Hidden Markov Models

Overview

- Signal Modeling
- Discrete Markov Process
- Hidden Markov Models (HMM)
- Elements of HMMs
- Three Basic Problems of HMMs
- Illustration of use of HMM for Isolated Word Recognition
- Solutions to Three Basic Problems of HMMs
- Types of HMMs
- Continuous Density HMMs

Signal Modeling

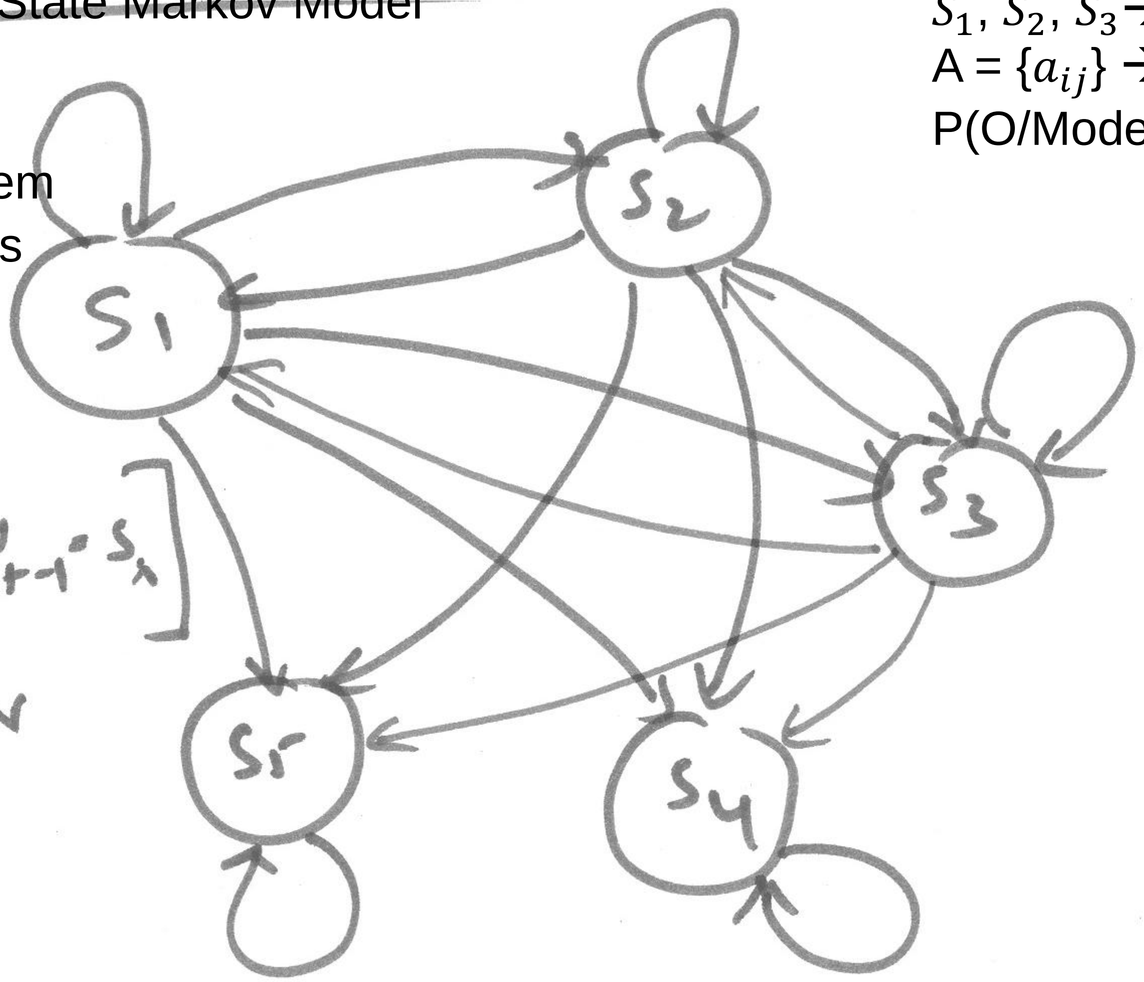
- ❑ Need for Signal Modeling
  - ✓ Provide the basis for theoretical description of a signal processing system
  - ✓ Remove noise & transmission distortion etc..
  - ✓ Learn about the signal source, and generate synthetic signals without real source
  - ✓ Important to realize the practical systems such as ASR, LID, etc..
- ❑ Deterministic Signal Models
  - ✓ Modeling a sine wave or Sum of exponentials
  - ✓ Parameters : Amplitude, Frequency, Phase & Rate of exponentials
- ❑ Statistical Signal Models
  - ✓ Signals characterized by parametric random process (Speech Production)
  - ✓ Examples: Gaussian, Poission, Markov & HMMs

Discrete Markov Process

Example : 5-State Markov Model

S_i, S_j → States of the system

N → Total number of states



$a_{ij} = P[q_t = S_j \mid q_{t-1} = S_i]$

$i \geq 1 ; j \leq N$

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

Example : 3-State Markov Model

S_1, S_2, S_3 → rain, cloudy, sunny

A = $\{a_{ij}\}$ → 3X3 state transition matrix

P(O/Model) = P(S_1, S_1, S_2, S_1, S_3, S_2, S_1/Model)

Hidden Markov Model

$O = (B, G, R, R, G, G, B, B, R, \ldots)$

$\Rightarrow$ # boxes, Sequence of boxes

symbol prob

Spectral vectors $\Leftrightarrow$ Sequence of VT shapes

Basic Elements of HMM

1. # states —— N

2. # distinct symbols / state —— M

$$V = \{v_1\ v_2 \cdots v_M\}$$

3. State transition probability $A = \{a_{ij}\}_{N\times N}$

$$a_{ij} = P\left[q_{t+1} = j \mid q_t = i\right]; \quad \begin{matrix} i \geq 1 \\ j \leq N \end{matrix}$$

4. Observation symbol prob. $B = \{b_j(k)\}$

$$b_j(k) = P\left[o_t = v_k \mid q_t = j\right]$$

5. Initial prob. $\pi_i = \pi$; $\pi_i = P\left[q_1 = i\right]$

$$\text{MODEL} = \lambda = [A, B, \pi]$$

Three Basic Problems of HMM

① How do we efficiently compute $P(O/\lambda)$

$P(o_1\, o_2 \cdots o_T / \lambda) \Rightarrow$ Testing, evaluation

Recognition, calculation

② $O = o_1\, o_2 \cdots o_T$, $\lambda = (A, B, \pi)$ $\Rightarrow Q = q_1\, q_2 \cdots q_T$

(seq of states)

best seq that explains the observations in better way

optimal seq for the given observation

states & # symbols can be modified using ~~step~~ step-2

③ How do we adjust $\lambda = (A, B, \pi)$ to max $P(O/\lambda)$

Training step, building models

Illustration of Use of HMM

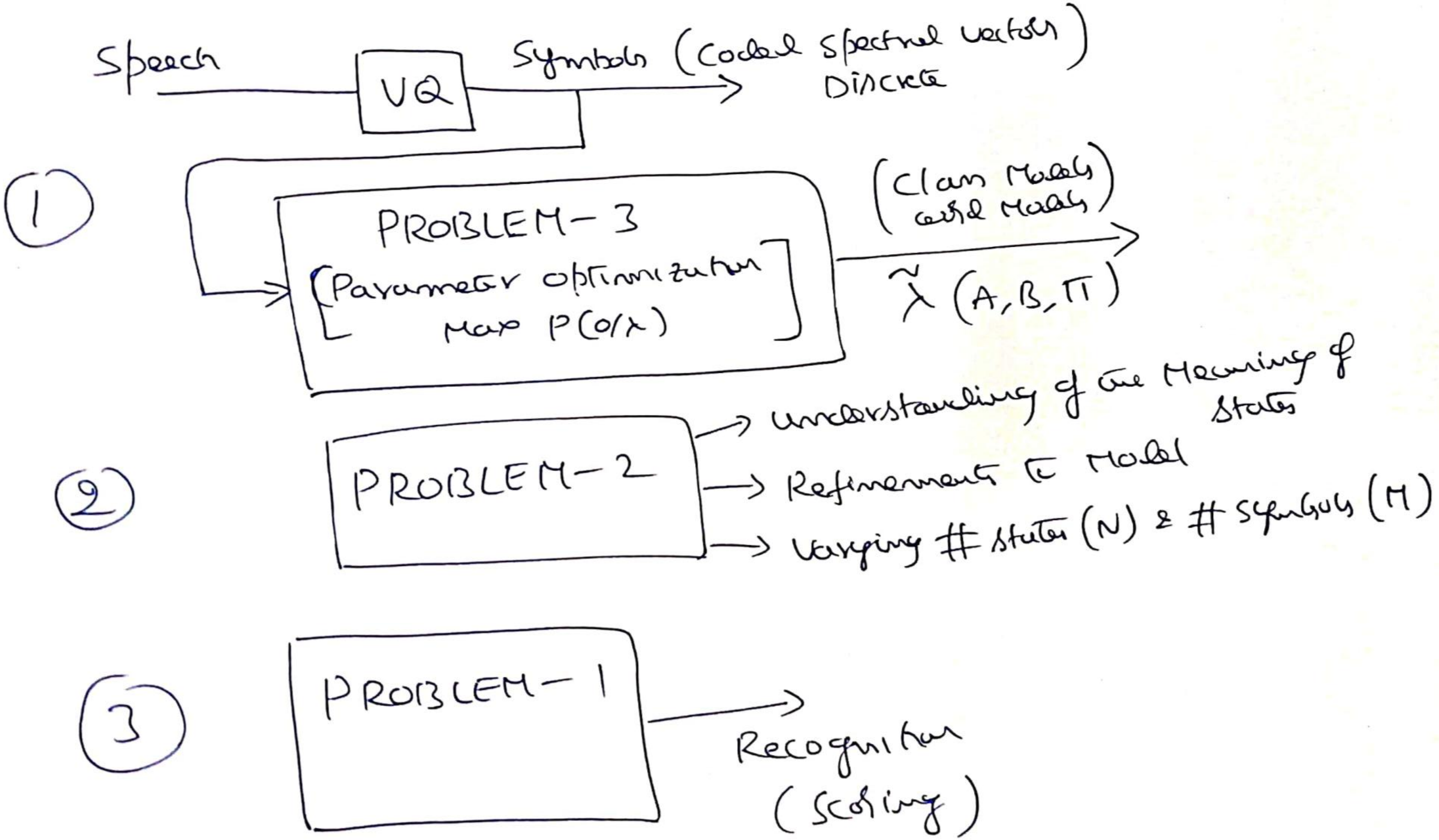
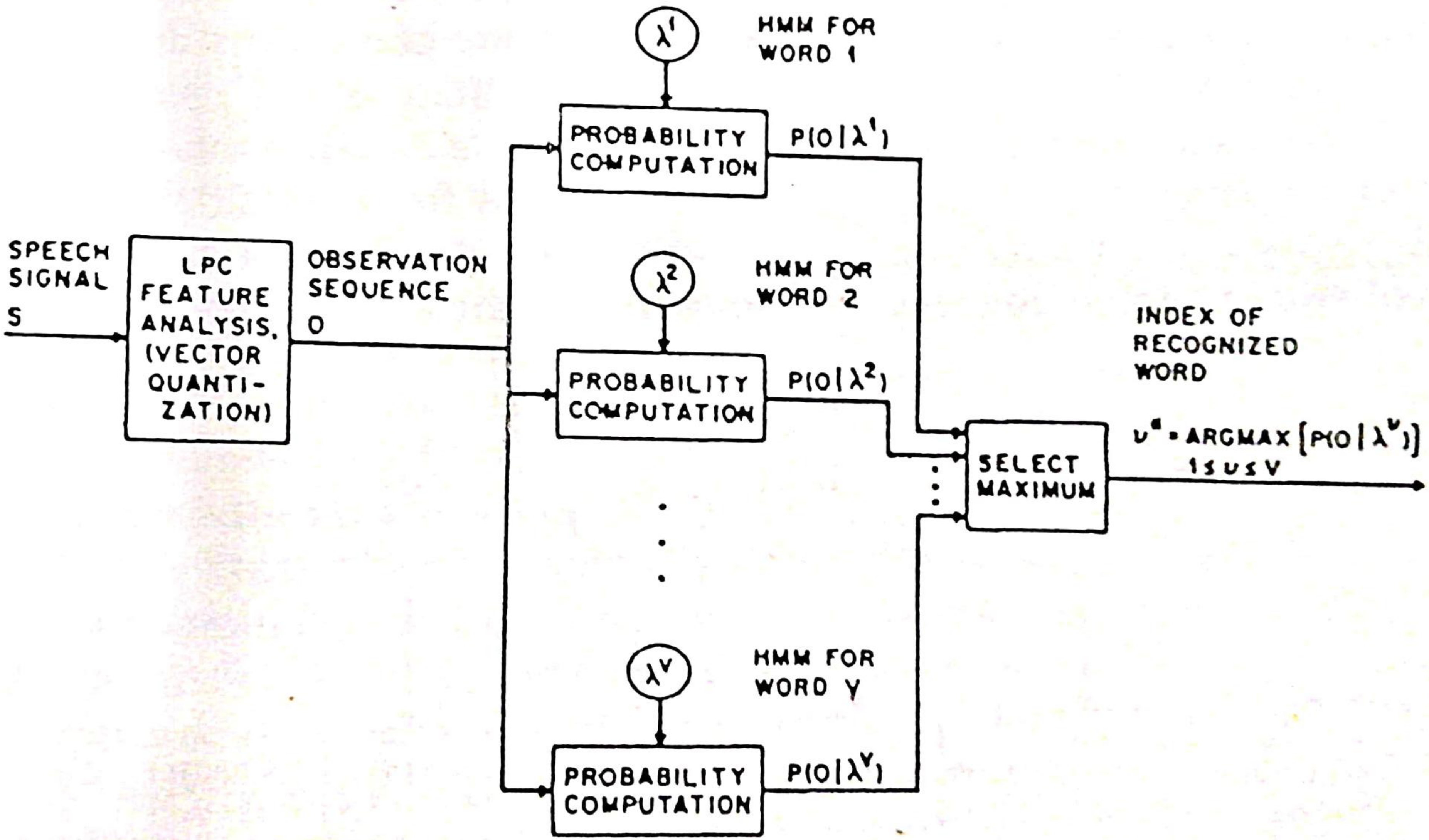


Illustration of Use of HMM : Isolated Word Rec



Solution to Problem – 1 (Computation of $P(^O/_\lambda)$)

$O = O_1 O_2 \cdots O_T \quad , \quad \lambda = A, B, \pi$

Possible state seq : $Q = (q_1 q_2 \cdots q_T) \Rightarrow N^T$

$$P(^O/_\lambda) = \sum_{q_1 q_2 \cdots q_T} \cancel{P(O/q, \lambda)}\; P(O, q/\lambda)$$

$$P(O, q/\lambda) = P(O/q, \lambda)\, P(q/\lambda)$$

$$P(O/q, \lambda) = \prod_{t=1}^{T} P(o_t / q_t, \lambda)$$

$$= b_{q_1}(o_1)\, b_{q_2}(o_2) \cdots b_{q_T}(o_T)$$

Solution to Problem – 1 (Computation of $P(^{O}/_{\lambda})$)

$$P(Q/\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T}$$

$$P(O, Q/\lambda) = P(O/Q, \lambda)\, P(Q/\lambda)$$

$$= \underbrace{\pi_{q_1} b_{q_1}(o_1)\, a_{q_1 q_2}\, b_{q_2}(o_2) \cdots b_{q_{T-1}}(o_{T-1})\, a_{q_{T-1} q_T}\, b_{q_T}(o_T)}_{(2T-1) \text{ multiplications}}$$

$$P(O/\lambda) = \sum_{q_1 q_2 \cdots q_T} P(O, Q/\lambda)$$

$q_1 q_2 \cdots q_T \Rightarrow N^T$ state sequences

Total no. of computations $<$ $2T N^T$ – multiplications; $N^T - 1$ – additions

Eg: $N = 5$, $T = 100$

\# computations $= 2 \times 100 \times 5^{100} \Rightarrow 10^{72}$

Solution to Problem – 1 (Forward Procedure)

Forward Procedure $\left[\boldsymbol{\alpha_t(i)}\right]$

$\boldsymbol{\alpha_t(i)}$ → Prob of partial observation sequence

$$\boldsymbol{\alpha_t(i)} = P(o_1 o_2 \ldots o_t, q_t = i \mid \lambda)$$

1. Initialization

$$\alpha_1(i) = P(o_1, q_1 = i \mid \lambda)$$
$$= \pi_i \, b_i(o_1) \; ; \quad 1 \le i \le N$$

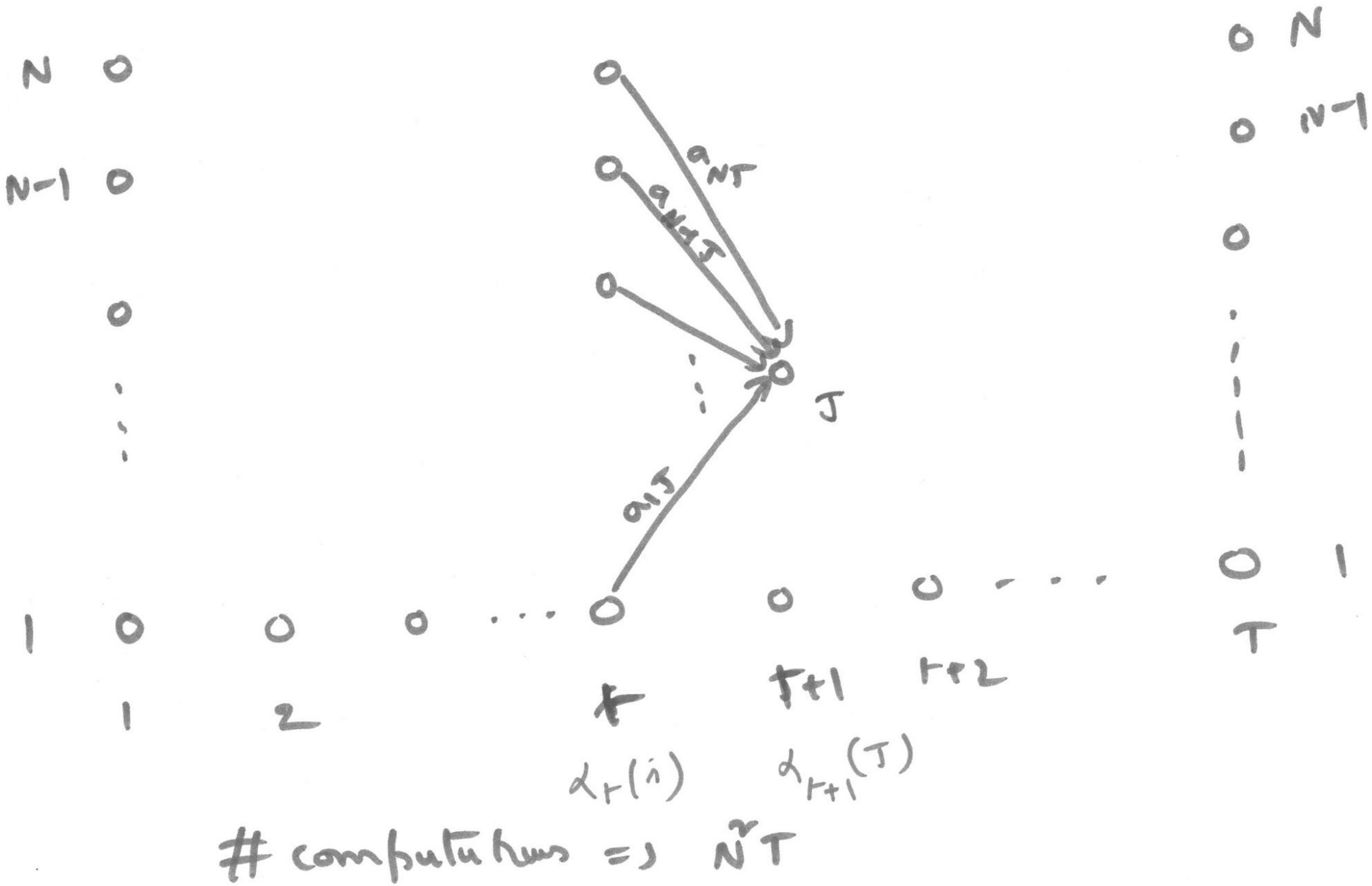
2. Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij}\right] b_j(o_{t+1}) \; ; \quad \begin{matrix} 1 \le t \le T-1 \\ 1 \le j \le N \end{matrix}$$

3. Termination

$$P(O/\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

Illustration of Computation of FV : $\alpha_t(i)$



Backward Procedure BV : $\beta_t(i)$

$$\beta_t(i) = P(o_{t+1}\, o_{t+2} \cdots o_T \mid q_t = i, \lambda)$$

Initialization

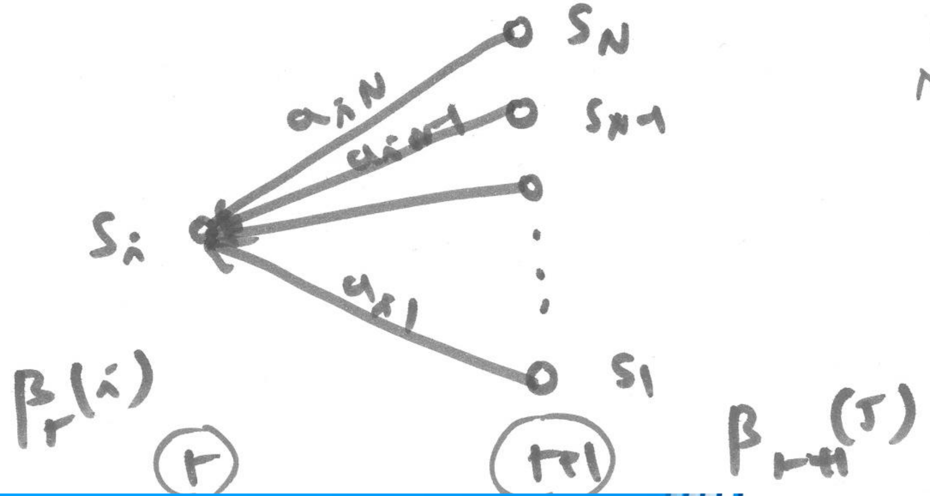
$$\beta_T(i) = 1 \qquad 1 \le i \le N$$

Induction

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

$$t = T-1, T-2, \ldots 1 \;; \quad 1 \le i \le N$$

N^2T computations



Solution to Problem – 2 Optimal State Sequence

- Individually most likely states at each time
- Max expectation of pairs of states $(q_t\ q_{t+1})$
- Max expectation of triplets of states $(q_{t-1}\ q_t\ q_{t+1})$

$$\gamma_t(i) = P(q_t = i \mid O, \lambda) = \frac{P(O, q_t = i \mid \lambda)}{\sum_{i=1}^{N} P(O, q_t = i \mid \lambda)}$$

$$= \frac{P(O, q_t = i \mid \lambda)}{P(O/\lambda)} = \frac{\alpha_t(i)\,\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\,\beta_t(i)}$$

$$q_t^* = \arg\max_i \left(\gamma_t(i)\right)$$

For Speech some of $a_{ij} = 0$; $\therefore$ optimal seq with q_t^* may not be valid sequence

Solution to Problem – 2 :Viterbi Algorithm (OSS)

$$\delta_t(i) = \max P[o_1 o_2 \cdots o_t \; q_1 q_2 \cdots q_{t-1} \; q_t = i \mid \lambda]$$

By induction $q_1 q_2 \cdots q_{t-1}$

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

$\psi_t(j)$ = tracking the argument that max for each t & j

$t = 2$ to T; $1 \le j \le N$

Initialization

$$\delta_1(i) = \pi_i \, b_i(o_1) \; ; \quad 1 \le i \le N$$

$$\psi_1(i) = 0$$

Recursion

$$\delta_t(j) = \max_i \left[\delta_{t-1}(i)\, a_{ij}\right] b_j(o_t) \qquad 2 \le t \le T$$

$$\psi_t(j) = \arg\max_i \left[\delta_{t-1}(i)\, a_{ij}\right] \qquad 1 \le j \le N$$

Solution to Problem – 2 :Viterbi Algorithm (OSS)

Termination

$$\delta_T(j) = \max_i \left[\delta_{T-1}(i)\, a_{ij} \right] b_j(O_T)$$

$$p^* = \max_i \delta_T(i) \; ; \; q_T^* = \arg\max_i \left[\delta_T(i) \right]$$

Path backtracking (state sequence)

$$q_t^* = \psi_{t+1}(q_{t+1}^*)$$

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

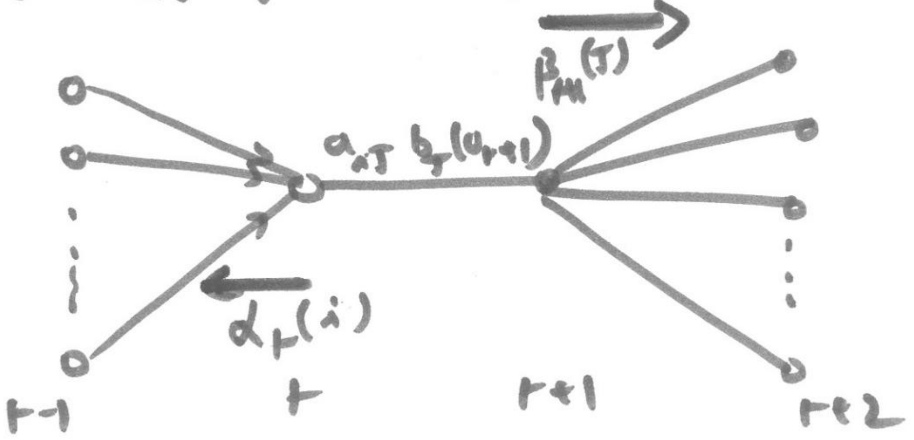
Solution to Problem – 3:Estimation of λ (A, B, π)

Baum-welch method (iterative approach)

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j \mid O, \lambda)$$

$$= P(O, q_t = i, q_{t+1} = j \mid \lambda) \,/\, P(O/\lambda)$$

$$= \frac{\alpha_t(i)\, a_{ij}\, b_j(O_{t+1})\, \beta_{t+1}(j)}{\sum_i \sum_j \alpha_t(i)\, a_{ij}\, b_j(O_{t+1})\, \beta_{t+1}(j)}$$



Solution to Problem – 3:Estimation of λ (A, B, π)

$$\gamma_t(i) = P[q_t = i \mid O, \lambda] = \sum_{j=1}^{N} \xi_t(i,j)$$

$$\text{Avg \# transitions from state 'i' in 'O'} = \sum_{t=1}^{T-1} \gamma_t(i)$$

$$\text{Avg \# transition from state } i \to j \text{ in 'O'} = \sum_{t=1}^{T-1} \xi_t(i,j)$$

$$\overline{\pi_i} = \text{expected \# times in state 'i' at } t=1 = \gamma_1(i)$$

$$\overline{a_{ij}} = \frac{\text{Expect \# transitions from } i \to j}{\text{Expected \# transition from } i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\overline{b_j(k)} = \frac{\text{Expected \# times in state 'j' \& } o_t = v_k}{\text{Expected \# times in state } j} = \frac{\sum_{t=1}^{T} \gamma_t(j) \big/_{o_t = v_k}}{\sum_{t=1}^{T} \gamma_t(j)}$$

$$\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi}); \quad P(O/\bar{\lambda}) > P(O/\lambda)$$

Types of HMMs

Types of HMMs

Based on structure of transition Matrix: Ergodic, Left-right

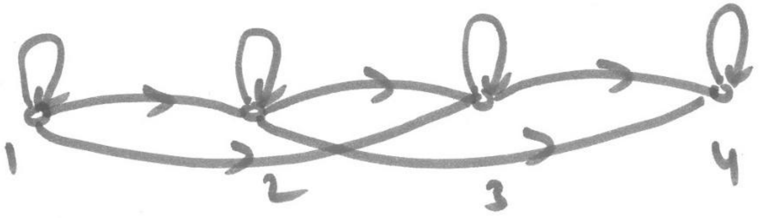
1. Ergodic (fully connected) HMM

   $\forall\ a_{ij} > 0$

2. Left-right model (Bakis's model)

   $$\left.\begin{array}{ll} a_{ij} = 0 & j < i \\ \pi_i = 0 & i \neq 1 \\ \quad = 1 & i = 1 \end{array}\right\} \qquad a_{ij} = 0 \quad j > i + \Delta i$$

   $\Delta i = 2$



3. Discrete vs Continuous HMMs

Continuous Density HMMs

Prob density of observation ~~freq~~ symbols at state J = $b_J(o)$

$$b_J(o) = \sum_{K=1}^{M} c_{JK} \, N(o, \mu_{JK}, \Sigma_{JK})$$

c_{JK}: Mixture coeff for K^{th} component

N: Gaussian density

μ_{JK}: Mean vector for ~~the~~ K^{th} component

Σ_{JK}: Co-variance matrix for K^{th} component

$$\sum_{k=1}^{M} c_{JK} = 1 \qquad c_{JK} \geq 0$$

$$1 \leq J \leq N \; ; \; 1 \leq K \leq M$$

$$\int_{-\infty}^{\infty} b_J(o)\, do = 1 \qquad 1 \leq J \leq N$$

Continuous Density HMMs : Parameter Estimation

$$\overline{c}_{JK} = \sum_{t=1}^{T} \gamma_t(J, K) \Big/ \sum_{t=1}^{T} \sum_{K=1}^{M} \gamma_t(J, K)$$

$$\overline{\mu}_{JK} = \sum_{t=1}^{T} \gamma_t(J, K) \cdot o_t \Big/ \sum_{t=1}^{T} \gamma_t(J, K)$$

$$\overline{\Sigma}_{JK} = \sum_{t=1}^{T} \gamma_t(J, K) \, (o_t - \mu_{JK})(o_t - \mu_{JK})^T \Big/ \sum_{t=1}^{T} \gamma_t(J, K)$$

$$\gamma_t(J, K) = \left[\frac{\alpha_t(J)\, \beta_t(J)}{\sum_{J=1}^{N} \alpha_t(J)\, \beta_t(J)} \right] \left[\frac{c_{JK}\, N(o_t, \mu_{JK}, \Sigma_{JK})}{\sum_{K=1}^{M} c_{JK}\, N(o_t, \mu_{JK}, \Sigma_{JK})} \right]$$

$\gamma_t(J, K) = \gamma_t(J)$ Mixture with only one component