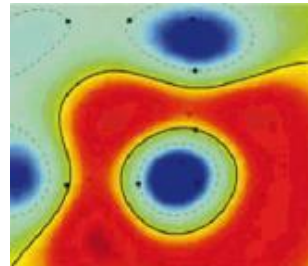


Hidden Markov Models

An Introduction



Markov Chain

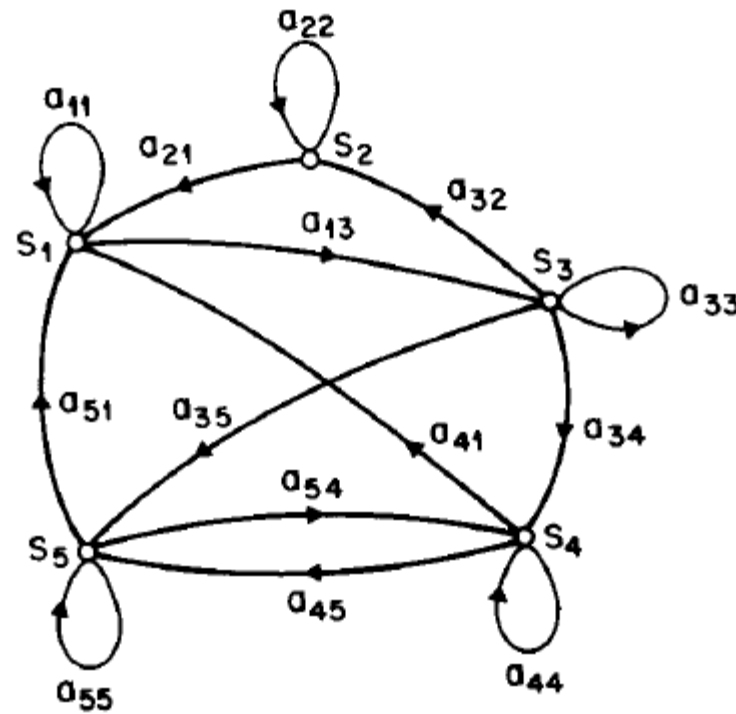


Fig. 1. A Markov chain with 5 states (labeled S_1 to S_5) with selected state transitions.

$$\begin{aligned}
 P[q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \dots] \\
 = P[q_t = S_j | q_{t-1} = S_i].
 \end{aligned}
 \tag{1}$$

Transition probabilities

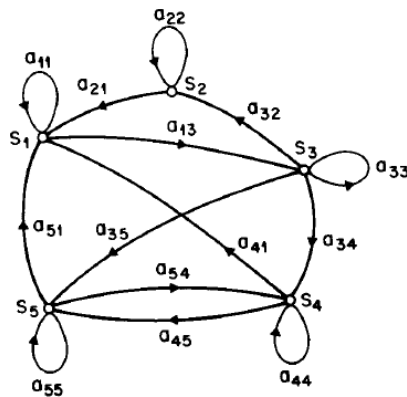


Fig. 1. A Markov chain with 5 states (labeled S_1 to S_5) with selected state transitions.

$$P[q_t = S_j | q_{t-1} = S_i], \quad 1 \leq i, j \leq N \tag{2}$$

$$a_{ij} \geq 0 \tag{3a}$$

$$\sum_{j=1}^N a_{ij} = 1 \tag{3b}$$

Example: weather I

State 1: rain or (snow)

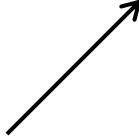
State 2: cloudy

State 3: sunny.

State Transitions

$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

$P(\text{sun-sun-rain-rain-sun-cloudy-sun})=?$

Observation 

$$\begin{aligned} P(O|\text{Model}) &= P[S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3|\text{Model}] \\ &= P[S_3] \cdot P[S_3|S_3] \cdot P[S_3|S_3] \cdot P[S_1|S_3] \\ &\quad \cdot P[S_1|S_1] \cdot P[S_3|S_1] \cdot P[S_2|S_3] \cdot P[S_3|S_2] \\ &= \pi_3 \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23} \\ &= 1 \cdot (0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2) \\ &= 1.536 \times 10^{-4} \end{aligned}$$

$$\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N \quad (4)$$

Example: weather II

Same weather for d days

$$O = \{S_{i,1}, S_{i,2}, S_{i,3}, \dots, S_{i,d}, S_{i,d+1} \neq S_{i,d}\},$$

$$P(O|\text{Model}, q_1 = S_i) = (a_{ii})^{d-1}(1 - a_{ii}) = p_i(d). \quad (5)$$

$$\begin{aligned} \bar{d}_i &= \sum_{d=1}^{\infty} d p_i(d) \\ &= \sum_{d=1}^{\infty} d (a_{ii})^{d-1} (1 - a_{ii}) = \frac{1}{1 - a_{ii}}. \end{aligned}$$

Expected number of consecutive days for sunny $1/0.2 = 5$, cloudy 2.5, rainy 1.67

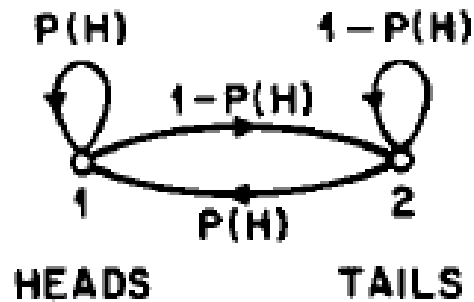


Hidden Markov Models

Coin toss experiment

$$\begin{aligned} \mathbf{O} &= O_1 O_2 O_3 \cdots O_T \\ &= \mathcal{H} \mathcal{H} \mathcal{T} \mathcal{T} \mathcal{T} \mathcal{H} \mathcal{T} \mathcal{T} \mathcal{H} \cdots \mathcal{H} \end{aligned}$$

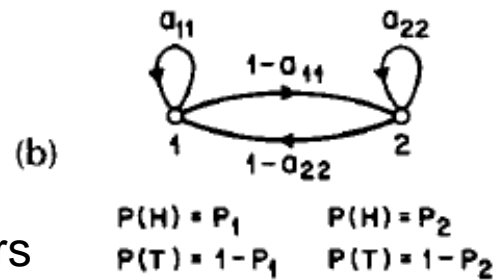
One solution



$$\begin{aligned} \mathbf{O} &= \mathcal{H} \mathcal{H} \mathcal{T} \mathcal{T} \mathcal{T} \mathcal{H} \mathcal{T} \mathcal{T} \mathcal{H} \dots \\ \mathbf{S} &= 1 \ 1 \ 2 \ 2 \ 1 \ 2 \ 1 \ 1 \ 2 \ 2 \ 1 \dots \end{aligned}$$

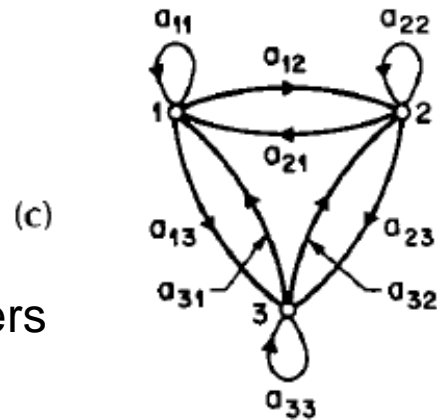
1 unknown parameter

Further alternatives



4 unknown parameters

$O = \text{HHTTHTHHTTH}..$
 $S = 21122212212..$



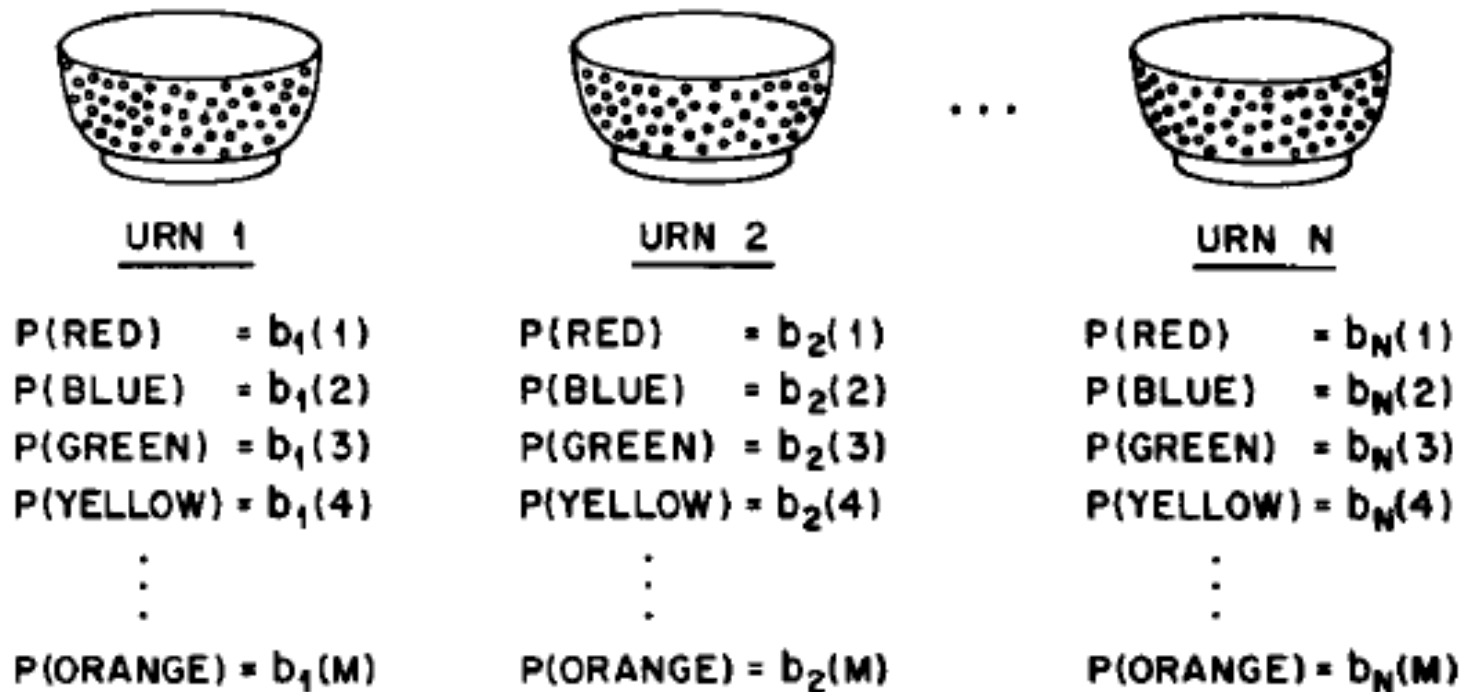
9 unknown parameters

$O = \text{HHTTHTHHTTH}..$
 $S = 31233112313..$

| | STATE | | |
|--------|-----------|-----------|-----------|
| | 1 | 2 | 3 |
| $P(H)$ | P_1 | P_2 | P_3 |
| $P(T)$ | $1 - P_1$ | $1 - P_2$ | $1 - P_3$ |

Fig. 2. Three possible Markov models which can account for the results of hidden coin tossing experiments. (a) 1-coin model. (b) 2-coins model. (c) 3-coins model.

A more general example for HMMs



$O = \{\text{GREEN, GREEN, BLUE, RED, YELLOW, RED, } \dots, \text{BLUE}\}$

Fig. 3. An N -state urn and ball model which illustrates the general case of a discrete symbol HMM.

HMM is characterized by

1) N , the number of states in the model. Although the states are hidden, for many practical applications there is often some physical significance attached to the states or to sets of states of the model. Hence, in the coin tossing experiments, each state corresponded to a distinct biased coin. In the urn and ball model, the states corresponded to the urns. Generally the states are interconnected in such a way that any state can be reached from any other state (e.g., an ergodic model); however, we will see later in this paper that other possible interconnections of states are often of interest. We denote the individual states as $S = \{S_1, S_2, \dots, S_N\}$, and the state at time t as q_t .

HMM is characterized by

2) M , the number of distinct observation symbols per state, i.e., the discrete alphabet size. The observation symbols correspond to the physical output of the system being modeled. For the coin toss experiments the observation symbols were simply heads or tails; for the ball and urn model they were the colors of the balls selected from the urns. We denote the individual symbols as $V = \{v_1, v_2, \dots, v_M\}$.

3) The state transition probability distribution $A = \{a_{ij}\}$ where

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \leq i, j \leq N. \quad (7)$$

For the special case where any state can reach any other state in a single step, we have $a_{ij} > 0$ for all i, j . For other types of HMMs, we would have $a_{ij} = 0$ for one or more (i, j) pairs.

HMM is characterized by

4) The observation symbol probability distribution in state j , $B = \{b_j(k)\}$, where

$$b_j(k) = P[v_k \text{ at } t | q_t = S_j], \quad 1 \leq j \leq N \\ 1 \leq k \leq M. \quad (8)$$

5) The initial state distribution $\pi = \{\pi_i\}$ where

$$\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N. \quad (9)$$

Given appropriate values of N , M , A , B , and π , the HMM can be used as a generator to give an observation sequence

$$O = O_1 O_2 \cdots O_T \quad (10)$$

(where each observation O_t is one of the symbols from V , and T is the number of observations in the sequence) as follows:

- 1) Choose an initial state $q_1 = S_i$ according to the initial state distribution π .
- 2) Set $t = 1$.
- 3) Choose $O_t = v_k$ according to the symbol probability distribution in state S_i , i.e., $b_i(k)$.
- 4) Transit to a new state $q_{t+1} = S_j$ according to the state transition probability distribution for state S_i , i.e., a_{ij} .
- 5) Set $t = t + 1$; return to step 3) if $t < T$; otherwise terminate the procedure.

$$\lambda = (A, B, \pi)$$

Three basic problems in HMMs

- Evaluation/scoring *Problem 1:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $P(O|\lambda)$, the probability of the observation sequence, given the model?
- States *Problem 2:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and the model λ , how do we choose a corresponding state sequence $Q = q_1 q_2 \cdots q_T$ which is optimal in some meaningful sense (i.e., best “explains” the observations)?
- Training *Problem 3:* How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

Solving the three basic problems in HMMs: Problem 1

- Compute probability of observing

$$O = O_1 O_2 \cdots O_T, \text{ given the model } \lambda, \text{ i.e., } P(O|\lambda)$$

State Sequence: $Q = q_1 q_2 \cdots q_T$

P(observed sequence): $P(O|Q, \lambda) = \prod_{t=1}^T P(O_t|q_t, \lambda)$

$$P(O|Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T).$$

P(State Sequence): $P(Q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T}.$

All together:

$$\begin{aligned} P(O|\lambda) &= \sum_{\text{all } Q} P(O|Q, \lambda) P(Q|\lambda) \\ &= \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \\ &\quad \cdots a_{q_{T-1} q_T} b_{q_T}(O_T). \end{aligned}$$

Problem 1: recap

All together:

$$\begin{aligned} P(O|\lambda) &= \sum_{\text{all } Q} P(O|Q, \lambda) P(Q|\lambda) \\ &= \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \end{aligned}$$

The interpretation of the computation in the above equation is the following. Initially (at time $t = 1$) we are in state q_1 with probability π_{q_1} , and generate the symbol O_1 (in this state) with probability $b_{q_1}(O_1)$. The clock changes from time t to $t + 1$ ($t = 2$) and we make a transition to state q_2 from state q_1 with probability $a_{q_1 q_2}$, and generate symbol O_2 with probability $b_{q_2}(O_2)$. This process continues in this manner until we make the last transition (at time T) from state q_{T-1} to state q_T with probability $a_{q_{T-1} q_T}$ and generate symbol O_T with probability $b_{q_T}(O_T)$.

Computation:

$$2T \cdot N^T$$

Example: $T = 100$ observations, $N = 5$ states

$$2 \cdot 100 \cdot 5^{100} \approx 10^{72}$$



Forward-backward Procedure

Define $\alpha_t(i) = P(O_1 O_2 \cdots O_t, q_t = S_i | \lambda)$

Induction

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N. \quad (19)$$

2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T-1$$
$$1 \leq j \leq N. \quad (20)$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i). \quad (21)$$

Forward-backward Procedure II

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N. \quad (19)$$

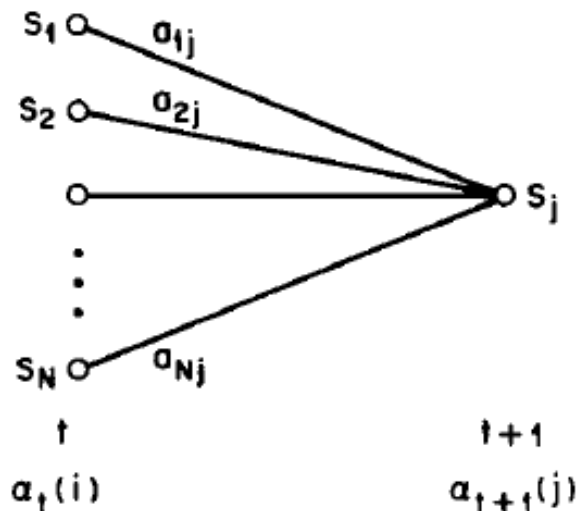
2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T-1$$

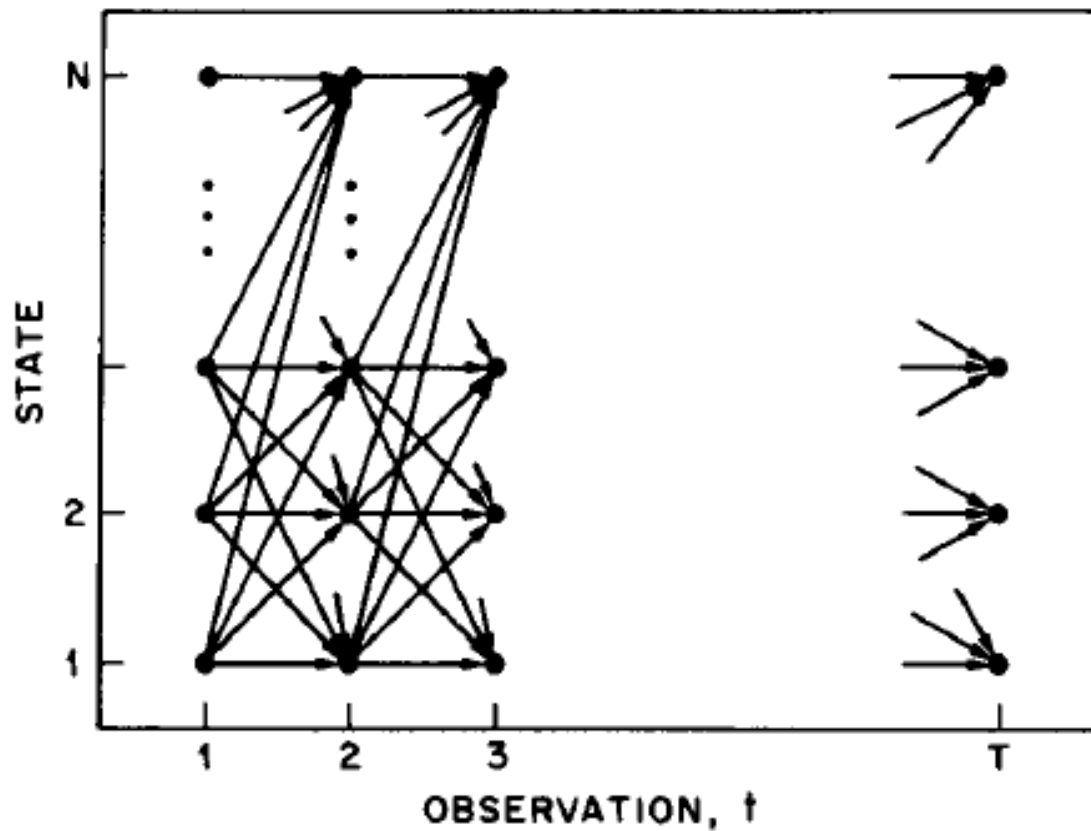
$$1 \leq j \leq N. \quad (20)$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i). \quad (21)$$



Forward-backward Procedure III



Computation: N^2T

Example: $T = 100$ observations, $N = 5$ states

Backward Procedure

$$\beta_t(i) = P(O_{t+1} O_{t+2} \cdots O_T | q_t = S_i, \lambda)$$

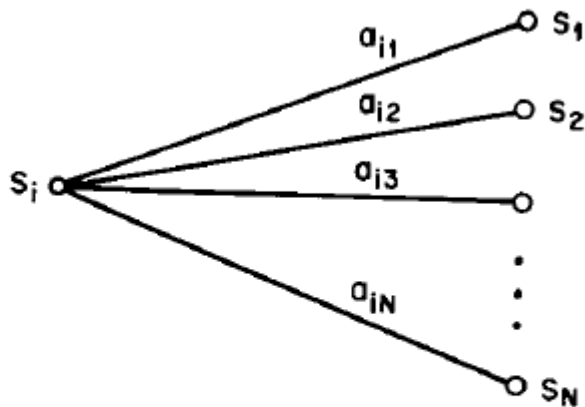
1) Initialization:

$$\beta_T(i) = 1, \quad 1 \leq i \leq N.$$

2) Induction:

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j),$$

$$t = T - 1, T - 2, \cdots, 1, 1 \leq i \leq N.$$



$$\begin{array}{ccc} t & & t+1 \\ \beta_t(i) & & \beta_{t+1}(j) \end{array}$$

Computation: $N^2 T$



Three basic problems in HMMs

- Evaluation/scoring *Problem 1:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $P(O|\lambda)$, the probability of the observation sequence, given the model?
- States *Problem 2:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and the model λ , how do we choose a corresponding state sequence $Q = q_1 q_2 \cdots q_T$ which is optimal in some meaningful sense (i.e., best “explains” the observations)?
- Training *Problem 3:* How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

Problem 2: computing optimal state sequence

Define: $\gamma_t(i) = P(q_t = S_i | O, \lambda)$

$$\gamma_t(i) = \frac{\alpha_t(i) \beta_t(i)}{P(O|\lambda)} = \frac{\alpha_t(i) \beta_t(i)}{\sum_{i=1}^N \alpha_t(i) \beta_t(i)} \quad \sum_{i=1}^N \gamma_t(i) = 1.$$

Most likely state at time t : $q_t = \operatorname{argmax}_{1 \leq i \leq N} [\gamma_t(i)], \quad 1 \leq t \leq T.$

BUT



Viterbi Algorithm

Define: $\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P[q_1 q_2 \dots q_t = i, O_1 O_2 \dots O_t | \lambda]$

Induction: $\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(O_{t+1}).$

Viterbi II

1) Initialization:

$$\delta_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N \quad (32a)$$

$$\psi_1(i) = 0. \quad (32b)$$

2) Recursion:

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \quad 2 \leq t \leq T$$
$$1 \leq j \leq N \quad (33a)$$

$$\psi_t(j) = \operatorname{argmax}_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \quad 2 \leq t \leq T$$
$$1 \leq j \leq N. \quad (33b)$$

3) Termination: (34a)

$$P^* = \max_{1 \leq i \leq N} [\delta_T(i)]$$

$$q_T^* = \operatorname{argmax}_{1 \leq i \leq N} [\delta_T(i)]. \quad (34b)$$

4) Path (state sequence) backtracking:

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \dots, 1. \quad (35)$$

Remember: Forward-backward Procedure

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N. \quad (19)$$

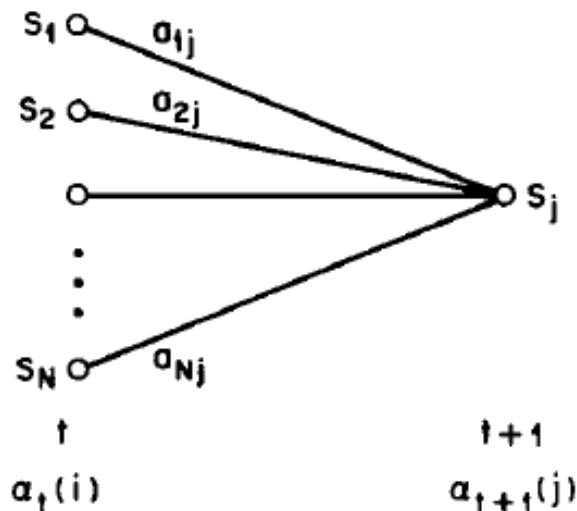
2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T-1$$

$$1 \leq j \leq N. \quad (20)$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i). \quad (21)$$



Problem 3: HMM training

Define: $\xi_t(i, j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda).$ (36)

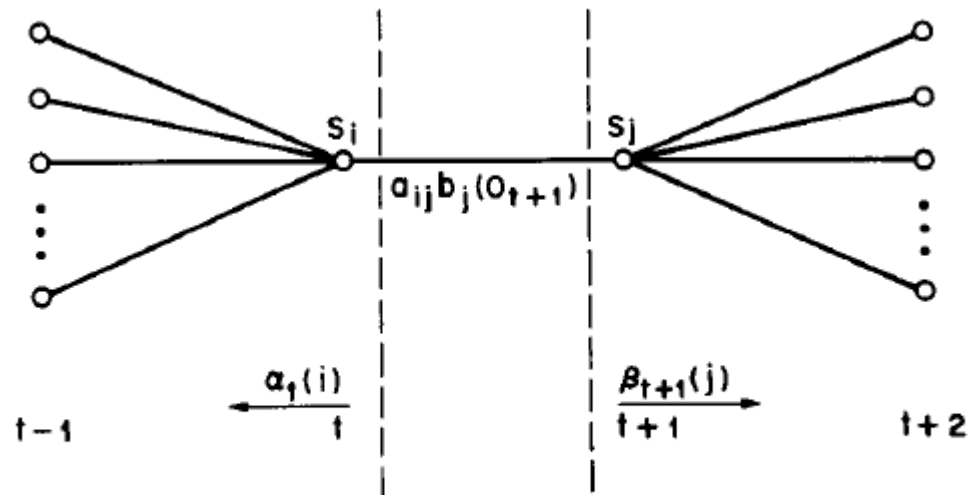
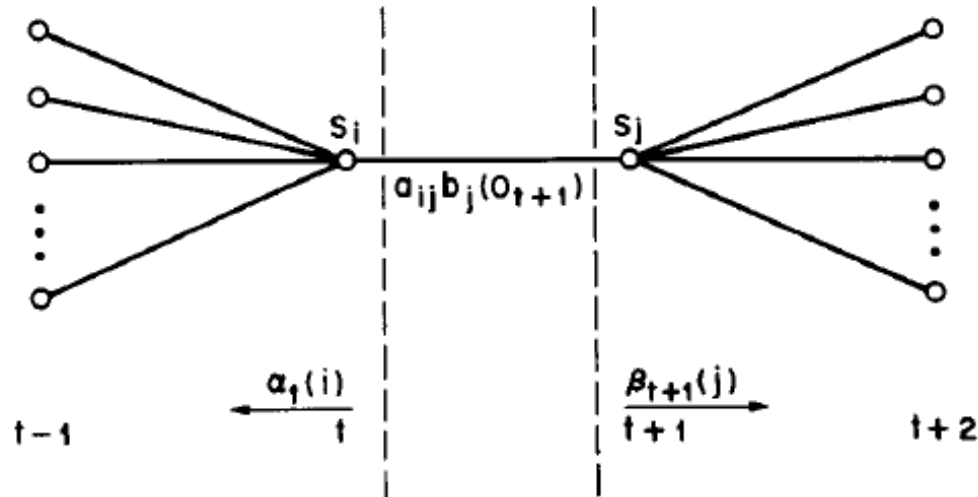


Fig. 6. Illustration of the sequence of operations required for the computation of the joint event that the system is in state S_i at time t and state S_j at time $t + 1$.

Problem 3: HMM training



$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)}$$

$P(q_t = S_i, q_{t+1} = S_j, O|\lambda)$

$$= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)} \quad (37)$$

HMM training II

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j).$$

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of transitions from } S_i \quad (39a)$$

$$\sum_{t=1}^{T-1} \xi_t(i, j) = \text{expected number of transitions from } S_i \text{ to } S_j. \quad (39b)$$

HMM training III: reestimation procedure

$\bar{\pi}_i$ = expected frequency (number of times) in state S_i at time ($t = 1$) = $\gamma_1(i)$

\bar{a}_{ij} = $\frac{\text{expected number of transitions from state } S_i \text{ to state } S_j}{\text{expected number of transitions from state } S_i}$

$$= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$\bar{b}_j(k)$ = $\frac{\text{expected number of times in state } j \text{ and observing symbol } v_k}{\text{expected number of times in state } j}$

$$= \frac{\sum_{t=1}^T \gamma_t(j) \text{ s.t. } O_t = v_k}{\sum_{t=1}^T \gamma_t(j)}.$$

Baum & Welch: reestimation procedure converges!



Remark on Baum Welch as EM algorithm

Aux Function: $Q(\lambda, \bar{\lambda}) = \sum_Q P(Q|O, \lambda) \log [P(O, Q|\bar{\lambda})]$

Theorem

$$\max_{\bar{\lambda}} [Q(\lambda, \bar{\lambda})] \Rightarrow P(O|\bar{\lambda}) \geq P(O|\lambda).$$

Stochastic constraints

$$\sum_{i=1}^N \bar{\pi}_i = 1$$
$$\sum_{j=1}^N \bar{a}_{ij} = 1, \quad 1 \leq i \leq N$$
$$\sum_{k=1}^M \bar{b}_j(k) = 1, \quad 1 \leq j \leq N$$

Also constraint optimization problem for $P(O|\lambda)$



Remark on Baum Welch as EM algorithm II

Constraint optimization problem says P is maximized if

$$\pi_i = \frac{\pi_i \frac{\partial P}{\partial \pi_i}}{\sum_{k=1}^N \pi_k \frac{\partial P}{\partial \pi_k}} \quad (44a)$$

$$a_{ij} = \frac{a_{ij} \frac{\partial P}{\partial a_{ij}}}{\sum_{k=1}^N a_{ik} \frac{\partial P}{\partial a_{ik}}} \quad (44b)$$

$$b_j(k) = \frac{b_j(k) \frac{\partial P}{\partial b_j(k)}}{\sum_{\ell=1}^M b_j(\ell) \frac{\partial P}{\partial b_j(\ell)}}. \quad (44c)$$

This EM perspective is equivalent to reestimation from slide 27



Three basic problems in HMMs

- Evaluation/scoring *Problem 1:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $P(O|\lambda)$, the probability of the observation sequence, given the model?
- States *Problem 2:* Given the observation sequence $O = O_1 O_2 \cdots O_T$, and the model λ , how do we choose a corresponding state sequence $Q = q_1 q_2 \cdots q_T$ which is optimal in some meaningful sense (i.e., best “explains” the observations)?
- Training *Problem 3:* How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

Acknowledgement

- The slides are using Rabiners Tutorial from Proc IEEE Feb 1989