Chapter 3:
Hidden Markov Models (part 4)

q Introduction

q Evaluation problem

q Decoding problem

q Learning problem

q Example

HIDDEN MARKOV MODELS

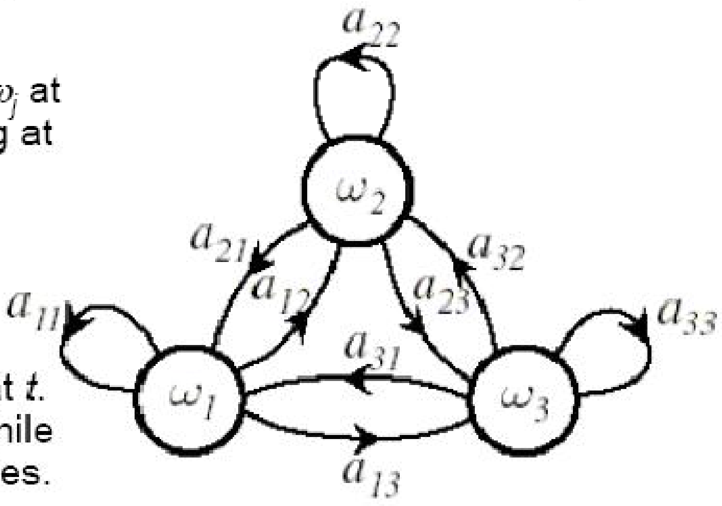
- **Problem:** determine parameters of class-conditional probabilities when classification decision is independent of previous history.
- **Solution**: ML or MAP estimation.

- **Problem:** estimate class probabilities when classification decision at moment *t+1* is directly influenced by decision at moment *t*.
- Examples: speech recognition, gesture recognition, etc.
- **Hidden Markov Model** (**HMM**)**:** sequence of system states is described by *transition probabilities* (from one moment to the next).

- **Representation:**
  - The transition probability that state ω_j at moment *t+1* follows state ω_i (existing at moment *t*) is noted with a_{ij} and it is assumed to be time-independent.

$$P(\omega_j(t+1)\,|\,\omega_i(t)) = a_{ij}$$

  - State at *t+1* depends only on state at *t*.
  - States are represented by nodes, while the links are the transition probabilities.



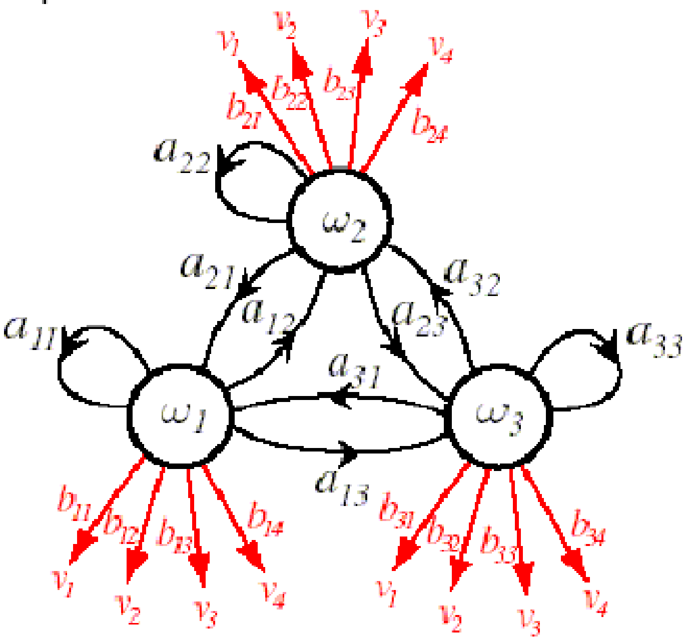
FIRST-ORDER HMM

Requirements:

- At any moment *t*, the actual state of the system ω(t) is not observable (i.e. it is said to be *hidden*). Instead, the the system emits a visible state *v(t)*.
- Probability of system state at *t+1* depends only on of the state at *t*.
- The *transition probability* from *t* to *t+1*, a_{ij}, and the probability b_{jk} of emitting a particular visible state are both time-independent.

$$\boxed{a_{ij} = P(\omega_j(t+1) \mid \omega_i(t))} \longrightarrow \boxed{\sum_j a_{ij} = 1}$$

$$\boxed{b_{jk} = P(v_k(t) \mid \omega_j(t))} \longrightarrow \boxed{\sum_k b_{jk} = 1}$$



Notations:

- A particular sequence of length *T* is denoted by ω^T =[ω(1), ω(2), …, ω(T)].
- The associated sequence of visible states is V^T =[v(1), v(2), …, v(T)].

MAIN PROBLEMS

❑ Evaluation Problem:

- Given a complete HMM, including probabilities a_{ij}, and b_{jk}, determine the probability of occurrence for a particular sequence of visible states V^T.

$$\boxed{P(V^T) = f(a_{ij}, b_{jk})}$$

❑ Decoding Problem:

- Given a complete HMM, determine the most likely sequence of hidden states ω^T that might generate the observed sequence of visible states V^T.

$$\boxed{\omega^T = f(a_{ij}, b_{jk}, V^T)}$$

❑ Learning Problem:

- Given a coarse structure of a HMM (in term of number of hidden states and number of visible states), determine the probabilities a_{ij}, and b_{jk} from the observed sequence of visible states V^T.

EVALUATION

- If the model has c hidden states, then the maximum number of possible sequences of T hidden states is $r_{max} = c^T$.
- The probability that the model produces a sequence of V^T visible states is:

$$P(V^T) = \sum_{r=1}^{r_{max}} P(V^T \mid \omega_r^T) P(\omega_r^T) = \sum_{r=1}^{r_{max}} \prod_{t=1}^{T} P(\mathbf{v}(t) \mid \omega(t)) P(\omega(t) \mid \omega(t-1))$$

(1) $$P(V^T \mid w_r^T) = \prod_{t=1}^{t=T} P(v(t) \mid w(t)) \text{ conditional independence}$$

(2) $$P(w_{\mathrm{r}}^{\mathrm{T}}) = \prod_{t=1}^{t=T} P(w(t) \mid w(t-1)) \text{ Markov chain of order 1}$$

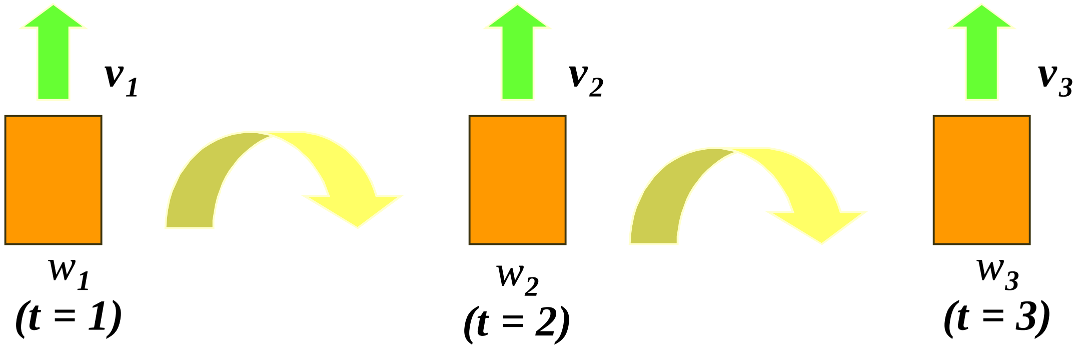
Interpretation: The probability that we observe the particular sequence of T visible states V^T is equal to the sum over all r_{max} possible sequences of hidden states of the conditional probability that the system has made a particular transition multiplied by the probability that it then emitted the visible symbol in our target sequence.

Example: Let ω_1, ω_2, ω_3 be the hidden states; v_1, v_2, v_3 be the visible states and $V^3 = \{v_1, v_2, v_3\}$ is the sequence of visible states

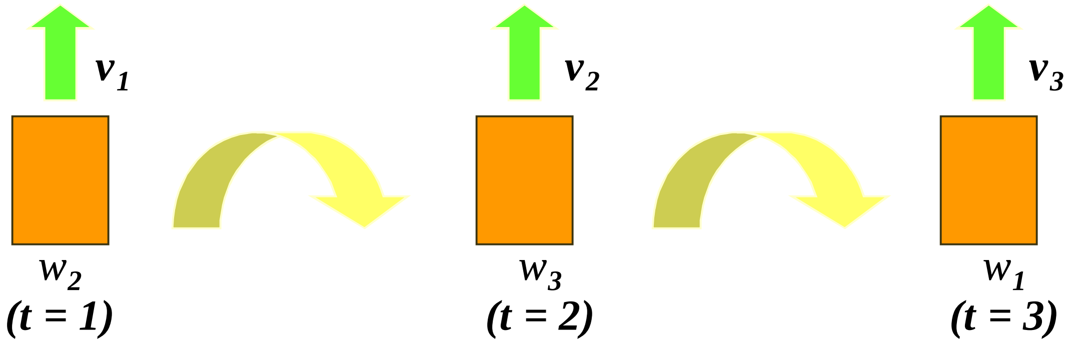
$P(\{v_1, v_2, v_3\}) = P(\omega_1).P(v_1 \mid \omega_1).P(\omega_2 \mid \omega_1).P(v_2 \mid \omega_2).P(\omega_3 \mid \omega_2).P(v_3 \mid \omega_3)$

$+\ldots+$ (possible terms in the sum = all possible (3^3= 27) cases !)

First possibility:



Second Possibility:



P({v1, v2, v3}) = P(w2).P(v1 | w2).P(w3 | w2).P(v2 | w3).P(w1 | w3).P(v3 | w1)
+ …+

Therefore:

$$P(\{v_1,v_2,v_3\}) = \underset{\substack{\text{possible sequence}\\\text{of hidden states}}}{\text{å}} \overset{t=3}{\underset{t=1}{\text{Õ}}} P(v(t)|w(t)).P(w(t)|w(t-1))$$

The evaluation problem is solved using the forward algorithm

- *$P(V^T)$* is computed recursively. Let *$\alpha_j(t)$* be the probability that the HMM is in hidden state ω_j at step *t* having generated the first *t* elements of V^T:

$$\alpha_j(t) = \begin{cases} 0 & t = 0 \ and \ \ j \neq initial \ state \\ 1 & t = 0 \ and \ \ j = initial \ state \\ \sum_{i-1}^{c} \alpha_i(t-1) a_{ij} b_{jk} v(t) & t > 0 \end{cases}$$

EVALUATION ALGORITHM

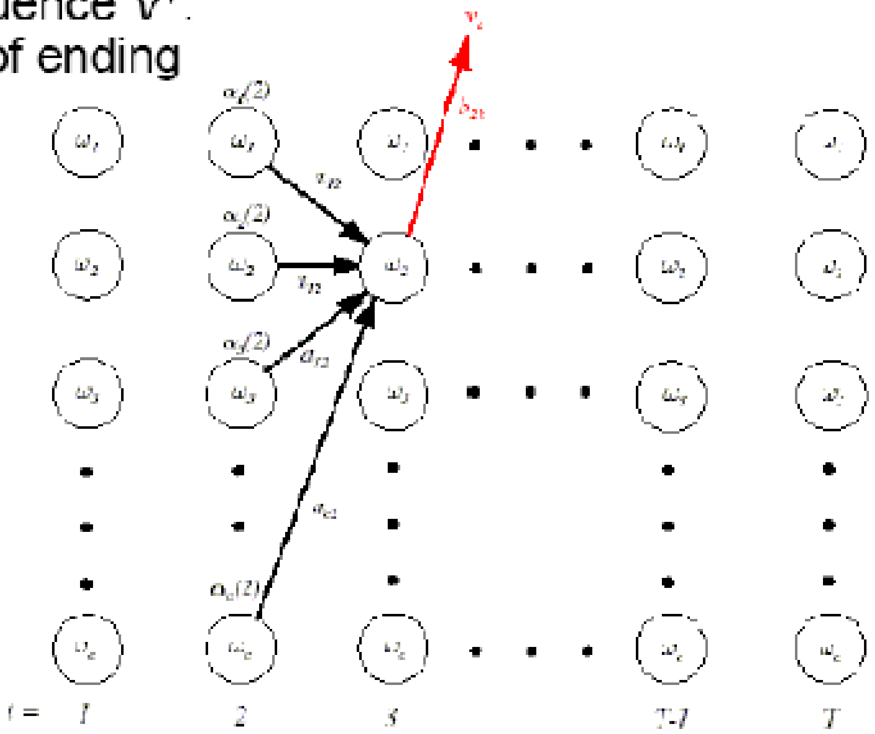
- □ **Input:** a_{ij}, b_{jk}, visible sequence V^T.
- □ **Output:** $\alpha(T)$= probability of ending in the known final state.

- □ **Algorithm:**

```
INITIALIZE:   α(0) = 1
DO            t ← t+1
```

$$\boxed{\alpha_j(t) \leftarrow b_{jk}v_k(t)\sum_{i=1}^{c}\alpha_i(t-1)a_{ij}}$$

```
UNTIL         t = T
RETURN        P(V^T) ← α(T)
```



Suppose we seek the probability that the HMM was in state ω_2 at $t - 3$ and generated the observed visible symbol up through that step (including the observed visible symbol v_k) The probability the HMM was in state $\omega_j(t = 2)$ and generated the observed sequence through $t = 2$ is $\alpha_j(2)$ for $j = 1, 2, \ldots, c$. To find $\alpha_2(3)$ we must sum these and multiply the probability that state ω_2 emitted the observed symbol v_k.

Decoding problem (optimal state sequence)

Given a sequence of visible states V^T, the decoding problem is to find the most probable sequence of hidden states.

This problem can be expressed mathematically as:

find the single “best” state sequence (hidden states)

$$\hat{w}(1),\hat{w}(2),\ldots,\hat{w}(T) \;\; such \;\; that:$$

$$\hat{w}(1),\hat{w}(2),\ldots,\hat{w}(T) = \underset{w(1),w(2),\ldots,w(T)}{\arg\max} \; P\left[w(1),w(2),\ldots,w(T),v(1),v(2),\ldots,V(T) \mid \Theta\right]$$

q Note that the summation disappeared, since we want to find only one unique best case !

Where: $\Theta = [\pi, A, B]$

$\pi = P(\omega(0) = \omega)$ (initial state probability)

$A = a_{ij} = P(\omega_j(t+1) \mid \omega_i(t))$

$B = b_{jk} = P(v_k(t) \mid \omega_j(t))$

In the preceding example, this computation corresponds to the selection of the best path amongst:

$$\{\omega_1(t = 1), \omega_2(t = 2), \omega_3(t = 3)\},\ \{\omega_2(t = 1), \omega_3(t = 2), \omega_1(t = 3)\}$$
$$\{\omega_3(t = 1), \omega_1(t = 2), \omega_2(t = 3)\},\ \{\omega_3(t = 1), \omega_2(t = 2), \omega_1(t = 3)\}$$
$$\{\omega_2(t = 1), \omega_1(t = 2), \omega_3(t = 3), \ldots\}$$

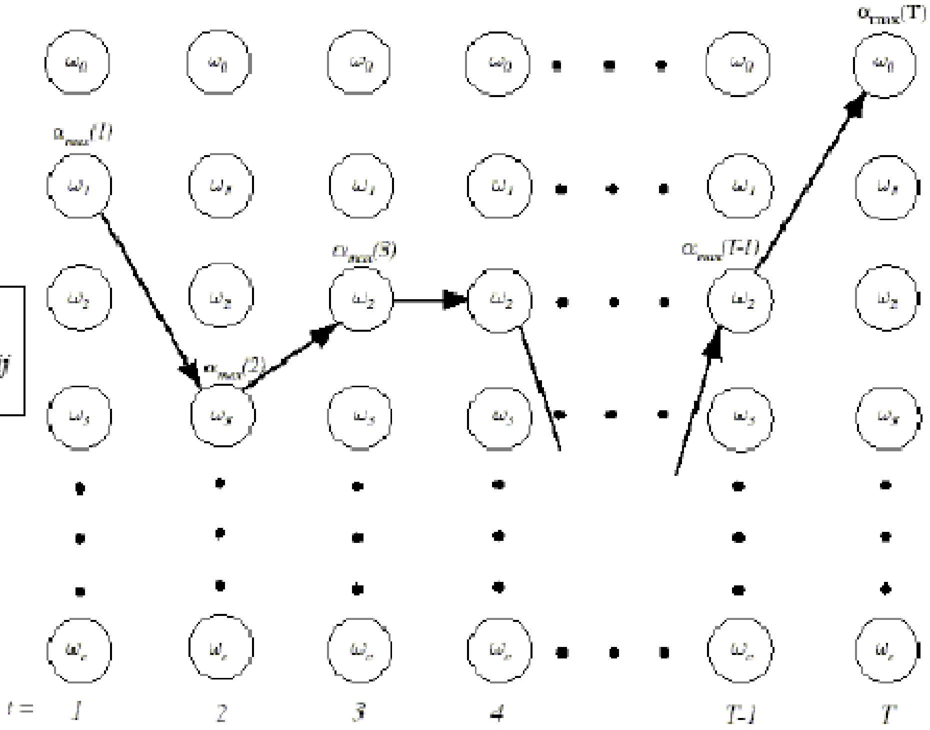
- The decoding problem is solved using the Viterbi Algorithm

DECODING

- The decoding algorithm finds at each time step *t* the state that has the highest probability of having come from the previous step and generated the observed visible state v_k. The full path is the sequence of such states.

- **Algorithm:**

```
INITIALIZE:    Path = $\varnothing$
DO             $t \leftarrow t+1$
        $j = 0,\ \alpha_0 = 1$
        DO     $j \leftarrow j+1$
     $\boxed{\alpha_j(t) \leftarrow b_{jk} v_k(t) \sum_{i=1}^{c} \alpha_i(t-1) a_{ij}}$
        UNTIL  $j = c$
           $\boxed{p = \arg\max_j \alpha_j(t)}$
        Append $\omega_p$ to Path
UNTIL          $t = T$
RETURN         Path
```



Learning problem (parameter estimation)

This third problem consists of determining a method to adjust the model parameters Θ = [π,A,B] to satisfy a certain optimization criterion. We need to find the best model

$$\hat{\Theta} = [\hat{p}, \hat{A}, \hat{B}]$$

Such that to maximize the probability of the observation sequence:

$$\underset{\Theta}{Max}\, P(V^T \mid \Theta)$$

We can use an iterative procedure such as Baum-Welch (Forward-Backward) or Gradient to find this local optimum

Parameter Updates:

Forward-Backward Algorithm

$$g_{ij}(t) = \frac{a_i(t-1)a_{ij}b_{jk}b_j(t)}{P(V^T|\Theta)}$$

- $\alpha_i(t)$= P(model generates visible sequence up to step t given hidden state $\omega_i(t)$)

- $\beta_i(t)$= P(model will generate the sequence from t+1 to T given $\omega_i(t)$)

$$a_j(t) = \begin{cases} 0 & t = 0 \, and \quad j \neq initial \quad state \\ 1 & t = 0 \, and \quad j = initial \quad state \\ \sum_{i=1}^{c} a_i(t-1)a_{ij}b_{jk}v(t) & t > 0 \end{cases}$$

$$b_i(t) = \begin{cases} 0 & t = T \, and \quad i \neq final \quad state \\ 1 & t = T \, and \quad i = final \quad state \\ \sum_{j=1}^{c} b_j(t+1)a_{ij}b_{jk}v(t+1) & t < T \end{cases}$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T} g_{ij}(t)}{\sum_{t=1}^{t=T}\sum_k g_{ik}(t)}$$

$$\hat{b}_{jk} = \frac{\sum_{\substack{t=1 \\ v(t)=v_k}}^{T} \sum_l g_{jl}(t)}{\sum_{t=1}^{T}\sum_l g_{jl}(t)}$$

Parameters Learning Algorithm

```
Begin initialize

  $a_{ij}$, $b_{jk}$, training sequence $V^T$, convergence criterion
    (cc), z=0
    Do z=z+1
          compute $\hat{a}(z)$ from a(z-1) and b(z-1)
          compute $\hat{b}(z)$ from a(z-1) and b(z-1)
            $a_{ij}(z)= \hat{a}_{ij}(z-1)$
            $b_{jk}(z)= \hat{b}_{jk}(z-1)$
    Until $\max\{a_{ij}(z)-a_{ij}(z-1),b_{jk}(z)-b_{jk}(z-1)\}<$ cc
    Return $a_{ij}=a_{ij}(z)$; $b_{jk}=b_{jk}(z)$
End
```

SPEECH RECOGNITION

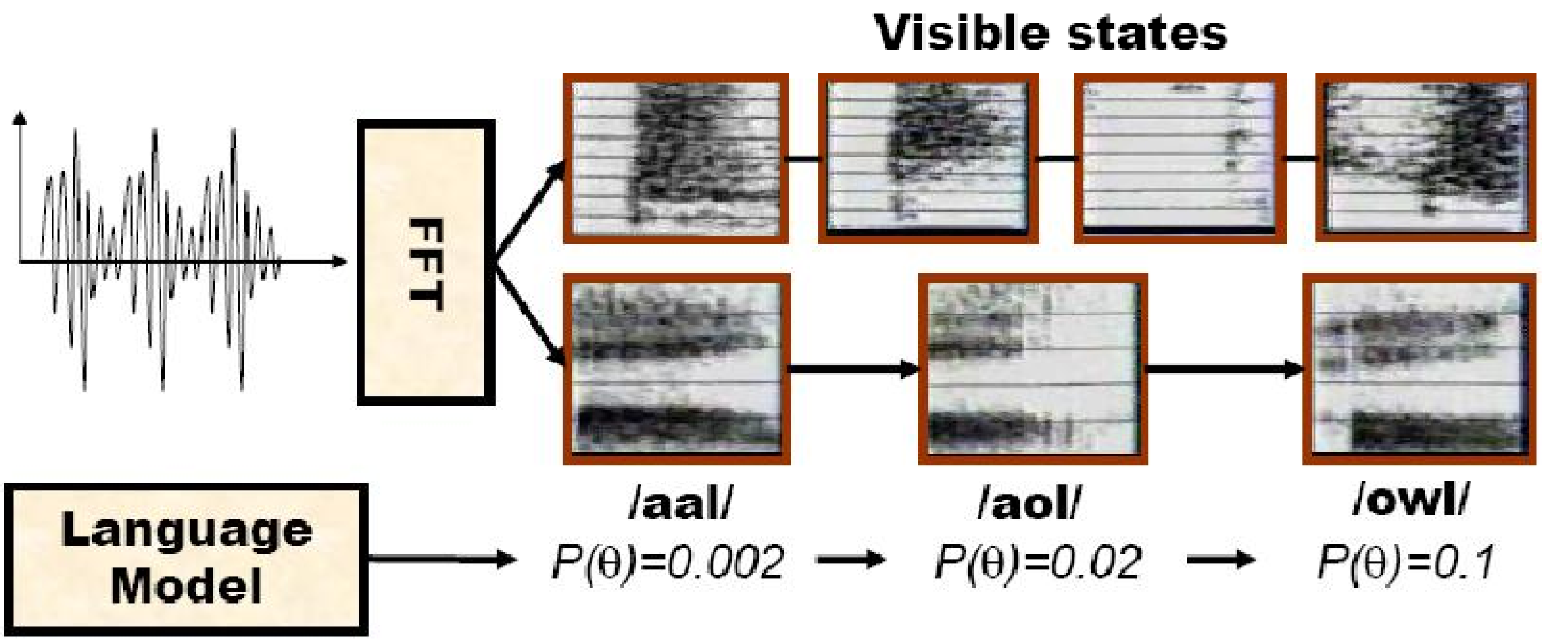
- Suppose that a given HMM model of $\{a_{ij}, b_{jk}\}$ is denoted by θ.
- For speech recognition we need such a model for each recognizable word. For example, there is a model for "cat", another for "dog", etc.

- According to Bayes theorem:

$$P(\theta \,|\, V^T) = \frac{P(V^T \,|\, \theta) P(\theta)}{P(V^T)}$$

- **Left-to-right model:**
  - Since:
    - $P(V^T)$ is independent of word classification and can be ignored.
    - $P(\theta)$ is provided by an external source such as a **language model**.
    - $P(V^T|\theta)$ is given by the HMM **evaluation algorithm**.
  - Classification can be achieved by maximizing $P(\theta|V^T)$: word selected corresponding to the highest probability.

EXAMPLE



Visible states

FFT

Language Model

/aal/ $P(\theta)=0.002$ → /aol/ $P(\theta)=0.02$ → /owl/ $P(\theta)=0.1$

The word classification as "owl" is achieved in the left-to-right model by maximizing the conditional probability $P(\theta|V^T)$ of the word model with respect to the sequence of visible states:

- $P(\theta)$ is provided by the language model.
- $P(V^T|\theta)$ is given by the HMM evaluation algorithm.