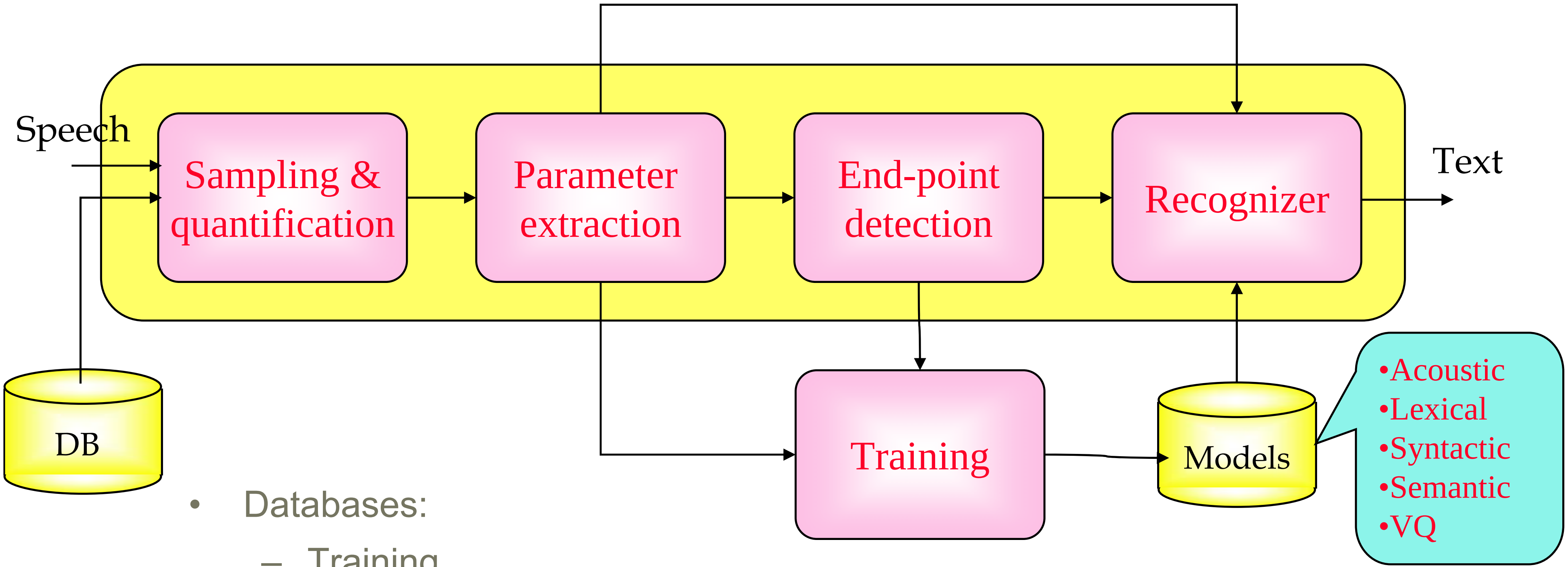
CEU
Universidad
San Pablo

Hidden Markov Models applied to Speech recognition:

Basic algorithms
Discrete, Continuous & Semicontinuous HMMs

Ricardo de Córdoba Herralde
Grupo de Tecnología del Habla
Departamento de Ingeniería Electrónica. UPM

Speech Recognition Architecture



- Databases:
  - Training
  - Test
  - Validation

MUST BE different

(size and "cheating" problems)

Hidden Markov Models (HMMs). Introduction (I)

- Problem:
  - A process generates an observable sequence of symbols (vectors, heads or tails, ball colors in an urn, etc.)
  - How a model that explains this sequence is built?
  - Using that model a system for generation, recognition, identification, etc., can be designed
- Model types:
  - Deterministic: exploit known characteristics of the signal
  - **Statistical:** try to characterize the statistical properties of the signal

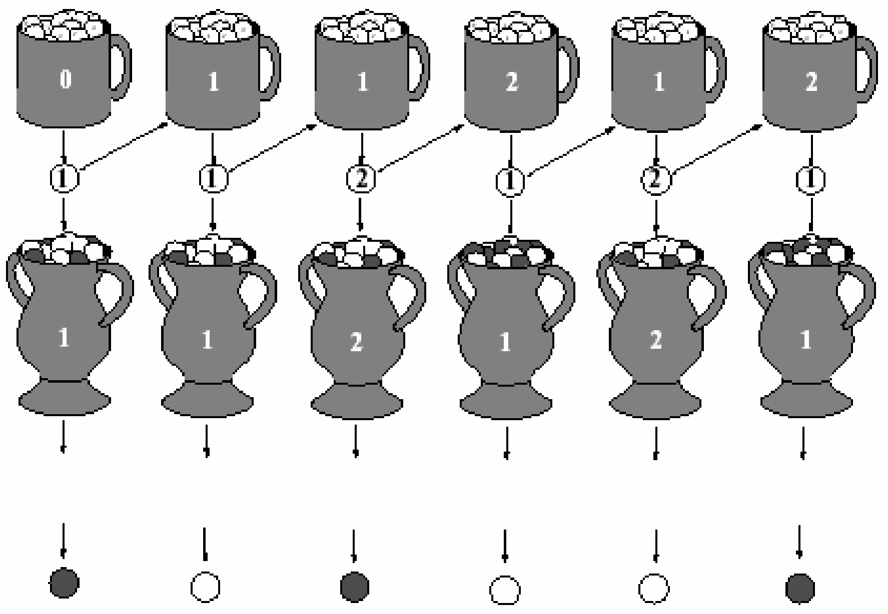
CEU Universidad San Pablo

Hidden Markov Models (HMMs). Introduction (II)

- Statistical models:
  - Gaussian, Poisson, Markov, Hidden Markov Models, etc.
    - Assumed that the signal is correctly characterized by a random process
- Example previous to the HMM definition:
  - Urns and colored balls, a subject is hidden
  - The subject selects an urn according to a random process (hidden process)
  - Selects a ball and finally shows it according to a random process (visible process)

Hidden Markov Models (HMMs). Introduction (III)

- Objective: given the model and the observation sequence O
  - How can the underlying state sequence Q be determined?



Observation Sequence: $\boldsymbol{O} = \{B, W, B, W, W, B\}$

State Sequence: $\boldsymbol{Q} = \{1, 1, 2, 1, 2, 1\}$

Hidden Markov Models (HMMs). Introduction (IV)

- Definition
  - Double stochastic process:
    - Hidden stochastic process, unseen
    - Visible stochastic process, generates the observation sequence
- Parametric model able to describe acoustic events in an efficient way
- We assume that the transition depends only on the previous state and the observation only on the current state (first order)

Hidden Markov Models (HMMs). Discrete HMMs

- Elements of a **discrete** HMM
  - N states $S = \{S_1, S_2, \ldots S_N\}$ in t, q_t ***TOPOLOGY***
  - M observation symbols $V = \{v_1, v_2, \ldots, v_M\}$ in t, O_t
  - State transition probability distribution

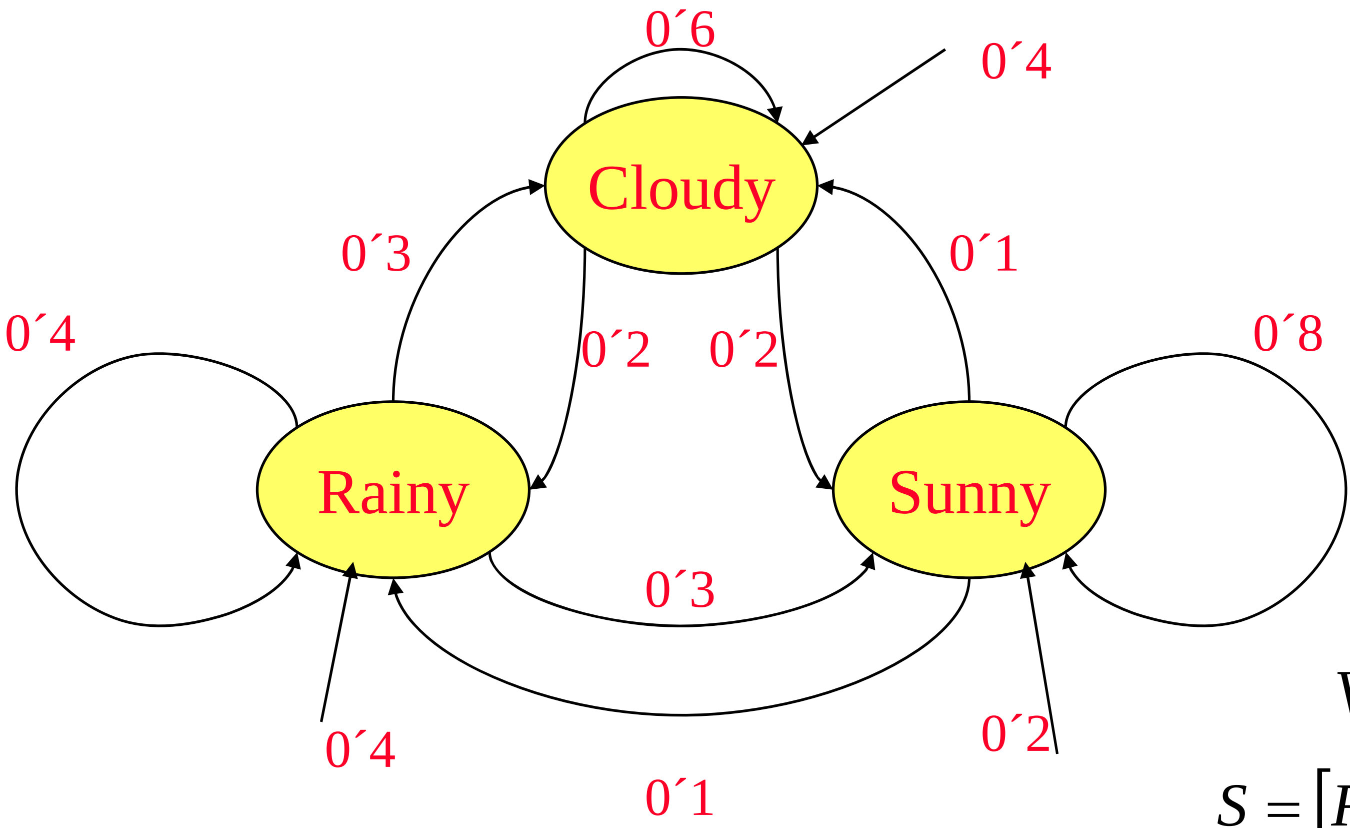
    $$A = \{ a_{ij} = p(q_{t+1}=S_j|q_t=S_i) \}$$
  - Observation symbol probability distribution in state j

    $$B = \{ b_j(k) = p(O_t=v_k|q_t=S_j) \}$$
  - Initial state distribution

    $$\Pi = \{ \pi_i = p(q_1=S_i) \}$$
- Notationally, an HMM is typically written as:
  $\lambda = \{\mathbf{A}, \mathbf{B}, \pi\}$
- ≈ Probabilistic finite automata

Hidden Markov Models (HMMs). Example

- $\lambda = \{ A, B, \Pi \}$



$$\Pi = \begin{bmatrix} 0'4 & 0'4 & 0'2 \end{bmatrix}$$

$$A = \begin{bmatrix} 0'4 & 0'3 & 0'3 \\ 0'2 & 0'6 & 0'2 \\ 0'1 & 0'1 & 0'8 \end{bmatrix}$$

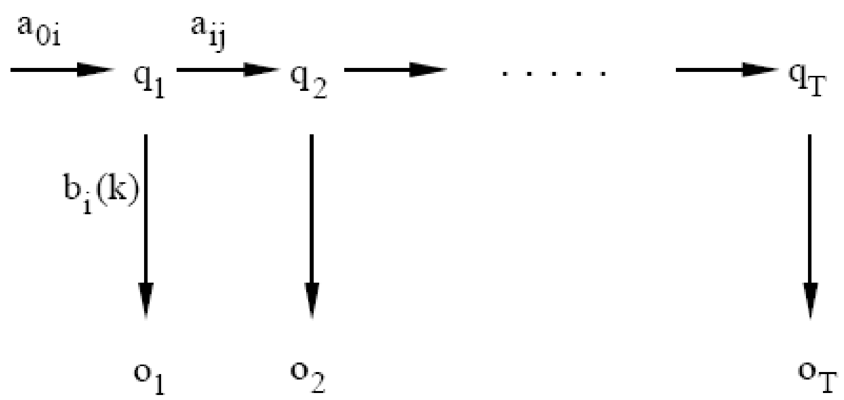
$$B = \begin{bmatrix} 0'7 & 0'2 & 0'1 \\ 0'3 & 0'6 & 0'1 \\ 0'1 & 0'2 & 0'7 \end{bmatrix}$$

$$V = \begin{bmatrix} rain & clouds & sun \end{bmatrix}$$

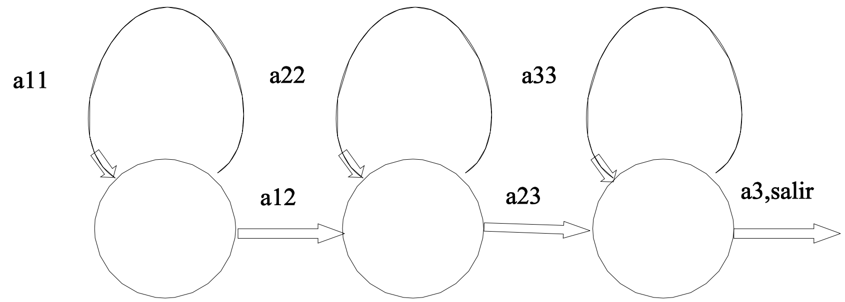
$$S = \begin{bmatrix} Rainy & Cloudy & Sunny \end{bmatrix}$$

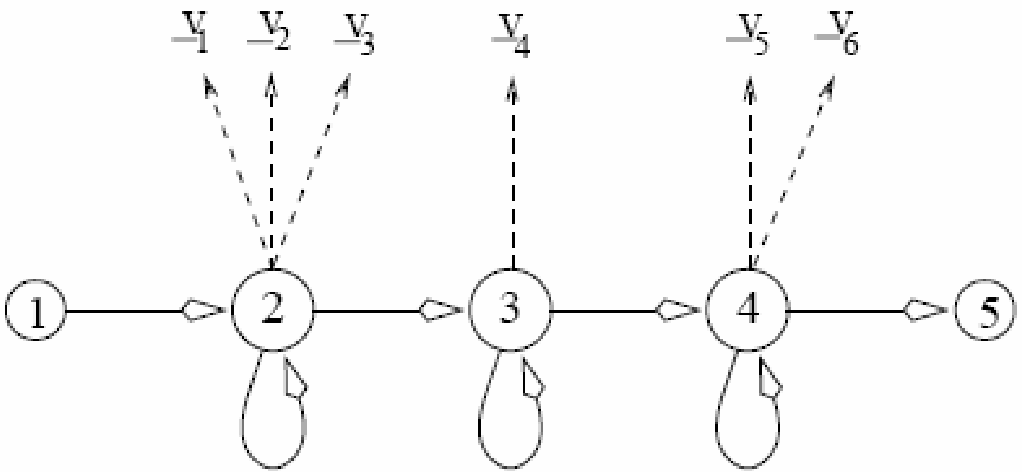
Hidden Markov Models (HMMs). Generation of HMM Observations

1. Choose an initial state, $q_1 = s_i$, based on the initial state distribution, π
2. For t =1 to T :
   - Choose $o_t = v_k$ according to the symbol probability distribution in state s_i, $b_i(k)$
   - Transition to a new state $q_{t+1} = s_j$ according to the state transition probability distribution for state s_i, a_{ij}
3. Increment t by 1, return to step 2 if $t \leq T$; else, terminate



Hidden Markov Models (HMMs). Typical topology for speech





Hidden Markov Models (HMMs). Problems to be solved (I)

- Three basic problems:
  - **Evaluation**:
    - Given the observation sequence $O=\{O_1, O_2, ..., O_T\}$ and the model λ
    - How do we compute $p(O \mid \lambda)$ = the probability of sequence O being generated by the model
    - To know which model better represents O $\Rightarrow$ recognition
  - **Segmentation**:
    - Given the observation sequence $O=\{O_1, O_2, ..., O_T\}$ and model λ
    - How do we choose a state sequence $Q=\{q_1, q_2, ..., q_T\}$ which is optimum in some sense?

CEU Universidad San Pablo

Hidden Markov Models (HMMs). Problems to be solved (II)

– **Training or estimation**:

  - Given the observation sequence $O=\{O_1, O_2, \ldots, O_T\}$
  - How do we adjust the model parameters λ to maximize $p(O \mid \lambda)$?
  - Objective: optimize λ parameters to better describe the sequence

- Application to isolated speech recognition: training + evaluation

Hidden Markov Models (HMMs). Evaluation (I)

- Evaluation using raw force
  - Given the observation sequence $O=\{O_1, O_2, ..., O_T\}$ and the model λ: ¿$p(O \mid \lambda)$?
  - Compute all possible sequences $Q = \{q_1, q_2, ..., q_T\}$:

$$p(O|Q,\lambda) = \prod_{t=1}^{T} p(O_t|q_t,\lambda) = b_{q_1}(O_1) b_{q_2}(O_2) \ldots b_{q_T}(O_T)$$

$$p(Q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \ldots a_{q_{T-1} q_T}$$

$$p(O,Q|\lambda) = p(O|Q,\lambda)\, p(Q|\lambda)$$

$$p(O|\lambda) = \sum_{\forall Q} p(O,Q|\lambda) = \sum_{\forall Q} p(O|Q,\lambda)\, p(Q|\lambda)$$

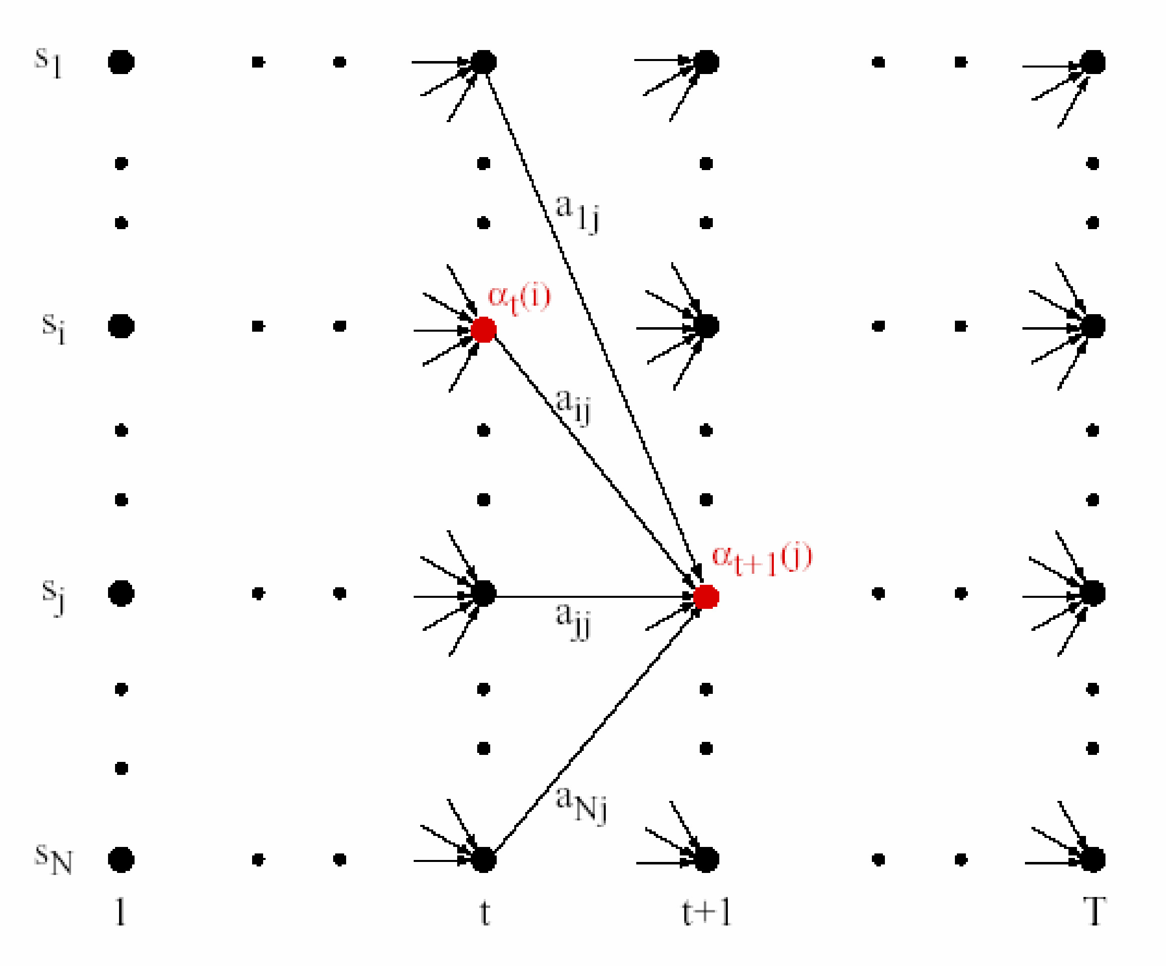
  - Very costly: $O(N^T)$
  - Underflow problems

Hidden Markov Models (HMMs). Evaluation (II)

- Forward $O(N^2T)$ $$\alpha_t(i) = p(O_1O_2\ldots O_t, q_t = S_i|\lambda)$$
  - The forward variable is defined as:
    - The probability of the partial observation sequence up to time t and state s_i at time t, given the model λ.
  - Initialization $$\alpha_1(i) = \pi_i b_i(O_1); \quad 1 \le i \le N$$
  - Recursion $$\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij}\right] b_j(O_t); \quad \begin{matrix} 1 \le t \le T \\ 1 \le j \le N \end{matrix}$$
  - Finalization $$p(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$
  - Computing cost: $O(N^2 T)$, instead of $O(N^T)$

Hidden Markov Models (HMMs). Evaluation (III)

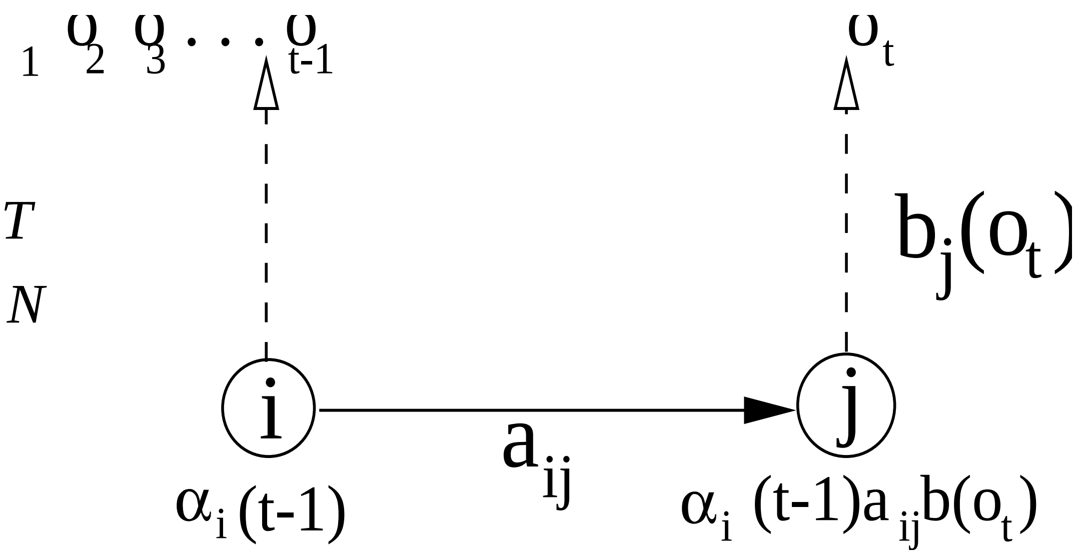
- Forward



Hidden Markov Models (HMMs). Evaluation (IV)

- Forward:
  - $\alpha_{t-1}(i)\, a_{ij}$ = joint probability of being in state i in time t-1 and making a transition to state j
  - The Σ for all previous states in t-1 = prob of being in state j in time t with the sequence until O_{t-1} being generated
  - With the final multiplication by $b_j(O_t)$ (prob of generating observation O_t in state j), we obtain $\alpha_t(j)$.

$$\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} \right] b_j(O_t); \qquad \begin{array}{l} 1 \le t \le T \\ 1 \le j \le N \end{array}$$

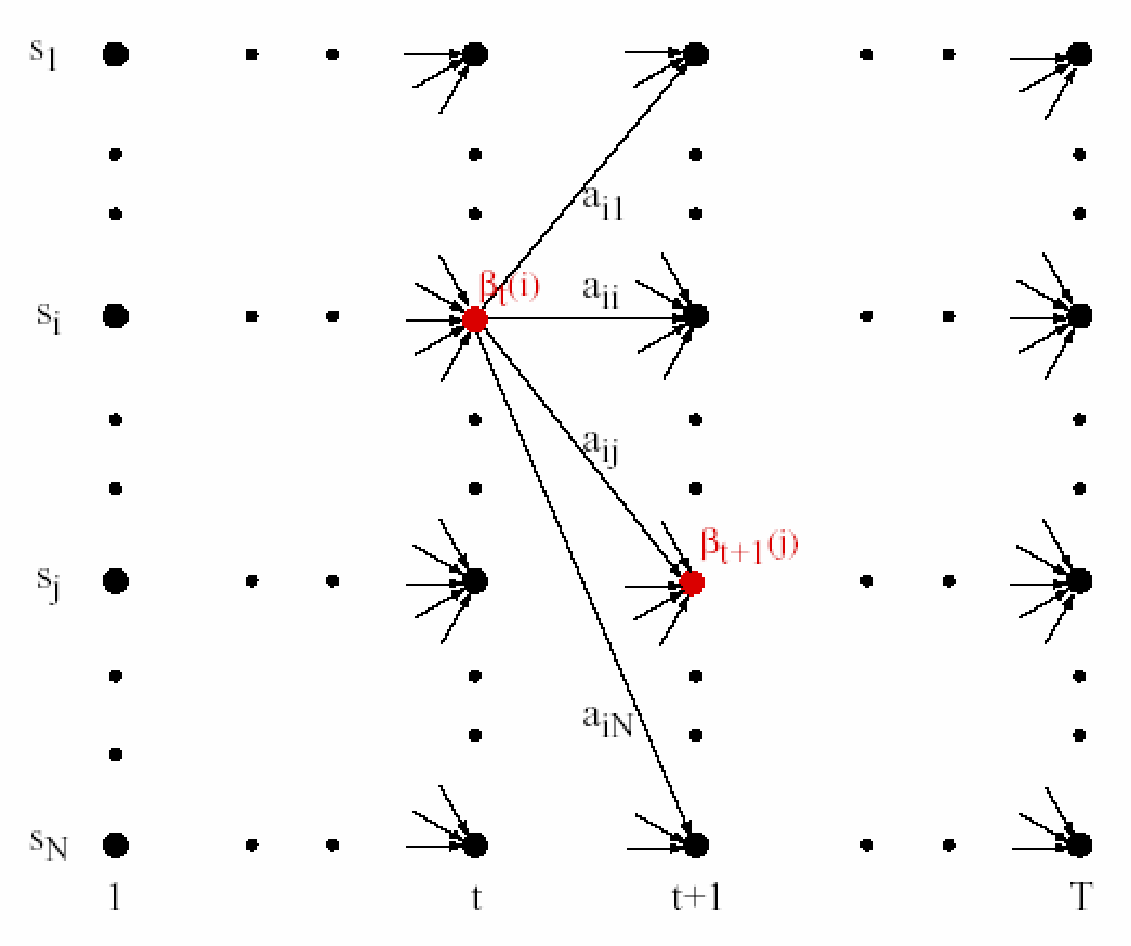


Hidden Markov Models (HMMs). Evaluation (V)

- Backward $O(N^2T)$ $$\beta_t(i) = p(O_{t+1}O_{t+2}\ldots O_T, q_t = S_i|\lambda)$$
  - The backward variable is defined as:
    - The probability of the partial observation sequence from time t+1 up to T, and state s_i at time t, given the model λ.
  - Initialization $$\beta_T(i) = 1; \quad 1 \le i \le N$$
  - Recursion $$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j); \quad \begin{matrix} 1 \le t \le T-1 \\ 1 \le i \le N \end{matrix}$$
  - Finalization $$p(O|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i)$$

Hidden Markov Models (HMMs). Evaluation (VI)

- Backward



Hidden Markov Models (HMMs). Segmentation (I)

– Given the observation sequence $O=\{O_1, O_2, \ldots, O_T\}$ and model λ

  - How do we choose a state sequence $Q=\{q_1, q_2, \ldots, q_T\}$ which is optimum in some sense?
  - Example: choose the most probable state sequence

– Viterbi algorithm

  - Based in **dynamic programming** (optimization of sequential decision processes). **Optimality principle.**
  - Similar to *forward* (maximization instead of addition)
  - To retrieve the state sequence, we must keep track of the state sequence which gave the best path, at time t, to state s_i

Hidden Markov Models (HMMs). Segmentation (II)

- **Viterbi algorithm**:
  - Initialization

Not used in t=1

$$\delta_1(i) = \pi_i b_i(O_1); \quad \psi_1(i) = 0; \quad 1 \le i \le N$$

  - Recursion (decision on a local optimum)

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

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

$$\psi_t(j) = \underset{1 \le i \le N}{\operatorname{argmáx}}\left[\delta_{t-1}(i) a_{ij}\right] \quad ; \quad 1 \le j \le N$$

  - Finalization

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

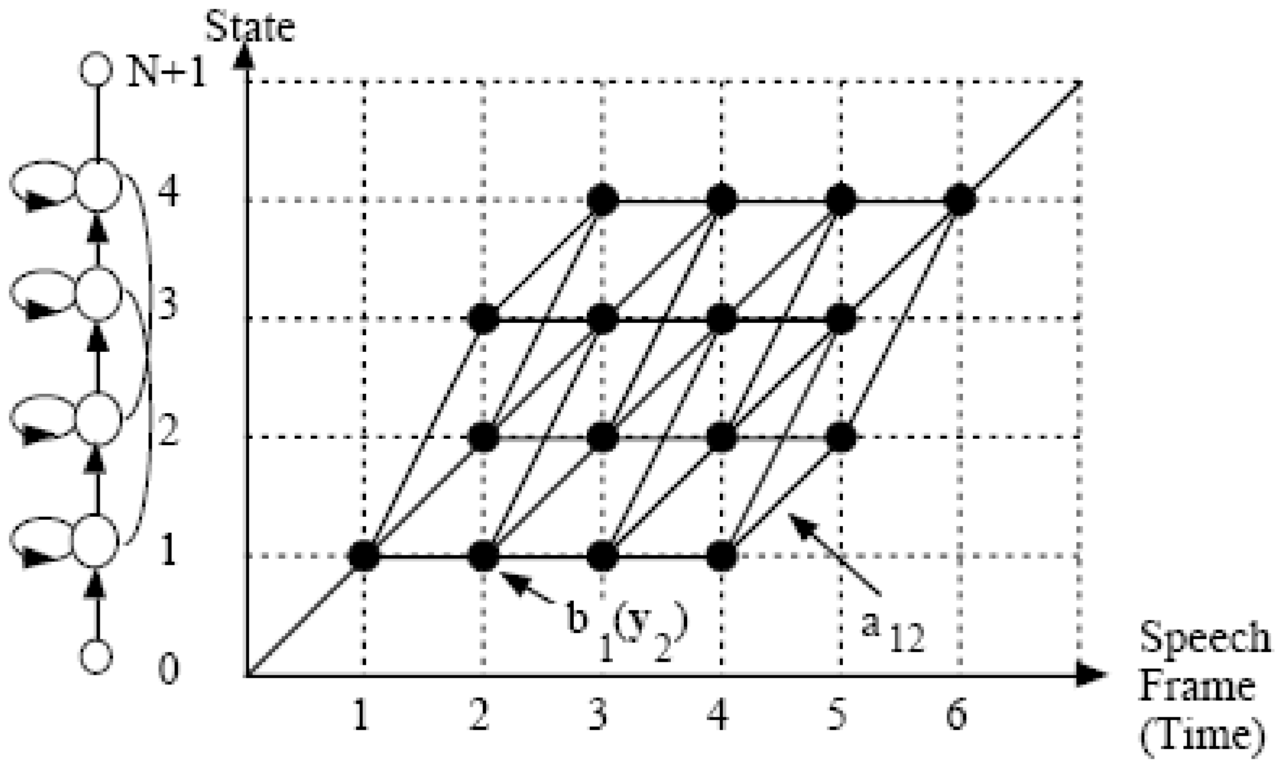
$$q_T^* = \underset{1 \le i \le N}{\operatorname{argmáx}}\left[\delta_T(i)\right]$$

Backtracking

$$q_t^* = \psi_{t+1}\left(q_{t+1}^*\right); \quad t = T-1, T-2, \ldots, 1$$

Hidden Markov Models (HMMs). Segmentation (III)

- **Viterbi algorithm:**



Hidden Markov Models (HMMs). Segmentation (IV)

- **Viterbi algorithm**:
  - The Segmentation problem is solved (the state sequence is obtained)
  - The Evaluation problem is also solved:
    - Even though the probability is not exact (as in forward-backward) because maximizations instead of additions are made
    - It can be used to compare the probabilities obtained for different models,
      - Which is the basic task in speech recognition
    - The recognized word is the one with the highest probability

CEU Universidad San Pablo