



Hidden Markov Model

L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257-286, Feb. 1989.

Based on slides by Mehmet Yunus Dönmez

Markov Random Processes

- ◆ A random sequence has the Markov property if its distribution is determined solely by its current state. Any random process having this property is called a *Markov random process*.
- ◆ For observable state sequences (state is known from data), this leads to a *Markov chain* model.
- ◆ For non-observable states, this leads to a *Hidden Markov Model* (HMM).

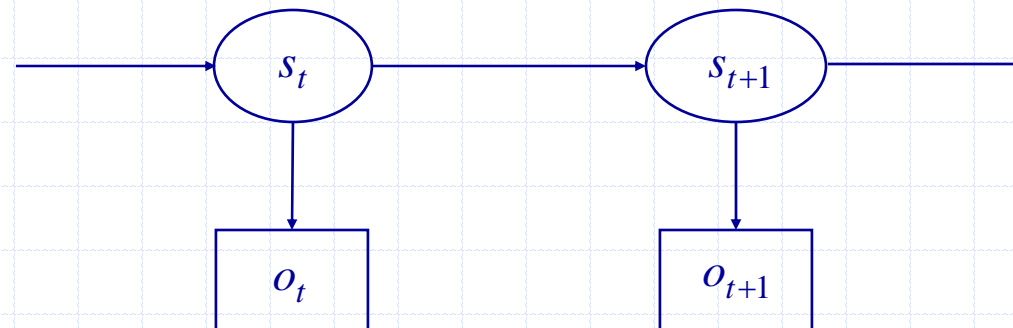
Hidden Markov model

◆ Hidden Markov model (HMM)

- x_k is called the hidden state and z_k is called the observation
- Markov chain $P(s_{t+1} | s_1, \dots, s_t) = P(s_{k+1} | s_t)$

- As the result $P(o_t | s_1, \dots, s_t) = P(o_t | s_t)$

$$P(s_1, \dots, s_t, o_1, \dots, o_t) = P(s_1)P(o_1 | s_1) \prod_{i=2}^t [P(s_i | s_{i-1})P(o_i | s_i)]$$



HMM Elements

◆ An HMM for discrete symbol observation

- N

the number of states in the model $\{1, 2, \dots, N\}$

the state at time $t \rightarrow s_t$

- M

the number of distinct observation symbols per state

$$V = \{v_1, v_2, \dots, v_M\}$$

HMM Elements (2)

- a_{ij} : the state-transition probability distribution : A

$$a_{ij} = p[s_{t+1} = j | s_t = i] \quad 1 \leq i, j \leq N$$

- $b_j(k)$: the observation symbol probability distribution : B

$$b_j(k) = p[o_t = v_k | s_t = j] \quad 1 \leq k \leq M$$

HMM Elements (3)

- π_i : the initial state distribution, π

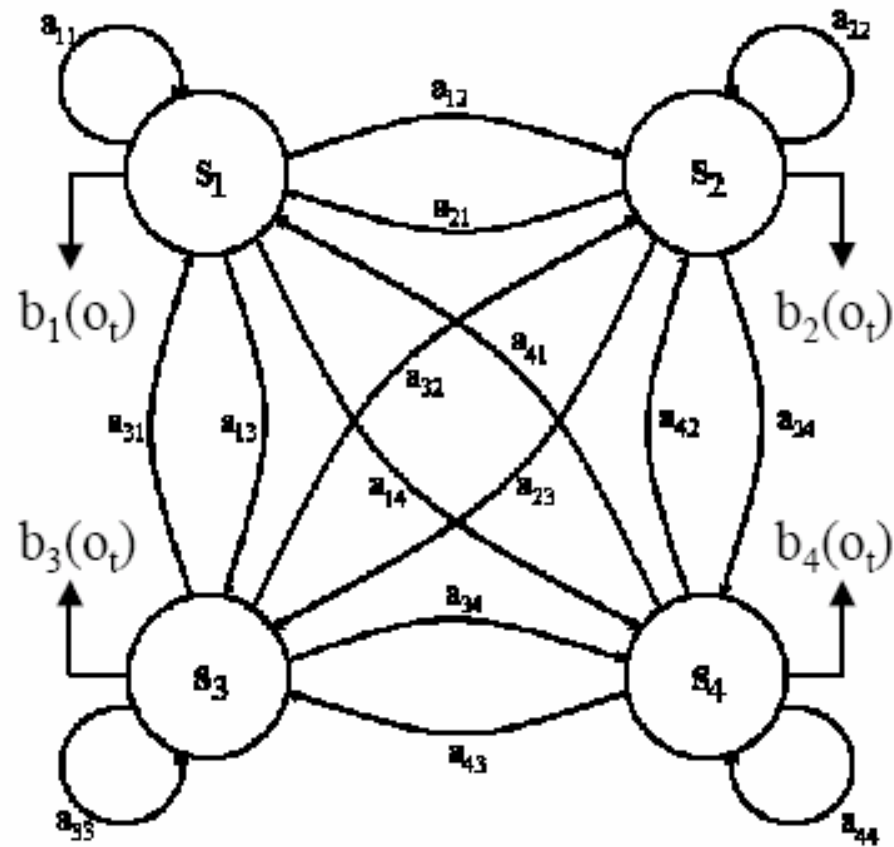
$$\pi_i = p[s_1 = i] \quad 1 \leq i \leq N$$

→ Compact Notation of a HMM Model

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

$$\begin{aligned} &\rightarrow p(o_1, o_2, \dots, o_T \mid \lambda, s_1, s_2, \dots, s_T) \\ &= \pi_1 b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) \dots a_{q_{(T-1)} q_T} b_{q_T}(o_T) \end{aligned}$$

A General Case HMM



HMM Generator

- ◆ Choose an initial state ($s_1 = i$) \leftarrow initial state distribution
- ◆ Set $t = 1$
- ◆ Choose $O_t = v_k$ \leftarrow symbol probability distribution
- ◆ Transit to a new state $s_{t+1} = j$ \leftarrow state transition probability distribution
- ◆ Set $t = t + 1$; return to step 3 if $t < T$; otherwise, terminate the procedure

HMM Properties

- ◆ Often simplified

$$\pi(s_1 = 1) = 1 \quad , \quad \text{and} \quad \pi(s_1 > 1) = 0$$

- ◆ Obviously

$$\sum_j a_{ij} = 1 \quad \text{for all } i$$

- ◆ Discrete HMMs :

$$V = \{v_1, v_2, \dots, v_M\}$$

- ◆ Continuous HMMs :

$$V = R^d$$

HMM Properties (2)

- ◆ The term “hidden”
 - we can only access to visible symbols (observations)
 - drawing conclusions without knowing the hidden sequence of states
- ◆ Causal: Probabilities depend on previous states
- ◆ Ergodic if every state is visited in transition sequence for any given initial state
- ◆ Final or absorbing state: the state which, if entered, is never left

3 Basic Problems

- ◆ The Evaluation Problem
 - given an HMM λ
 - given an observation o_1, o_2, \dots, o_T
 - compute the probability of the observation

$$p\{o_1, o_2, \dots, o_T \mid \lambda\}$$

3 Basic Problems (2)

◆ The Decoding Problem

- given an HMM λ
- given an observation o_1, o_2, \dots, o_T
- compute the most likely state sequence

$$s_1, s_2, \dots, s_T$$

i.e. $\arg \max_{s_1, \dots, s_T} p(s_1, \dots, s_T \mid o_1, o_2, \dots, o_T, \lambda)$

3 Basic Problems (3)

- ◆ The learning / optimization problem
 - given an HMM λ
 - given an observation o_1, o_2, \dots, o_T
 - find an HMM such that

$$p\{o_1, o_2, \dots, o_T \mid \lambda_1\} > p\{o_1, o_2, \dots, o_T \mid \lambda\}$$

The Evaluation Problem

◆ We know :

$$p(o_1, o_2, \dots, o_T \mid \lambda, s_1, s_2, \dots, s_T)$$
$$= \pi(s_1) b_{s_1}(o_1) \prod_{k=1, \dots, T-1} a_{s_k s_{k+1}} b_{s_{k+1}}(o_{k+1})$$

- From this :

$$p(o_1, o_2, \dots, o_T \mid \lambda)$$
$$= \sum_{\substack{s_2=1, \dots, N \\ s_3=1, \dots, N \\ \vdots \\ s_T=1, \dots, N}} \sum_{s_1=1, \dots, N} \pi(s_1) b_{s_1}(o_1) \prod_{k=1, \dots, T-1} a_{s_k s_{k+1}} b_{s_{k+1}}(o_{k+1})$$

The Evaluation Problem(2)

- ◆ Obvious:
for sufficiently large values of T , it is infeasible to compute the above term for all possible state sequences \rightarrow need other solution

The Forward Algorithm

- ◆ At time t and state i , probability of partial observation sequence

$$\alpha_t(i) = P(o_1, \dots, o_T, s_t = i \mid \lambda)$$

- ◆ $\alpha_1(i) = \pi_i b_i(o_1) \quad 1 \leq i \leq N$

→ $A[\text{time}][\text{state}]$: array

- ◆
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(o_{t+1})$$

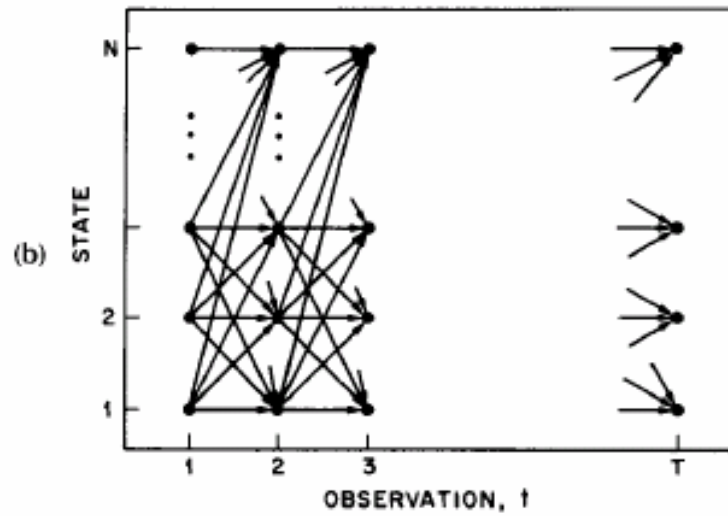
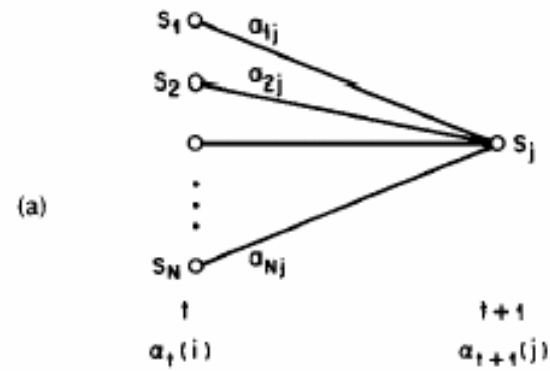
The Forward Algorithm (2)

- ◆ $A[time][state] = \alpha_{time}(state)$

- ◆ As a result at the last time T

$$p(o_1, o_2, \dots, o_T | \lambda) = \sum_{state} A[T][state]$$

Figure



The Backward Algorithm

$$\beta_t(i) = P(o_{t+1}, \dots, o_T \mid s_t = i, \lambda)$$

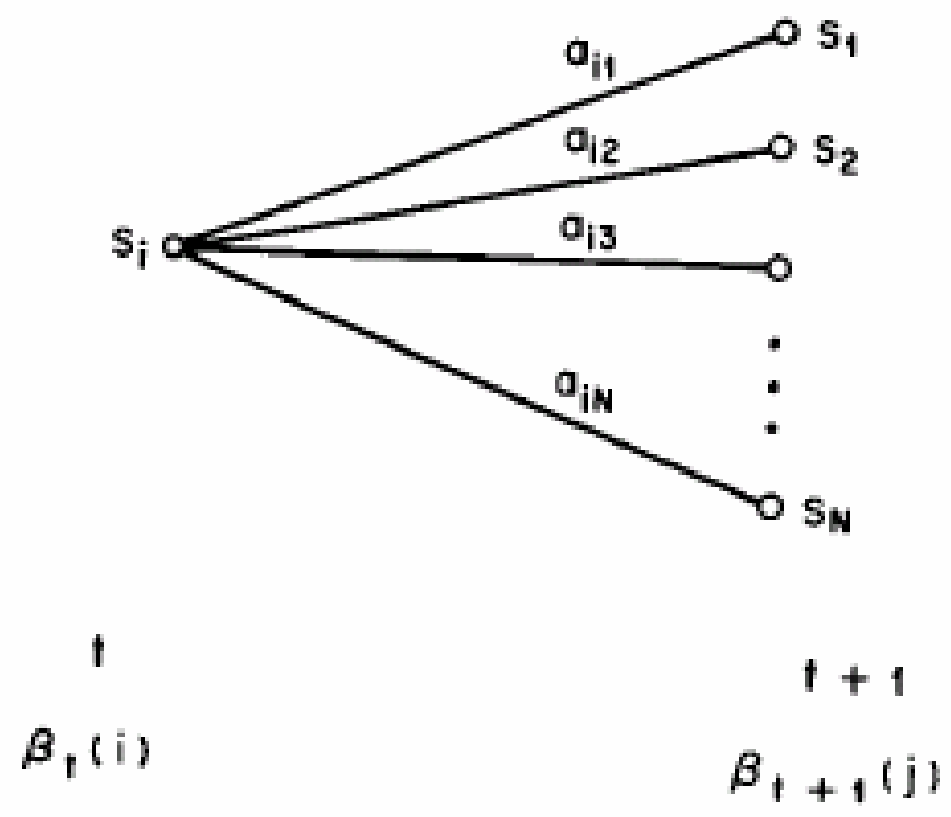
$$\beta_T(i) = 1$$

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

$$t = T - 1, T - 2, \dots, 1$$

$$p(o_1, o_2, \dots, o_T \mid \lambda) = \sum_{j=1}^N \pi(j) b_j(o_1) \beta_1(j)$$

Figure



The Decoding Problem

- ◆ Finding the “optimal” state sequence associated with the given observation sequence

Forward-Backward

- ◆ Optimality criterion : to choose the states q_t that are individually most likely at each time t

- ◆ The probability of being in state i at time t

$$\begin{aligned}\gamma_i(t) &= p(s_t = i | O, \lambda) \\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^N \alpha_t(i)\beta_t(i)}\end{aligned}$$

- ◆ $\alpha_t(i)$: accounts for partial observation sequence O_1, O_2, \dots, O_t
- ◆ $\beta_t(i)$: account for remainder $O_{t+1}, O_{t+2}, \dots, O_T$

The Viterbi Algorithm

- ◆ The best score along a single path, at time t , which accounts for the first t observations and ends in state i

$$\delta_t(i) = \max p(s_1, \dots, s_t = i, o_1, \dots, o_t | \lambda)$$

- ◆ It can be derived that

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] b_j(o_{t+1})$$

- ◆ Keep track of the argument that maximize above equation

$$\psi_t(j)$$

- ◆ Viterbi Algorithm is similar in implementation to the forward calculation, but the major difference is the maximization over previous states

The Complete Procedure

(for finding the best state sequence)

◆ Initialization

$$\delta_1(i) = \pi_i b_i(o_1) \quad 1 \leq i \leq N$$

◆ Recursion

$$\psi_1(i) = 0$$

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

$$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] \quad 1 \leq j \leq N$$

The Complete Procedure (2)

(for finding the best state sequence)

- ◆ Termination

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

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

- ◆ Path(state sequence) backtracking

$$s_t^* = \psi_{t+1}(s_{t+1}^*)$$

$$t = T-1, T-2, \dots, 1$$

The Learning / Optimization problem

◆ How do we adjust the model parameters to maximize $P(O | \lambda)$??

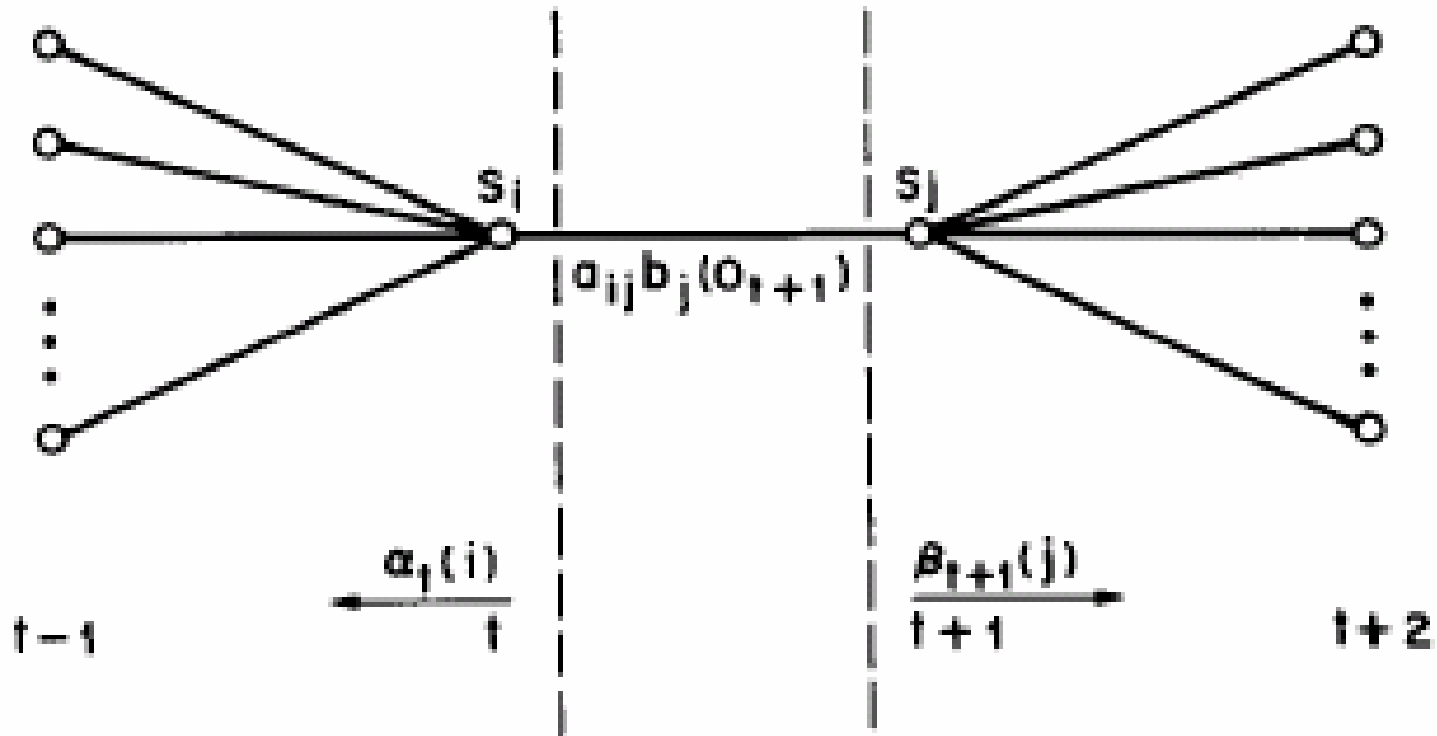
- Parameter Estimation
- Baum-Welch Algorithm (EM : Expectation Maximization)
- Iterative Procedure

Parameter Estimation

- ◆ Probability of being in state i at time t , and state j at time $t+1$

$$\begin{aligned}\xi_t(i, j) &= P(s_t = i, s_{t+1} = j | O, \lambda) \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}\end{aligned}$$

Figure



Parameter Estimation (2)

- ◆ Probability of being in state i at time t , given the entire observation sequence and the model

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

- ◆ We can relate these by summing over j

Parameter Estimation (3)

- ◆ By summing over time index t ...
 - expected number of times that state i visited
 - expected number of transitions made from state i
- That is ...

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of times that state } i \text{ in } O$$

$$\sum_{t=1}^{T-1} \xi_t(i, j) = \text{expected number of transitions made from state } i \text{ to } j \text{ in } O$$

Parameter Estimation (4)

- ◆ Update $\lambda = (A, B, \pi)$ using $\xi_t(i, j)$ & $\gamma_i(t)$

$$\bar{\pi}_i = \gamma_1(i)$$

: expected frequency (number of times) in state i at time ($t=1$)

Parameter Estimation (5)

◆ New Transition Probability ...

expected number of transitions from state i to j

expected number of transitions from state i

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

Parameter Estimation (6)

◆ New Observation Probability...

expected number of times in state j and observing symbol v_k

expected number of times in j

$$= \bar{b}_j(k) = \frac{\sum_{t=1}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j) \text{ s.t. } o_t = v_k}$$

Parameter Estimation (7)

◆ From $\lambda = (A, B, \pi)$, if we define new $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$

- New model is more likely than old model in the sense that

$$P(O | \bar{\lambda}) > P(O | \lambda)$$

- The observation sequence is more likely to be produced by new model

- has been proved by Baum & his colleagues

- iteratively use new model in place of old model, and repeat the reestimation calculation → "ML estimation"