{t+d_1}(j_1, d_1) \left[\prod_{m=2}^{n} a_{(j_{m-1},d_{m-1})(j_m,d_m)} b_{j_m,d_m}(o_{\tau_{m-1}+1:\tau_m}) \right] \beta_{t+k}(j_n, d_n),$$

where $\tau_m = t + d_1 + \cdots + d_m$. When $n = 1$ the equation is reduced to (6). If we release the condition that the first state of the subsequence starts at $t+1$ or the one that the last state of the subsequence ends at $t+k$, the computation can be done by allowing the first state duration $d \geqslant d_1$ and the last state duration $d' \geqslant d_n$.

A simple way for computing the posterior probability of a segment is modifying the forward–backward algorithm to conform the constraints. Similar to the constrained forward–backward algorithm for a CRF (conditional random field) proposed by Culotta and McCallum [41], the forward–backward formulas given by (4) and (5) for the HSMM can be modified. Let $\alpha_{t'}(j, d) = 0$ and $\beta_{t'}(j, d) = 0$ when $t+1 \leqslant t' \leqslant t+k$ and $j \neq s_{t'}$, where $s_{t'} \in s_{t+1:t+k}$ is a constraint that each path must pass through and $s_{t+1:t+k}$ is the constraint subpath. If we further constrain that the first state of the constraint subpath starts at

$t+1$, we must let $\alpha_{t'}(j,d)=0$ and $\beta_{t'}(j,d)=0$ when $t+1\leqslant t'\leqslant t+k$ and $t'-d+1<t+1$. Similarly, if we constrain that the last state of the constraint subpath ends at $t+k$, we let $\alpha_{t'}(j,d)=0$ and $\beta_{t'}(j,d)=0$ when $t+1\leqslant t'-d+1\leqslant t+k$ and $t'>t+k$. Using this modified forward recursion, we obtain $\alpha'_T(j,d)$. Let $L'=P[s_{t+1:t+k},o_{1:T}|\lambda]=\sum_{j\in\mathcal{S}}\sum_{d\in\mathcal{D}}\alpha'_T(j,d)$ be the constrained lattice, the set of all paths that conform to the constraints $s_{t+1:t+k}$. Then the posterior probability is yielded by L'/L, where $L=P[o_{1:T}|\lambda]$. Obviously, the probabilities given by (6) to (8) can be computed using the modified forward recursion as well.

2.3. Estimation

In the preceding problems, such as the forward–backward algorithm, MAP and ML estimation of states, we assumed that the set of model parameters λ is given. If λ is unknown, we need to learn about λ from the observations $o_{1:T}$: λ is initially estimated and then re-estimated so that the likelihood function $L(\lambda)=P[o_{1:T}|\lambda]$ increases and converges to its maximum value. If the system is slowly varying (i.e., non-stationary), the model parameters λ may need to be updated adaptively. Such training and updating process is referred to as *parameter re-estimation*.

2.3.1. Parameter estimate of HSMM

For the parameter estimation/re-estimation problem, there is no known analytical method to find the λ that maximizes the likelihood function. Thus, some iterative procedure must be employed.

The model parameters λ can be re-estimated using the expectations. For instance,

1) the initial distribution $\hat{\pi}_{j,d}$ can be updated by $\eta_t(j,d)/\sum_{j,d}\eta_t(j,d)$ for $t\leqslant 0$,
2) the transition probabilities $\hat{a}_{(i,d')(j,d)}$ by $\sum_t\xi_t(i,d';j,d)/\sum_{j\neq i,d}\sum_t\xi_t(i,d';j,d)$, and
3) the observation probabilities $\hat{b}_{j,d}(v_{k_1:k_d})$ by $\sum_t[\eta_t(j,d)\cdot\mathbb{I}(o_{t+1:t+d}=v_{k_1:k_d})]/\sum_t\eta_t(j,d)$, where $\mathbb{I}(o_{t+1:t+d}=v_{k_1:k_d})=1$ if $o_{t+1}=v_{k_1},\ldots,o_{t+d}=v_{k_d}$ and zero otherwise.

Except these parameters for the general HSMM, parameters for other HSMMs can be estimated as well, such as

i) the transition probabilities $\hat{a}_{ij}$ by $\sum_t\xi_t(i,j)/\sum_{j\neq i}\sum_t\xi_t(i,j)$,
ii) the duration probabilities $\hat{p}_j(d)$ of state j by $\sum_t\eta_t(j,d)/\sum_d\sum_t\eta_t(j,d)$,
iii) the observation probabilities $\hat{b}_j(v_k)$ by $\sum_t[\gamma_t(j)\cdot\mathbb{I}(o_t=v_k)]/\sum_t\gamma_t(j)$, and
iv) the initial distribution $\hat{\pi}_j$ by $\gamma_0(j)/\sum_j\gamma_0(j)$.

Those probability mass function or probability density function satisfy: $\sum_{j,d}\hat{\pi}_{j,d}=1$, $\sum_d\hat{p}_j(d)=1$, $\sum_{j\neq i,d}\hat{a}_{(i,d')(j,d)}=1$, $\sum_{j\neq i}\hat{a}_{ij}=1$, $\sum_{v_{k_1},\ldots,v_{k_d}}b_{j,d}(v_{k_1:k_d})=1$, and $\sum_{v_k}b_j(v_k)=1$.

The re-estimation procedure:

a) Assume an initial model parameter set λ_0;
b) For given model parameter set λ_k, use the forward–backward formulas (4) and (5) to compute the forward and backward variables $\{\alpha_t(j,d)\}$ and $\{\beta_t(j,d)\}$. Then use the forward and backward variables to compute the related probabilities $\eta_t(j,d)$, $\xi_t(i,d';j,d)$, $\xi_t(i,j)$ and $\gamma_t(j)$ by (6) through (9). Finally re-estimate the model parameters to get $\hat{\lambda}_{k+1}$;
c) Let $\lambda_{k+1}=\hat{\lambda}_{k+1}$, $k++$, and go back to step b);
d) Repeat b) and c) until the likelihood $L(\lambda_k)=P[o_{1:T}|\lambda_k]$ converges to a fixed point.

2.3.2. Order estimate of HSMM

In the re-estimation algorithms discussed above, the number of hidden states, M, the maximum length of state duration, D, the number of observable values, K, and the length of the observation sequence, T, are usually assumed known in the context of applications. However, the learning issues when the order of an HSMM is unknown is sometimes particularly important in practice. A detailed discussion on the order estimate of HMMs can be found in Ephraim and Merhav [57, Section VIII], and the issues of overfitting and model selection in Ghahramani [66, Section 7]. However, the order estimate of HSMMs is somewhat different from that of HMMs, because HSMMs have variable durations. Therefore, for an HSMM we must estimate both the number of states, M, and the maximum length of state durations, D.

In fact, some special HSMMs can be described by a dynamic Bayesian network (DBN) using a directed graphical model. For simplicity, one usually assumes the observations are conditionally independent, i.e.,

$$b_{j,d}(o_{t+1:t+d})=P[o_{t+1:t+d}|S_{[t+1:t+d]}=j]=\prod_{\tau=t+1}^{t+d}b_j(o_\tau), \tag{12}$$

where $b_j(v_k)\triangleq P[o_t=v_k|S_t=j]$. To easily identify when a segment of states starts, one usually further assumes a state transition is independent to the previous state duration. This is just the assumption made for the explicit duration HMM and

the residential time HMM, as described on page 3. The conditional probability distribution (CPD) function for the explicit duration HMM is [126]:

$$P[S_t = j|S_{t-1} = i, R_{t-1} = \tau] = \begin{cases} a_{ij} & \text{if } \tau = 1 \text{ (transition)}, \\ \mathbb{I}(i = j) & \text{if } \tau > 1 \text{ (decrement)}, \end{cases}$$

$$P\big[R_t = \tau'|S_t = j, R_{t-1} = \tau\big] = \begin{cases} p_j(\tau') & \text{if } \tau = 1 \text{ (transition)}, \\ \mathbb{I}(\tau' = \tau - 1) & \text{if } \tau > 1 \text{ (decrement)}, \end{cases}$$

and for the residential time HMM

$$P\big[Q_t = \big(j, \tau'\big)|Q_{t-1} = (i, \tau)\big] = \begin{cases} a_{i(j,\tau')} & \text{if } \tau = 1 \text{ (transition)}, \\ \mathbb{I}(\tau' = \tau - 1) & \text{if } \tau > 1 \text{ (decrement)} \end{cases}$$

where R_t is the remaining duration of state S_t, $Q_t = (j, \tau)$ represents $S_t = j$ and $R_t = \tau$, and the self-transition probability $a_{i(i,\tau')} = 0$ as defined in the beginning of this section. The indicator function $\mathbb{I}(x) = 1$ if x is true and zero otherwise. Several DBNs for HSMMs are presented in Murphy [126].

As discussed in Ghahramani [66], a Bayesian approach to learning treats all unknown quantities as random variables. These unknown quantities comprise the number of states, the parameters, and the hidden states. By integrating over both the parameters and the hidden states, the unknown quantities can be estimated. For the explicit duration HMM, the number of states, M, and the maximum length of state durations, D, can be determined after S_t and R_t, for $t = 1$ to T, are estimated. For the residential time HMM, after the set of hidden states that Q_t can take and the transition probabilities are estimated, the values of M and D can be determined by checking the transition probabilities of $P[q_t|q_{t-1}]$, where q_t is the estimated hidden state of Q_t. Obviously from the CPDs, $P[q_t|q_{t-1}] = 1$ represents a self-transition, and $P[q_t|q_{t-1}] < 1$ a state transition. Therefore, by counting the number of consecutive self-transitions we can determine the maximum duration of states, D, and then determine the number of HSMM states, M.

Sometimes, one uses a simple method to find out the order of an HSMM by trying various values of M and D. Denote $\lambda^{(M,D)}$ as the model parameter with assumed order M and D. The maximum likelihood estimate of $\lambda^{(M,D)}$ is

$$\hat{\lambda}^{(M,D)} = \arg\max_{\lambda^{(M,D)}} \log P\big[o_{1:T}|\lambda^{(M,D)}\big],$$

which can be determined using the re-estimation algorithms discussed in this subsection for given M and D. Then the order estimators given in Ephraim and Merhav [57] can be used in the selection of the model order. For instance, the order estimator proposed by Finesso [61] can be used as the objective function for the selection of the model order:

$$(\hat{M}, \hat{D}) = \min\left\{\arg\min_{M,D\geqslant 1}\left\{-\frac{1}{T}\log P\big[o_{1:T}|\hat{\lambda}^{(M\cdot D)}\big] + 2c_{MD}^2\frac{\log T}{T}\right\}\right\},$$

where $c_{MD} = MD(MD + K - 2)$ is a penalty term that favors simpler models over more complex models, T the total number of observations, and K the total number of values that an observation can take.

In fact, we can alternatively use an undirected graphical model to describe the HSMMs and to learn the unknown quantities, such as semi-Markov conditional random fields (semi-CRFs) introduced by Sarawagi and Cohen [161]. In this model, the assumption that the observations are conditional independent is not needed.

2.4. EM algorithm and online estimation

Using the theory associated with the well-known EM (expectation–maximization) algorithm [42], it can be proved that the re-estimation procedure for the HSMMs increases the likelihood function of the model parameters. However, these algorithms require the backward procedures and the iterative calculations, and so are not practical for online learning. A few of online algorithms for HSMM have been developed in the literature, including an adaptive EM algorithm by Ford et al. [62], an online algorithm based on recursive prediction error (RPE) techniques by Azimi et al. [9,11], and recently a recursive maximum likelihood estimation (RMLE) algorithm by Squire and Levinson [168].

2.4.1. Re-estimation vs. EM algorithm

Let λ represent the complete set of the model parameters to be estimated in the re-estimation procedure. The purpose is to find maximum likelihood estimates of the model parameter set λ such that the likelihood function $P[o_{1:T}|\lambda]$ is maximized for given $o_{1:T}$.

Let us consider two *a posteriori probabilities* (APPs) of the state sequence variable $s_{1:T} = s_1, \ldots, s_T$, given an instance of the observation sequence $o_{1:T}$; one under model parameter λ and the other under its improved version λ'. Denote $L(\lambda) \triangleq P[o_{1:T}|\lambda]$ as the *likelihood function* of the model parameter λ. Following the discussion given in Ferguson [60], an auxiliary function is defined as a conditional *expectation* [121]

$$Q\left(\lambda, \lambda'\right) \triangleq E\big[\log P\big[s_{1:T}, o_{1:T}|\lambda'\big]|o_{1:T}, \lambda\big] = \sum_{s_{1:T}\in\mathcal{S}^T} P[s_{1:T}, o_{1:T}|\lambda]\log P\big[s_{1:T}, o_{1:T}|\lambda'\big].$$

Because

$$\frac{Q(\lambda,\lambda') - Q(\lambda,\lambda)}{L(\lambda)} \leqslant \log\frac{L(\lambda')}{L(\lambda)},$$

if $Q(\lambda,\lambda') > Q(\lambda,\lambda)$, then $L(\lambda') > L(\lambda)$. This implies that the best choice of λ' (for given λ) is found by solving the *maximization* problem of $Q(\lambda,\lambda')$. Therefore, by iterating the *expectation step* (E-step) and the *maximization step* (M-step), as in the *expectation–maximization* (EM) algorithm, an optimum λ' can be found. In fact, EM algorithm holds in a more general setting than HSMMs.

Instead of using the forward–backward algorithms for the E-step and the re-estimate formulas for the M-step, Krishnamurthy and Moore [92] use the EM algorithm to directly re-estimate the model parameters. In this case, the unknowns are the model parameters λ as well as the state sequence $s_{1:T}$. When the order of the model is large, the computational amount involved in the re-estimate is huge. To reduce this computational amount, MCMC sampling is used for approximate re-estimate [46,47]. MCMC sampling is a general methodology for generating samples from a desired probability distribution function and the obtained samples are used for various types of inference. Therefore, MCMC sampling can also be used in the estimation of the state sequence and the model parameters. MCMC sampling draws samples of the unknowns from their posteriors so that the posteriors can be approximated using these samples. The prior distributions of all the unknowns are required to specify before applying the MCMC sampling methods. The emission probabilities and the durations of various states are often modelled using some parametric distributions.

2.4.2. Forward algorithm for online estimation

Define the maximum log-likelihood of the observation sequence $o_{1:t}$ as [168]

$$L_t(\lambda) \triangleq \max_n \max_{\substack{d_{1:n} \\ d_1+\cdots+d_n=t}} \log P[o_{1:t}, d_{1:n}|\lambda] = \max_n L_t^{(n)}(\lambda)$$

where $d_{1:n}$ denotes $d_1,\ldots,d_n$ with $d_k \in \mathcal{D}$, and

$$L_t^{(n)}(\lambda) \triangleq \max_{\substack{d_{1:n} \\ d_1+\cdots+d_n=t}} \log P[o_{1:t}, d_{1:n}|n,\lambda].$$

By maximizing $L_t(\lambda)$ with respect to $\lambda = \{a_{ij}, b_i(v_k), p_i(d), \pi_i\}$, the set of model parameters $\hat{\lambda}$ can be re-estimated [168].

Signal modelling using HSMM proposed by Azimi et al. [9–11] is a different online estimation algorithm. The state S_t of the signal at time t is assumed to take its values from the set $\{\boldsymbol{e}_1,\ldots,\boldsymbol{e}_M\}$ where $\boldsymbol{e}_i$ is an $M\times 1$ vector with unity as the ith element and zeros elsewhere. In this case, the state at time t is appropriate to be denoted using an $M\times 1$ vector, i.e., $\boldsymbol{s}_t \in \{\boldsymbol{e}_1,\ldots,\boldsymbol{e}_M\}$.

Define the log-likelihood of the observations up to time t given λ [9,11]:

$$l_t(\lambda) \triangleq \log P[o_{1:t}|\lambda] = \sum_{\tau=1}^{t} \log P[o_\tau|o_{1:\tau-1},\lambda] = \sum_{\tau=1}^{t} x_\tau(\lambda),$$

where $x_{t+1}(\lambda) \triangleq \log P[o_{t+1}|o_{1:t},\lambda]$. Denote the estimate of the Hessian matrix by $R_t \triangleq \frac{\partial^2}{\partial\lambda^2} l_t(\lambda)$, and the gradient of $x_t(\lambda)$ with respect to λ by $\psi_t \triangleq \frac{\partial}{\partial\lambda} x_t(\lambda)$, then the model parameters can be updated using $\hat{\lambda} = \lambda + \varepsilon_{t+1}\cdot R_{t+1}^{-1}\cdot\psi_{t+1}$, where ε_{t+1} is a step size.

A sequential online learning of HMM state duration using quasi-Bayes (QB) estimate was presented in Chien and Huang [36,37], in which the Gaussian, Poisson, and gamma distributions were investigated to characterize the duration models.

2.5. Implementation

It is well known that the joint probabilities associated with observation sequence often decay exponentially as the sequence length increases. The implementation of the forward–backward algorithms by programming in a real computer would suffer a severe underflow problem. This subsection considers the issues related to the practical implementation of the algorithms.

A general heuristic method to solve the underflow problem is to re-scale the forward–backward probabilities by multiplying a large factor whenever an underflow is likely to occur [106,40]. However, the scale factors cannot guarantee the backward variables being bounded or immunizing from the underflow problem, as pointed out by Murphy [126].

If the forward–backward algorithm is implemented in the logarithmic domain, like the MAP and Viterbi algorithms used for turbo-decoding in digital communications, then the multiplications become additions and the scaling becomes unnecessary [20]. In fact, the logarithmic form of the extended Viterbi algorithm can be considered as an approximation to that of the forward–backward algorithm.

The notion of posterior probabilities is used to overcome the underflow problem involved in the recursive calculation of the joint probabilities of observations [68,202]. This is similar to the standard HMM. The HMM's forward–backward

algorithms can automatically avoid the underflow problem by replacing the joint probabilities with conditional ones [45,6]. The refined forward–backward algorithm for the HSMM becomes robust against the underflow problem, without increasing the computational complexity.

In notion of posterior probabilities, the forward variables are redefined [202] using the *predicted* probabilities by

$$\bar{\alpha}_t(j,d) \triangleq P[S_{[t-d+1:t]} = j | o_{1:t-d}, \lambda],$$

and the backward variables by

$$\bar{\beta}_t(j,d) \triangleq \frac{P[o_{t-d+1:T} | S_{[t-d+1:t]} = j, \lambda]}{P[o_{t-d+1:T} | o_{1:t-d}, \lambda]}.$$

Denote

$$\bar{b}_{j,d}(o_{t-d+1:t}) \triangleq \frac{b_{j,d}(o_{t-d+1:t})}{P[o_{t-d+1:t} | o_{1:t-d}, \lambda]}.$$

We have $\alpha_t(j,d) = P[o_{1:t-d}|\lambda]\bar{\alpha}_t(j,d)b_{j,d}(o_{t-d+1:t})$ and $\bar{\beta}_t(j,d)P[o_{t-d+1:T}|o_{1:t-d},\lambda] = b_{j,d}(o_{t-d+1:t})\beta_t(j,d)$. Then the forward–backward formulas (4) and (5) become

$$\bar{\alpha}_t(j,d) = \sum_{i \in \mathcal{S}\setminus\{j\}} \sum_{d' \in \mathcal{D}} \bar{\alpha}_{t-d}(i,d') \bar{b}_{i,d'}\big(o_{t-d-d'+1}^{t-d}\big) a_{(i,d')(j,d)}, \tag{13}$$

and

$$\bar{\beta}_t(j,d) = \bar{b}_{j,d}(o_{t-d+1:t}) \sum_{i \in \mathcal{S}\setminus\{j\}} \sum_{d' \in \mathcal{D}} a_{(j,d)(i,d')} \bar{\beta}_{t+d'}(i,d'). \tag{14}$$

The probability $P[o_{1:t}|\lambda]$ can be determined by

$$P[o_{1:t}|\lambda] = \sum_{j \in \mathcal{S}} \sum_{d \in \mathcal{D}} P[S_{[t-d+1:t]} = j, o_{1:t}|\lambda] = \sum_{j \in \mathcal{S}} \sum_{d \in \mathcal{D}} P[o_{1:t-d}|\lambda]\bar{\alpha}_t(j,d)b_{j,d}(o_{t-d+1:t}),$$

and $P[o_{t-d+1:t}|o_{1:t-d},\lambda]$ by $P[o_{1:t}|\lambda]/P[o_{1:t-d}|\lambda]$.

After the forward–backward variables are determined, the probabilities defined by (6) through (9) can be yielded, such as

$$\frac{\eta_t(j,d)}{P[o_{1:T}|\lambda]} = \bar{\alpha}_t(j,d)\bar{\beta}_t(j,d)$$

and

$$\frac{\xi_t(i,d';j,d)}{P[o_{1:T}|\lambda]} = \bar{\alpha}_t(i,d')b_{i,d'}(o_{t-d'+1:t})a_{(i,d')(j,d)}\bar{\beta}_{t+d}(j,d).$$

If we denote $\bar{\alpha}'_t(j,d) = \bar{\alpha}_t(j,d)b_{j,d}(o_{t-d+1:t})$ and $\bar{\beta}'_{t+1}(j,d) = \bar{\beta}_{t+d}(j,d)$, then the forward recursion (13) becomes

$$\bar{\alpha}'_t(j,d) = \bar{b}_{j,d}(o_{t-d+1:t}) \sum_{i \in \mathcal{S}\setminus\{j\}} \sum_{d' \in \mathcal{D}} \bar{\alpha}'_{t-d}(i,d') a_{(i,d')(j,d)},$$

and the backward recursion (14) becomes

$$\bar{\beta}'_t(j,d) = \bar{b}_{j,d}(o_{t:t+d-1}) \sum_{i \in \mathcal{S}\setminus\{j\}} \sum_{d' \in \mathcal{D}} \bar{\beta}'_{t+d}(i,d') a_{(j,d)(i,d')}.$$

That is, the backward recursion is symmetric to the forward one in the time reversed form. This can potentially reduce the requirement for the silicon area on a chip if the backward logic module uses the forward one. A symmetric forward–backward algorithm for the residential time model was introduced by Yu and Kobayashi [199].

3. Conventional models

As pointed out in the last section on page 3, the general HSMM is reduced to the conventional models when specific assumptions are made on the dependency of states and durations. This section overviews three conventional models, including the explicit duration HMM, variable transition HMM and residential time HMM. These models have fewer parameters and lower computational complexity than the general model. They are the HSMMs often found in the literature.

3.1. Explicit duration HMM

Ferguson [60] was the first to consider the HSMM, which he called an "HMM with variable duration." Since then a number of studies have been reported on the subject. See for example, Mitchell and Jamieson [123], Yu and Kobayashi [199,202] and references therein.

The explicit duration hidden Markov model assumes that a state transition is independent to the duration of the previous state, i.e., $a_{(i,d')(j,d)} = a_{i(j,d)}$, without self-transitions, i.e., $a_{i(i,d)} = 0$. The state duration is assumed to be dependent on the current state and independent to the previous state. That is, state j will last for duration variable d according to the conditional probability $p_j(d)$, as defined in (2). Therefore, we have $a_{(i,d')(j,d)} = a_{ij} p_j(d)$ with $a_{ii} = 0$, for $i, j \in \mathcal{S}, d \in \mathcal{D}$, where $a_{ij} \triangleq P[S_{[t} = j | S_{t-1]} = i]$ is the state transition probability from state i to state j. It also assumes the "conditional independence" of outputs as defined in (12). Due to all those independent assumptions, the explicit duration HMM is one of the most simple models among all the HSMMs. Therefore, it is the most popular HSMM in applications.

Replace $a_{(i,d')(j,d)}$ with $a_{ij} p_j(d)$, $b_{j,d}(o_{t+1:t+d})$ with $\prod_{\tau=t+1}^{t+d} b_j(o_\tau)$, $\beta_t(j,d)$ with $\beta_t(j) \triangleq P[o_{t+1:T} | S_{t]} = i, \lambda]$ in the general forward–backward formulas (4) and (5), and define $\alpha_t(j) \triangleq P[S_{t]} = j, o_{1:t} | \lambda] = \sum_{d \in \mathcal{D}} \alpha_t(j,d)$. Then we readily obtain the forward–backward formulas for the explicit duration HMM [60]:

$$\alpha_t(j) = \sum_{d \in \mathcal{D}} \alpha^*_{t-d}(j) p_j(d) u_t(j,d), \tag{15}$$

$$\alpha^*_t(j) \triangleq P[S_{[t+1} = j, o_{1:t} | \lambda] = \sum_{i \in \mathcal{S} \setminus \{j\}} \alpha_t(i) a_{ij}, \tag{16}$$

for $j \in \mathcal{S}$, $t = 1, \ldots, T$, and

$$\beta^*_t(j) \triangleq P[o_{t+1:T} | S_{[t+1} = j, \lambda] = \sum_{d \in \mathcal{D}} p_j(d) u_{t+d}(j,d) \beta_{t+d}(j), \tag{17}$$

$$\beta_t(j) = \sum_{i \in \mathcal{S} \setminus \{j\}} a_{ji} \beta^*_t(i), \tag{18}$$

for $j \in \mathcal{S}$, $t = T-1, \ldots, 0$, where

$$u_t(j,d) \triangleq \prod_{\tau=t-d+1}^{t} b_j(o_\tau). \tag{19}$$

The forward variable $\alpha_t(j)$ represents the joint probability that state j ends at t and the partial observation sequence is $o_{1:t}$, and $\alpha^*_t(j)$ the joint probability that state j starts at $t+1$ and the partial observation sequence is $o_{1:t}$. The backward variable $\beta_t(j)$ represents the conditional probability that given state i ending at t, the future observation sequence is $o_{t+1:T}$, and $\beta^*_t(j)$ the conditional probability that given state j starting at $t+1$ the future observation sequence is $o_{t+1:T}$.

The boundary conditions use the simplifying assumption described on page 5, i.e., $\alpha_0(i) = \pi_i$ and $\alpha_\tau(i) = 0$ for $\tau < 0$, and $\beta_T(i) = 1$ and $\beta_\tau(i) = 0$ for all $\tau > T$, $i \in \mathcal{S}$, where π_i is the initial distribution of state i.

3.1.1. Computational complexity

From (19), we can see that $u_t(j,d)$ for $j \in \mathcal{S}$, $d \in \mathcal{D}$, given t require $O(MD^2/2)$ multiplications, and the forward–backward formulas (16), (15), (18) and (17) require extra $O(M^2 + MD)$ multiplications. Therefore, the computational complexity of the explicit duration HMM is $O((M^2 + MD + MD^2)T)$. The storage requirement is $O(M^2 + MD + MK + MT)$, where K is the total number of observable values that O_t can take. Due to the computational complexity is high, the explicit duration HMM is not appropriate to be applied in some applications when D is large. To reduce this computational complexity, the key is to reduce the computational complexity of $u_t(j,d)$. Levinson [107] suggested a recursive method that can calculate the product more efficiently, i.e.,

$$u_t(j,d) = \prod_{\tau=t-d+1}^{t} b_j(o_\tau) = u_t(j,d-1) \cdot b_j(o_{t-d+1}) \tag{20}$$

with $u_t(j,1) = b_j(o_t)$, which requires $O(MD)$ multiplications. This recursive method was also used by Mitchell et al. [124]. However, in their method D recursive steps must be performed at every t. Therefore, the total number of recursive steps required in their method increases by a factor of D compared with the Ferguson algorithm [60]. In fact, a better way to reduce both the computational complexity and the total number of recursive steps is letting

$$u_t(j,d) = u_{t-1}(j,d-1) \cdot b_j(o_t) \tag{21}$$

which can be implemented in parallel manner and has no need to retrieve previous observation probabilities $b_j(o_{t-d+1})$. This idea was realized in a parallel implementation of the explicit duration HMM for spoken language recognition on a

hardware architecture in Mitchell et al. [122]. The computational load $p_j(d) \cdot u_t(j,d)$ can also be reduced by approximation such as segmental beam pruning and duration pruning as proposed by Russell [154]. It shows that they can combine to give a 95% reduction in segment probability computations at a cost of a 3% increase in phone error rate.

3.2. Variable transition HMM

In this model, an HSMM is realized in the HMM framework, including the 2-vector HMM [93], the duration-dependent state transition model [179,181,180], the inhomogeneous HMM [151], and the non-stationary HMM [164,46,47]. These approaches take the vector (i,d) as an HMM state, where i is one of the HSMM states and d sojourn time since entering the state. The explicit duration HMM can also be expressed in this model by letting the triples (i,w,d) to be HMM states, where d is a duration and w a counter, $1 \leqslant w \leqslant d$, which indicates the number of observations produced so far while in state i [60]. In addition to the state and its sojourn time, Pieczynski et al. [144] added the observation as the third component. This makes it possible to generalize the model to the triplet Markov chain [144–147,98–101,2]. The constraints among the three components are released in the triplet Markov chain model and the components are extended to be general processes. The price is the loss of physical meaning in the sense of hidden semi-Markov process. One has to add some constraints back on the triplet Markov chain and re-define the meaning of the three processes when it is applied for the HSMM. The triplet Markov chain model can be further generalized to be a non-stationary fuzzy Markov chain by letting the underlying Markov chain be a fuzzy Markov random chain [158].

Compared with the explicit duration HMM, this model assumes the state transition is dependent on the state duration, and hence it is more suitable for describing inhomogeneous or non-stationary hidden Markov processes. This makes it useful for some applications that cannot be modelled by a homogeneous process.

A state transition is allowed only for either $(i,d) \to (j,1)$, for $i \neq j$, or $(i,d) \to (i,d+1)$ for self-transitions. It assumes the "conditional independence" of outputs as given by (12). The boundary conditions use the simplifying assumption described on page 5. The state transition probability from state i to state j given that the sojourn time in state i at time t is d is defined by [151,93]

$$a_{ij}(d) \triangleq P[S_{t+1} = j | S_{[t-d+1:t} = i],$$

subject to $\sum_{j \in \mathcal{S}} a_{ij}(d) = 1$, for $i, j \in \mathcal{S}$, $d \in \mathcal{D}$, where the self-transition with probability $a_{ii}(d) > 0$ can occur. We note that $a_{ij}(d)$ is different from $a_{(i,d)j}$ defined by (1). The latter does not allow self-transition. We have $a_{ij}(d) = (1 - a_{ii}(d))a_{(i,d)j}$.

It is not straightforward to derive the forward–backward formulas for the variable transition HMM from the general ones given by (4) and (5) because $a_{(i,d)(j,d')} = a_{ij}(d) \prod_{\tau=1}^{d'-1} a_{jj}(\tau)[1 - a_{jj}(d')]/[1 - a_{ii}(d)]$. Instead, they are easy to derive from the definitions of the forward variables $\breve{\alpha}_t(j,d)$ and the backward variables $\breve{\beta}_t(j,d)$ [151,93], i.e.,

$$\breve{\alpha}_t(j,d) \triangleq P[S_{[t-d+1:t} = j, o_{1:t} | \lambda] = \begin{cases} \sum_{d' \in \mathcal{D}} \sum_{i \in \mathcal{S} \setminus \{j\}} \breve{\alpha}_{t-1}(i,d') a_{ij}(d') b_j(o_t), & d = 1, \\ \breve{\alpha}_{t-1}(j,d-1) a_{jj}(d-1) b_j(o_t), & d > 1 \end{cases} \tag{22}$$

for $j \in \mathcal{S}$, $d \in \mathcal{D}$, $t = 2, \ldots, T$, and

$$\breve{\beta}_t(j,d) \triangleq P[o_{t+1:T} | S_{[t-d+1:t} = j, \lambda] = \sum_{i \in \mathcal{S} \setminus \{j\}} a_{ji}(d) \breve{\beta}_{t+1}(i,1) b_i(o_{t+1}) + a_{jj}(d) \breve{\beta}_{t+1}(j,d+1) b_j(o_{t+1}) \tag{23}$$

for $j \in \mathcal{S}$, $d \in \mathcal{D}$, $t = T-1, \ldots, 1$. The forward variable $\breve{\alpha}_t(j,d)$ represents the joint probability that the sojourn time in state j at time t is d and the partial observation sequence is $o_{1:t}$. The backward variable $\breve{\beta}_t(j,d)$ is the conditional probability that the future observation sequence is $o_{t+1:T}$ given that the sojourn time in state j at time t is d.

The boundary conditions are $\breve{\alpha}_1(j,1) = \pi_j b_j(o_1)$, $\breve{\alpha}_1(j,d) = 0$ for $d > 1$ and $\breve{\beta}_T(j,d) = 1$, for $j \in \mathcal{S}$, $d \in \mathcal{D}$. Similar to the forward recursion formula, a Viterbi algorithm for the inhomogeneous HMM can be readily obtained by replacing the sum $\sum_{d' \in \mathcal{D}} \sum_{i \in \mathcal{S} \setminus \{j\}}$ of (22) with the maximum operations $\max_{d' \in \mathcal{D}} \max_{i \in \mathcal{S} \setminus \{j\}}$, as did in Ramesh and Wilpon [151] and Deng and Aksmanovic [44].

Though the super state space of the pairwise process (i,d) is $\mathcal{S} \times \mathcal{D}$ in the order of MD, the computational complexity is $O((MD + M^2 D)T)$, where M is the number of HSMM states, D the maximum duration of any HSMM state and K the total number of observable values. The storage requirement is $O(M^2 D + MK + M + MDT)$. Compared with $O((M^2 + MD + MD^2)T)$ of the explicit duration HMM, the computational complexity of the variable transition HMM is higher when the order of the state space is higher, and is lower when the maximum length of the state durations is smaller. However, its space complexity is definitely higher than $O(M^2 + MD + MK + MT)$ of the explicit duration HMM.

The variable transition HMM and the explicit duration HMM have different assumptions for their models. However, the model parameters of the variable transition HMM can be expressed by those of the explicit duration HMM [47,11]. Reversely, the model parameters of the explicit duration HMM can be expressed by those of the variable transition HMM, i.e., $p_i(d) = \prod_{\tau=1}^{d-1} a_{ii}(\tau) \cdot [1 - a_{ii}(d)]$ and $a_{ij} = a_{ij}(d)/[1 - a_{ii}(d)]$. The model [119] defined by (1) and (2) can be considered as a combination of the variable transition HMM and the explicit duration HMM. Therefore, it is more general.

3.3. Residential time HMM

The residential time HMM [199] assumes a state transition is either $(i,1) \to (j,\tau)$ for $i \neq j$ or $(i,\tau) \to (i,\tau-1)$ for a self-transition with $\tau > 1$, where τ is the residential time of state i. The state transition probabilities are assumed to be independent to the duration of the previous state. The residential time HMM also assumes the "conditional independence" of outputs as yielded by (12). The boundary conditions use the simplifying assumption described on page 5. Therefore, this model is useful in the application areas when the residential time that the current state will stay in the future is of interest. This is useful for predicting the residential time of the current state. It is contrast to the variable transition HMM which concerns the sojourn time that the current state has been stayed in the past.

As defined by (3), the state transition probability from state i to state j that will have residential time τ is $a_{i(j,\tau)} \triangleq P[S_{[t:t+\tau-1]} = j | S_{t-1}] = i]$ for $i \neq j$, with $\sum_{j\in\mathcal{S}\setminus\{i\}} \sum_{\tau\in\mathcal{D}} a_{i(j,\tau)} = 1$. The self-transition probability from (i,τ) to $(i,\tau-1)$ is $P[S_{t+1:t+\tau-1]} = i | S_{t:t+\tau-1]} = i] = 1$, for $\tau > 1$. Therefore, the general forward–backward formulas given by (4) and (5) can be reduced to the ones for the residential time HMM. Instead, we derive the formulas directly from the definitions of the forward and backward variables. Define the *forward variable* and *backward variable* by [199]

$$\breve{\alpha}_t(i,\tau) \triangleq P[S_{t:t+\tau-1]} = i, o_{1:t} | \lambda]$$

and

$$\breve{\beta}_t(i,\tau) \triangleq P[o_{t+1:T} | S_{t:t+\tau-1]} = i, \lambda].$$

The forward variable $\breve{\alpha}_t(i,\tau)$ is the joint probability that the partial observation sequence is $o_{1:t}$ and the current state i will stay for the next τ steps and end at $t+\tau-1$. The backward variable $\breve{\beta}_t(i,\tau)$ is the conditional probability that the future observations will be $o_{t+1:T}$ given the current state i that has τ steps of remaining time.

The following forward and backward recursion formulas can be readily obtained:

$$\breve{\alpha}_t(i,\tau) = \breve{\alpha}_{t-1}(i,\tau+1) b_i(o_t) + \sum_{j\in\mathcal{S}\setminus\{i\}} \breve{\alpha}_{t-1}(j,1) a_{j(i,\tau)} b_i(o_t), \tag{24}$$

for $i \in \mathcal{S}$, $\tau \in \mathcal{D}$, $t = 1,\ldots,T$, and

$$\breve{\beta}_t(i,\tau) = b_i(o_{t+1}) \breve{\beta}_{t+1}(i,\tau-1), \quad \tau > 1, \tag{25}$$

$$\breve{\beta}_t(i,1) = \sum_{j\in\mathcal{S}\setminus\{i\}} \sum_{\tau\geqslant 1} a_{i(j,\tau)} b_j(o_{t+1}) \breve{\beta}_{t+1}(j,\tau), \tag{26}$$

for $i \in \mathcal{S}$, $\tau \in \mathcal{D}$, $t = T-1,\ldots,1$. The boundary conditions are $\breve{\alpha}_0(i,1) = \pi_i$, $\breve{\alpha}_0(i,\tau) = 0$ for $\tau > 1$, and $\breve{\beta}_T(i,1) = 1$, $\breve{\beta}_T(i,\tau) = 0$ for $\tau > 1$.

3.3.1. Computational complexity

The computational complexity involved in the residential time HMM is in the same order of the variable transition HMM. However, it can be reduced significantly if the state duration is assumed to be independent to the previous state. In this case, we have $a_{i(j,\tau)} = a_{ij} p_j(\tau)$. From the definition of $\alpha^*_t(i)$ given by (16) and the definition of $\beta^*_t(j)$ given by (17), we have

$$\alpha^*_{t-1}(i) = P[S_{[t} = i, o_{1:t-1} | \lambda] = \sum_{j\in\mathcal{S}\setminus\{i\}} \breve{\alpha}_{t-1}(j,1) a_{ji} \tag{27}$$

and

$$\beta^*_t(j) = P[o_{t+1:T} | S_{[t+1} = j] = b_j(o_{t+1}) \sum_{\tau\in\mathcal{D}} p_j(\tau) \breve{\beta}_{t+1}(j,\tau). \tag{28}$$

Then the forward formula (24) and the backward formula (26) are reduced to [199]

$$\breve{\alpha}_t(i,\tau) = \breve{\alpha}_{t-1}(i,\tau+1) b_i(o_t) + \alpha^*_{t-1}(i) p_i(\tau) b_i(o_t) \tag{29}$$

and

$$\breve{\beta}_t(i,1) = \sum_{j\in\mathcal{S}\setminus\{i\}} a_{ij} \beta^*_t(j). \tag{30}$$

Now the forward recursion is given by (27) and (29), and the backward recursion by (28), (30) and (25).

In this case, computing the forward variables $\alpha^*_{t-1}(i)$ for all i requires $O(M^2)$ steps, and $\breve{\alpha}_t(i,\tau)$ for all i and τ requires extra $O(MD)$ steps. Similarly, computing the backward variables $\beta^*_t(j)$ for all j requires $O(MD)$ steps, and $\breve{\beta}_t(i,1)$ for

all i requires extra $O(M^2)$ steps. Hence, the total number of computation steps for evaluating the forward and backward variables is $O((MD + M^2)T)$. This computational complexity is much lower than those of the explicit duration HMM and the variable transition HMM.

Because the backward variables $\breve{\beta}_t(i,\tau)$ and the probabilities $\eta_t(i,\tau), \xi_t(i,j)$ and $\gamma_t(i)$ do not have to be stored for estimate of the model parameters $\{p_j(\tau), a_{ij}, b_i(o_t), \pi_i\}$, and only the forward variables $\breve{\alpha}_t(i,1)$ and $\alpha^*_{t-1}(i)$ for all i and t need to be stored, with the storage requirement of $O(MT)$. Therefore, the storage requirement for the residential time HMM is $O(M^2 + MD + MK + MT)$, similar to the explicit duration HMM.

The Matlab code for the forward–backward algorithm is quite simple which can be found from the website http://sist.sysu.edu.cn/~syu/. An R package for analyzing hidden semi-Markov models can be found in Bulla et al. [23].

4. Specific modelling issues

This section discusses the issues related to the duration distributions, the observation distributions, variants of the HSMMs and the relationship with standard HMMs.

4.1. Different duration distributions

The choice of distribution family for the state duration is central to the use of the HSMM [56]. So far, we have discussed the general distribution of state duration, where $p_i(d)$ is non-parametric. In some applications, a parametric distribution may be preferred so that only a few parameters that specify the selected distribution functions are required to be estimated. This subsection presents exponential family distributions, convex monotonic distributions, and discrete Coxian distributions of duration. A uniform distribution of duration can be found in Hongeng and Nevatia [79]. More complex duration models can be found in Ostendorf et al. [136], and a discussion on the capacity and complexity of duration modelling techniques can be found in Johnson [86]. Besides, the state duration distributions can be taken into account in the Viterbi algorithm as in Burshtein [25,26], Yoma and McInnes [195] and Yoma and Sanchez [196], where the state duration distributions can be estimated efficiently for a left–right HMM.

4.1.1. Exponential family distribution of duration

The state duration can be modelled by Poisson [155], Gaussian [5] and gamma distributions [107]. All these distributions belong to the exponential family [107,123]. The probability density function (pdf) or probability mass function (pmf) for the duration of state j belonging to the exponential family can be expressed as [123]

$$p_j(d) = \frac{1}{B(\boldsymbol{\theta}_j)}\xi(d)\exp\left(-\sum_{p=1}^{P}\theta_{j,p}S_p(d)\right)$$

where P is the number of natural parameters, $\theta_{j,p}$ is the pth natural parameter for state j and $\boldsymbol{\theta}_j = [\theta_{j,1}, \ldots, \theta_{j,P}]$, $S_p(d)$ and $\xi(d)$ are sufficient statistic, and $B(\boldsymbol{\theta}_j)$ is a normalizing term. Since HSMMs are discrete-time stochastic processes, a pmf is used rather than a pdf. For simplicity, the pmf is obtained by letting

$$B(\boldsymbol{\theta}_j) = \sum_{d=1}^{D}\xi(d)\exp\left(-\sum_{p=1}^{P}\theta_{j,p}S_p(d)\right).$$

According to Ferguson [60], the new duration parameters for state j can be found by maximizing

$$f(\boldsymbol{\theta}_j) \triangleq \sum_{d=1}^{D}\hat{p}_j(d)\log p_j(d)$$

subject to the constraint $\sum_{d=1}^{D} p_j(d) = 1$, where $\{\hat{p}_j(d)\}$ is the non-parametric probability mass function estimated by the re-estimation formulas, and $\sum_{d=1}^{D}\hat{p}_j(d) = 1$. Since the exponential family is log-concave, the global maximum can be found by setting the derivative equal to zero, yielding the maximum likelihood equations [123]

$$\frac{\partial}{\partial\theta_{j,p}}f(\boldsymbol{\theta}_j) = \sum_{d=1}^{D}\hat{p}_j(d)\left[-\frac{\partial\log B(\boldsymbol{\theta}_j)}{\partial\theta_{j,p}} - S_p(d)\right] = 0$$

where, for the pmf,

$$-\frac{\partial\log B(\boldsymbol{\theta}_j)}{\partial\theta_{j,p}} = \sum_{d=1}^{D}p_j(d)S_p(d).$$

Therefore, the new duration parameters can be found by solving the following equations:

$$\sum_{d=1}^{D} p_j(d) S_p(d) = \sum_{d=1}^{D} \hat{p}_j(d) S_p(d), \quad p = 1, \ldots, P.$$

The MAP estimation of the mean and variance of Gaussian and gamma distributions of state durations for a left–right HMM can be found in Yoma and Sanchez [196].

4.1.2. Convex monotonic distributions

For the explicit duration HMM, the extended Viterbi algorithm for HSMM given by (11) becomes

$$\delta_t(j) = \max_{d\in\mathcal{D}} \max_{i\in\mathcal{S}\setminus\{j\}} \left\{ \delta_{t-d}(i) a_{ij} p_j(d) \prod_{t'=t-d+1}^{t} b_j(o_{t'}) \right\}, \tag{31}$$

for $1 \leqslant t \leqslant T$, $j \in \mathcal{S}$, where $\delta_t(j) \triangleq \max_{s_{1:t-1}} P[s_{1:t-1}, S_t = j, o_{1:t}|\lambda]$. Tweed et al. [178] find that if $p_j(d)$ is *concave monotonic*, i.e.,

$$C_1 p_j(d_1) \leqslant C_2 p_j(d_2) \quad \Longrightarrow \quad C_1 p_j(d_1+\tau) \leqslant C_2 p_j(d_2+\tau),$$

for $\tau > 0$ and arbitrary constants C_1 and C_2, then

$$\begin{aligned} &\delta_{t-d_1}(i_1) a_{i_1 j} p_j(d_1) \prod_{t'=t-d_1+1}^{t} b_j(o_{t'}) \leqslant \delta_{t-d_2}(i_2) a_{i_2 j} p_j(d_2) \prod_{t'=t-d_2+1}^{t} b_j(o_{t'}) \\ &\Longrightarrow \quad \delta_{t-d_1}(i_1) a_{i_1 j} p_j(d_1+\tau) \prod_{t'=t-d_1+1}^{t+\tau} b_j(o_{t'}) \leqslant \delta_{t-d_2}(i_2) a_{i_2 j} p_j(d_2+\tau) \prod_{t'=t-d_2+1}^{t+\tau} b_j(o_{t'}) \end{aligned} \tag{32}$$

for $d_1 > d_2$ and any $\tau > 0$. This means that for given state j if the longer segmentation has a lower probability, then it will always have a lower probability as both segmentations are further extended [178]. This fact can be used to reduce the number of items in (31). That is, if (32) is satisfied, then $\delta_{t-d_1}(i_1)$ can never give optimal solutions in the future and the index $(t-d_1, i_1)$ can be removed from the jth set, $Q(j)$, for state j. Therefore, there exits $t_0^* = \min_{(t-d_1)\in Q(j)}(t-d_1)$ such that $(t_0, i_1) \notin Q(j)$, for any $t_0 < t_0^*$, $i_1 \in \mathcal{S}$. This implies that $Q(j)$ is a queue that only older ones can be discarded. The newer items (t, i), $i \in \mathcal{S}$, are pushed into the queue after $\delta_t(j)$ for all $j \in \mathcal{S}$ are determined by

$$\delta_t(j) = \max_{(t_0,i)\in Q(j)} \left\{ \delta_{t_0}(i) a_{ij} p_j(t-t_0) \prod_{t'=t_0+1}^{t} b_j(o_{t'}) \right\},$$

for $1 \leqslant t \leqslant T$, $j \in \mathcal{S}$, where for simplicity it assumes $D \geqslant T$.

4.1.3. Discrete Coxian distribution of duration

A discrete Coxian distribution of duration is an extension of the non-parametric distribution of duration for a conventional HSMM that the duration distribution is $p_i(d)$, for given state $i \in \mathcal{S}$.

Denote the discrete Coxian distribution by $\mathrm{Cox}(\mu, \theta)$, where $\mu = \mu_1, \ldots, \mu_N$ and $\theta = \theta_1, \ldots, \theta_N$ are parameters [54,55]. A left-to-right Markov chain with N states (phases) is used to describe $\mathrm{Cox}(\mu, \theta)$. Each phase n has the self-transition probability $A_{nn} = 1 - \theta_n$, $0 < \theta_n \leqslant 1$, $n \in \{1, \ldots, N\}$ and the duration of geometric distribution $X_n \sim \mathrm{Geom}(\theta_n)$. If the left-to-right Markov chain starts from phase n with initial probability μ_n, then $X_n + \cdots + X_N$ is the duration of the left-to-right Markov chain, for $0 \leqslant \mu_n \leqslant 1$, $\sum_n \mu_n = 1$.

In this case, the state duration distribution, $p_i(d)$, of the HSMM becomes $\mathrm{Cox}(\mu^{(i)}, \theta^{(i)})$ with $d = X_n + \cdots + X_N$, for $i \in \mathcal{S}$, where $\mu^{(i)}$ and $\theta^{(i)}$ are the Coxian parameter set for the HSMM state i. When $\theta_n^{(i)} = 1$ for all n, it reduces to the conventional duration distribution of HSMM with $X_n \equiv 1$, $p_i(d) = \mu_{N-d+1}^{(i)}$, $D = N$.

Obviously, (i, n) can be considered as an HMM state, where $i \in \mathcal{S}$ is the state of the HSMM and $n \in \{1, \ldots, N\}$ is the phase of the Coxian distribution. Therefore, the traditional forward–backward algorithm for HMM can be applied for the model parameter re-estimation and state sequence estimation.

An HMM state (i, n) can transit to (i, n) with self-transition probability $A_{nn}^{(i)} = 1 - \theta_n^{(i)}$ for any n, $(i, n+1)$ with probability $A_{n,n+1}^{(i)} = \theta_n^{(i)}$ for $n < N$, or (j, n') with probability $a_{ij}\mu_{n'}^{(j)}$ for $n = N$, where $A_{nm}^{(i)}, \theta_n^{(i)}, \mu_n^{(i)}$ are parameters for state i. The computational complexity is $O(M^2 NT)$.

A modification of this model is assuming that the left-to-right Markov chain of each HSMM state always starts from phase 1 and ends at any phase n. Then the forward algorithm can be expressed as

$$\alpha_t\big[(i,n),d\big] = \begin{cases} \sum_{j\in\mathcal{S}} \sum_{n'=1}^{N} \sum_{d\in\mathcal{D}} \alpha_{t-1}[(j,n'),d'] p_j(d') a_{ji} b_{i,1}(o_t), & n = 1,\ d = 1, \\ \alpha_{t-1}[(i,1),d-1] A_{1,1}^{(i)} b_{i,1}(o_t), & n = 1,\ d > 1, \\ 0, & n > 1,\ d = 1, \\ \sum_{n'=n-1}^{n} \alpha_{t-1}[(i,n'),d-1] A_{n'n}^{(i)} b_{i,n}(o_t), & n > 1,\ d > 1 \end{cases}$$

where $p_i(d)$ is the duration probability, $b_{i,n}(o_t)$ the observation probability, $A^{(i)}_{n'n}$ the transition probability from phase (i,n') to (i,n), and $\alpha_t[(i,n),d]$ the forward variable at time t when the state is i, the phase is n, and the duration of state i is d [182,183,166]. The Viterbi version of this forward formula can be straightforward by replacing the $\sum$ with max as shown in Kwon and Un [97] and Peng et al. [141].

Another modification of the model is allowing the Markov chain of each HSMM state is not necessary to be left-to-right. That is, the transition probability from phase n to any phase m is allowed, with $A_{nm} \geqslant 0$, for $n \neq m$ and $n, m \in \{1, \ldots, N\}$. This is called "extended HMM" in Russell and Cook [156]. In this case, the Markov chain of each HSMM state is in fact a series-parallel network of geometric processes. Based on Coxian theory, the overall duration pdf of the series-parallel network can construct any discrete pdf with rational *z-transform* [182,18,183].

4.2. Different observation distributions

This subsection presents various observation distributions that have been used in applications.

4.2.1. Parametric distribution of observations

The observation variable O_t is usually assumed as a discrete variable with finite alphabet $|\mathcal{V}| = K$. In some applications, however, a parametric distribution with possibly infinite support may be required or preferred. For instance, the probability distribution $b_i(v_k)$ may be represented as a Poisson distribution. In some other applications, the observation variable O_t may be treated as a continuous variable, e.g., Gaussian random variable. In such cases the number of model parameters can be reduced substantially [60].

For example, if the probability density function $b_i(v_k)$, for $v_k = 0, 1, \ldots, \infty$, is Poisson with mean μ_i, i.e., $b_i(k) = \mu_i^k e^{-\mu_i}/k!$, then the parameter μ_i can be re-estimated by $\hat{\mu}_i = \sum_k \hat{b}_i(k)k$, or equivalently,

$$\hat{\mu}_i = \frac{\sum_{t=1}^{T} \gamma_t(i) o_t}{\sum_i \sum_{t=1}^{T} \gamma_t(i)}.$$

A similar result can be obtained by directly maximizing the likelihood function $P[o_{1:T}|\lambda]$ [108].

For the continuous random variable $O_t = v_k$, the probability distribution $b_j(v_k)$ should then be replaced by a probability density function. More generally, the observation distribution $b_j(v_k)$ of any state j is often represented by a mixture of distributions such as Gaussian distributions [137]. For a given state j, the probability density function that state j produces an observation v_k can then be written as [83]

$$b_j(v_k) = \sum_{n=1}^{N} p_{jn} f_{jn}(v_k)$$

where N denotes the number of the mixture probability density functions, $\{f_{jn}(v_k)\}$, and $\sum_{n=1}^{N} p_{jn} = 1$. A linear mixed models for the observation distribution can be found in Chaubert-Pereira et al. [30].

4.2.2. Segmental model

Usually the observation distributions are assumed to be dependent on the states. The segmental model [92,157,63,87,43, 74,136,138,78,203,64,1] extend them to be dependent on the states as well as the state durations.

Suppose there is a sequence of distribution regions corresponds to the sequence of observations for a given state. Then the observation probabilities are defined by $b_{i,d}(o_t) \triangleq P[o_t|S_{[t-d+1:t} = i]$, where $b_{i,d}(\cdot)$ denotes the dth distribution region of given state i, for $d = 1, \ldots, D$. $\{b_{i,d}(\cdot)\}$ can be a set of model regions or a continuum of distributions via trajectory sampling. An instance of the continuum of distributions is

$$b_{i,d}(o_t) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp \frac{-(o_t - c_i^{d-1}\mu_i)^2}{2\sigma_i^2},$$

where c_i, μ_i, σ_i are the parameters of the Gaussian distribution. This model was called "exponentially decay state" by Krishnamurthy and Moore [92]. An application in detection of shape anomalies can be found in Z. Liu et al. [114].

More generally, the observation probabilities can be defined by $b^{(\tau)}_{i,d}(o_{t+d}) \triangleq P[o_{t+d}|S_{[t+1:t+\tau]} = i]$, for $d = 1, \ldots, \tau$, where τ is given and denotes the length of the segment or the duration of state i, o_{t+d} is the dth observation of the segment $o_{t+1:t+\tau}$, and $b^{(\tau)}_{i,d}(\cdot)$ the dth distribution region for given state i and the length τ. According to this definition, $b^{(\tau_1)}_{i,d}(o_{t+d})$ may different from $b^{(\tau_2)}_{i,d}(o_{t+d})$ if $\tau_1 \neq \tau_2$. For instance, Kim and Smyth [89] use the segmental HMM in waveform modelling, which models the rth segment of observations of length τ, $\boldsymbol{o}_r = o_{t+1:t+\tau}$, generated by state i, as a linear function of time

$$\boldsymbol{o}_r = a_i + c_i \boldsymbol{x}_r + \boldsymbol{e}_r, \qquad \boldsymbol{e}_r \sim N_\tau\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_\tau\right)$$

where a_i and c_i are regression coefficients for the intercept and slope of the waveform, respectively. $\boldsymbol{x}_r$ is a $\tau \times 1$ vector representing discrete time values. $\boldsymbol{e}_r$ a $\tau \times 1$ vector of Gaussian noise with variance σ^2 for each component. If random effects

are added to the segment distribution to model parameter variability across waveforms, then the regression coefficients can be $a_i \sim N(\bar{a}_i, \sigma_i^2)$ and $c_i \sim N(\bar{c}_i, \sigma_i^2)$ [89].

Generally, extra-segmental variability associated with a state i can be characterized by a probability density function g_i called the *state target PDF* [157,77]. A target distribution $b_i(\cdot)$ is chosen according to g_i [7]. Then the joint probability of the segment $o_{t+1:t+\tau}$ and a particular target $b_i(\cdot)$ given state i is given by [157]:

$$P[o_{t+1:t+\tau}, b_i | S_{[t+1:t+\tau]} = i] = g_i(b_i) \prod_{d=1}^{\tau} b_i(o_{t+d}).$$

Therefore, $P[o_{t+1:t+\tau} | S_{[t+1:t+\tau]} = i] = \sum_{b_i} P[o_{t+1:t+\tau}, b_i | S_{[t+1:t+\tau]} = i]$.

If the distribution regions $\{b_{i,d}^{(\tau)}(\cdot)\}$ are given, then it is a deterministic distribution mappings that associate the dth observation o_{t+d}, for $d = 1, \ldots, \tau$, in the τ-length segment $o_{t+1:t+\tau}$ with the dth specific region $b_{i,d}^{(\tau)}(\cdot)$; otherwise, it is a dynamic distribution mappings, which can be implemented using dynamic programming to find the maximum likelihood mapping that associate the segment $o_{t+1:t+\tau}$ to a fixed number of regions. If we still assume that observations are conditionally independent given the segment length, i.e.,

$$P[o_{t+1:t+\tau} | S_{[t+1:t+\tau]} = i] = \prod_{d=1}^{\tau} b_{i,d}^{(\tau)}(o_{t+d})$$

then a segment model with an unconstrained dynamic mapping is equivalent to an HMM network [103]. A detailed discussion of segmental models can be found in Ostendorf et al. [136].

An extension of the segmental model is defining the observation distributions to be dependent on both the state and its substates [97]. This model is in fact a special case of the original HSMM if we define a complex state that includes both the state and its substate.

4.2.3. Event sequence model

Thoraval et al. [175], Thoraval [176] and Faisan et al. [58,59] introduced a hidden semi-Markov event sequence model (HSMESM), which is a special instance of hidden semi-Markov model for the modelling and analysis of event-based random processes. At a given time t there may be an event $o_t = v_k$ observed, with an occurring probability $1 - e_{s_t}$, or a null observation (missing observation) $o_t = \phi$ with missing probability e_{s_t}, where s_t is the state at time t. Therefore, the modified observation probabilities in the case that there exist missing observations, denoted by $b_i^+(o_t)$, are given by

$$b_i^+(o_t) = (1 - e_i) b_i(o_t) \cdot \mathbb{I}(o_t \in \mathcal{V}) + e_i \cdot \mathbb{I}(o_t \notin \mathcal{V}), \quad i \in \mathcal{S},$$

where $\mathcal{V}$ is the set of observable events, and $\mathbb{I}(x) = 1$ if x is true and zero otherwise.

This event sequence model is called "state-dependent observation misses" by Yu and Kobayashi [200] because the null observation ϕ is treated as one of the observable, i.e., the full set of observations is $\phi \cup \mathcal{V}$. They classify patterns of observation misses into five types, where except the state-dependent observation misses, the other four types are as follows:

a) Output-dependent observation misses: the probability that a given o_t becomes "null" depends on the output value o_t itself. For instance, when the output is too weak (in comparison with noise) at time t, such output may not be observed. In this case, the "output-dependent miss probability" is defined by $e(v_k) = P[\phi | v_k]$. Then the probability for a null observation $o_t = \phi$ is $\sum_k b_i(v_k) e(v_k)$.

b) Regular observation misses: the outputs $\{o_t\}$ are observed only at predetermined epochs. Regular or periodic sampling is a typical example. The rest of $\{o_t\}$ will be missed, and such portion will be considerable, if the sampling is done infrequently. In this case, the observation probability for a null observation $o_t = \phi$ is $\sum_k b_i(v_k) = 1$.

c) Random observation misses: the outputs $\{o_t\}$ are observed at randomly chosen instants. Such observation pattern may apply, when the measurement is costly or we are not interested in keeping track of state transitions so closely. In this case, some outputs may become "null" observations randomly. If the sampling probability is $1 - e$, then the modified observation probability is $b_i^+(o_t) = \mathbb{I}(o_t \in \mathcal{V})(1 - e) b_i(o_t) + \mathbb{I}(o_t \notin \mathcal{V}) e$.

d) Miss match between multiple observation sequences: multiple observation sequences are associated with the hidden state sequence, and these observations may not be synchronized to each other. For instance, two sequences $\{o_t\}$ and $\{q_t\}$ are available as the outputs of an HSMM state sequence, but there exits some random delay τ between the two output sequences. Therefore, the observations we can obtain at time t are o_t and $q_{t-\tau}$ though the emissions of a given state at time t are o_t and q_t. In this case, delay τ has to be estimated by maximizing the joint likelihood of the two observation sequences.

4.3. Variants of HSMM

This subsection discusses the extension of HSMMs, which includes switching HSMM, multi-channel HSMM, and adaptive factor HSMM.

4.3.1. Switching HSMM

A switching hidden semi-Markov model is defined as the concatenation of many HSMMs, with model parameter sets $\lambda_1, \ldots, \lambda_{|Q^*|}$, each initiated by a different "switching" state $q \in Q^*$, where the set of states Q^* defines a Markov chain, as described in Duong et al. [54] and Phung et al. [142,143]. A two stage inhomogeneous HMM was proposed by Sitaram and Sreenivas [165] to capture the variabilities in speech for phoneme recognition. The first stage models the acoustic and durational variabilities for all distinct sub-phonemic segments and the second for the whole phoneme.

Assumes at each time step the computational complexity is C_q for computing the forward–backward variables of the qth HSMM. Because the current state sequence of the qth HSMM may start at any time step before t, the computational complexity for the qth HSMM is in the order of $O(t^2 C_q)$. Therefore, the total computational complexity for the switching HSMM is $O(T^3 \sum_{q=1}^{|Q^*|} C_q + |Q^*|^2)$, where T is the total length of the observation sequence. This means the switching HSMM is applicable only for short observation sequences.

4.3.2. Multi-channel HSMM

Multi-channel HSMM was proposed to model multiple interacting processes [129]. In contrast to the basic HSMM that a process has a single state at any instant, this extension generalizes the HSMM state to be a vector $\boldsymbol{S}_t = [S_t^{(1)}, \ldots, S_t^{(C)}]$ representing the states of multiple processes. Each of the processes, say process c, has an observation sequence, $O^{(c)}$, produced by a hidden semi-Markov state sequence, $S^{(c)}_{[1:d_1]}, \ldots, S^{(c)}_{[T-d_n+1:T]}$, taking values in $\mathcal{S}^{(c)}$, with a set of model parameters, $\lambda^{(c)}$, where T is the length of each of the observation sequences. Though this model can be realized in the framework of inhomogeneous HMM, the state space will be M^C. To reduce the complexity of the model, some simplifying assumptions should be made for the transition probabilities. Assume $P[\boldsymbol{s}_{t+1}, \boldsymbol{d}_{t+1} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda]$, and the observation probabilities, $P[\boldsymbol{o}_t | \boldsymbol{s}_t, \lambda]$, where $\boldsymbol{d}_t = [d_t^{(1)}, \ldots, d_t^{(C)}]$ and $d_t^{(c)}$ is the duration having been spent in state $s_t^{(c)}$ by time t. $\boldsymbol{o}_t = [o_t^{(1)}, \ldots, o_t^{(C)}]$ is the observation vector at time t. Usually, each channel is allowed to evolve independently, i.e.,

$$P[\boldsymbol{s}_{t+1}, \boldsymbol{d}_{t+1} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda] = \prod_{c=1}^{C} P\big[s_{t+1}^{(c)}, d_{t+1}^{(c)} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda\big]$$

and

$$P[\boldsymbol{o}_t | \boldsymbol{s}_t, \lambda] = \prod_{c=1}^{C} P\big[o_t^{(c)} | s_t^{(c)}, \lambda\big].$$

To further simplify the expressions, $P[s_{t+1}^{(c)}, d_{t+1}^{(c)} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda]$ is approximated by [129]

$$P\big[s_{t+1}^{(c)}, d_{t+1}^{(c)} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda\big] \sim \prod_{c'=1}^{C} P\big[s_{t+1}^{(c)}, d_{t+1}^{(c)} | s_t^{(c')}, d_t^{(c')}, \lambda\big]$$

or

$$P\big[s_{t+1}^{(c)}, d_{t+1}^{(c)} | \boldsymbol{s}_t, \boldsymbol{d}_t, \lambda\big] = \sum_{c'=1}^{C} \theta_{c,c'} P\big[s_{t+1}^{(c)}, d_{t+1}^{(c)} | s_t^{(c')}, d_t^{(c')}, \lambda\big],$$

where $\theta_{c,c'}$ is the weight.

According to this simplified assumption, the current state-duration pair $s_{t+1}^{(c)}, d_{t+1}^{(c)}$ is dependent on the previous state-duration pair $s_t^{(c')}, d_t^{(c')}$. There are MD current state-duration pairs for given c and MD previous state-duration pairs for given c'. Therefore, the computational complexity for evaluating the forward or backward variables at each time step is $O(M^2 D^2 C^2)$.

4.3.3. Adaptive factor HSMM

Adaptive factor HSMM is assumed to have variable model parameters for its parametric distributions of observation and/or state durations. For instance, the parameters of the observation distributions $b_i(v_k)$ and the duration distributions $p_i(d)$ can be changed with time or in different situations. For example, the mean μ_i of $b_i(v_k)$ given state i is changed for different speakers, such that for the fth speaker the mean becomes $a_i^{(f)} + c_i^{(f)} \mu_i$, for $f = 1, \ldots, F$, where $\{a_i^{(f)}, c_i^{(f)}\}$ are adaptive factor, μ_i the common parameters for all speakers, and F the total number of speakers. Let $O^{(f)} = (o_1^{(f)}, o_2^{(f)}, \ldots, o_{T_f}^{(f)})$ be the fth observation sequence of length T_f, for $f = 1, \ldots, F$, λ be the set of the common parameters of the HSMM, and $\Lambda = \{\Lambda^{(1)}, \ldots, \Lambda^{(F)}\}$ be the set of the adaptive factors, where $\mu_i \subset \lambda$ and $(a_i^{(f)}, c_i^{(f)}) \subset \Lambda^{(f)}$. Then the model parameters can be jointly estimated by

$$\{\hat{\lambda}, \hat{\Lambda}\} = \arg\max_{\lambda, \Lambda} P\big[O^{(1)}, \ldots, O^{(F)} | \lambda, \Lambda\big] = \arg\max_{\lambda, \Lambda} \prod_{f=1}^{F} P\big[O^{(f)} | \lambda, \Lambda^{(f)}\big].$$

Specific examples for the joint estimate of the model parameters are given in Yamagishi and Kobayashi [189] and Yamazaki et al. [192], where $o_t^{(f)}$, $a_i^{(f)}$ and μ_i are vectors, and $c_i^{(f)}$ is a matrix.

4.4. HSMM vs. HMM

HSMM can be considered as an extension of an HMM. Therefore, the HMM can be reversely considered as a special case of the HSMM, in considering that the duration distributions of HMM states are implicitly geometric. This subsection discusses the relationship between HSMM and the conventional HMM. A discussion about both HMM and HSMM can also be found in Kobayashi and Yu [91].

A hybrid HMM/HSMM proposed by Guedon [71,72] can be viewed as a special case of the HSMM where the occupancy distributions of some states are constrained to be geometric distributions while others are still generally distributed.

4.4.1. HSMM using HMM algorithms

In some application areas, such as speech recognition, one often uses HMM algorithms to estimate HSMM parameters to obtain benefits from both the small computational complexity of HMM algorithms and the explicit duration expression of HSMM states. As discussed in the last subsection, HMM implicitly has a geometric distribution that limits its capability in many applications whose duration is not geometrically distributed.

Duration estimation using HMM algorithms. After the optimum state sequence, $i_1^* i_2^* \dots i_T^* = (j_1^*, d_1^*) \dots (j_N^*, d_N^*)$, is found via the ordinary Viterbi algorithm or the Baum–Welch algorithm [13], based on the current HMM parameters $\{\pi_i, a_{ij}, b_i(v_k)\}$, where $\sum_{n=1}^{N} d_n^* = T$ and $N \leqslant T$, $i_t^*, j_n^* \in \mathcal{S}$, $d_n^* \in \mathcal{D}$, the model parameters $\hat{\pi}_i$, $\hat{a}_{ij}$, $\hat{b}_i(v_k)$, and $\hat{p}_i(d)$ (for HSMM) can be estimated [36] by letting

$$\eta_t(i,d) = \mathbb{I}\big(j_n^* = i\big) \cdot \mathbb{I}\big(d_n^* = d\big) \cdot \mathbb{I}\left(\sum_{k=1}^{n} d_k^* = t\right),$$

$$\xi_t(i,j) = \mathbb{I}\big(i_t^* = i\big) \cdot \mathbb{I}\big(i_{t+1}^* = j\big),$$

and

$$\gamma_t(i) = \mathbb{I}\big(i_t^* = i\big),$$

where $\eta_t(i,d)$, $\xi_t(i,j)$ and $\gamma_t(i)$ are defined by (6), (7) and (9), and $\mathbb{I}(x) = 1$ if x is true and zero otherwise. Then the model parameters $\hat{a}_{ij}$, $\hat{b}_i(v_k)$, $\hat{p}_i(d)$, and $\hat{\pi}_i$ can be re-estimated. The re-estimated model parameters can be used to produce a new segmentation from which new sets of parameters are obtained until the model converges.

Certainly, the optimum state sequence, $i_1^* i_2^* \dots i_T^* = (j_1^*, d_1^*) \dots (j_N^*, d_N^*)$, can also be found via the extended Viterbi algorithm for the HSMM [138,195,196], as discussed in Section 2.2.3, and then the model parameters are similarly re-estimated.

The estimated duration probabilities can be applied to modify the scores in the Viterbi algorithm on each departure from a state. This approach is non-optimal but can be ensured that the resulting state segmentation sequence is at least reasonable according to the duration specifications. During the recognition phase, the distance metric used in the Viterbi algorithm is modified as [150]:

$$\delta_t(j) = \max_{d \in \mathcal{D}} \max_{i \in \mathcal{S}} \left\{ \delta_{t-d}(i) a_{ij} \big[p_j(d)\big]^{\alpha} \prod_{\tau=t-d+1}^{t} b_j(o_\tau) \right\},$$

for $2 \leqslant t \leqslant T$, $j \in \mathcal{S}$, where α is a modification factor, which is usually assumed as 1. A similar modification to the forward algorithm of HMM was given in Hanazawa et al. [73]. With state duration-dependent transition probabilities the recursion in Viterbi algorithm becomes [179]

$$\delta_t\big(j, d_t^j\big) = \max_{i \in \mathcal{S}} \big\{ \delta_{t-1}\big(i, d_{t-1}^i\big) a_{ij}\big(d_{t-1}^i\big) b_j(o_t) \big\},$$

for $j \in \mathcal{S}$, $2 \leqslant t \leqslant T$, where $a_{ij}(d_t^i)$ denotes the probability of a transition from state i to state j of the model, given that state i has been in the current state for d_t^i consecutive time units. d_t^j denotes the current residency duration of state j. $d_t^j = d_{t-1}^j + 1$ if the state does not change and $d_t^j = 1$ otherwise.

A modification to the conventional Viterbi algorithm suggested by Lee and Rabiner [104] is letting:

$$\delta_t(i) = b_i(o_t) \max \left\{ \delta_{t-1}(i) a_{ii},\ \max_{j \in \mathcal{S} \setminus \{i\}} \delta_{t-1}(j) p_j\big(d_{t-1}^j\big) a_{ji} \right\}.$$

The problem of this algorithm is obviously that it gives preference to the first case that there is no state change. To rectify this situation Preez [149] suggested a modified recursion by letting

$$\delta_t(i) = b_i(o_t)\delta_{t-1}(i)a_{ii}, \qquad d_t^i = d_{t-1}^i + 1$$

if $\delta_{t-1}(i)a_{ii}\sum_{d\geqslant d_{t-1}^i} p_i(d) > \max_{j\in\mathcal{S}\setminus\{i\}} \delta_{t-1}(j)p_j(d_{t-1}^j)a_{ji}$; otherwise $d_t^i = 1$ and

$$\delta_t(i) = b_i(o_t)\max_{j\in\mathcal{S}\setminus\{i\}} \delta_{t-1}(j)p_j\big(d_{t-1}^j\big)a_{ji}.$$

Another way of using $\{p_i(d)\}$ in the Viterbi algorithm is through introducing temporal constraints in the HMM, that is, let [194]

$$a_{ii} = \frac{\sum_{d=\tau+1}^{D} p_i(d)}{\sum_{d=\tau}^{D} p_i(d)},$$
$$a_{ij} = \frac{p_i(\tau)}{\sum_{d=\tau}^{D} p_i(d)}$$

where τ is the duration of a given state i.

Bounded state durations. It is possible to implement upper and lower bounds on duration without specific probabilistic modelling [67,181,90,102]. The bounded state durations can be estimated in the training phase [102] or after the training phase by finding the global minimum and maximum durations for each state [67,90]. Following the definitions in the previous sections, which define $\alpha_t(j) \triangleq P[S_t = j, o_{1:t}|\lambda]$ and $\alpha_t(j,d) \triangleq P[S_{[t-d+1:t]} = j, o_{1:t}|\lambda]$, Laurila [102] yields

$$\alpha_{t+1}(j,1) = \sum_{i\in\mathcal{S}\setminus\{j\}} \alpha_t(i)a_{ij}b_j(o_{t+1}),$$
$$\alpha_{t+1}(j,d+1) = \alpha_t(j,d)a_{jj}b_j(o_{t+1}),$$

and

$$\alpha_{t+1}(j) = \sum_{d=D_{\min}}^{D_{\max}} \alpha_{t+1}(j,d),$$

where $D_{\min}$ and $D_{\max}$ are the bounds of the state durations. The bounded state duration model resulted in quite loose state duration constraints and were not effective enough in the recognition phase.

5. Applications of HSMM

The HSMM has been applied to thirty areas, such as human activity recognition [38,55,56,79,80,119,120,128–131,140, 197,200,205,206], handwriting recognition or printed text recognition [4,15,16,27,32–35,95,96,163], network traffic characterization and anomaly detection [105,111,116,117,139,153,172,177,187,188,198,201], speech recognition [31,39,67,76,107, 127,137,148,152,155], speech synthesis [125,132–134,169–171,186,189–192,204], functional MRI brain mapping [58,59,174–176], electrocardiograph (ECG) [53,85,84,174], recognition of human genes in DNA [19,24,94,173], language identification [118], ground target tracking [88], document image comparison and classification at the spatial layout level [81,82], change-point/end-point detection for semi-conductor manufacturing [64,65], protein structure prediction [8,12,162], the analysis of branching and flowering patterns in plants [69–72], rain events time series model [3,159,160], satellite propagation channel modelling [112], event recognition in videos [79], mobility tracking in cellular networks [120,197,200], Internet traffic modelling [198], image segmentation [21,98,184], semantic learning for a mobile robot [167], symbolic plan recognition [54], terrain modelling [185], adaptive cumulative sum test for change detection in non-invasive mean blood pressure trend [193], equipment prognosis [14,48–51,75], financial time series modelling [22], classification of music [113], remote sensing [147], and prediction of particulate matter in the air [52].

Among those applications, the major applications include speech recognition, speech synthesis, human activity recognition, handwriting recognition, network traffic modelling & anomaly detection, and functional MRI brain mapping.

5.1. Human activity recognition

How to learn and recognize human activities of daily living (ADL) is an important research issue in building a pervasive and smart environment. Yu and Kobayashi [200] proposed an explicit duration HMM for missing data and multiple observation sequences, and applied the model into mobility tracking in wireless networks [120,197]. Hongeng and Nevatia [79], Hongeng et al. [80] and Zhang et al. [206] applied HSMM for recognizing events in a video surveillance. Pavel et al. [140] used the model in unobtrusive assessment of mobility. Marhasev et al. [119] used a non-stationary HSMM in activity recognition. Niwase et al. [131] used the HSMM in human walking motion synthesis.

Different from the usual application of HSMM, Duong et al. [54–56] introduced a two-layered extension of HSMM for modelling the ADL. In this model, sequences of major activities of human daily routine, such as making breakfast, eating

breakfast, going to work, coming back home, are modelled by a Markov chain, with each state representing a major activity. For a given major activity, atomic activities such as spending time at the cupboard, stove, fridge, or moving between these designated places, are described by a hidden semi-Markov model. Each HSMM state represents an atomic activity and its duration represents the atomic action time.

Similarly, Zhang et al. [205] used a layered HSMM in an intelligent surveillance system deployed in parking lots. Chung and Liu [38] applied a hierarchical context hidden Markov model for behavior understanding from video streams in a nursing center. The two-layered HSMM has been discussed in Section 4.3.1 "Switching HSMM" of this paper. A three-layered variable transition hidden Markov model was introduced by Natarajan and Nevatia [130] for representing the composite actions at the top-most layer, the primitive actions at the middle layer and the body pose transitions at the bottom-most layer.

As an extension to the layered HSMM, Natarajan and Nevatia [128] proposed a hierarchical multi-channel HSMM. A multi-channel HSMM is used to model multiple interacting processes, which has been discussed in Section 4.3.2 "Multi-channel HSMM." As a special case of multi-channel HSMM, a coupled HSMM is used for continuous sign-language recognition in Natarajan and Nevatia [129].

5.2. Handwriting recognition

In the application of HSMM in handwriting recognition, the 26 letters in the alphabet are defined as 26 different states of an HSMM, and the number of subcharacter symbols in a letter as the duration of the state. For instance, "B" has three subcharacter symbols and "P" has two. Therefore, the key problem is to develop a segmentation algorithm to translate the 2-D image of written text into a 1-D sequence of subcharacter symbols. The 1-D sequence is used as the observation sequence of the HSMM.

Chen et al. [32,33,35] proposed a robust segmentation algorithm based on mathematical morphology and used a mixture Gaussian distribution to model the subcharacter symbol probability distribution for each state. The Viterbi algorithm given by (11) is used in finding the best path of states for the recognition of letters or words [34,35,95,96,27,28,15]. Senior et al. [163] compared normal, gamma and Poisson distributions to the original histogram of lexeme durations. They found that the Poisson distribution most closely matched the desired distribution, giving improved performance over the un-smoothed histogram. The other distributions did not perform as well.

5.3. Network traffic characterization and anomaly detection

In this application, the HSMM is applied to characterize the network traffic. Measurements of real traffic often indicate that a significant amount of variability is present in the traffic observed over a wide range of time scales, exhibiting self-similar or long range dependent characteristics [105]. Such characteristics can have a significant impact on the performance of networks and systems [177,139]. Therefore, better understanding of the nature of network traffic is critical to the proper design and implementation of servers and anomaly detection systems for network security purpose. A major advantage of using an HSMM is the capability of capturing various statistical properties of the traffic, including the long-range dependence [198]. They can also be used together with, for example, matrix-analytic methods to obtain analytically tractable solutions to queueing-theoretic models of server performance [153].

In this application, the observation sequence $\{o_t\}$ represents the number of user requests, packets, bytes, connections, etc., arriving in the tth second, or the interarrival time between requests or packets. The observation sequence is characterized as a discrete-time random process modulated by an underlying (hidden state) semi-Markov process. The hidden state represents the density of traffic or mass of active users.

Using the model parameters of the HSMM trained by the normal behavior, one can detect anomaly embedded in the network behavior according to its likelihood or entropy against the model [201,111,116,117,187,188].

5.4. Speech recognition and speech synthesis

In this application, observation vectors are obtained by extracting features from the speech signal through a spectral and/or temporal analysis. The observation vectors can be used to train the HSMMs which characterize various speech units. Speech units can be linguistically based sub-word units such as phones and acoustic units, whole word units, and units which contain a group of words. Each unit is characterized by an HSMM whose states can be considered as its distinct sounds (e.g., phonemes, syllables). A lot of applications of HSMM in speech recognition can be found in the literature, such as Levinson [107], Codogno and Fissore [39], Nakagawa and Hashimoto [127], Gu et al. [67], Ratnayake et al. [152], Hieronymus et al. [76] and Oura et al. [137]. The applications in speech synthesis can be found in Zen et al. [204], Yamagishi and Kobayashi [189,191], Yamagishi et al. [190], Yamazaki et al. [192], Tachibana et al. [169–171], Nose et al. [132–134], Moore and Savic [125] and Wu et al. [186].

To associate time with model states, a left–right model of HSMM is usually assumed. The number of states can be selected to correspond roughly to the number of sounds (e.g., phonemes) within the unit, or to the average number of observations in a spoken version of the speech unit. The output probability $b_j(v_k)$ of any state j is often represented by a mixture of Gaussian distributions with a diagonal covariance matrix [137]. The duration distribution of state j is

usually assumed as a parametric one such as Gaussian distribution [137], Poisson distribution [155], and gamma distribution [107,39]. The uniform distribution with lower and upper bounding parameters was also applied, which inhibited a state occupying too few or too many speech frames [67].

The model parameters can be trained by the re-estimation algorithms for an HSMM. But in fact in the area of speech recognition, they are often estimated using the Viterbi algorithms, as discussed in Section 4.4.1.

5.5. Functional MRI brain mapping

Applying an HSMM in functional magnetic resonance imaging (fMRI) brain mapping is to reveal components of interest in the fMRI data, as did in Faisan et al. [58,59], Thoraval [176] and Thoraval et al. [175,174]. This enables the model to automatically detect neural activation embedded in a given set of fMRI signals. It allows to enrich brain mapping with activation lag mapping, activation mode visualizing, and hemodynamic response function analysis. The three problems of HRF (the hemodynamic response function) shape variability, neural event timing, and fMRI response linearity can be solved by the model.

In this application, the sequence of hemodynamic response onsets (HROs) observed in the fMRI signal is used as the observation sequence, which is usually composed of events mixed with missing observations (null). A left–right Markov chain is selected for the hidden process of task-induced neural activations. The state index i reflects the order of appearance of the event in the chain. The chain is registered temporally with a sequence of discrete time OFF–ON blocks. The probability that the first observation at the start time t_l of state i is o_{t_l} is denoted as $b_i(o_{t_l})$. The inter-state duration distributions are specified as one-dimensional gaussians. At time t during the transition time interval from state i to state j, the probability of missing an observation is $1 - e_{ij}$ and having an observed event $o_t = e_l$ is e_{ij} with the observation probability $b_{ij}(e_l)$. All observation distributions are specified as one-dimensional gaussians.

6. Conclusions and remarks

The issues related to a general HSMM include (a) the forward–backward algorithm; (b) the computation of the predicted/filtered/smoothed probabilities, expectations and the likelihood of observations; (c) the MAP estimate of states and the MLE estimate of state sequence by extended Viterbi algorithm; (d) the parameter estimate/update and the order estimate of the model; (e) the implementation of the forward–backward algorithms.

By introducing certain assumptions and some constraints on the state transitions, the general HSMM becomes the traditional explicit duration HMM, variable transition HMM, or residential time HMM. Those conventional models have different capability in modelling applications. They also have different computational complexity and memory requirement involved in the forward–backward algorithms and the model estimate, as shown in the following table:

The model	Complexity	Mem. for F–B	Mem. for estimate
E. D. HMM	$O((M^2 + MD^2)T)$	$O(MT)$	$O(M^2 + MD + MK + MT)$
V. T. HMM	$O((M^2D)T)$	$O(MDT)$	$O(M^2D + MK + MDT)$
R. T. HMM	$O((MD + M^2)T)$	$O(MT)$	$O(M^2 + MD + MK + MT)$

where M is the number of hidden states, D the maximum duration between successive state transitions, K the number of observable values, and T the period of the observation data. However, the three models are interchangeable to each other if we express the state transition probabilities and the state duration distributions of one model with the parameters of the other one.

The state duration distributions and the observation distributions can be non-parametric or parametric dependent on the specific preference of the applications. Among the parametric distributions, the most popular ones are the exponential family distribution, such as Poisson, exponential, Gaussian, and the mixture of Gaussian distributions. The Coxian distribution of duration can represent any discrete pdf, and the underlying series-parallel network also reveals the structure of different HSMMs.

Observations are usually assumed to be dependent on states that produce them and are conditionally independent to each other for given states. However, the segmental model assumes the observations to be dependent not only on the emission state but also on the state duration.

The variants of HSMM include switching HSMM, multi-channel HSMM, and adaptive factor HSMM, which are suitable for the applications that cannot be described by a homogenous process. In considering that the duration of an HMM state is geometric, it is possible to use the algorithms of HSMMs for the HMM, which show different forms from the conventional Baum–Welch algorithm and the Viterbi algorithm.

There are three methods that can be used for overcoming the underflow problem embedded in the implementation of the forward–backward algorithms for the HSMMs. Among those three methods, the posterior notion performs best because the heuristic scaling cannot guarantee the backward variables from underflow and the logarithmic formulation cannot be very accurate based on the lookup table.

It has proved that the iterative re-estimation procedure of the model parameters using the expectations is equivalent to the EM algorithm. It maximizes the likelihood function for given observations. In a non-stationary situation, the model

parameters have to be updated online with time or with the increase of observation sequence length. Therefore, the re-estimation algorithms based on the forward–backward algorithms become unsuitable. Instead, one should use the online algorithms for the realtime update of the model parameters, which are usually done by maximizing the likelihood functions. Sometimes, to reduce the computation amount required in the re-estimation of the HSMM parameters, non-maximum likelihood estimator is used. For instance, the Viterbi algorithms or the forward–backward algorithms for the HMM are used to estimate the state duration distributions for the HSMMs. The method is splitting the states into segments.

The HSMMs have been applied in thirty areas. More and more papers are published in the literature. The number of papers published in each five years, which are cited in this survey, is listed in the following table:

Years	1980–1984	1985–1989	1990–1994	1995–1999	2000–2004	2005–2008
No. of papers	3	16	29	35	38	77

We note that there are about one hundred papers (mainly in speech recognition) that use the HSMMs but do not contribute to the theory or algorithm of the HSMMs are not cited in this paper.

Acknowledgements

We would like to acknowledge support for this project from the State Key Program of NSFC-Guangdong Joint Funds (U0735002) and The National High Technology Research and Development Program of China (2007AA01Z449).

References

[1] K. Achan, S. Roweis, A. Hertzmann, B. Frey, A segment-based probabilistic generative model of speech, in: Proc. of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 5, Philadelphia, PA, 2005, pp. 221–224.
[2] B. Ait-el-Fquih, F. Desbouvries, Kalman filtering for triplet Markov chains: Applications and extensions, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, ICASSP 05, vol. 4, Philadelphia, USA, 2005, pp. 685–688.
[3] C. Alasseur, L. Husson, F. Perez-Fontan, Simulation of rain events time series with Markov model, in: Proc. of 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2004, vol. 4, 2004, pp. 2801–2805.
[4] N.B. Amara, A. Belaid, Printed PAW recognition based on planar hidden Markov models, in: Proceedings of the 13th International Conference on Pattern Recognition, vol. 2, 1996, pp. 220–224.
[5] Y. Ariki, M.A. Jack, Enhanced time duration constraints in hidden Markov modelling for phoneme recognition, Electronics Letters 25 (13) (22 June 1989) 824–825.
[6] M. Askar, H. Derin, A recursive algorithm for the Bayes solution of the smoothing problem, IEEE Trans. Automat. Contr. AC-26 (Apr. 1981) 558–561.
[7] S.C. Austin, F. Fallside, Frame compression in hidden Markov models, in: Proc. of 1988 International Conference on Acoustics, Speech, and Signal Processing, ICASSP-88, 11–14 April 1988, pp. 477–480.
[8] Z. Aydin, Y. Altunbasak, M. Borodovsky, Protein secondary structure prediction for a single-sequence using hidden semi-Markov models, BMC Bioinformatics 7 (2006) 178. Available: http://www.biomedcentral.com/1471-2105/7/178.
[9] M. Azimi, P. Nasiopoulos, R.K. Ward, Online identification of hidden semi-Markov models, in: Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, ISPA 2003, vol. 2, 18–20 Sept. 2003, pp. 991–996.
[10] M. Azimi, P. Nasiopoulos, R.K. Ward, A new signal model and identification algorithm for hidden semi-Markov signals, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004, (ICASSP '04), vol. 2, 17–21 May 2004, pp. ii-521–ii-524.
[11] M. Azimi, P. Nasiopoulos, R.K. Ward, Offline and online identification of hidden semi-Markov models, IEEE Transactions on Signal Processing 53 (8) (Aug. 2005) 2658–2663, Part 1.
[12] K. Bae, B.K. Mallick, C.G. Elsik, Prediction of protein interdomain linker regions by a nonstationary hidden Markov model, Journal of the American Statistical Association 103 (483) (Sep. 2008) 1085–1099.
[13] L.E. Baum, T. Petrie, Statistical inference for probabilistic functions of finite state Markov chains, Ann. Math. Stat. 37 (1966) 1554–1563.
[14] E. Bechhoefer, A. Bernhard, D. He, P. Banerjee, Use of hidden semi-Markov models in the prognostics of shaft failure, in: Proceedings of the American Helicopter Society 62th Annual Forum, Phoenix, AZ, 2006. Available: http://www.vtol.org/pdf/62se.pdf.
[15] A. Benouareth, A. Ennaji, M. Sellami, Arabic handwritten word recognition using HMMs with explicit state duration, EURASIP Journal on Advances Signal Processing 2008 (2008) 1–13.
[16] R. Bippus, V. Margner, Script recognition using inhomogeneous P2DHMM and hierarchical search space reduction, in: Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999 (ICDAR '99), 20–22 Sept. 1999, pp. 773–776.
[17] A. Bonafonte, X. Ros, J.B. Marino, An efficient algorithm to find the best state sequence in HSMM, in: Proceedings of Eurospeech'93, Berlin, 1993, pp. 1547–1550.
[18] A. Bonafonte, J. Vidal, A. Nogueiras, Duration modeling with expanded HMM applied to speech recognition, in: Fourth International Conference on Spoken Language, 1996 (ICSLP '96), vol. 2, 3–6 Oct. 1996, pp. 1097–1100.
[19] M. Borodovsky, A.V. Lukashin, GeneMark.hmm: New solutions for gene finding, Nucleic Acids Res. 26 (1998) 1097–1100.
[20] E. Boutillon, W.J. Gross, P.G. Gulak, VLSI architectures for the MAP algorithm, IEEE Transactions on Communications 51 (2) (Feb. 2003) 175–185.
[21] Z. Bouyahia, L. Benyoussef, S. Derrode, Change detection in synthetic aperture radar images with a sliding hidden Markov chain model, Journal of Applied Remote Sensing 2 (2008) 023526.
[22] J. Bulla, I. Bulla, Stylized facts of financial time series and hidden semi-Markov models, Computational Statistics and Data Analysis 51 (4) (December 2006) 2192–2209.
[23] J. Bulla, I. Bulla, O. Nenadic, HSMM—An R package for analyzing hidden semi-Markov models, Computational Statistics and Data Analysis (2009).
[24] C. Burge, S. Karlin, Prediction of complete gene structures in human genomic DNA, J. Mol. Biol. 268 (1997) 78–94.
[25] D. Burshtein, Robust parametric modeling of durations in hidden Markov models, in: Proc. of 1995 International Conference on Acoustics, Speech, and Signal Processing, (ICASSP-95), vol. 1, 9–12 May 1995, p. 548.
[26] D. Burshtein, Robust parametric modeling of durations in hidden Markov models, IEEE Transactions on Speech and Audio Processing 4 (3) (May 1996) 240–242.
[27] J. Cai, Z.-Q. Liu, Integration of structural and statistical information for unconstrained handwritten numeral recognition, in: Proceedings of Fourteenth International Conference on Pattern Recognition, 1998, vol. 1, 16–20 Aug. 1998, pp. 378–380.

[28] J. Cai, Z.-Q. Liu, Integration of structural and statistical information for unconstrained handwritten numeral recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (3) (March 1999) 263–270.
[29] O. Cappe, E. Moulines, T. Ryden, Inference in Hidden Markov Models, Springer, New York, 2005.
[30] F. Chaubert-Pereira, Y. Guedon, C. Lavergne, C. Trottier, Markov and semi-Markov switching linear mixed models for identifying forest tree growth components, Research Report. Available: http://hal.ird.fr/docs/00/31/15/88/PDF/RR-6618.pdf.
[31] K. Chen, M. Hasegawa-Johnson, A. Cohen, S. Borys, S.-S. Kim, J. Cole, J.-Y. Choi, Prosody dependent speech recognition on radio news corpus of American English, IEEE Transactions on Audio, Speech, and Language Processing 14 (1) (Jan. 2006) 232–245, see also IEEE Transactions on Speech and Audio Processing.
[32] M.-Y. Chen, A. Kundu, S.N. Srihari, Handwritten word recognition using continuous density variable duration hidden Markov model, in: Proc. of 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-93), vol. 5, 27–30 April 1993, pp. 105–108.
[33] M.-Y. Chen, A. Kundu, S.N. Srihari, Variable duration hidden Markov model and morphological segmentation for handwritten word recognition, in: Proc. of 1993 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '93), 15–17 June 1993, pp. 600–601.
[34] M.-Y. Chen, A. Kundu, A complement to variable duration hidden Markov model in handwritten word recognition, in: Proceedings of IEEE International Conference on Image Processing, 1994 (ICIP-94), vol. 1, 13–16 Nov. 1994, pp. 174–178.
[35] M.Y. Chen, A. Kundu, S.N. Srihari, Variable duration hidden Markov model and morphological segmentation for handwritten word recognition, IEEE Trans. Image Processing 4 (Dec. 1995) 1675–1688.
[36] J.-T. Chien, C.-H. Huang, Bayesian learning of speech duration models, IEEE Transactions on Speech and Audio Processing 11 (6) (Nov. 2003) 558–567.
[37] J.-T. Chien, C.-H. Huang, Bayesian duration modeling and learning for speech recognition, in: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004 (ICASSP '04), vol. 1, 17–21 May 2004, pp. I-1005–I-1008.
[38] P.C. Chung, C.D. Liu, A daily behavior enabled hidden Markov model for human behavior understanding, Pattern Recognition 41 (2008) 1572–1580.
[39] M. Codogno, L. Fissore, Duration modelling in finite state automata for speech recognition and fast speaker adaptation, in: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '87), vol. 12, Apr. 1987, pp. 1269–1272.
[40] Y. Cohen, A. Erell, Y. Bistritz, Enhancement of connected words in an extremely noisy environment, IEEE Transactions on Speech and Audio Processing 5 (2) (March 1997) 141–148.
[41] A. Culotta, A. McCallum, Confidence estimation for information extraction, in: Human Language Technology Conference (HLT), 2004.
[42] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society, Series B (Methodological) 39 (1977) 1–38.
[43] L. Deng, M. Aksmanovic, X. Sun, J. Wu, Speech recognition using hidden Markov models with polynomial regression functions as nonstationary states, IEEE Transactions on Speech and Audio Processing 2 (4) (1994) 507–520.
[44] L. Deng, M. Aksmanovic, Speaker-independent phonetic classification using hidden Markov models with mixtures of trend functions, IEEE Transactions on Speech and Audio Processing 5 (4) (July 1997) 319–324.
[45] P.A. Devijver, Baum's forward–backward algorithm revisited, Pattern Recognition Letters 3 (1985) 369–373.
[46] P.M. Djuric, J.-H. Chun, Estimation of nonstationary hidden Markov models by MCMC sampling, in: 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '99), vol. 3, 15–19 March 1999, pp. 1737–1740.
[47] P.M. Djuric, J.-H. Chun, An MCMC sampling approach to estimation of nonstationary hidden Markov models, IEEE Transactions on Signal Processing 50 (5) (May 2002) 1113–1123.
[48] M. Dong, D. He, P. Banerjee, J. Keller, Equipment health diagnosis and prognosis using hidden semi-Markov models, The International Journal of Advanced Manufacturing Technology 30 (7–8) (October 2006) 738–749 (12).
[49] M. Dong, D. He, A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology, Mechanical Systems and Signal Processing 21 (5) (July 2007) 2248–2266.
[50] M. Dong, D. He, Hidden semi-Markov model-based methodology for multi-sensor equipment health diagnosis and prognosis, European Journal of Operational Research 178 (3) (May 2007) 858–878.
[51] M. Dong, A novel approach to equipment health management based on auto-regressive hidden semi-Markov model (AR-HSMM), Science in China Series F: Information Sciences 51 (9) (Sep. 2008) 1291–1304.
[52] M. Dong, D. Yang, Y. Kuang, D. He, S. Erdal, D. Kenski, PM2. 5 concentration prediction using hidden semi-Markov model-based times series data mining, Expert Systems with Applications 36 (2009) 9046–9055.
[53] J. Dumont, A.I. Hernandez, J. Fleureau, G. Carrault, Modelling temporal evolution of cardiac electrophysiological features using hidden semi-Markov models, in: Proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Aug. 2008, pp. 165–168.
[54] T.V. Duong, H.H. Bui, D.Q. Phung, S. Venkatesh, Activity recognition and abnormality detection with the switching hidden semi-Markov model, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005 (CVPR 2005), vol. 1, 20–25 June 2005, pp. 838–845.
[55] T.V. Duong, D.Q. Phung, H.H. Bui, S. Venkatesh, Efficient Coxian duration modelling for activity recognition in smart environments with the hidden semi-Markov model, in: Proceedings of the 2005 International Conference on Intelligent Sensors, Sensor Networks and Information Processing Conference, 2005, 5–8 Dec. 2005, pp. 277–282.
[56] T.V. Duong, D.Q. Phung, H.H. Bui, S. Venkatesh, Human behavior recognition with generic exponential family duration modeling in the hidden semi-Markov model, in: Proc. of 18th International Conference on Pattern Recognition, 2006 (ICPR 2006), vol. 3, 20–24 Aug. 2006, pp. 202–207.
[57] Y. Ephraim, N. Merhav, Hidden Markov processes, IEEE Trans. Information Theory 48 (6) (June 2002) 1518–1569.
[58] S. Faisan, L. Thoraval, J.-P. Armspach, F. Heitz, Hidden semi-Markov event sequence models: Application to brain functional MRI sequence analysis, in: Proceedings of 2002 International Conference on Image Processing, vol. 1, 22–25 Sept. 2002, pp. I-880–I-883.
[59] S. Faisan, L. Thoraval, J.-P. Armspach, M.-N. Metz-Lutz, F. Heitz, Unsupervised learning and mapping of active brain functional MRI signals based on hidden semi-Markov event sequence models, IEEE Transactions on Medical Imaging 24 (2) (Feb. 2005) 263–276.
[60] J.D. Ferguson, Variable duration models for speech, in: Symp. Application of Hidden Markov Models to Text and Speech, Institute for Defense Analyses, Princeton, NJ, Oct. 1980, pp. 143–179.
[61] L. Finesso, Consistent estimation of the order for Markov and hidden Markov chains, Ph.D. dissertation, Univ. Maryland, College Park, 1990.
[62] J. Ford, V. Krishnamurthy, J.B. Moore, Adaptive estimation of hidden semi-Markov chains with parameterised transition probabilities and exponential decaying states, in: Proc. of Conf. on Intell. Signal Processing and Communication Systems (ISPACS), Sendai, Japan, Oct. 1993, pp. 88–92.
[63] M. Gales, S. Young, The theory of segmental hidden Markov models, Technical Report CUED/F-INFENG/TR 133, Cambridge University, Engineering Department, 1993.
[64] X. Ge, P. Smyth, Deformable Markov model templates for time-series pattern matching, in: Proc. of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, 2000, pp. 81–90.
[65] X. Ge, P. Smyth, Segmental semi-Markov models for change-point detection with applications to semiconductor manufacturing, Technical Report UCI-ICS 00-08, http://www.ics.uci.edu/~datalab/papers/trchange.pdf, March 2000, http://citeseer.ist.psu.edu/ge00segmental.html.
[66] Z. Ghahramani, An introduction to hidden Markov models and Bayesian networks, International Journal of Pattern Recognition and Artificial Intelligence 15 (1) (2001) 9–42.
[67] H.-Y. Gu, C.-Y. Tseng, L.-S. Lee, Isolated-utterance speech recognition using hidden Markov models with bounded state durations, IEEE Transactions on Signal Processing 39 (8) (Aug. 1991) 1743–1752, see also IEEE Transactions on Acoustics, Speech, and Signal Processing.

[68] Y. Guedon, C. Cocozza-Thivent, Use of the Derin's algorithm in hidden semi-Markov models for automatic speech recognition, in: Proc. of 1989 International Conference on Acoustics, Speech, and Signal Processing (ICASSP-89), 23–26 May 1989, pp. 282–285.
[69] Y. Guedon, D. Barthelemy, Y. Caraglio, E. Costes, Pattern analysis in branching and axillary flowering sequences, Journal of Theoretical Biology 212 (4) (Oct. 2001) 481–520.
[70] Y. Guedon, Estimating hidden semi-Markov chains from discrete sequences, Journal of Computational and Graphical Statistics 12 (3) (2003) 604–639.
[71] Y. Guedon, Hidden hybrid Markov/semi-Markov chains, Computational Statistics and Data Analysis 49 (3) (June 2005) 663–688.
[72] Y. Guedon, Exploring the state sequence space for hidden Markov and semi-Markov chains, Computational Statistics and Data Analysis 51 (5) (Feb. 2007) 2379–2409.
[73] T. Hanazawa, K. Kita, S. Nakamura, T. Kawabata, K. Shikano, ATR HMM-LR continuous speech recognition system, in: Proc. of 1990 International Conference on Acoustics, Speech, and Signal Processing, 1990, ICASSP-90, 3–6 April 1990, pp. 53–56.
[74] J. He, H. Leich, A unified way in incorporating segmental feature and segmental model into HMM, in: Proc. of 1995 International Conference on Acoustics, Speech, and Signal Processing, 1995, ICASSP-95, vol. 1, 9–12 May 1995, pp. 532–535.
[75] H. He, S. Wu, P. Banerjee, E. Bechhoefer, Probabilistic model based algorithms for prognostics, in: Proc. of 2006 IEEE Aerospace Conference, 4–11 March 2006.
[76] J.L. Hieronymus, D. McKelvie, F. McInnes, Use of acoustic sentence level and lexical stress in HSMM speech recognition, in: Proc. of 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1992, ICASSP-92, vol. 1, 23–26 March 1992, pp. 225–227.
[77] W.J. Holmes, M.J. Russell, Experimental evaluation of segmental HMMs, in: Proc. of International Conference on Acoustics, Speech, and Signal Processing, 1995, ICASSP-95, vol. 1, 9–12 May 1995, pp. 536–539.
[78] W.J. Holmes, M.J. Russell, Probabilistic-trajectory segmental HMMs, Computer Speech and Language 13 (1) (1999) 3–37.
[79] S. Hongeng, R. Nevatia, Large-scale event detection using semi-hidden Markov models, in: Proceedings of Ninth IEEE International Conference on Computer Vision, 13–16 Oct. 2003, pp. 1455–1462.
[80] S. Hongeng, R. Nevatia, F. Bremond, Video-based event recognition: Activity representation and probabilistic methods, Comp. Vis. and Image Understanding 96 (2004) 129–162.
[81] J. Hu, R. Kashi, G. Wilfong, Document classification using layout analysis, in: Proc. of First Intl. Workshop on Document Analysis and Understanding for Document Databases, Florence, Italy, September 1999.
[82] J. Hu, R. Kashi, G. Wilfong, Comparison and classification of documents based on layout similarity, Information Retrieval 2 (2) (May 2000) 227–243.
[83] X.D. Huang, Phoneme classification using semicontinuous hidden Markov models, IEEE Transactions on Signal Processing 40 (5) (May 1992) 1062–1067, see also IEEE Transactions on Acoustics, Speech, and Signal Processing.
[84] N.P. Hughes, S.J. Roberts, L. Tarassenko, Semi-supervised learning of probabilistic models for ECG segmentation, in: Proc. of 26th Annual International Conference of the Engineering in Medicine and Biology Society, 2004, EMBC 2004, vol. 1, 2004, pp. 434–437.
[85] N.P. Hughes, L. Tarassenko, S.J. Roberts, Markov models for automated ECG interval analysis, Advances in Neural Information Processing Systems (2003). Available: http://citeseer.ist.psu.edu/hughes03markov.html.
[86] M.T. Johnson, Capacity and complexity of HMM duration modeling techniques, IEEE Signal Processing Letters 12 (5) (May 2005) 407–410.
[87] S. Katagiri, C.-H. Lee, A new hybrid algorithm for speech recognition based on HMM segmentation and learning vector quantization, IEEE Transactions on Speech and Audio Processing 1 (4) (Oct. 1993) 421–430.
[88] C.C. Ke, J. Llinas, Literature survey on ground target tracking problems, Research Project Report, Center for Multisource Information Fusion, State University of New York at Buffalo, 1999.
[89] S. Kim, P. Smyth, Segmental hidden Markov models with random effects for waveform modeling, Journal of Machine Learning Research 7 (2006) 945–969.
[90] W.-G. Kim, J.-Y. Yoon, D.H. Youn, HMM with global path constraint in Viterbi decoding for isolated word recognition, in: Proc. ICASSP 1994, 1994, pp. 605–608.
[91] H. Kobayashi, S.-Z. Yu, Hidden semi-Markov models and efficient forward–backward algorithms, in: 2007 Hawaii and SITA Joint Conference on Information Theory, Honolulu, Hawaii, 29–31 May 2007, pp. 41–46.
[92] V. Krishnamurthy, J.B. Moore, Signal processing of semi-Markov models with exponentially decaying states, in: Proceedings of the 30th Conference on Decision and Control, Brighton, England, Dec. 1991, pp. 2744–2749.
[93] V. Krishnamurthy, J.B. Moore, S.H. Chung, Hidden fractal model signal processing, Signal Processing 24 (2) (Aug. 1991) 177–192.
[94] D. Kulp, D. Haussler, M.G. Reese, F.H. Eeckman, A generalized hidden Markov model for the recognition of human genes in DNA, in: Proc. 4th Int. Conf. Intell. Syst. Molecular Bio., 1996, pp. 134–142.
[95] A. Kundu, Y. He, M.-Y. Chen, Efficient utilization of variable duration information in HMM based HWR systems, in: Proceedings of International Conference on Image Processing, 1997, vol. 3, 26–29 Oct. 1997, pp. 304–307.
[96] A. Kundu, Y. He, M.-Y. Chen, Alternatives to variable duration HMM in handwriting recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (11) (Nov. 1998) 1275–1280.
[97] O.W. Kwon, C.K. Un, Context-dependent word duration modelling for Korean connected digit recognition, Electronics Letters 31 (19) (Sept. 1995) 1630–1631.
[98] P. Lanchantin, W. Pieczynski, Unsupervised nonstationary image segmentation using triplet Markov chains, in: Proc. of Advanced Concepts for Intelligent Vision Systems (ACVIS 04), Brussels, Belgium, Aug. 31–Sept. 3, 2004.
[99] P. Lanchantin, W. Pieczynski, Unsupervised restoration of hidden nonstationary Markov chain using evidential priors, IEEE Transactions on Signal Processing 53 (8) (2005) 3091–3098.
[100] P. Lanchantin, J. Lapuyade-Lahorguea, W. Pieczynski, Unsupervised segmentation of triplet Markov chains hidden with long-memory noise, Signal Processing 88 (5) (May 2008) 1134–1151.
[101] J. Lapuyade-Lahorgue, W. Pieczynski, Unsupervised segmentation of hidden semi-Markov non-stationary chains, in: Twenty Sixth International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, MaxEnt 2006, Paris, France, 8–13 July 2006.
[102] K. Laurila, Noise robust speech recognition with state duration constraints, in: Proc. of 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1997, ICASSP-97, vol. 2, 21–24 April 1997, pp. 871–874.
[103] C.-H. Lee, F.K. Soong, B.-H. Juang, A segment model based approach to speech recognition, in: Proc. Int'l. Conf. on Acoust., Speech and Signal Processing, 1988, pp. 501–504.
[104] C.-H. Lee, L.R. Rabiner, A frame-synchronous network search algorithm for connected word recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing 37 (11) (Nov. 1989) 1649–1658, see also IEEE Transactions on Signal Processing.
[105] W. Leland, M. Taqqu, W. Willinger, D. Wilson, On the self-similar nature of Ethernet traffic (extended version), IEEE/ACM Transactions on Networking 2 (1) (February 1994) 1–15.
[106] S.E. Levinson, L.R. Rabiner, M.M. Sondhi, An introduction to the application of the theory of probabilistic functions of a Markov process in automatic speech recognition, B.S.T.J. 62 (1983) 1035–1074.
[107] S.E. Levinson, Continuously variable duration hidden Markov models for automatic speech recognition, Computer Speech and Language 1 (1) (1986) 29–45.

[108] S.E. Levinson, Continuously variable duration hidden Markov models for speech analysis, in: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '86., vol. 11, Apr. 1986, pp. 1241–1244.
[109] S.E. Levinson, A. Ljolje, L.G. Miller, Large vocabulary speech recognition using a hidden Markov model for acoustic/phonetic classification, in: Proc. of 1988 International Conference on Acoustics, Speech, and Signal Processing, ICASSP-88, 11–14 April 1988, pp. 505–508.
[110] S.E. Levinson, M.Y. Liberman, A. Ljolje, L.G. Miller, Speaker independent phonetic transcription of fluent speech for large vocabulary speech recognition, in: Proc. of 1989 International Conference on Acoustics, Speech, and Signal Processing, ICASSP-89, 23–26 May 1989, pp. 441–444.
[111] M. Li, S.-Z. Yu, A network-wide traffic anomaly detection method based on HSMM, in: Proc. of 2006 International Conference on Communications, Circuits and Systems Proceedings, vol. 3, June 2006, pp. 1636–1640.
[112] H.-P. Lin, M.-J. Tseng, F.-S. Tsai, A non-stationary hidden Markov model for satellite propagation channel modeling, in: Proceedings of 2002 IEEE 56th Vehicular Technology Conference, VTC 2002-Fall, vol. 4, 24–28 Sept. 2002, pp. 2485–2488.
[113] X.B. Liu, D.S. Yang, X.O. Chen, New approach to classification of Chinese folk music based on extension of HMM, in: International Conference on Audio, Language and Image Processing, ICALIP 2008, 7–9 July 2008, pp. 1172–1179.
[114] Z. Liu, J.X. Yu, L. Chen, D. Wu, Detection of shape anomalies: A probabilistic approach using hidden Markov models, in: IEEE 24th International Conference on Data Engineering, ICDE 2008, 7–12 April 2008, pp. 1325–1327.
[115] A. Ljolje, S.E. Levinson, Development of an acoustic–phonetic hidden Markov model for continuous speech recognition, IEEE Transactions on Signal Processing 39 (1) (Jan. 1991) 29–39, see also IEEE Transactions on Acoustics, Speech, and Signal Processing.
[116] W.-Z. Lu, S.-Z. Yu, An HTTP flooding detection method based on browser behavior, in: Proc. of 2006 International Conference on Computational Intelligence and Security, vol. 2, Nov. 2006, pp. 1151–1154.
[117] W.-Z. Lu, S.-Z. Yu, Clustering web traffic of request bursts, in: Proc. of 2006 IEEE Region 10 Conference on Communications, TENCON 2006, Nov. 2006, pp. 1–4.
[118] E. Marcheret, M. Savic, Random walk theory applied to language identification, in: Proc. of 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-97, vol. 2, 21–24 April 1997, pp. 1119–1122.
[119] E. Marhasev, M. Hadad, G.A. Kaminka, Non-stationary hidden semi-Markov models in activity recognition, in: Proceedings of the AAAI Workshop on Modeling Others from Observations (MOO-06), 2006.
[120] B.L. Mark, Z.R. Zaidi, Robust mobility tracking for cellular networks, in: Proc. of IEEE International Conference on Communications, 2002, ICC 2002, vol. 1, April 28–May 2, 2002, pp. 445–449.
[121] G.J. McLachlan, T. Krishnan, The EM Algorithm and Extensions, 2nd ed., Wiley, New York, 2008.
[122] C.D. Mitchell, R.A. Helzerman, L.H. Jamieson, M.P. Harper, A parallel implementation of a hidden Markov model with duration modeling for speech recognition, in: Proceedings of the Fifth IEEE Symposium on Parallel and Distributed Processing, 1993, 1–4 Dec. 1993, pp. 298–306.
[123] C. Mitchell and L. Jamieson, Modeling duration in a hidden Markov model with the exponential family, in: Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-93, 1993, pp. 331–334.
[124] C. Mitchell, M. Harper, L. Jamieson, On the complexity of explicit duration HMMs, IEEE Transactions on Speech and Audio Processing 3 (2) (May 1995) 213–217.
[125] M.D. Moore, M.I. Savic, Speech reconstruction using a generalized HSMM (GHSMM), Digital Signal Processing 14 (1) (2004) 37–53.
[126] K.P. Murphy, Hidden semi-Markov models (HSMMs), http://www.ai.mit.edu/murphyk, Nov. 2002.
[127] S. Nakagawa, Y. Hashimoto, A method for continuous speech segmentation using HMM, in: Proc. of 9th International Conference on Pattern Recognition, vol. 2, 14–17 Nov. 1988, pp. 960–962.
[128] P. Natarajan, R. Nevatia, Hierarchical multi-channel hidden semi-Markov models, in: The Twentieth International Joint Conference on Artificial Intelligence, Hyderabad, India, Jan. 2007, pp. 2562–2567.
[129] P. Natarajan, R. Nevatia, Coupled hidden semi-Markov models for activity recognition, in: IEEE Workshop on Motion and Video Computing, 2007, WMVC '07, Feb. 2007.
[130] P. Natarajan, R. Nevatia, Online, real-time tracking and recognition of human actions, in: IEEE Workshop on Motion and Video Computing, WMVC 2008, 8–9 Jan. 2008, pp. 1–8.
[131] N. Niwase, J. Yamagishi, T. Kobayashi, Human walking motion synthesis with desired pace and stride length based on HSMM, IEICE Transactions on Information and Systems (2005).
[132] T. Nose, J. Yamagishi, T. Kobayashi, A style control technique for speech synthesis using multiple regression HSMM, in: Proc. INTERSPEECH 2006-ICSLP, Sept. 2006, pp. 1324–1327.
[133] T. Nose, J. Yamagishi, T. Masuko, T. Kobayashi, A style control technique for HMM-based expressive speech synthesis, IEICE Transactions on Information and Systems E90-D (9) (Sept. 2007) 1406–1413.
[134] T. Nose, Y. Kato, T. Kobayashi, A speaker adaptation technique for MRHSMM-based style control of synthetic speech, in: Proc. ICASSP 2007, vol. IV, Apr. 2007, pp. 833–836.
[135] M. Ostendorf, S. Roukos, A stochastic segment model for phoneme-based continuous speech recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing 37 (12) (Dec. 1989) 1857–1869, see also IEEE Transactions on Signal Processing.
[136] M. Ostendorf, V.V. Digalakis, O.A. Kimball, From HMM's to segment models: A unified view of stochastic modeling for speech recognition, IEEE Transactions on Speech and Audio Processing 4 (5) (Sep. 1996) 360–378.
[137] K. Oura, H. Zen, Y. Nankaku, A. Lee, K. Tokuda, Hidden semi-Markov model based speech recognition system using weighted finite-state transducer, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, 2006, ICASSP 2006, vol. 1, 14–19 May 2006, pp. I-33–I-36.
[138] Y.K. Park, C.K. Un, O.W. Kwon, Modeling acoustic transitions in speech by modified hidden Markov models with state duration and state duration-dependent observation probabilities, IEEE Transactions on Speech and Audio Processing 4 (5) (Sept. 1996) 389–392.
[139] K. Park, G.T. Kim, M.E. Crovella, On the effect of traffic self-similarity on network performance, in: Proceedings of SPIE International Conference on Performance and Control of Network Systems, November 1997, pp. 296–310.
[140] M. Pavel, T.L. Hayes, A. Adami, H.B. Jimison, J. Kaye, Unobtrusive assessment of mobility, in: 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, New York, NY, 30 August–3 September 2006.
[141] G. Peng, B. Zhang, W. S.-Y. Wang, Performance of mandarin connected digit recognizer with word duration modeling, in: ASR2000 – Automatic Speech Recognition: Challenges for the New Millenium, Paris, France, 18–20 Sep. 2000, pp. 140–144.
[142] D. Phung, T. Duong, H. Bui, S. Venkatesh, Activity recognition and abnormality detection with the switching hidden semi-Markov model, in: Int. Conf. on Comp. Vis. & Pat. Recog, 2005.
[143] D.Q. Phung, T.V. Duong, S. Venkatesh, H.H. Bui, Topic transition detection using hierarchical hidden Markov and semi-Markov models, in: Proceedings of the 13th Annual ACM International Conference, 2005, pp. 11–20.
[144] W. Pieczynski, C. Hulard, T. Veit, Triplet Markov chains in hidden signal restoration, in: SPIE's International Symposium on Remote Sensing, Crete, Greece, 22–27 September 2002.
[145] W. Pieczynski, Modeling nonstationary hidden semi-Markov chains with triplet Markov chains and theory of evidence, in: 2005 IEEE/SP 13th Workshop on Statistical Signal Processing, 17–20 July 2005, pp. 727–732.
[146] W. Pieczynski, F. Desbouvries, On triplet Markov chains, in: International Symposium on Applied Stochastic Models and Data Analysis (ASMDA 2005), Brest, France, May 2005.

[147] W. Pieczynski, Multisensor triplet Markov chains and theory of evidence, International Journal of Approximate Reasoning 45 (1) (May 2007) 1–16.
[148] A. Pikrakis, S. Theodoridis, D. Kamarotos, Classification of musical patterns using variable duration hidden Markov models, IEEE Transactions on Audio, Speech, and Language Processing 14 (5) (Sept. 2006) 1795–1807, see also IEEE Transactions on Speech and Audio Processing.
[149] J.A. du Preez, Modelling durations in hidden Markov models with application to word spotting, in: Proceedings of South African Symposium on Communications and Signal Processing, 1991, COMSIG 1991, 30 Aug. 1991, pp. 1–5.
[150] L.R. Rabiner, A tutorial on hidden Markov models and selected application in speech recognition, Proceedings of the IEEE 77 (2) (Feb. 1989) 257–286.
[151] P. Ramesh, J.G. Wilpon, Modeling state durations in hidden Markov models for automatic speech recognition, in: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92, vol. 1, 23–26 March 1992, pp. 381–384.
[152] N. Ratnayake, M. Savic, J. Sorensen, Use of semi-Markov models for speaker-independent phoneme recognition, in: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92, vol. 1, 23–26 March 1992, pp. 565–568.
[153] A. Riska, M. Squillante, S.-Z. Yu, Z. Liu, L. Zhang, Matrix-analytic analysis of a MAP/PH/1 queue fitted to web server data, in: G. Latouche, P. Taylor (Eds.), Fourth International Conference on Matrix Analytic Methods in Stochastic Models, Matrix-Analytic Methods: Theory and Applications, World Scientific, Adelaide, Australia, July 2002, pp. 333–356.
[154] M.J. Russell, Reducing computational load in segmental hidden Markov model decoding for speech recognition, Electronics Letters 41 (25) (Dec. 2005) 1408–1409.
[155] M.J. Russell, R.K. Moore, Explicit modelling of state occupancy in hidden Markov models for automatic speech recognition, in: Proc. IEEE Int. Conf. Acoust. Speech Signal Processing, vol. 10, Apr. 1985, pp. 5–8.
[156] M.J. Russell, A. Cook, Experimental evaluation of duration modelling techniques for automatic speech recognition, in: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1987, pp. 2376–2379.
[157] M.J. Russell, A segmental HMM for speech pattern modelling, in: 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-93, vol. 2, 27–30 April 1993, pp. 499–502.
[158] F. Salzenstein, C. Collet, S. Lecam, M. Hatt, Non-stationary fuzzy Markov chain, Pattern Recognition Letters 28 (16) (Dec. 2007) 2201–2208.
[159] J. Sansom, P. Thomson, Fitting hidden semi-Markov models to breakpoint rainfall data, J. Appl. Probab. A 38 (2001) (2001) 142–157.
[160] J. Sansom, C.S. Thompson, Spatial and temporal variation of rainfall over New Zealand, J. Geophys. Res. D 113 (6) (Apr. 2008).
[161] S. Sarawagi, W.W. Cohen, Semi-Markov conditional random fields for information extraction, in: Advances in Neural Information Processing Systems, vol. 17, NIPS, 2004.
[162] S.C. Schmidler, J.S. Liu, D.L. Brutlag, Bayesian segmentation of protein secondary structure, J. Comp. Biol. 7 (2000) 233–248.
[163] A. Senior, J. Subrahmonia, K. Nathan, Duration modeling results for an on-line handwriting recognizer, in: Proceedings of 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-96, vol. 6, 7–10 May 1996, pp. 3482–3485.
[164] B. Sin, J.H. Kim, Nonstationary hidden Markov model, Signal Processing 46 (1995) 31–46.
[165] R.N.V. Sitaram, T.V. Sreenivas, Phoneme recognition in continuous speech using large inhomogeneous hidden Markov models, in: 1994 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-94, vol. i, 19–22 April 1994, pp. I/41–I/44.
[166] R. Sitaram, T. Sreenivas, Connected phoneme HMMs with implicit duration modelling for better speech recognition, in: Proceedings of 1997 International Conference on Information, Communications and Signal Processing, ICICS 1997, 9–12 Sept. 1997, pp. 1024–1028.
[167] K. Squire, HMM-based semantic learning for a mobile robot, Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2004. Available: http://www.ifp.uiuc.edu/~k-squire/thesis/Kevin_thesis_full.pdf.
[168] K. Squire, S.E. Levinson, Recursive maximum likelihood estimation for hidden semi-Markov models, in: 2005 IEEE Workshop on Machine Learning for Signal Processing, 28–30 Sept. 2005, pp. 329–334.
[169] M. Tachibana, J. Yamagishi, T. Masuko, T. Kobayashi, Performance evaluation of style adaptation for hidden semi-Markov model based speech synthesis, in: INTERSPEECH-2005, 2005, pp. 2805–2808.
[170] M. Tachibana, J. Yamagishi, T. Masuko, T. Kobayashi, A style adaptation technique for speech synthesis using HSMM and suprasegmental features, IEICE Transactions on Information and Systems E89-D (3) (2006) 1092–1099.
[171] M. Tachibana, S. Izawa, T. Nose, T. Kobayashi, Speaker and style adaptation using average voice model for style control in hmm-based speech synthesis, in: Proc. ICASSP 2008, pp. 4633–4636.
[172] X.R. Tan, H.S. Xi, Hidden semi-Markov model for anomaly detection, Applied Mathematics and Computation 205 (2008) 562–567.
[173] V. Ter-Hovhannisyan, Unsupervised and semi-supervised training methods for eukaryotic gene prediction, Ph.D. dissertation, Georgia Institute of Technology, 2008.
[174] L. Thoraval, G. Carrault, F. Mora, Continuously variable duration hidden Markov models for ECG segmentation, in: Proceedings of the Annual International Conference of the IEEE, vol. 2, Oct. 29–Nov. 1, 1992, in: Engineering in Medicine and Biology Society, vol. 14, 1992, pp. 529–530.
[175] L. Thoraval, G. Carrault, J.J. Bellanger, Heart signal recognition by hidden Markov models: The ECG case, Meth. Inform. Med. 33 (1994) 10–14.
[176] L. Thoraval, Technical Report: Hidden semi-Markov event sequence models, 2002. Available: http://picabia.ustrasbg.fr/lsiit/perso/thoraval.htm.
[177] T. Tuan, K. Park, Multiple time scale congestion control for self-similar network traffic, Performance Evaluation 36 (1999) 359–386.
[178] D. Tweed, R. Fisher, J. Bins, T. List, Efficient hidden semi-Markov model inference for structured video sequences, in: 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005, 15–16 Oct. 2005 pp. 247–254.
[179] S.V. Vaseghi, Hidden Markov models with duration-dependent state transition probabilities, Electronics Letters 27 (8) (April 1991) 625–626.
[180] S.V. Vaseghi, P. Conner, On increasing structural complexity of finite state speech models, in: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92, vol. 1, 23–26 March 1992, pp. 537–540.
[181] S.V. Vaseghi, State duration modeling in hidden Markov models, Signal Processing 41 (1) (1995) 31–41.
[182] X. Wang, Durationally constrained training of hmm without explicit state durational pdf, in: Proceedings of the Institute of Phonetic Sciences, vol. 18, University of Amsterdam, 1994, pp. 111–130.
[183] X. Wang, L.F.M. ten Bosch, L.C.W. Pols, Integration of context-dependent durational knowledge into HMM-based speech recognition, in: Proceedings of Fourth International Conference on Spoken Language, 1996, ICSLP 96, vol. 2, 3–6 Oct. 1996, pp. 1073–1076.
[184] J.B. Wang, V. Athitsos, S. Sclaroff, M. Betke, Detecting objects of variable shape structure with hidden state shape models, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (3) (March 2008) 477–492.
[185] C. Wellington, A. Courville, A. Stentz, Interacting Markov random fields for simultaneous terrain modeling and obstacle detection, in: Proceedings of Robotics: Science and Systems, 2005.
[186] C.-H. Wu, C.-C. Hsia, T.-H. Liu, J.-F. Wang, Voice conversion using duration-embedded bi-HMMs for expressive speech synthesis, IEEE Transactions on Audio, Speech, and Language Processing 14 (4) (July 2006) 1109–1116, see also IEEE Transactions on Speech and Audio Processing.
[187] Y. Xie, S.-Z. Yu, A dynamic anomaly detection model for web user behavior based on HsMM, in: 10th International Conference on Computer Supported Cooperative Work in Design, May 2006, pp. 1–6.
[188] Y. Xie, S.-Z. Yu, A novel model for detecting application layer DDoS attacks, in: First International Multi-Symposiums on Computer and Computational Sciences, IMSCCS '06, vol. 2, 20–24 April 2006, pp. 56–63.
[189] J. Yamagishi, T. Kobayashi, Adaptive training for hidden semi-Markov model, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005 (ICASSP '05), vol. 1, 18–23 March 2005, pp. 365–368.

[190] J. Yamagishi, K. Ogata, Y. Nakano, J. Isogai, T. Kobayashi, HSMM-based model adaptation algorithms for average-voice-based speech synthesis, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2006, ICASSP 2006, vol. 1, 14–19 May 2006, pp. I-77–I-80.
[191] J. Yamagishi, T. Kobayashi, Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training, IEICE Transactions on Information and Systems E90-D (2) (2007) 533–543.
[192] T. Yamazaki, N. Niwase, J. Yamagishi, T. Kobayashi, Human walking motion synthesis based on multiple regression hidden semi-Markov model, in: International Conference on Cyberworlds, 23–25 Nov. 2005.
[193] P. Yang, G. Dumont, J.M. Ansermino, An adaptive Cusum test based on a hidden semi-Markov model for change detection in non-invasive mean blood pressure trend, in: Proceedings of the 28th IEEE EMBS Annual International Conference, New York City, USA, Aug. 30–Sept. 3 2006, pp. 3395–3398.
[194] N.B. Yoma, F.R. McInnes, M.A. Jack, Weighted Viterbi algorithm and state duration modelling for speech recognition in noise, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 1998, ICASSP '98, vol. 2, 12–15 May 1998, pp. 709–712.
[195] N.B. Yoma, F.R. McInnes, M.A. Jack, S.D. Stump, L.L. Ling, On including temporal constraints in Viterbi alignment for speech recognition in noise, IEEE Transactions on Speech and Audio Processing 9 (2) (Feb. 2001) 179–182.
[196] N.B. Yoma, J.S. Sanchez, MAP speaker adaptation of state duration distributions for speech recognition, IEEE Transactions on Speech and Audio Processing 10 (7) (Oct. 2002) 443–450.
[197] S.-Z. Yu, B. L. Mark, H. Kobayashi, Mobility tracking and traffic characterization for efficient wireless internet access, in: IEEE MMT'2000, Multiaccess, Mobility and Teletraffic in Wireless Communications, vol. 5, Duck Key, Florida, 3–6 Dec. 2000, pp. 279–290.
[198] S.-Z. Yu, Z. Liu, M. Squillante, C. Xia, L. Zhang, A hidden semi-Markov model for web workload self-similarity, in: 21st IEEE International Performance, Computing, and Communications Conference, IPCCC 2002, Phoenix, Arizona, 3–5 April 2002, pp. 65–72.
[199] S.-Z. Yu, H. Kobayashi, An efficient forward–backward algorithm for an explicit duration hidden Markov model, IEEE Signal Processing Letters 10 (1) (Jan. 2003) 11–14.
[200] S.-Z. Yu, H. Kobayashi, A hidden semi-Markov model with missing data and multiple observation sequences for mobility tracking, Signal Processing 83 (2) (Feb. 2003) 235–250.
[201] S.-Z. Yu, Multiple tracking based anomaly detection of mobile nodes, in: 2nd International Conference on Mobile Technology, Applications and Systems, 2005, 15–17 Nov. 2005, pp. 5–9.
[202] S.-Z. Yu, H. Kobayashi, Practical implementation of an efficient forward–backward algorithm for an explicit-duration hidden Markov model, IEEE Transactions on Signal Processing 54 (5) (January 2006) 1947–1951.
[203] Y.-S. Yun, Y.-H. Oh, A segmental-feature HMM for speech pattern modeling, IEEE Signal Processing Letters 7 (6) (2000) 135–137.
[204] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, T. Kitamura, Hidden semi-Markov model based speech synthesis, in: Proc. of 8th International Conference on Spoken Language Processing, ICSLP, Jeju Island, Korea, 4–8 Oct. 2004, pp. 1393–1396.
[205] W. Zhang, F. Chen, W. Xu, E. Zhang, Real-time video intelligent surveillance system, in: 2006 IEEE International Conference on Multimedia and Expo, July 2006, pp. 1021–1024.
[206] W. Zhang, F. Chen, W. Xu, Y. Du, Learning human activity containing sparse irrelevant events in long sequence, in: 2008 Congress on Image and Signal Processing, CISP'08, 2008, pp. 211–215.