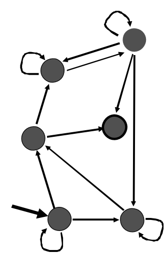
HMMs

CS224N
2004
(based on slides by David Blei, UCB)

1

HMM formalism



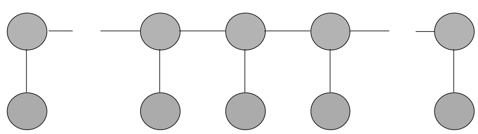
HMM = probabilistic FSA

HMM = states $s_1, s_2, \ldots$
(special start state s_1
special end state s_n)
token alphabet $a_1, a_2, \ldots$
state transition probs $P(s_i | s_j)$
token emission probs $P(a_i | s_j)$

Widely used in many language processing tasks, *e.g.*, speech recognition [Lee, 1989], POS tagging [Kupiec, 1992], topic detection [Yamron *et al*, 1998].

2

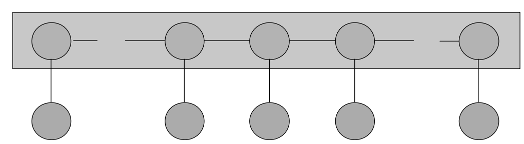
What is an HMM?



- Graphical Model Representation: Variables by time
- Circles indicate states
- Arrows indicate probabilistic dependencies between states

3

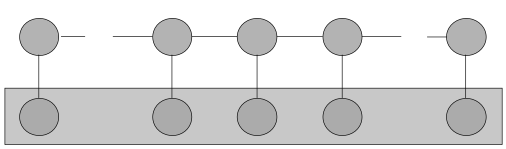
What is an HMM?



- Green circles are ***hidden states***
- Dependent only on the previous state: Markov process
- "The past is independent of the future given the present."

4

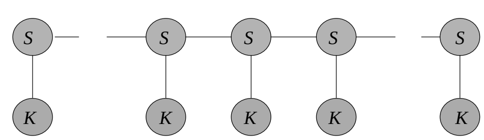
What is an HMM?



- Purple nodes are ***observed states***
- Dependent only on their corresponding hidden state

5

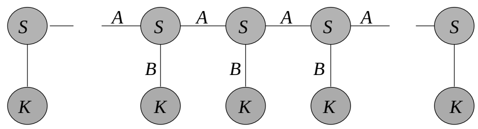
HMM Formalism



- $\{S, K, \Pi, A, B\}$
- $S : \{s_1 \ldots s_N\}$ are the values for the hidden states
- $K : \{k_1 \ldots k_M\}$ are the values for the observations

6

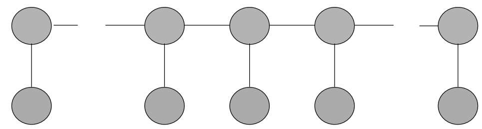
HMM Formalism



- $\{S, K, \Pi, A, B\}$
- $\Pi = \{\pi_i\}$ are the initial state probabilities
- $A = \{a_{ij}\}$ are the state transition probabilities
- $B = \{b_{ik}\}$ are the observation state probabilities

7

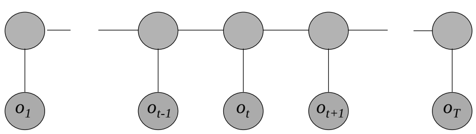
Inference for an HMM



- Compute the probability of a given observation sequence
- Given an observation sequence, compute the most likely hidden state sequence
- Given an observation sequence and set of possible models, which model most closely fits the data?

8

Sequence Probability

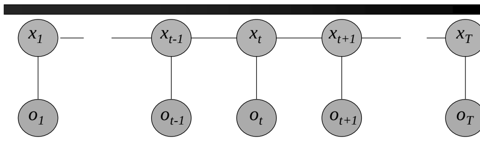


Given an observation sequence and a model, compute the probability of the observation sequence

$$O = (o_1, ..., o_T),\ \mu = (A, B, \Pi)$$
$$\text{Compute } P(O \mid \mu)$$

9

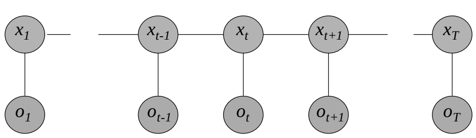
Sequence probability



$$P(O \mid X, \mu) = b_{x_1 o_1} b_{x_2 o_2} \ldots b_{x_T o_T}$$
$$P(X \mid \mu) = \pi_{x_1} a_{x_1 x_2} a_{x_2 x_3} \ldots a_{x_{T-1} x_T}$$
$$P(O, X \mid \mu) = P(O \mid X, \mu) P(X \mid \mu)$$
$$P(O \mid \mu) = \sum_X P(O \mid X, \mu) P(X \mid \mu)$$

10

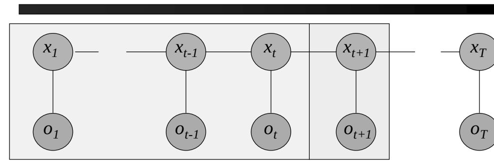
Sequence probability



$$P(O \mid \mu) = \sum_{\{x_1 \ldots x_T\}} \pi_{x_1} b_{x_1 o_1} \prod_{t=1}^{T-1} a_{x_t x_{t+1}} b_{x_{t+1} o_{t+1}}$$

11

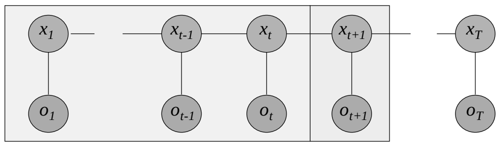
Sequence probability



- Special structure gives us an efficient solution using *dynamic programming.*
- **Intuition**: Probability of the first t observations is the same for all possible $t + 1$ length state sequences.
- **Define:** $\alpha_i(t) = P(o_1 \ldots o_t, x_t = i \mid \mu)$

12

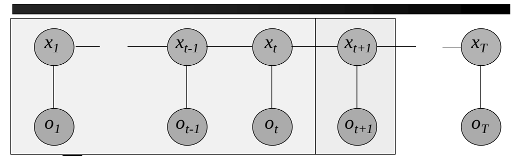
Forward Procedure



$$\begin{aligned}\alpha_j(t+1) &= P(o_1...o_{t+1}, x_{t+1}=j)\\ &= P(o_1...o_{t+1} \mid x_{t+1}=j)P(x_{t+1}=j)\\ &= P(o_1...o_t \mid x_{t+1}=j)P(o_{t+1} \mid x_{t+1}=j)P(x_{t+1}=j)\\ &= P(o_1...o_t, x_{t+1}=j)P(o_{t+1} \mid x_{t+1}=j)\end{aligned}$$

13

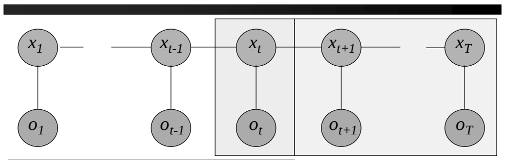
Forward Procedure



$$\begin{aligned} &= \sum_{i=1...N} P(o_1...o_t, x_t = i, x_{t+1}=j)P(o_{t+1} \mid x_{t+1}=j)\\ &= \sum_{i=1...N} P(o_1...o_t, x_{t+1}=j \mid x_t = i)P(x_t = i)P(o_{t+1} \mid x_{t+1}=j)\\ &= \sum_{i=1...N} P(o_1...o_t, x_t = i)P(x_{t+1}=j \mid x_t = i)P(o_{t+1} \mid x_{t+1}=j)\\ &= \sum_{i=1...N} \alpha_i(t) a_{ij} b_{jo_{t+1}}\end{aligned}$$

14

Backward Procedure



$$\beta_i(T+1) = 1$$

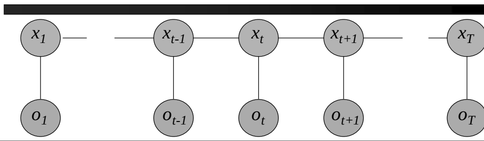
$$\beta_i(t) = P(o_t...o_T \mid x_t = i)$$

$$\beta_i(t) = \sum_{j=1...N} a_{ij} b_{io_t} \beta_j(t+1)$$

Probability of the rest of the states given the first state

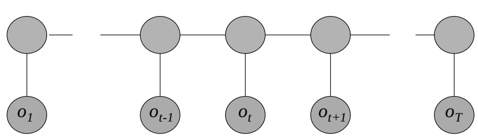
15

Sequence probability



$P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(T)$	**Forward Procedure**
$P(O \mid \mu) = \sum_{i=1}^{N} \pi_i \beta_i(1)$	**Backward Procedure**
$P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(t)\beta_i(t)$	**Combination**

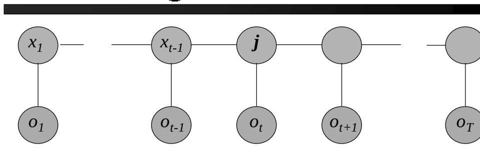
Best State Sequence



- Find the state sequence that best explains the observations
- **Viterbi** algorithm (1967)
- $\arg\max_X P(X \mid O)$

17

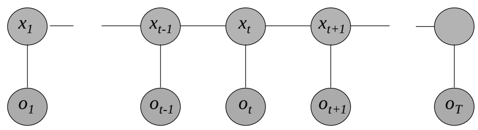
Viterbi Algorithm



$$\delta_j(t) = \max_{x_1...x_{t-1}} P(x_1...x_{t-1}, o_1...o_{t-1}, x_t = j, o_t)$$

The state sequence which maximizes the probability of seeing the observations to time t-1, landing in state j, and seeing the observation at time t

Viterbi Algorithm



$$\delta_j(t) = \max_{x_1 \dots x_{t-1}} P(x_1 \dots x_{t-1}, o_1 \dots o_{t-1}, x_t = j, o_t)$$

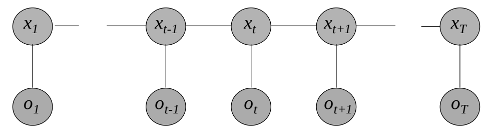
$$\delta_j(t+1) = \max_i \delta_i(t) a_{ij} b_{jo_{t+1}}$$

$$\psi_j(t+1) = \arg\max_i \delta_i(t) a_{ij} b_{jo_{t+1}}$$

Recursive Computation

19

Viterbi Algorithm



$$\hat{X}_T = \arg\max_i \delta_i(T)$$

$$\hat{X}_t = \psi_{\hat{X}_{t+1}}(t+1)$$

$$P(\hat{X}) = \arg\max_i \delta_i(T)$$

Compute the most likely state sequence by working backwards

20

Is it that easy?

- As often with text, the biggest problem is the *sparseness* of observations (words)
- Need to use many techniques to do it well
  - *Smoothing* (as in NB) to give suitable nonzero probability to unseens
  - *Featural decomposition* (capitalized?, number?, etc.) gives a better estimate
  - *Shrinkage* allows pooling of estimates over multiple states of same type (e.g., prefix states)
  - Well designed (or learned) HMM topology

21