Hidden Markov Models and Information Retrieval

Tapas Kanungo

Center for Automation Research

University of Maryland

Web: www.cfar.umd.edu/~kanungo

Email: kanungo@cfar.umd.edu

Outline

1. Markov models
2. Hidden Markov models
3. Forward/Backward algorithm
4. Viterbi algorithm
5. Baum-Welch estimation algorithm
6. HMM-based information retrieval

Markov Models

- **Observable states:**

$$1, 2, \ldots, N$$

- **Observed sequence:**

$$q_1, q_2, \ldots, q_t, \ldots, q_T$$

- **First order Markov assumption:**

$$P(q_t = j | q_{t-1} = i, q_{t-2} = k, \ldots) = P(q_t = j | q_{t-1} = i)$$

- **Stationarity:**

$$P(q_t = j | q_{t-1} = i) = P(q_{t+l} = j | q_{t+l-1} = i)$$

Markov Models

- **State transition matrix** A :

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2N} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{iN} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{Nj} & \cdots & a_{NN} \end{bmatrix}$$

where

$$a_{ij} = P(q_t = j | q_{t-1} = i) \qquad 1 \leq i, j, \leq N$$

- **Constraints on** a_{ij} :

$$\begin{aligned} a_{ij} &\geq 0, & \forall i, j \\ \sum_{j=1}^{N} a_{ij} &= 1, & \forall i \end{aligned}$$

Markov Models: Example

- **States:**
  1. **Rainy (R)**
  2. **Cloudy (C)**
  3. **Sunny (S)**
- **State transition probability matrix:**

$$A = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

- **Compute the probability of observing $SSRRSCS$ given that today is S.**

Markov Models: Example

Basic conditional probability rule:

$$P(A, B) = P(A|B)P(B)$$

The Markov chain rule:

$$\begin{aligned}
&P(q_1, q_2, \ldots, q_T) \\
&\quad = P(q_T|q_1, q_2, \ldots, q_{T-1})P(q_1, q_2, \ldots, q_{T-1}) \\
&\quad = P(q_T|q_{T-1})P(q_1, q_2, \ldots, q_{T-1}) \\
&\quad = P(q_T|q_{T-1})P(q_{T-1}|q_{T-2})P(q_1, q_2, \ldots, q_{T-2}) \\
&\quad = P(q_T|q_{T-1})P(q_{T-1}|q_{T-2}) \cdots P(q_2|q_1)P(q_1)
\end{aligned}$$

Markov Models: Example

- **Observation sequence** O :

$$O = (S, S, S, R, R, S, C, S)$$

- **Using the chain rule we get:**

$$\begin{aligned}
&P(O|model) \\
&= P(S, S, S, R, R, S, C, S|model) \\
&= P(S)P(S|S)P(S|S)P(R|S)P(R|R) \times \\
&\quad P(S|R)P(C|S)P(S|C) \\
&= \pi_3 a_{33} a_{33} a_{31} a_{11} a_{13} a_{32} a_{23} \\
&= (1)(0.8)^2(0.1)(0.4)(0.3)(0.1)(0.2) \\
&= 1.536 \times 10^{-4}
\end{aligned}$$

- **The prior probability** $\pi_i = P(q_1 = i)$

Markov Models: Example

- **What is the probability that the sequence remains in state i for exactly d time units?**

$$\begin{aligned} p_i(d) &= P(q_1 = i, q_2 = i, \ldots, q_d = i, q_{d+1} \neq i, \ldots) \\ &= \pi_i (a_{ii})^{d-1}(1 - a_{ii}) \end{aligned}$$

- **Exponential Markov chain duration density.**
- **What is the expected value of the duration d in state i?**

$$\begin{aligned} \bar{d}_i &= \sum_{d=1}^{\infty} d p_i(d) \\ &= \sum_{d=1}^{\infty} d (a_{ii})^{d-1}(1 - a_{ii}) \\ &= (1 - a_{ii}) \sum_{d=1}^{\infty} d (a_{ii})^{d-1} \\ &= (1 - a_{ii}) \frac{\partial}{\partial a_{ii}} \sum_{d=1}^{\infty} (a_{ii})^d \\ &= (1 - a_{ii}) \frac{\partial}{\partial a_{ii}} \left(\frac{a_{ii}}{1 - a_{ii}} \right) \\ &= \frac{1}{1 - a_{ii}} \end{aligned}$$

Markov Models: Example

- **Avg. number of consecutive sunny days =**

$$\frac{1}{1-a_{33}} = \frac{1}{1-0.8} = 5$$

- **Avg. number of consecutive cloudy days = 2.5**

- **Avg. number of consecutive rainy days = 1.67**

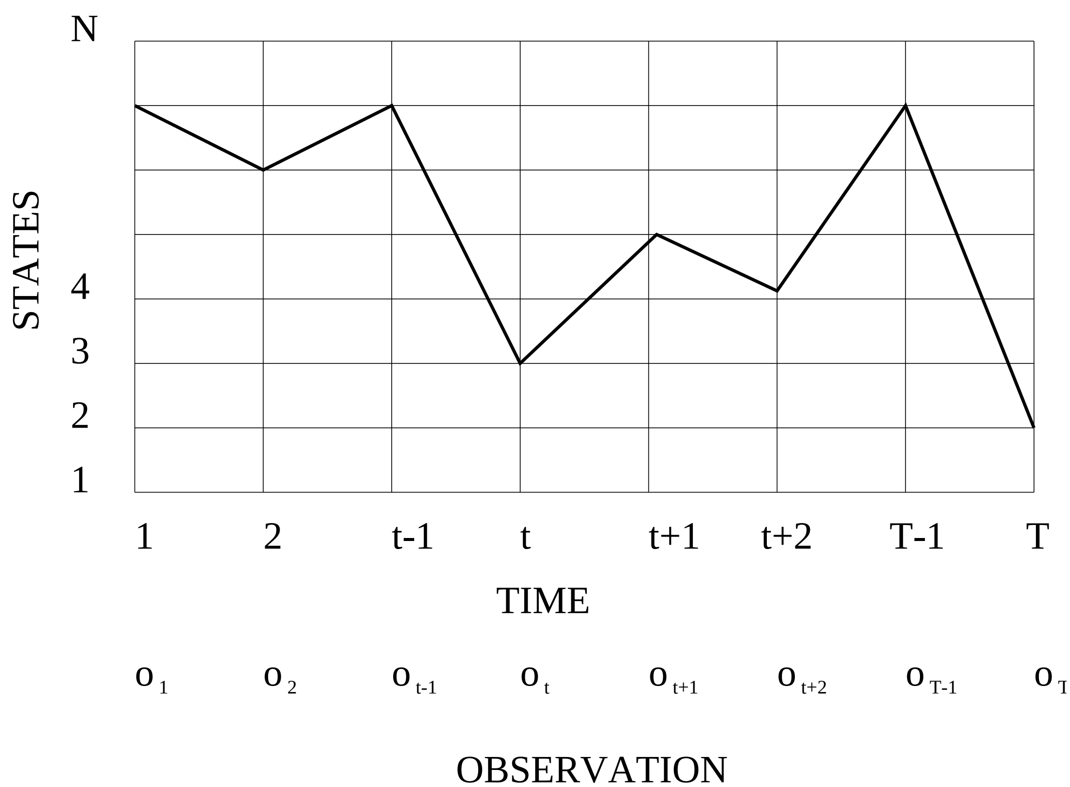
Hidden Markov Models

- States are not observable
- Observations are probabilistic functions of state
- State transitions are still probabilistic

Cities and Weather Model

- N cities
- M distinct (observable) weather conditions
- Each city has a (possibly) different distribution of weather conditions
- Sequence generation algorithm:
  1. Pick initial city according to some random process.
  2. Randomly pick a weather condition
  3. Select another city according a random selection process associated with the current city
  4. Repeat steps 2 and 3

The Trellis


N
STATES
4
3
2
1
1
2
t-1
t
t+1
t+2
T-1
T
TIME
o 1
o 2
o t-1
o t
o t+1
o t+2
o T-1
o T
OBSERVATION

Elements of Hidden Markov Models

- N – **the number of hidden states**
- Q – **set of states** $Q = \{1, 2, \ldots, N\}$
- M – **the number of symbols**
- V – **set of symbols** $V = \{1, 2, \ldots, M\}$
- A – **the state-transition probability matrix.**

$$a_{ij} = P(q_{t+1} = j | q_t = i) \quad 1 \leq i, j, \leq N$$

- B – **Observation probability distribution:**

$$B_j(k) = P(o_t = k | q_t = j) \quad 1 \leq k \leq M$$

- π – **the initial state distribution:**

$$\pi_i = P(q_1 = i) \quad 1 \leq i \leq N$$

- λ – **the entire model** $\lambda = (A, B, \pi)$

Three Basic Problems

1. **Given observation** $O = (o_1, o_2, \ldots, o_T)$ **and model** $\lambda = (A, B, \pi)$, **efficiently compute** $P(O|\lambda)$.
   - **Hidden states complicate the evaluation**
   - **Given two models** λ_1 **and** λ_2, **this can be used to choose the better one.**
2. **Given observation** $O = (o_1, o_2, \ldots, o_T)$ **and model** λ **find the optimal state sequence** $q = (q_1, q_2, \ldots, q_T)$.
   - **Optimality criterion has to be decided (e.g. maximum likelihood)**
   - **"Explanation" for the data.**
3. **Given** $O = (o_1, o_2, \ldots, o_T)$, **estimate model parameters** $\lambda = (A, B, \pi)$ **that maximize** $P(O|\lambda)$.

Solution to Problem 1

- **Problem: Compute** $P(o_1, o_2, \ldots, o_T|\lambda)$
- **Algorithm:**
  - **Let** $q = (q_1, q_2, \ldots, q_T)$ **be a state sequence.**
  - **Assume the observations are independent:**

$$\begin{aligned} P(O|q,\lambda) &= \prod_{i=1}^{T} P(o_t|q_t,\lambda) \\ &= b_{q_1}(o_1) b_{q_2}(o_2) \cdots b_{q_T}(o_T) \end{aligned}$$

  - **Probability of a particular state sequence is:**

$$P(q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T}$$

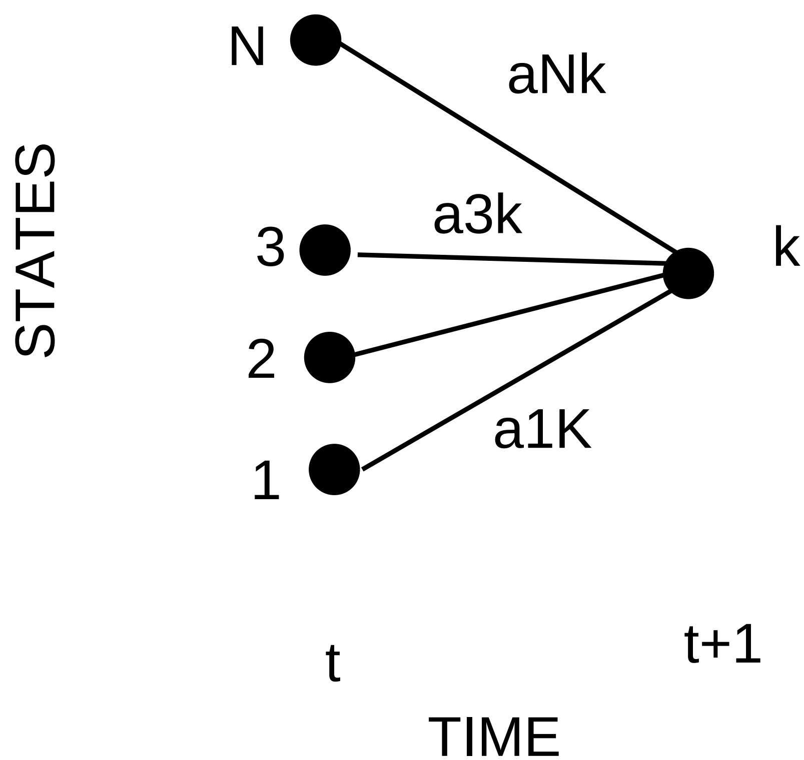
  - **Also,** $P(O, q|\lambda) = P(O|q,\lambda) P(q|\lambda)$
  - **Enumerate paths and sum probabilities:**

$$P(O|\lambda) = \sum_q P(O|q,\lambda) P(q|\lambda)$$

- N^T **state sequences and** $O(T)$ **calculations.**

  Complexity: $O(TN^T)$ **calculations.**

Forward Procedure: Intuition



Forward Algorithm

- **Define forward variable $\alpha_t(i)$ as:**

$$\alpha_t(i) = P(o_1, o_2, \ldots, o_t, q_t = i|\lambda)$$

- **$\alpha_t(i)$ is the probability of observing the partial sequence $(o_1, o_2 \ldots, o_t)$ such that the state q_t is i.**

- **Induction:**

  1. **Initialization:** $\alpha_1(i) = \pi_i b_i(o_1)$
  2. **Induction:**

  $$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1})$$

  3. **Termination:**

  $$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

- **Complexity:** $O(N^2T)$.

Example

Consider the following coin-tossing experiment:

	State 1	State 2	State 3
P(H)	0.5	0.75	0.25
P(T)	0.5	0.25	0.75

– **state-transition probabilities equal to 1/3**

– **initial state probabilities equal to 1/3**

1. **You observe** $O = (H, H, H, H, T, H, T, T, T, T)$. **What state sequence,** q, **is most likely? What is the joint probability,** $P(O, q|\lambda)$, **of the observation sequence and the state sequence?**

2. **What is the probability that the observation sequence came entirely of state 1?**

3. Consider the observation sequence

$$\tilde{O} = (H, T, T, H, T, H, H, T, T, H).$$

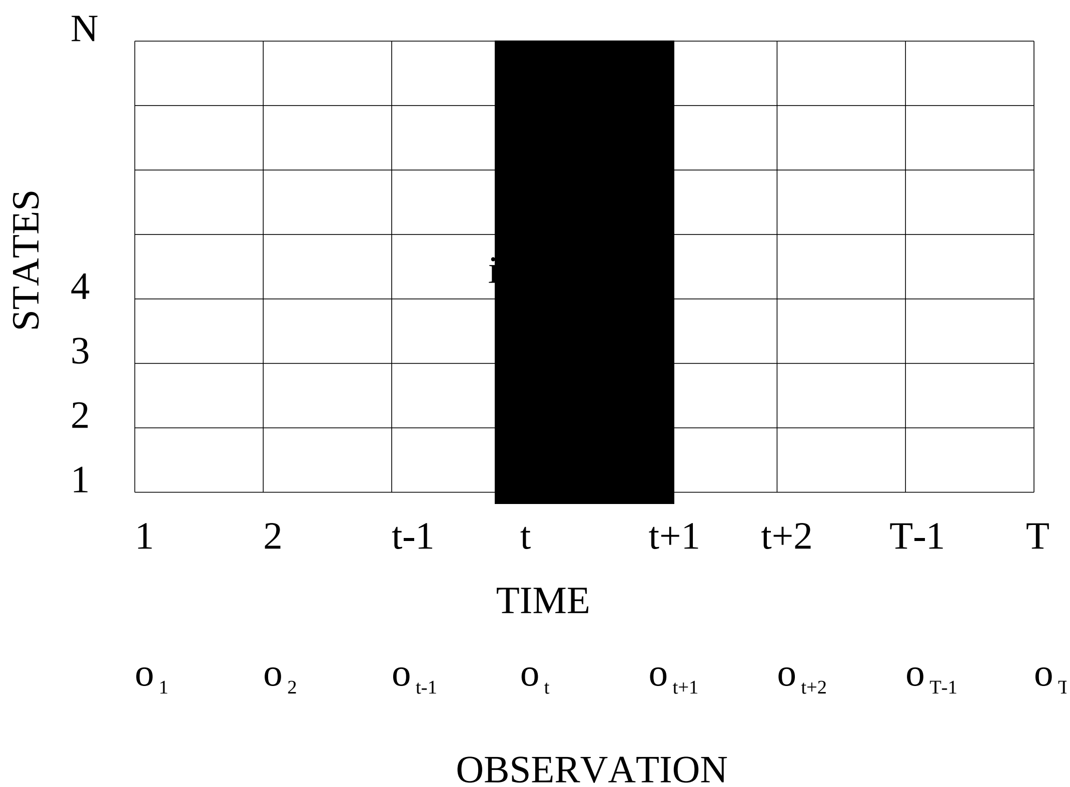
How would your answers to parts 1 and 2 change?

4. If the state transition probabilities were:

$$A' = \begin{bmatrix} 0.9 & 0.45 & 0.45 \\ 0.05 & 0.1 & 0.45 \\ 0.05 & 0.45 & 0.1 \end{bmatrix},$$

how would the new model λ' change your answers to parts 1-3?

Backward Algorithm



Backward Algorithm

- **Define backward variable $\beta_t(i)$ as:**

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T | q_t = i, \lambda)$$

- **$\beta_t(i)$ is the probability of observing the partial sequence $(o_{t+1}, o_{t+2} \ldots, o_T)$ such that the state q_t is i.**

- **Induction:**

  1. **Initialization:** $\beta_T(i) = 1$
  2. **Induction:**

$$\begin{aligned} \beta_t(i) &= \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \\ & \qquad 1 \leq i \leq N, \\ & \qquad t = T-1, \ldots, 1 \end{aligned}$$

Solution to Problem 2

- **Choose the most likely path**
- **Find the path $(q_1, q_t, \ldots, q_T)$ that maximizes the likelihood:**

$$P(q_1, q_2, \ldots, q_T | O, \lambda)$$

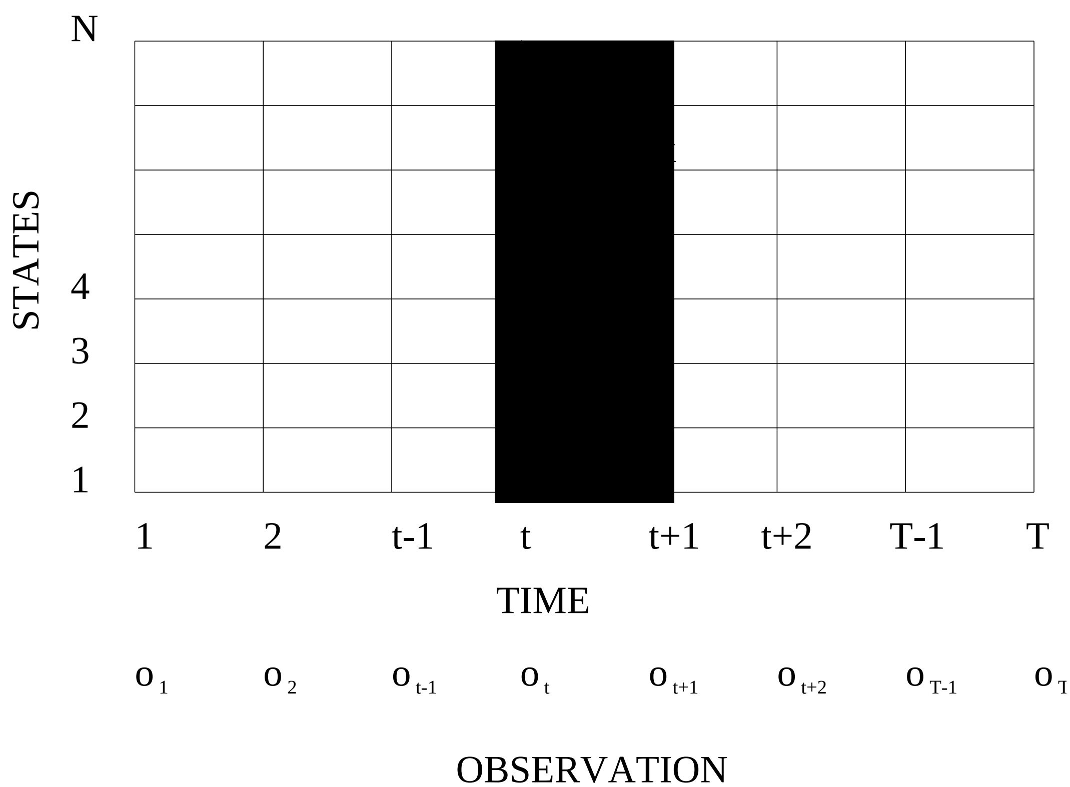
- **Solution by Dynamic Programming**
- **Define:**

$$\delta_t(i) = \max_{q_1, q_2, \cdots, q_{t-1}} P(q_1, q_2, \ldots, q_t = i, o_1, o_2, \ldots, o_t | \lambda)$$

- $\delta_t(i)$ **is the highest prob. path ending in state** i
- **By induction we have:**

$$\delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] \cdot b_j(o_{t+1})$$

Viterbi Algorithm



Viterbi Algorithm

- **Initialization:**

$$\begin{aligned} \delta_1(i) &= \pi_i b_i(o_1), \quad 1 \leq i \leq N \\ \psi_1(i) &= 0 \end{aligned}$$

- **Recursion:**

$$\begin{aligned} \delta_t(j) &= \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t) \\ \psi_t(j) &= \arg\max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] \\ & \qquad 2 \leq t \leq T, 1 \leq j \leq N \end{aligned}$$

- **Termination:**

$$\begin{aligned} P^* &= \max_{1 \leq i \leq N} [\delta_T(i)] \\ q_T^* &= \arg\max_{1 \leq i \leq N} [\delta_T(i)] \end{aligned}$$

- **Path (state sequence) backtracking:**

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \ldots, 1$$

Solution to Problem 3

- **Estimate** $\lambda = (A, B, \pi)$ **to maximize** $P(O|\lambda)$
- **No analytic method because of complexity – iterative solution.**
- **Baum-Welch Algorithm:**
  1. **Let initial model be** λ_0.
  2. **Compute new** λ **based on** λ_0 **and observation** O.
  3. **If** $\log P(O|\lambda) - \log P(O|\lambda_0) < DELTA$ **stop.**
  4. **Else set** $\lambda_0 \leftarrow \lambda$ **and goto step 2.**

Baum-Welch: Preliminaries

- **Define** $\xi_t(i,j)$ **as the probability of being in state** i **at time** t **and in state** j **at time** $t+1$**.**

$$\begin{aligned}\xi_t(i,j) &= P(q_t = i, q_{t+1} = j | O, \lambda) \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}\end{aligned}$$

- **Define** $\gamma_t(i)$ **as probability of being in state** i **at time** t**, given the observation sequence.**

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i,j)$$

- $\sum_{t=1}^T \gamma_t(i)$ **is the expected number of times state** i **is visited.**

- $\sum_{t=1}^{T-1} \xi_t(i,j)$ **is the expected number of transitions from state** i **to state** j**.**

Baum-Welch: Update Rules

- $\bar{\pi}_i$ = **expected frequency in state** i **at time** $(t = 1)$ = $\gamma_1(i)$.
- $\bar{a}_{ij}$ = **(expected number of transition from state** i **to state** j**)/ (expected number of transitions from state** i**):**

$$\bar{a}_{ij} = \frac{\sum \xi_t(i,j)}{\sum \gamma_t(i)}$$

- $\bar{b}_j(k)$ = **(expected number of times in state** j **and observing symbol** k**) / (expected number of times in state** j**:**

$$\bar{b}_j(k) = \frac{\sum_{t,o_t=k} \gamma_t(j)}{\sum_t \gamma(j)}$$

Properties

- **Covariance of the estimated parameters**
- **Convergence rates**

Types of HMM

- Continuous density
- Ergodic
- State duration

Implementation Issues

- Scaling
- Initial parameters
- Multiple observation/Pooling

Comparison of HMMs

- **What is a natural distance function?**
- **If $\rho(\lambda_1, \lambda_2)$ is large, does it mean that the models are really different?**

Probabilistic Model for IR

Miller, Leek and Schwartz, SIGIR 99

Compute: $P(D \text{ is } R|Q)$

By Bayes' rule:

$$P(D \text{ is } R|Q) = \frac{P(D \text{ is } R, Q)}{P(Q)} = \frac{P(Q|D \text{ is } R)P(D \text{ is } R)}{P(Q)}$$

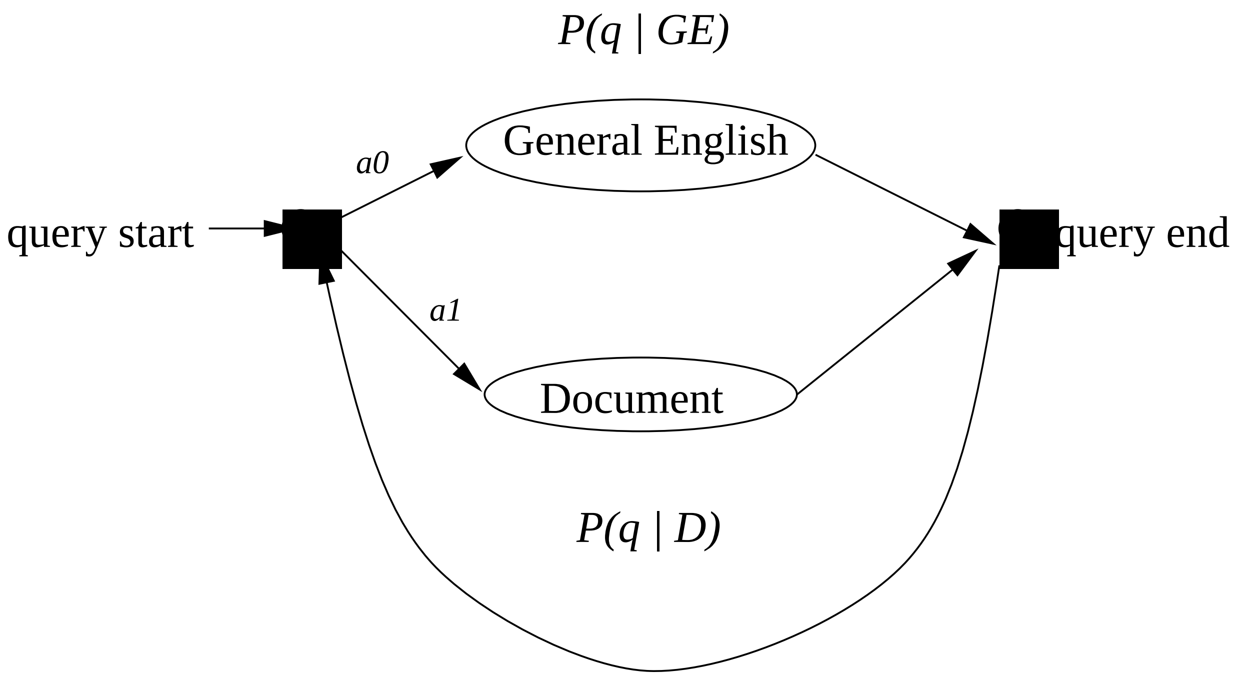
$P(Q|D \text{ is } R)$ – **prob. that the query is posed, given that the document is relevant**

$P(Q)$ – **prior prob. that the query will be posed;** $P(Q)$ **is identical for all queries.**

$P(D \text{ is } R)$ – **prior prob. that the document is relevant;** $P(D \text{ is } R)$ **is fixed.**

Worry about: $P(Q|D \text{ is } R)$

A Simple HMM-IR Mapping



HMM-IR Assumptions/Details

- **Transition probabilities are same across all docs**
- **Output distribution for "Document" state:**

$$P(q|D_k) = \frac{\textbf{num. of times } q \textbf{ appears in } D_k}{\textbf{length of } D_k}$$

- **Output distribution for "General English" state:**

$$P(q|GE) = \frac{\sum_k \textbf{ num. of times } q \textbf{ appears in } D_k}{\sum_k \textbf{ length of } D_k}$$

- **Now compute the probability:**

$$P(Q|D_k \textbf{ is } R) = \prod_{q \in Q} (a_0 P(q|GE) + a_1 P(q|D_k))$$

Experiments

- **Two state model**
- **TREC-6 dataset: 556,077 news docs**
- **TREC-7 dataset: subset of TREC-6, 528,155 docs**
- **50 test topics (queries) average of 88.4 words for TREC-6 and 57.6 words for TREC-7**
- **Porter stemming, token for 397 stopwords, token for money, number.**
- **After processing, 26.5 unique query terms for TREC-6, 17.6 for TREC-7**
- **Baum-Welch to train:** $a_1 = 0.3$

Results: Average Precision

	TREC-6			TREC-7		
	HMM	**tf.idf**	**Diff**	**HMM**	**tf.idf**	**Diff**
Title	**21.6**	**15.9**	+5.8	**16.1**	**11.6**	+4.5
Desc	**18.1**	**11.9**	+6.2	**18.3**	**14.2**	+4.1
Narr	**21.5**	**15.8**	+5.7	**17.7**	**14.7**	+3.0
Full	**27.1**	**18.9**	+8.2	**23.9**	**19.0**	+4.9