HMM: Three Fundamental Problems

Problem 1 (Likelihood):

Given an HMM $\lambda = (\mathcal{T}, \mathcal{B})$ and an observation sequence O, determine the likelihood $P(O \mid \lambda)$.

Problem 2 (Decoding):

Given an observation sequence O and an HMM $\lambda = (\mathcal{T}, \mathcal{B})$, discover the best hidden state sequence.

Problem 3 (Learning):

Given an observation sequence O and the set of states in the HMM, learn the HMM parameters $\mathcal{T}$ and $\mathcal{B}$.

Brute Force Enumeration

Algorithm 1 Brute Force Likelihood

1: Enumerate all possible hidden state sequences (N^T of them)
2: **for** each hidden state sequence **do**
3: Compute $p(o_{1:T} \mid s_{1:T})p(s_{1:T})$
4: **end for**
5: Add up all the above obtained $p(o_{1:T}, s_{1:T})$ to marginalize over all possible hidden state sequences

Do you see a problem with this approach?

Forward Algorithm

$$\alpha_t(j) = p(o_1, o_2, \ldots, o_t, s_t = j)$$

1. Initialization:

$$\alpha_1(j) = p(s_1 = j) \cdot p(o_1 \mid s_1 = j) \qquad 1 \leq j \leq N$$

2. Recursion:

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) \cdot p(s_t = j \mid s_{t-1} = i) \cdot p(o_t \mid s_t = j) \qquad 1 \leq j \leq N, 1 < t \leq T$$

3. Termination:

$$p(O) = \sum_{i=1}^{N} \alpha_T(i) \qquad \sum_{i=1}^{N} p(o_1, \ldots, o_T, s_T = i)$$

Marginalise over final state

Forward Algo: Rationale

$$\alpha_t(j) = p(o_1, o_2, \ldots, o_t, s_t = j)$$

$$= \sum_{i=1}^{N} p(o_1, o_2, \ldots, o_{t-1}, s_{t-1} = i, o_t, s_t = j)$$

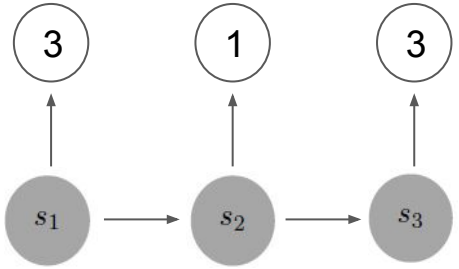
$$= \sum_{i=1}^{N} p(o_{1:t-1}, s_{t-1} = i) \cdot p(o_t, s_t = j \mid o_{1:t-1}, s_{t-1} = i)$$

$$= \sum_{i=1}^{N} p(o_{1:t-1}, s_{t-1} = i) \cdot p(s_t = j \mid o_{1:t-1}, s_{t-1} = i) \cdot p(o_t \mid o_{1:t-1}, s_{t-1} = i, s_t = j)$$

$$= \sum_{i=1}^{N} p(o_{1:t-1}, s_{t-1} = i) \cdot p(o_t \mid s_t = j) \cdot p(s_t = j \mid s_{t-1} = i)$$

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) \cdot p(s_t = j \mid s_{t-1} = i) \cdot p(o_t \mid s_t = j)$$

Brute Force vs. Forward Algorithm



$$p(o_{1:T}) = \sum_{\forall\{s_{1:T}\}} p(o_{1:T}, s_{1:T})$$

$$= \sum_{\forall\{s_{1:T}\}} p(o_{1:T}|s_{1:T}) \cdot p(s_{1:T})$$

$$= \sum_{\forall\{s_{1:T}\}} \left(\prod_{i=1}^{T} p(o_i \mid s_i) \cdot p(s_{1:T}) \right)$$

$$p(o_{1:T}) = \sum_{i=1}^{N} p(o_{1:T}, s_T = s^i)$$

$$= \sum_{i=1}^{N} p(o_{1:T}, s_T = i)$$

Notation alert

Forward Algorithm

$$\alpha_t(j) = p(o_1, o_2, \ldots, o_t, s_t = j)$$

1. Initialization:

$$\alpha_1(j) = p(s_1 = j) \cdot p(o_1 \mid s_1 = j) \qquad 1 \leq j \leq N$$

2. Recursion:

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) \cdot p(s_t = j \mid s_{t-1} = i) \cdot p(o_t \mid s_t = j) \qquad 1 \leq j \leq N, 1 < t \leq T$$

3. Termination:

$$p(O) = \sum_{i=1}^{N} \alpha_T(i)$$

How do we translate this to pseudocode?

Forward Algo: Pseudocode

$$\alpha_t(j) = p(o_1, o_2, \ldots, o_t, s_t = j)$$

function FORWARD(*observations* of len T, *state-graph* of len N) **returns** *forward-prob*

create a probability matrix *forward[N,T]*
for each state s **from** 1 **to** N **do**
    $forward[s,1] \leftarrow \pi_s * b_s(o_1)$

$$\alpha_1(j) = p(s_1 = j) \cdot p(o_1 \mid s_1 = j)$$

for each time step t **from** 2 **to** T **do**
  for each state s **from** 1 **to** N **do**

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) \cdot p(s_t = j \mid s_{t-1} = i) \cdot p(o_t \mid s_t = j)$$

$$forward[s,t] \leftarrow \sum_{s'=1}^{N} forward[s',t-1] * a_{s',s} * b_s(o_t)$$

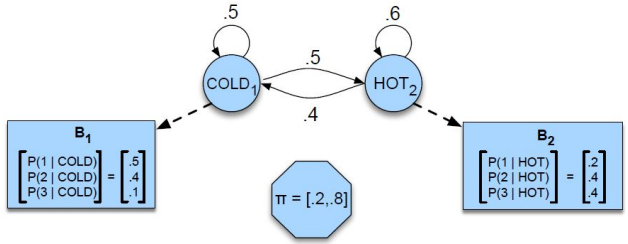
$$forwardprob \leftarrow \sum_{s=1}^{N} forward[s,T]$$

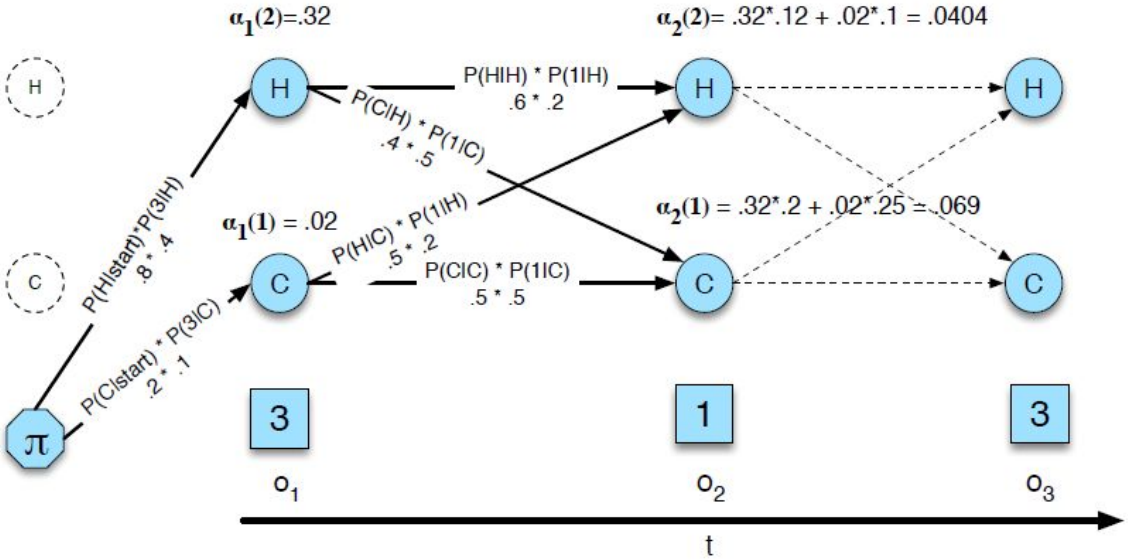
$$p(O) = \sum_{i=1}^{N} \alpha_T(i)$$

return *forwardprob*

Source: SLP, Dan Jurafsky

Forward Trellis





How do you compute the original goal: observation sequence likelihood?

Source: SLP, Dan Jurafsky

HMM: Three Fundamental Problems

Problem 1 (Likelihood):

Given an HMM $\lambda = (\mathcal{T}, \mathcal{B})$ and an observation sequence O, determine the likelihood $P(O \mid \lambda)$.

Problem 2 (Decoding):

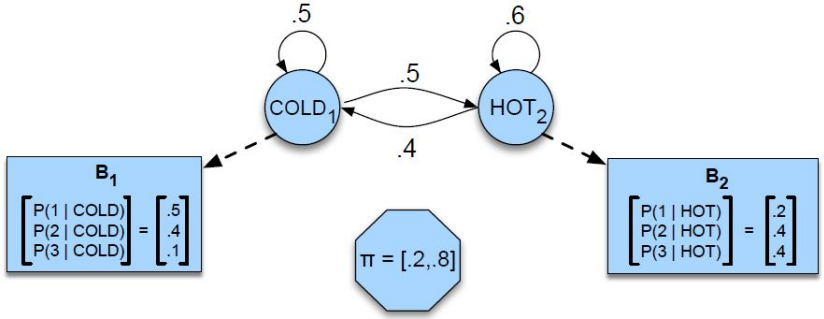
Given an observation sequence O and an HMM $\lambda = (\mathcal{T}, \mathcal{B})$, discover the best hidden state sequence.

Problem 3 (Learning):

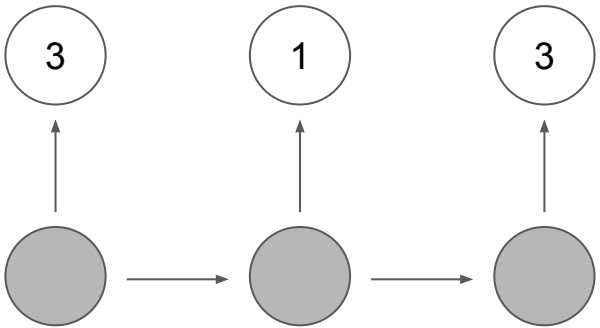
Given an observation sequence O and the set of states in the HMM, learn the HMM parameters $\mathcal{T}$ and $\mathcal{B}$.

Problem 2: Decoding

Problem 2: Decoding



➔ Find the hidden state sequence that was most likely to have generated the input observation sequence.



$p(3, 1, 3)?$ **Earlier goal**

What should the goal be now?

$$\arg\max_{s_1,s_2,s_3} p(s_1, s_2, s_3 \mid o_1 = 3, o_2 = 1, o_3 = 3)?$$

Decoding: Equivalent Objective

$$\arg\max_{s_1,s_2,s_3} p(s_1, s_2, s_3 \mid o_1 = 3, o_2 = 1, o_3 = 3)?$$

$$p(s_1, s_2, s_3 \mid o_1, o_2, o_3) = \frac{p(o_1, o_2, o_3, s_1, s_2, s_3)}{p(o_1, o_2, o_3)}$$

$$\arg\max_{s_1,s_2,s_3} p(s_1, s_2, s_3 \mid o_1, o_2, o_3) = \arg\max_{s_1,s_2,s_3} p(o_1, o_2, o_3, s_1, s_2, s_3)$$

Where have we seen this quantity before?

$$p(o_{1:3}, s_{1:3})$$

$$p(o_{1:3}) = \sum\nolimits_{\forall\{s_{1:3}\}} p(o_{1:3}, s_{1:3})$$

Can you think of a brute force algorithm for decoding?