Outline of Today’s Lecture

- Hidden Markov Models: Definition and Properties
- Solving problems with HMMs and why.
books & sources

• Huang, Acero, Hon, “Spoken Language Processing”, Chapter 6.
• Deller et. al. “Discrete-time Processing of speech signals”
• Duda, Hart, & Stork, Pattern Recognition, 2001.
• Bilmes, “What HMMs can do”, technical report.

Scribes so far

<table>
<thead>
<tr>
<th>Day 1: Jeff Bilmes</th>
<th>Day 12:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day 2: Xin Lie</td>
<td>Day 13:</td>
</tr>
<tr>
<td>Day 3: Tom Mackel</td>
<td>Day 14:</td>
</tr>
<tr>
<td>Day 4: Scott Philips</td>
<td>Day 15:</td>
</tr>
<tr>
<td>Day 5: Kevin Duh</td>
<td>Day 16:</td>
</tr>
<tr>
<td>Day 6:</td>
<td>Day 17:</td>
</tr>
<tr>
<td><strong>Day 7: Travis Wilkins</strong></td>
<td>Day 18:</td>
</tr>
<tr>
<td>Day 8: Amar Subramanya</td>
<td></td>
</tr>
<tr>
<td><strong>Day 9: Andrei Alexandres</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Day 10: Dustin Lennon</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Day 11: Jeremy Holleman</strong></td>
<td></td>
</tr>
</tbody>
</table>

Missing scribes in RED
Final Projects

- There will be a deadline every week for the rest of the quarter.
- S/NS people must do a final project.
- Due dates:
  - Friday, Feb 18th: 1-page email abstract due
  - Friday, Feb 25th: 2-page progress report due
  - Friday, March 4th: 2-page progress report due
  - Friday, March 11th: Final 4-page writeup due
  - Tuesday, March 15th: Final Projects
- Stop by office hours (Tuesdays, 2:30-4:00pm) if you are unsure of what to do, or send me email (I am available most early evenings to meet).

DTW→HMM

- Cost Matrices → Stochastic Transition Matrices:

\[ A = [a_{ij}] \quad a_{ij} = \frac{e^{-w_{ij}}}{\sum_{j'} e^{-w_{ij'}}} \]

\[ \forall i, \sum_j a_{ij} = 1 \quad w_{ij} = \infty \Rightarrow a_{ij} = 0 \]

State transitions drawn only for non-zero probability state transitions
Note special “start” S and “end” state E
• State duration distribution:
  – geometric  
  \[ P(Dur_i = N) = a_{ii}^{N-1}(1 - a_{ii}) \]

  – Is this a good duration distribution for phones?
    • vowels vs. consonants?
    • Sum of geometric distributions is a negative binomial

• SFSA with probability distribution at each state:
1st order Markov Models

- 1st order Markov chain as Bayesian Network

![Bayesian Network Diagram]

- Graph represents factorization: node for each random variable, lack of edge $\approx$ some conditional independence statement. Doesn’t show time-homogeneity.

- This is **NOT** a stochastic FSA (so there are no self-loops in this kind of graph).

Hidden Markov Models

- So distribution is random given current $Q$, but once given nothing else matters. This is representable as:

$$P(X_t|Q_t, Q_1, \ldots, Q_{t-1}, Q_{t+1}, \ldots, Q_T, X_1, \ldots, X_{t-1}, X_{t+1}, \ldots, X_t) = P(X_t|Q_t)$$

- These kinds of equations get long and unwieldy. We use Dawid’s notation for conditional independence:

$$X_t \perp \{Q_{-t}, X_{-t}\} \mid Q_t$$

$$Q_{-t} \triangleq \{Q_1, \ldots, Q_{t-1}, Q_{t+1}, \ldots, Q_T\}$$

- The above + the below are the two fundamental properties that define an HMM over a set of $2T$ random variables:
Hidden Markov Models

• The past is independent of the future given the present.

\[ \{Q_{1:t-1}, X_{1:t-1}\} \perp\!\!\!\!\perp \{Q_{t:T}, X_{t:T}\} | Q_t \]

• **Definition**: an HMM is a collection of T discrete RVs and T other RVs such that the above two conditional independence properties hold true for all t. No other CI statements are true in general, unless they follow from the above two. T is itself a random variable with a mixture of sums of negative binomials distribution.

• Other CI properties might hold, but not in general.

---

Hidden Markov Models

• Simple way to describe HMM properties is using a graphical model or Bayesian network.

• In a Bayesian network, a joint distribution w.r.t. a given BN factorizes according to the network: product of probs. of children given parents.

\[
p(x_{1:T}, q_{1:T}) = p(x_1|q_1)p(q_1) \prod_{t=2}^{T} p(x_t|q_t)p(q_t|q_{t-1})
\]
Hidden Markov Models

• Or in shorthand notation:

\[ p(x_{1:T}, q_{1:T}) = \prod_{t=1}^{T} p(x_t|q_t)p(q_t|q_{t-1}) \]

• Implicitly assumes time-inhomogeneous since distributions are not function of t. Time-homog. vrsn:

\[ p(x_{1:T}, q_{1:T}) = \prod_{t=1}^{T} p_t(x_t|q_t)p_t(q_t|q_{t-1}) \]

• In time-homogeneous (or shared parameter) version, three sets of parameters associated with HMM.
  – 1: Initial state distribution:
    \[ \Pi = [\pi_i], \quad \pi_i = P(Q_1 = i) \]

Hidden Markov Models

• Three sets of HMM parameters
  – 2: state transition probabilities:
    \[ A = [a_{ij}]_{ij} \]
    \[ p(q_t|q_{t-1}) = P(Q_t = q_t|Q_{t-1} = q_{t-1}) = a_{q_{t-1}q_t} \]
  – 3: observation distributions. For each \( q \in Q \) we have dist.
    \[ P(X_t = x_t|Q_t = q) = p(x_t|q) = b_q(x_t) \]
    \[ B \triangleq \{b_1(x), \ldots, b_{|Q|}(x)\} \]

Discrete distributions:
\[ p(x|q) = \prod_{i=1}^{M} p_{iq}^{1(x=x^{(iq)})} \]

Continuous Distributions:
\[ p(x|q) = \sum_{k=1}^{K_q} c_{kq}p(x|k, q) \]
Hidden Markov Models

• All three parameters (initial distribution, transition matrix, and observation distribution) together we have $\lambda$: $\lambda \triangleq (\Pi, A, B)$

• There are three ways to view HMMs:
  - 1: As a stochastic finite-state automata. Indicates generative process of an HMM.
  - 2: As a graphical model. Indicates conditional independence properties of an HMM.
  - 3: as a lattice: Indicates valid state transitions at each time.

• Each view shows a different perspective.

Hidden Markov Models

• As SFSA:

• Example: 4 urns, with red/black balls. Choose urn randomly, choose ball (with replacement) randomly.

$P(R|U = 1) = 3/7$

$P(B|U = 1) = 4/7$

$P(R|U = 2) = 1/2$

$P(R|U = 3) = 5/6$

$P(R|U = 4) = 1/5$
In HMM, some sequence of urns is chosen, and from each urn, a ball chosen w. replacement.

We know only the balls, but not which urns were used for each ball.

Underlying generative process:

<table>
<thead>
<tr>
<th>Time t:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>Urn Sequence q_t:</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Ball Sequence x_t:</td>
<td>x_1</td>
<td>x_2</td>
<td>x_3</td>
<td>x_4</td>
<td>x_5</td>
<td>x_6</td>
<td>x_7</td>
<td>x_8</td>
<td>x_9</td>
<td>x_{10}</td>
<td>x_{11}</td>
<td>x_{12}</td>
<td>x_{13}</td>
<td>x_{14}</td>
</tr>
</tbody>
</table>

Transition Probs:

<table>
<thead>
<tr>
<th></th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>2/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
<th>1/3</th>
</tr>
</thead>
</table>

Obs. Probs:

|   | 3/7 | 4/7 | 4/7 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 | 5/6 |

As a Graphical Model (Bayesian Network):

Graph shows RVs and conditional independencies.

\[
p(x_1:T, q_1:T) = p(x_1|q_1)p(q_1) \prod_{t=2}^{T} p(x_t|q_t)p(q_t|q_{t-1})
\]

Why called an HMM? Because underlying 1st order Markov chain is hidden, and only observations are known.

In HMM applications, Markov chain isn’t always hidden (sometimes it is known).
Hidden Markov Models

- As a Lattice (just like DTW, shows possible transitions)
- Can show time-inhomogeneous Markov chains
- (really only shows underlying Markov chain, not the observation part)

Main problems to solve associated with HMMs:

- 1) Efficiently compute: $p(x_{1:T}|M, \lambda)$
  - we can build an HMM for each word $M$, and use this in Bayes decision theory as isolated word or utterance model.

- 2) Find most likely state sequence (Viterbi path)
  
  $$q_{1:T}^* = \arg\max_{q_{1:T}} p(x_{1:T}, q_{1:T}|M, \lambda)$$
  
  - the Viterbi path in some sense "best" explains the utterance $x_{1:T}$ (i.e., best set of urns for a given set of balls). Note: we are given $x_{1:T}$, and we need to find best $q_{1:T}$

- 3) Given data set: $D = \{x_{1:T_1}^{(1)}, x_{1:T_2}^{(2)}, \ldots, x_{1:T_N}^{(N)}\}$
  - find most likely parameters:
  
  $$\lambda^* = \arg\max_\lambda p(D|\lambda)$$
  
  - This is maximum likelihood training, solved by EM algorithm.
Hidden Markov Models

- Main problems to solve associated with HMMs:
  - 4) Discriminative Training: Given training data \( D \), find the following quantity:

\[
\lambda^* = \arg\max_\lambda \sum_{i=1}^{N} \frac{p(x_{1:T_i}^{(i)}|M_i, \lambda_i)p(M_i)}{\sum_j p(x_{1:T_i}^{(i)}|M_j, \lambda_j)p(M_j)}
\]

- where, for \( K \) different possible utterances:

\[
\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_K, \ p(M_j) \forall j\}
\]

- This problem is much harder than the previous three (this one will be covered later)

---

Hidden Markov Models

- Problem 1: write \( p(x_{1:T}|M, \lambda) = p(x_{1:T}) \)

- \( x \) could be multi-D vector, \( T \) could be very large (thousands), so we’ve got very high-D space.

\[
p(x_{1:T}) = \sum_{q_{1:T}} p(x_{1:T}, q_{1:T})
\]

\[
= \sum_{q_{1:T}} p(x_{1:T}|q_{1:T})p(q_{1:T})
\]

- \( |Q| \) is also very large (thousands or millions), so the sum above requires \( O(|Q|^T) \) operations, exponential in the length of the utterance \( T \).

- Solution: use the conditional independence properties, and distribute sums inside of products to significantly reduce computation.
Hidden Markov Models

- First approach: \( p(x_{1:T}|M, \lambda) = p(x_{1:T}) \)
  - 1) read off from the graphical model definition:
    \[
    \sum_{q_{1:T}} p(x_{1:T}, q_{1:T}) = \sum_{q_{1:T}} \prod_{t=1}^{T} p(x_t|q_t)p(q_t|q_{t-1})
    \]
    - distribute the sums into the product as far as it can go, leading to the desired solution.
    - This is easy, but when you first encounter HMMs, it might not be the most obvious way to derive the solution

- Second approach:
  - define quantity that if available makes solution easy.
  - Derive derivation of this quantity that can be computed recursively in only \( T \) recursions (dynamic programming)
  - Use HMM conditional independence properties to show how derivation is valid.

\[
\begin{align*}
p(x_{1:t}, q_t) &\triangleq p(x_{1:t}, q_t) \triangleq p(X_{1:t} = x_{1:t}, Q_t = q) \\
p(x_{1:t}; q_t) &= \sum_{q_{t-1}} p(x_{1:t}, q_t, q_{t-1}) \\
p(x_{1:t}, q_t, q_{t-1}) &= p(x_{1:t-1}, q_{t-1}, x_t, q_t) \\
&= p(x_t, q_t|x_{1:t-1, q_{t-1}})p(x_{1:t-1}, q_{t-1}) \\
&= p(x_t|q_t, x_{1:t-1}, q_{t-1})p(q_t|x_{1:t-1}, q_{t-1})\alpha_{q_{t-1}}(t-1) \\
&= p(x_t|q_t)p(q_t|q_{t-1})\alpha_{q_{t-1}}(t-1) \\
&\text{valid since HMM properties say:} \\
X_{t|\perp\{Q_{t-1}, X_{1:t-1}\}|Q_t} &\quad Q_{t|\perp\{X_{1:t-1}\}|Q_{t-1}}
\end{align*}
\]
Hidden Markov Models

- So we have recursion for alpha:
  \[ \alpha_{q_t}(t) = \sum_{q_{t-1}} p(x_t|q_t)p(q_t|q_{t-1})\alpha_{q_{t-1}}(t-1) \]
  - or in other words:  \[ \alpha_q(t) = p(x_t|q) \sum_r p(q|r)\alpha_r(t-1) \]
  - To get final quantity:
    \[ p(x_{1:T}) = \sum_q \alpha_q(T) \]
    - Cost: at each step we do \(|Q|^2\) operations, \(T\) steps, so we get \(O(T|Q|^2)\) ops \(<\) \(O(|Q|^T)\) ops.

- We can view this using a picture very similar to the DTW pictures we saw before.

\[ \alpha_3(2) = p(x_2|Q_2 = 3) \sum_{q=1}^{\lfloor Q \rfloor} p(Q_2 = 3|Q_1 = q)\alpha_q(1) \]

*local distortion*  
*dynamic programming recursion*
Hidden Markov Models

- Zeros in $p(q_t|q_{t-1})$ determine local path constraints (prob. 0).
- What do CI statements give us? They make it mathematically “valid” to do the efficient dynamic programming solution to computing $p(x)$.
- Efficiency is one of the key reasons HMMs have been so successful (similar to the FFT).
- Initial conditions: $\alpha_q(1) = p(x_1|q)p(q)$
- Note: we used independence relation: $X_{t:t-1}\perp\!\perp X_1:1:T|Q_t$
- But for HMMs, we also need: $X_{t:t+1:T}\perp\!\perp X_t|Q_t$
- Why? Used for “Backwards” recursion:

\[
\begin{align*}
\text{Hidden Markov Models} \\
\text{Backwards recursion: } & \beta_q(t) \triangleq p(x_{t+1:T}|Q_t = q) \\
p(x_{1:T}) &= \sum_{q_1} p(x_1, q_1, x_{2:T}) \\
&= \sum_{q_1} p(x_{2:T}|q_1, x_1)p(x_1|q_1)p(q_1) \\
&= \sum_{q_1} \beta_{q_1}(t)p(x_1|q_1)p(q_1)
\end{align*}
\]

- This follows since: $X_{2:T}\perp\!\perp X_1|Q_1$
Hidden Markov Models

• Backwards recursion:
  – we can get a backwards recursion for $\beta$ using the
    conditional independence properties of the HMM.

$$p(x_{t+1:T}^T|q_t) = \sum_{q_{t+1}} p(x_{t+2:T}^T, x_{t+1}, q_{t+1}|q_t)$$
$$= \sum_{q_{t+1}} p(x_{t+2:T}|q_{t+1}, x_{t+1}, q_t)p(x_{t+1}|q_{t+1}, q_t)p(q_{t+1}|q_t)$$
$$= \sum_{q_{t+1}} p(x_{t+2:T}|q_{t+1})p(x_{t+1}|q_{t+1})p(q_{t+1}|q_t)$$
$$= \sum_{q_{t+1}} \beta_{q_{t+1}}(t + 1)p(x_{t+1}|q_{t+1})p(q_{t+1}|q_t)$$
$$= \beta_{q_t}(t)$$

Hidden Markov Models

• $\beta$ “backwards recursion”:

$$\beta_3(T - 2) = \sum_{q=1}^{[Q]} \beta_q(T - 1)p(x_{T-1}|Q_{T-1} = q)p(Q_{T-1} = q|Q_{T-2} = 3)$$
Hidden Markov Models

- β “backwards recursion”
- β_q(t) includes local probability values only from later frames (not current frame t)
- Initial conditions: \( \beta_q(T) = 1 \quad \forall q \)
- Using forward & backwards recursion together \( \forall t \) and HMM conditional independence properties:

\[
p(x_{1:T}) = \sum_{q_t} p(q_t, x_{1:t}, x_{t+1:T}) \\
= p(x_{t+1:T}|q_t, x_{1:t})p(x_{1:t}, q_t) \\
= p(x_{t+1:T}|q_t)p(x_{1:t}, q_t) \\
= \sum_{q_t} \beta_{qt}(t)\alpha_{qt}(t)
\]

**Hidden Markov Models**

- \( \alpha/\beta \) recursions summary:

\[
p(x_{1:T}) = \sum_{q_t} \beta_{qt}(t)\alpha_{qt}(t) \\
p(x_{1:T}, q_t) = \beta_{qt}(t)\alpha_{qt}(t) \\
\alpha_q(t) = \sum_r \alpha_r(t - 1)p(q|r)p(x_t|q) \\
\beta_q(t) = \sum_r \beta_r(t + 1)p(x_{t+1}|r)p(r|q)
\]

- Therefore, posteriors are easily computable:

\[
p(Q_t = q|x_{1:T}) = \frac{p(Q_t = q, x_{1:T})}{p(x_{1:T})} \\
= \frac{\alpha_q(t)\beta_q(t)}{\sum_q \alpha_q(t)\beta_q(t)} \\
\Delta = \gamma_q(t)
\]
Hidden Markov Models

- Finding Viterbi (most likely) path:
- Is this the same as finding: \( \{q_1, q_2, \ldots, q_T\} \)
  
  where
  \[
  q_t = \arg\max_q p(Q_t = q|x_1:T), \quad t = 1, \ldots, T
  \]
- When is it the same and when is it different?
- Our goal:
  \[
  \{q_{1:T}^*\} = \arg\max_{q_{1:T}} p(x_{1:T}, q_{1:T})
  \]

\[
\delta_{q_t}(t) = p(x_t|q_t) \left[ \max_{q_{t-1}} \delta_{q_{t-1}}(t-1)p(q_t|q_{t-1}) \right]
\]

\[
\psi_{q_t}(t) = \arg\max_{q_{t-1}} \left[ p(x_t|q_t)\delta_{q_{t-1}}(t-1)p(q_t|q_{t-1}) \right]
\]
Hidden Markov Models

• About HMMs
  – is \(X_{1:T}\) i.i.d.?
  – is \(X_{1:T}\) i.i.d. given \(Q_{1:T}\)?
  – is \(P(X_{1:T}|q^*_{1:T})\) i.i.d.? (\(q^*\) is Viterbi path)
  – are observations \(X_{1:T}\) correlated?
  – Is \(X_{\{1:T\}}\) under a stationary distribution necessarily?
  – Are durations necessarily geometric?
  – What is “capacity” of dependence between neighboring observations?

• In general, HMMs are quite powerful.

Hidden Markov Models

• Still two problems to solve, namely training:
  – Problem 3: Maximum Likelihood training
  – Problem 4: Discriminative Training