Parametric Models
Part III: Hidden Markov Models

Selim Aksoy
Bilkent University
Department of Computer Engineering
saksoy@cs.bilkent.edu.tr

CS 551, Spring 2006

Discrete Markov Processes (Markov Chains)

- The goal is to make a sequence of decisions where a particular decision may be influenced by earlier decisions.
- Consider a system that can be described at any time as being in one of a set of N distinct states $w_1, w_2, \ldots, w_N$.
- Let $w(t)$ denote the actual state at time t where $t = 1, 2, \ldots$.
- The probability of the system being in state $w(t)$ is $P(w(t)|w(t-1), \ldots, w(1))$.

First-Order Markov Models

- We assume that the state $w(t)$ is conditionally independent of the previous states given the predecessor state $w(t-1)$, i.e.,

$$P(w(t)|w(t-1), \ldots, w(1)) = P(w(t)|w(t-1)).$$

- We also assume that the Markov Chain defined by $P(w(t)|w(t-1))$ is time homogeneous (independent of the time t).

First-Order Markov Models

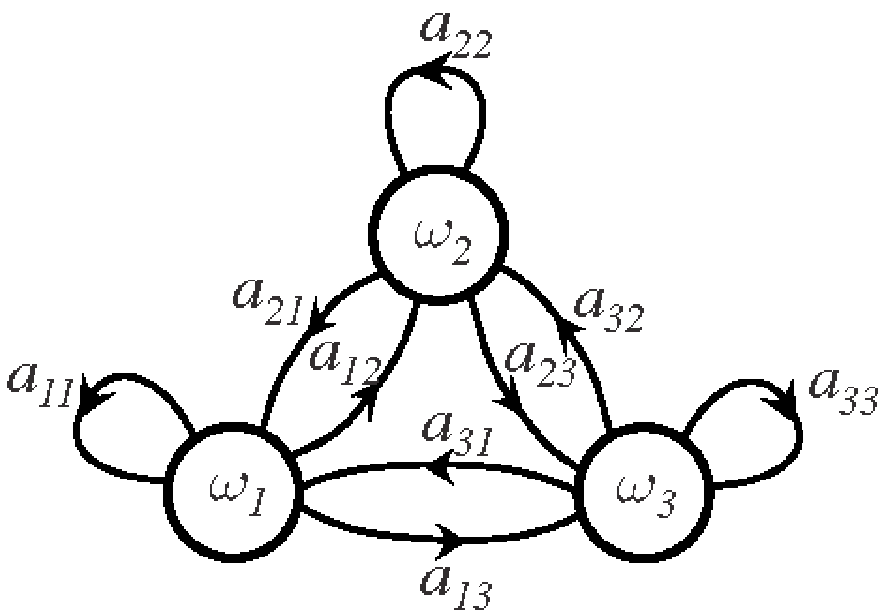
- A particular *sequence of states* of length T is denoted by
$$\mathcal{W}^T = \{w(1), w(2), \ldots, w(T)\}.$$
- The model for the production of any sequence is described by the *transition probabilities*
$$a_{ij} = P(w(t) = w_j | w(t-1) = w_i)$$
where $i, j \in \{1, \ldots, N\}$, $a_{ij} \geq 0$, and $\sum_{j=1}^{N} a_{ij} = 1, \forall i$.

First-Order Markov Models

- There is no requirement that the transition probabilities are symmetric ($a_{ij} \neq a_{ji}$, in general).
- Also, a particular state may be visited in succession ($a_{ii} \neq 0$, in general) and not every state need to be visited.
- This process is called an *observable Markov model* because the output of the process is the set of states at each instant of time, where each state corresponds to a physical (observable) event.

First-Order Markov Model Examples

- Consider the following 3-state first-order Markov model of the weather in Ankara:
  - w_1: rain/snow
  - w_2: cloudy
  - w_3: sunny



$$\mathbf{\Theta} = \{a_{ij}\}$$
$$= \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{pmatrix}$$

First-Order Markov Model Examples

- We can use this model to answer the following question: Starting with sunny weather on day 1 (given), what is the probability that the weather for the next seven days will be "sunny-sunny-rainy-rainy-sunny-cloudy-sunny" ($\mathcal{W}^8 = \{w_3, w_3, w_3, w_1, w_1, w_3, w_2, w_3\}$)?
- Solution:

$$
\begin{aligned}
P(\mathcal{W}^8 | \boldsymbol{\Theta}) &= P(w_3, w_3, w_3, w_1, w_1, w_3, w_2, w_3) \\
&= P(w_3) P(w_3|w_3) P(w_3|w_3) P(w_1|w_3) \\
&\qquad P(w_1|w_1) P(w_3|w_1) P(w_2|w_3) P(w_3|w_2) \\
&= P(w_3)\, a_{33}\, a_{33}\, a_{31}\, a_{11}\, a_{13}\, a_{32}\, a_{23} \\
&= 1 \times 0.8 \times 0.8 \times 0.1 \times 0.4 \times 0.3 \times 0.1 \times 0.2 \\
&= 1.536 \times 10^{-4}
\end{aligned}
$$

First-Order Markov Model Examples

- Consider another question: Given that the model is in a known state, what is the probability that it stays in that state for exactly d days?
- Solution:

$$\mathcal{W}^{d+1} = \{w(1) = w_i, w(2) = w_i, \ldots, w(d) = w_i, w(d+1) = w_j \neq w_i\}$$

$$P(\mathcal{W}^{d+1}|\mathbf{\Theta}, w(1) = w_i) = (a_{ii})^{d-1}(1 - a_{ii})$$

$$E[d|w_i] = \sum_{d=1}^{\infty} d\,(a_{ii})^{d-1}\,(1 - a_{ii}) = \frac{1}{1 - a_{ii}}$$

- For example, the expected number of consecutive days of sunny weather is 5, cloudy weather is 2.5, rainy weather is 1.67.

First-Order Hidden Markov Models

- We can extend this model to the case where the observation (output) of the system is a probabilistic function of the state.
- The resulting model, called a *Hidden Markov Model (HMM)*, has an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produce a sequence of observations.

First-Order Hidden Markov Models

- We denote the observation at time t as $v(t)$ and the probability of producing that observation in state $w(t)$ as $P(v(t)|w(t))$.
- There are many possible state-conditioned observation distributions.
- When the observations are discrete, the distributions

$$b_{jk} = P(v(t) = v_k | w(t) = w_j)$$

are probability mass functions where $j \in \{1, \ldots, N\}$, $k \in \{1, \ldots, M\}$, $b_{jk} \geq 0$, and $\sum_{k=1}^{M} b_{jk} = 1, \forall j$.

First-Order Hidden Markov Models

- When the observations are continuous, the distributions are typically specified using a parametric model family where the most common family is the Gaussian mixture

$$b_j(\mathbf{x}) = \sum_{k=1}^{M_j} \alpha_{jk} \, p(\mathbf{x} | \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})$$

  where $\alpha_{jk} \geq 0$ and $\sum_{k=1}^{M_j} \alpha_{jk} = 1, \forall j$.

- We will restrict ourselves to discrete observations where a particular sequence of visible states of length T is denoted by

$$\mathcal{V}^T = \{v(1), v(2), \ldots, v(T)\}.$$

First-Order Hidden Markov Models

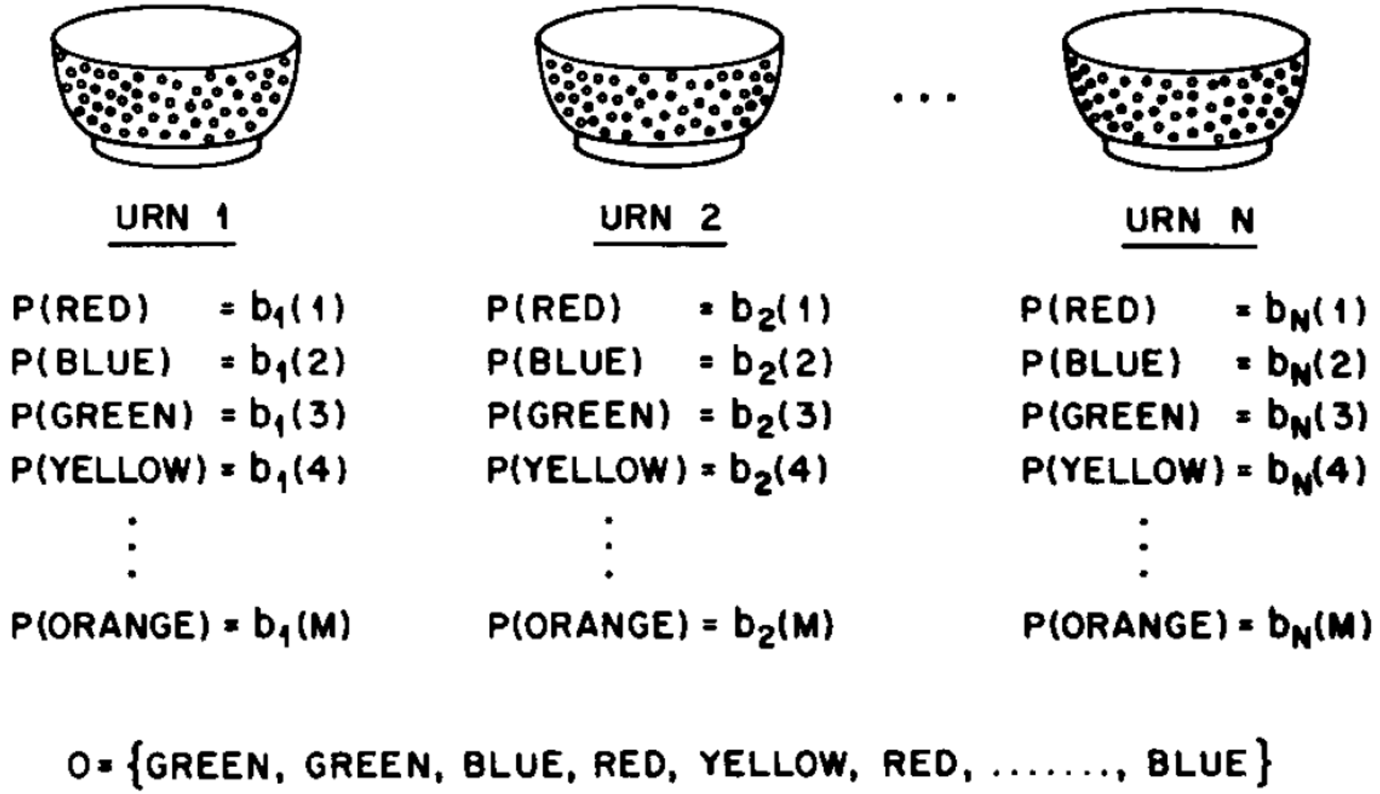
- An HMM is characterized by:
  - N, the number of hidden states
  - M, the number of distinct observation symbols per state
  - $\{a_{ij}\}$, the state transition probability distribution
  - $\{b_{jk}\}$, the observation symbol probability distribution
  - $\{\pi_i = P(w(1) = w_i)\}$, the initial state distribution
  - $\mathbf{\Theta} = (\{a_{ij}\}, \{b_{jk}\}, \{\pi_i\})$, the complete parameter set of the model

First-Order HMM Examples

- Consider the "urn and ball" example (Rabiner, 1989):
  - There are N large urns in the room.
  - Within each urn, there are a large number of colored balls where the number of distinct colors is M.
  - An initial urn is chosen according to some random process, and a ball is chosen at random from it.
  - The ball's color is recorded as the observation and it is put back to the urn.
  - A new urn is selected according to the random selection process associated with the current urn and the ball selection process is repeated.

First-Order HMM Examples

- The simplest HMM that corresponds to the urn and ball selection process is the one where
  - each state corresponds to a specific urn,
  - a ball color probability is defined for each state.



First-Order HMM Examples

- Let's extend the weather example.
  - Assume that you have a friend who lives in İstanbul and you talk daily about what each of you did that day.
  - Your friend has a list of activities that she/he does every day (such as playing sports, shopping, studying) and the choice of what to do is determined exclusively by the weather on a given day.
  - Assume that İstanbul has a weather state distribution similar to the one in the previous example.
  - You have no information about the weather where your friend lives, but you try to guess what it must have been like according to the activity your friend did.

First-Order HMM Examples

- This process can be modeled using an HMM where the state of the weather is the hidden variable, and the activity your friend did is the observation.
- Given the model and the activity of your friend, you can make a guess about the weather in İstanbul that day.
- For example, if your friend says that she/he played sports on the first day, went shopping on the second day, and studied on the third day of the week, you can answer questions such as:
  - What is the overall probability of this sequence of observations?
  - What is the most likely weather sequence that would explain these observations?

Applications of HMMs

- Speech recognition
- Optical character recognition
- Natural language processing (e.g., text summarization)
- Bioinformatics (e.g., protein sequence modeling)
- Video analysis (e.g., story segmentation, motion tracking)
- Robot planning (e.g., navigation)
- Economics and finance (e.g., time series, customer decisions)

Three Fundamental Problems for HMMs

- *Evaluation problem:* Given the model, compute the probability that a particular output sequence was produced by that model (solved by the forward algorithm).
- *Decoding problem:* Given the model, find the most likely sequence of hidden states which could have generated a given output sequence (solved by the Viterbi algorithm).
- *Learning problem:* Given a set of output sequences, find the most likely set of state transition and output probabilities (solved by the Baum-Welch algorithm).

HMM Evaluation Problem

- A particular *sequence of observations* of length T is denoted by

$$\mathcal{V}^T = \{v(1), v(2), \ldots, v(T)\}.$$

- The probability of observing this sequence can be computed by enumerating every possible state sequence of length T as

$$\begin{aligned} P(\mathcal{V}^T|\mathbf{\Theta}) &= \sum_{\text{all } \mathcal{W}^T} P(\mathcal{V}^T, \mathcal{W}^T|\mathbf{\Theta}) \\ &= \sum_{\text{all } \mathcal{W}^T} P(\mathcal{V}^T|\mathcal{W}^T, \mathbf{\Theta}) P(\mathcal{W}^T|\mathbf{\Theta}). \end{aligned}$$

HMM Evaluation Problem

- This summation includes N^T terms in the form

$$P(\mathcal{V}^T|\mathcal{W}^T)P(\mathcal{W}^T) = \left(\prod_{t=1}^{T} P(v(t)|w(t))\right)\left(\prod_{t=1}^{T} P(w(t)|w(t-1))\right)$$
$$= \prod_{t=1}^{T} P(v(t)|w(t))P(w(t)|w(t-1))$$

  where $P(w(t)|w(t-1))$ for $t = 1$ is $P(w(1))$.
- It is unfeasible with computational complexity $O(N^T T)$.
- However, a computationally simpler algorithm called the *forward algorithm* computes $P(\mathcal{V}^T|\boldsymbol{\Theta})$ recursively.

HMM Evaluation Problem

- Define $\alpha_j(t)$ as the probability that the HMM is in state w_j at time t having generated the first t observations in $\mathcal{V}^T$

$$\alpha_j(t) = P(v(1), v(2), \ldots, v(t), w(t) = w_j | \mathbf{\Theta}).$$

- $\alpha_j(t), j = 1, \ldots, N$ can be computed as

$$\alpha_j(t) = \begin{cases} \pi_j b_{jv(1)} & t = 1 \\ \left(\sum_{i=1}^{N} \alpha_i(t-1) a_{ij} \right) b_{jv(t)} & t = 2, \ldots, T. \end{cases}$$

- Then, $P(\mathcal{V}^T | \mathbf{\Theta}) = \sum_{j=1}^{N} \alpha_j(T)$.

HMM Evaluation Problem

- Similarly, we can define a *backward algorithm* where

$$\beta_i(t) = P(v(t+1), v(t+2), \ldots, v(T) | w(t) = w_i, \mathbf{\Theta})$$

  is the probability that the HMM will generate the observations from $t+1$ to T in $\mathcal{V}^T$ given that it is in state w_i at time t.

- $\beta_i(t), i = 1, \ldots, N$ can be computed as

$$\beta_i(t) = \begin{cases} 1 & t = T \\ \sum_{j=1}^{N} \beta_j(t+1) a_{ij} b_{jv(t+1)} & t = T-1, \ldots, 1. \end{cases}$$

- The computations of both $\alpha_j(t)$ and $\beta_i(t)$ have complexity $O(N^2 T)$.

HMM Evaluation Problem

- For classification, we can compute the posterior probabilities

$$P(\boldsymbol{\Theta}|\mathcal{V}^T) = \frac{P(\mathcal{V}^T|\boldsymbol{\Theta})P(\boldsymbol{\Theta})}{P(\mathcal{V}^T)}$$

  where $P(\boldsymbol{\Theta})$ is the prior for a particular class, and $P(\mathcal{V}^T|\boldsymbol{\Theta})$ is computed using the forward algorithm with the HMM for that class.

- Then, we can select the class with the highest posterior.

HMM Decoding Problem

- Given a sequence of observations $\mathcal{V}^T$, we would like to find the most probable sequence of hidden states.
- One possible solution is to enumerate every possible hidden state sequence and calculate the probability of the observed sequence with $O(N^T T)$ complexity.
- We can also define the problem of finding the optimal state sequence as finding the one that includes the states that are individually most likely.
- This also corresponds to maximizing the expected number of correct individual states.

HMM Decoding Problem

- Define $\gamma_i(t)$ as the probability that the HMM is in state w_i at time t given the observation sequence $\mathcal{V}^T$

$$\gamma_i(t) = P(w(t) = w_i | \mathcal{V}^T, \mathbf{\Theta})$$
$$= \frac{\alpha_i(t)\beta_i(t)}{P(\mathcal{V}^T|\mathbf{\Theta})} = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^N \alpha_j(t)\beta_j(t)}$$

where $\sum_{i=1}^N \gamma_i(t) = 1$.

- Then, the individually most likely state $w(t)$ at time t becomes

$$w(t) = w_{i'} \text{ where } i' = \arg \max_{i=1,\ldots,N} \gamma_i(t).$$

HMM Decoding Problem

- One problem is that the resulting sequence may not be consistent with the underlying model because it may include transitions with zero probability ($a_{ij} = 0$ for some i and j).
- One possible solution is the *Viterbi algorithm* that finds the single best state sequence $\mathcal{W}^T$ by maximizing $P(\mathcal{W}^T | \mathcal{V}^T, \mathbf{\Theta})$ (or equivalently $P(\mathcal{W}^T, \mathcal{V}^T | \mathbf{\Theta})$).
- This algorithm recursively computes the state sequence with the highest probability at time t and keeps track of the states that form the sequence with the highest probability at time T (see Rabiner (1989) for details).

HMM Learning Problem

- The goal is to determine the model parameters $\{a_{ij}\}$, $\{b_{jk}\}$ and $\{\pi_i\}$ from a collection of training samples.
- Define $\xi_{ij}(t)$ as the probability that the HMM is in state w_i at time $t-1$ and state w_j at time t given the observation sequence $\mathcal{V}^T$

$$\begin{aligned}
\xi_{ij}(t) &= P(w(t-1) = w_i, w(t) = w_j | \mathcal{V}^T, \boldsymbol{\Theta}) \\
&= \frac{\alpha_i(t-1)\, a_{ij}\, b_{jv(t)}\, \beta_j(t)}{P(\mathcal{V}^T | \boldsymbol{\Theta})} \\
&= \frac{\alpha_i(t-1)\, a_{ij}\, b_{jv(t)}\, \beta_j(t)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_i(t-1)\, a_{ij}\, b_{jv(t)}\, \beta_j(t)}.
\end{aligned}$$

HMM Learning Problem

- $\gamma_i(t)$ defined in the decoding problem and $\xi_{ij}(t)$ defined here can be related as

$$\gamma_i(t-1) = \sum_{j=1}^{N} \xi_{ij}(t).$$

- Then, $\hat{a}_{ij}$, the estimate of the probability of a transition from w_i at $t-1$ to w_j at t, can be computed as

$$\hat{a}_{ij} = \frac{\text{expected number of transitions from } w_i \text{ to } w_j}{\text{expected total number of transitions away from } w_i} = \frac{\sum_{t=2}^{T} \xi_{ij}(t)}{\sum_{t=2}^{T} \gamma_i(t-1)}.$$

HMM Learning Problem

- Similarly, $\hat{b}_{jk}$, the estimate of the probability of observing the symbol v_k while in state w_j, can be computed as

$$\hat{b}_{jk} = \frac{\text{expected number of times observing symbol } v_k \text{ in state } w_j}{\text{expected total number of times in } w_j}$$

$$= \frac{\sum_{t=1}^{T} \delta_{v(t),v_k} \gamma_j(t)}{\sum_{t=1}^{T} \gamma_j(t)}$$

where $\delta_{v(t),v_k}$ is the Kronecker delta which is 1 only when $v(t) = v_k$.

- Finally, $\hat{\pi}_i$, the estimate for the initial state distribution, can be computed as $\hat{\pi}_i = \gamma_i(1)$ which is the expected number of times in state w_i at time $t = 1$.

HMM Learning Problem

- These are called the *Baum-Welch* equations (also called the *EM estimates for HMMs* or the *forward-backward algorithm*) that can be computed iteratively until some convergence criterion is met (e.g., sufficiently small changes in the estimated values in subsequent iterations).
- See (Bilmes, 1998) for the estimates $\hat{b}_j(\mathbf{x})$ when the observations are continuous and their distributions are modeled using Gaussian mixtures.