Modeling for Discrete Features
Hidden Markov Models I

Matthew Stone
CS 520, Spring 2000
Lecture 8

Relaxing Independence Assumptions

- **Want to specify**

$$P(\mathbf{x} \mid \omega_i) \qquad \mathbf{x} \in \Delta^k$$

  – few parameters for training and inference
  – but accurate representation of distribution

- **Seen two extremes**
  – full list and naïve Bayes

Relaxing Independence Assumptions

- **Intermediate specs depend on problem**
- **Start with an important special case: sequential features**

- **Key assumption: Markov property**
  – At each step in the sequence, the state depends only on the previous state

Some Terminology

- **We'll reserve class or category to refer to the *c* alternative ω_i**
- **We'll use state to refer to the changing variable that governs successive features**
  – concrete possible states: $\delta_1, \delta_2, \cdots$
  – event of being in state *i* at step *t*: $\delta_i^{(t)}$
  – variable for events at step *t*: $\delta^{(t)}$
  – variable over sequences of events: δ

Simple Question

- **Say we observe a state sequence directly**

$$\mathbf{x} = \delta = \left\langle \delta^{(1)}, \delta^{(2)}, \cdots, \delta^{(m)} \right\rangle$$

- **Must model how likely x is for this class**

$$P(\delta \mid \omega_i, \text{len} = m)$$

(We restrict attention to sequences of length *m* for ease of normalization.)

- **For Bayes discrimination**

$$P(\omega_i \mid \delta) \propto P(\delta \mid \omega_i, \text{len} = m) P(\omega_i)$$

Modeling

- **Factor in the causal direction:**

$$P(\delta) = P(\delta^{(1)}) \prod_{t=2}^{m} P(\delta^{(t)} \mid \delta^{(1)}, \cdots, \delta^{(t-1)})$$

- **Markov property, I:** $\delta^{(t)}$ **depends only on** $\delta^{(t-1)}$

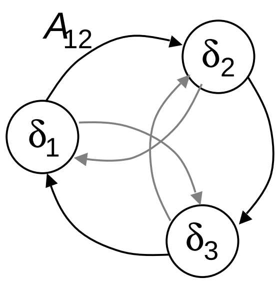
$$P(\delta) = P(\delta^{(1)}) \prod_{t=2}^{m} P(\delta^{(t)} \mid \delta^{(t-1)})$$

- **Markov property, II:**

$P(\delta^{(t)} \mid \delta^{(t-1)})$ does not vary with t

Visualization

- **Diagram of states and arcs**



Arcs determine matrix A

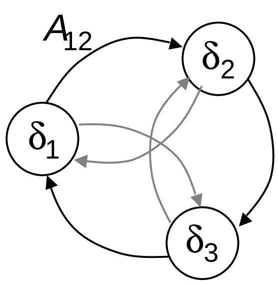
$$A_{ij} = P(\delta_j^{(t)} \mid \delta_i^{(t-1)})$$

Meas. x gives events, e.g.:

$$\mathbf{x} = \left\langle \delta_3^{(1)}, \delta_1^{(2)}, \delta_2^{(3)}, \delta_1^{(4)} \right\rangle$$

Visualization

- **Diagram of states and arcs**



Arcs determine matrix A

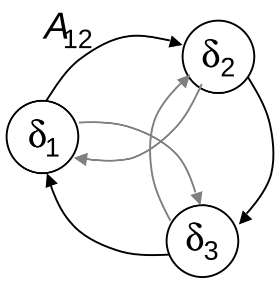
$$A_{ij} = P(\delta_j^{(t)} \mid \delta_i^{(t-1)})$$

x determines arcs used

$$A^{[\mathbf{x},t]} := A_{ij} \text{ such that } \left\langle x^{(t-1)}, x^{(t)} \right\rangle = \left\langle \delta_i, \delta_j \right\rangle$$

Visualization

- **Diagram of states and arcs**



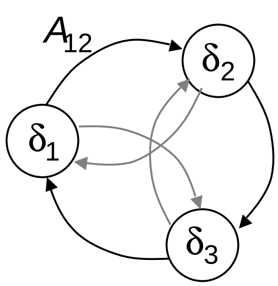
Calculate $P(\mathbf{x})$ by

$$P(\mathbf{x}) = P(x^{(1)}) \prod_{t=2}^{m} P(x^{(t)} \mid x^{(t-1)})$$

$$= P(x^{(1)}) \prod_{t=2}^{m} A^{[\mathbf{x},t]}$$

Visualization

- **Diagram of states and arcs**



Example: x gives path

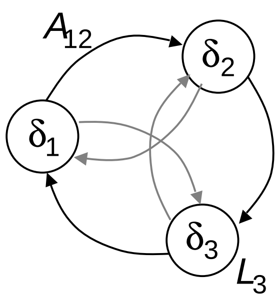
$$\mathbf{x} = \langle \delta_3^{(1)}, \delta_1^{(2)}, \delta_2^{(3)}, \delta_1^{(4)} \rangle$$

$$P(\mathbf{x}) = P(x^{(1)}) \prod_{t=2}^{m} A^{[\mathbf{x},t]}$$

$$= P(\delta_3^{(1)}) A_{31} A_{12} A_{21}$$

Visualization

- **Diagram of states and arcs**



Info about initial state

$$L_i = P(\delta_i^{(1)})$$

Example, ctd:

$$P(\mathbf{x}) = P(\delta_3^{(1)}) A_{31} A_{12} A_{21}$$
$$= L_3 A_{31} A_{12} A_{21}$$

Notation $L^{[\mathbf{x}]} := L_i$ if $x^{(1)} = \delta_i$

Classification Situation

- **Distribution on measurements by class**

$$P(\mathbf{x} \mid \omega_i, \text{len} = m)$$

- **Given by**
  - – Priors on initial states L
  - – Transition matrix A
- **Assuming**
  - – Set of n (observable) states Δ
  - – Fixed length m for sequences

Classification Situation
(CONTINUED)

- **Opportunities for finer representation**
  – Naïve Bayes has $m(n\text{-}1)$ parameters
  – Markov model has $n(n+1)$ parameters
- **Better independence assumptions**

Markov Model Uses

- **There are some problems where you can measure the changing state directly**
  – text compression
  – correcting text (OCR, typographical errors)

Markov Model Uses
(CONTINUED)

- **Treat texts as word sequences**
  - set Δ of observations (and states): words
  - matrix **A** contains estimates of

    bigram frequency by class – probability, given you see word *i* now, of seeing word *j* immediately following
  - obtained from training sequences in class by counting and smoothing

But

- **In the more frequent case:**
  - You can't observe the state directly –
  - You must infer the state given indirect measurements

- **Hidden Markov Models (HMMs) take this into account**

Extended Terminology

- **We retain states and state variables:**
  - event of being in state *i* at step *t*: $\delta_i^{(t)}$
  - variable for events at step *t*: $\delta^{(t)}$
- **We observe a symbol at each step:**
  - concrete symbols: $v_1, v_2, \cdots$
  - event of observing *i* at step *t*: $v_i^{(t)}$
  - variable for symbol at step *t*: $v^{(t)}$
  - variable over observed sequences: **v**

Extended Assumption

- **The symbol observed at time *t* depends only on the state at time *t***
  - and does not vary with *t*
  - specified by matrix **B**

$$B_{jk} = P(v_k^{(t)} \mid \delta_j^{(t)})$$

$$B^{[\mathbf{v},\delta,t]} := B_{jk} \text{ such that}$$

$$\langle \delta^{(t)}, v^{(t)} \rangle = \langle \delta_j, v_k \rangle$$

HMM Trajectory

- **Three problems must be solved to use these more flexible models**
  - Evaluation:

    Compute $P(\mathbf{v} \mid \omega_i, \text{len} = m)$
  - Decoding:

    Find $\underset{\delta}{\arg\max}\ P(\delta \mid \mathbf{v}, \omega_i)$
  - Learning:

    Train **A** and **B** given observations only

Evaluation – Theory

- **List all** s **state sequences with** m **elements:**

  $\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_s$
- **Use Markov assumption to find:**

$$P(\mathbf{s}_a) = L^{[\mathbf{s}_a]} \prod_{u=2}^{m} A^{[\mathbf{s}_a, u]}$$

- **Use observation assumption to find:**

$$P(\mathbf{v} \mid \mathbf{s}_a) = \prod_{u=1}^{m} B^{[v, \mathbf{s}_a, u]}$$

Evaluation – Theory
(CONTINUED)

- **Given observation v:**

$$P(\mathbf{v},\mathbf{s}_u) = P(\mathbf{v} \mid \mathbf{s}_u)P(\mathbf{s}_u)$$

$$= L^{[\mathbf{s}_a]} \prod_{u=2}^{m} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{m} B^{[v,\mathbf{s}_a,u]}$$

- **Thus –**

$$P(\mathbf{v}) = \sum_{a=1}^{s} \left(L^{[\mathbf{s}_a]} \prod_{u=2}^{m} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{m} B^{[v,\mathbf{s}_a,u]} \right)$$

Fortunately We Can Table Sums

- **First, two pieces of notation**
  - Probability of being in state j at step t having seen first t observations:

$$P(\delta_j^{(t)}, \mathbf{v}^{(\le t)})$$

  - Access from **B**:

$$B_j^{[\mathbf{v},t]} := B_{jk} \text{ if } v^{(t)} = v_k$$

  - Fixing a sequence to match α after t

$$\mathbf{s}^{(>t)} = \alpha$$

Tabling Sums

- **We find $P(\mathbf{v}^{(\le t)})$ as before, making an arbitrary selection among sequences:**

$$P(\mathbf{v}^{(\le t)}) = \sum_{\mathbf{s}_a^{(>t)}=\delta} \left(L^{[\mathbf{s}_a]} \prod_{u=2}^{t} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{t} B^{[v,\mathbf{s}_a,u]} \right)$$

- **Narrow to one state by restricting sum:**

$$P(\delta_j^{(t)}, \mathbf{v}^{(\le t)}) = \sum_{\mathbf{s}_a^{(>t-1)}=\delta'} \left(L^{[\mathbf{s}_a]} \prod_{u=2}^{t} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{t} B^{[v,\mathbf{s}_a,u]} \right)$$

- **Ensure match with** $\delta' := \delta[t : \delta_j]$

Tabling Sums
(CONTINUED)

- **Take current formula:**

$$P(\delta_j^{(t)}, \mathbf{v}^{(\le t)}) = \sum_{\mathbf{s}_a^{(>t-1)}=\delta'} \left(L^{[\mathbf{s}_a]} \prod_{u=2}^{t} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{t} B^{[v,\mathbf{s}_a,u]} \right)$$

- **And condition on t-1:** $\mathbf{s}_a^{(>t-2)} = \delta[t-1 : \delta_i]$

$$= \sum_{i=1}^{n} \left[\sum_{\mathbf{s}_a^{(>t-2)}=\delta'[t-1:\delta_i]} \left(L^{[\mathbf{s}_a]} \prod_{u=2}^{t} A^{[\mathbf{s}_a,u]} \prod_{u=1}^{t} B^{[v,\mathbf{s}_a,u]} \right) \right]$$

Tabling Sums
(CONTINUED)

- **Rewrite:**

$$= \sum_{i=1}^{n}\left[\sum_{\mathbf{s}_a^{(>t-2)}=\delta'[t-1:\delta_i]}\left(L^{[\mathbf{s}_a]}A^{[\mathbf{s}_a,t]}B^{[\mathbf{v},\mathbf{s}_a,t]}\prod_{u=2}^{t-1}A^{[\mathbf{s}_a,u]}\prod_{u=1}^{t-1}B^{[\mathbf{v},\mathbf{s}_a,u]}\right)\right]$$

- **And factor:**

$$= \sum_{i=1}^{n}\left[A_{ij}B_j^{[\mathbf{v},t]}\sum_{\mathbf{s}_a^{(>t-2)}=\delta'[t-1:\delta_i]}\left(L^{[\mathbf{s}_a]}\prod_{u=2}^{t-1}A^{[\mathbf{s}_a,u]}\prod_{u=1}^{t-1}B^{[\mathbf{v},\mathbf{s}_a,u]}\right)\right]$$

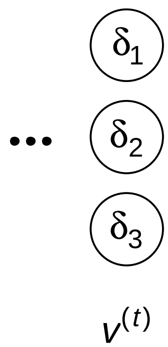
Tabling Sums
(CONTINUED)

- **And factor again:**

$$P(\delta_j^{(t)},\mathbf{v}^{(\le t)}) = \sum_{i=1}^{n}\left[A_{ij}B_j^{[\mathbf{v},t]}P(\delta_i^{(t-1)},\mathbf{v}^{(\le t-1)})\right]$$

The Big Picture

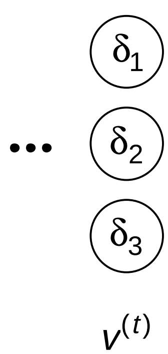
- **Recurrence says how to step forward…**



Suppose you've reached step *t*, and you've tabulated the probabilities of seeing each of the possible states...

The Big Picture

- **Recurrence says how to step forward…**



$$p_1 = P(\delta_1^{(t)}, \mathbf{v}^{(\le t)})$$

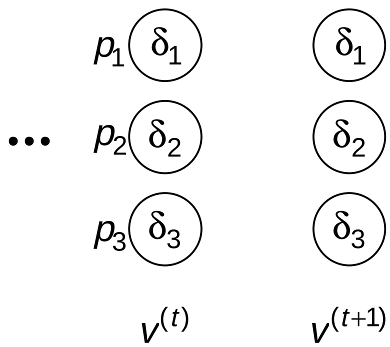
$$p_2 = P(\delta_2^{(t)}, \mathbf{v}^{(\le t)})$$

$$p_3 = P(\delta_3^{(t)}, \mathbf{v}^{(\le t)})$$

…like so

The Big Picture

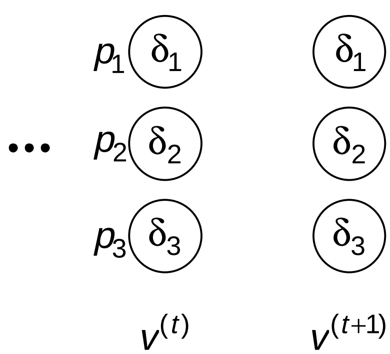
- **Recurrence says how to step forward…**



Consider the output symbol at the next step, and each of the states that might have produced it.

The Big Picture

- **Recurrence says how to step forward…**



$? = P(\delta_1^{(t+1)}, \mathbf{v}^{(\leq t+1)}) = p_1'$

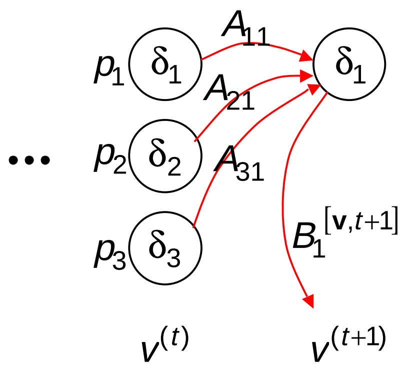
$? = P(\delta_2^{(t+1)}, \mathbf{v}^{(\leq t+1)}) = p_2'$

$? = P(\delta_3^{(t+1)}, \mathbf{v}^{(\leq t+1)}) = p_3'$

Want to assign probabilities to the new states.

The Big Picture

- **Recurrence says how to step forward…**



For each new state

$$p'_j = \sum_{i=1}^{n} \left[A_{ij} B_j^{[\mathbf{v},t+1]} p_i \right]$$

E.g.:

$$p'_1 = A_{11} B_1^{[\mathbf{v},t+1]} p_1 + A_{12} B_1^{[\mathbf{v},t+1]} p_2 + A_{13} B_1^{[\mathbf{v},t+1]} p_3$$

Evaluation – Summary

- **We have defined and justified**
  - HMM forward algorithm
  - determining probabilities of observations
- **Build table**
  - Initialize: $p_{j,0} = L_j B_j^{[\mathbf{v},0]}$
  - Step forward: $p_{j,t+1} = \sum_{i=1}^{n} \left[A_{ij} B_j^{[\mathbf{v},t+1]} p_{i,t} \right]$
  - Finish: $P(\mathbf{v} \mid \text{len} = m) = \sum_{i=1}^{n} p_{i,m}$

Use of Evaluation

- **We have *c* models**

$$\omega_a \Rightarrow \langle \mathbf{L}^a, \mathbf{A}^a, \mathbf{B}^a \rangle$$

  - Each model represents distribution over sequences in the class, e.g. –
    - likely word sequences
    - likely sound sequences for saying a word
    - likely motion patterns in gesture

Use of Evaluation (CONTINUED)

- **We get some observed sequence v**
- **We can classify v by Bayes's formula:**

$$\text{Choose } \underset{\omega_a}{\operatorname{argmax}} P(\mathbf{v} \mid \omega_a) P(\omega_a)$$

where $P(\mathbf{v} \mid \omega_a)$ is got by HMM forward