HMM: 3 basic problems

Leonid E. Zhukov

School of Applied Mathematics and Information Science
National Research University Higher School of Economics

16.11.2012

3 basic problems

- The Evaluation problem.
  Given:
  1. Observable sequence $O = O_1 O_2 O_3 ... O_T$
  2. model $\lambda = (A, B, \pi)$

  Find: $P(O|\lambda)$
- The Decoding problem.
  Given:
  1. Observable sequence $O = O_1 O_2 O_3 ... O_T$
  2. model $\lambda = (A, B, \pi)$

  Find: $Q^* = q_1 q_2 q_3 .. q_T$: $Q^* = \arg\max_Q P(Q|O, \lambda)$
- The Learning problem (training).
  Given:
  1. Observable sequence $O = O_1 O_2 O_3 ... O_T$

  Find: $\lambda^* = \arg\max_\lambda P(O|\lambda)$

The Evaluation problem

Find: $P(O|\lambda)$

Sequence of signals

$$P(O|\lambda) = \sum_{Q} P(O|Q,\lambda)P(Q|\lambda)$$

$$P(O|Q\lambda) = P(O_1|q_1,\lambda)P(O_2|q_2,\lambda)..P(O_T|q_T,\lambda) = \\ = b_{q_1}(O_1)b_{q_2}(O_2)...b_{q_T}(O_T)$$

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

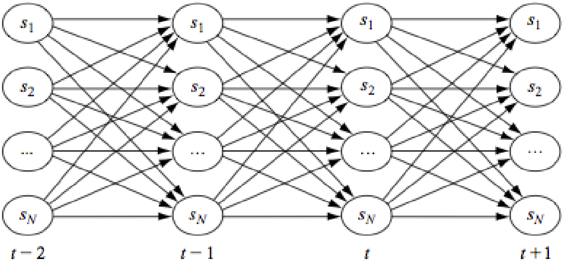
$$P(O|\lambda) = \sum_{q_1,..q_T} \pi_{q_1} b_{q_1}(O_1)b_{q_2}(O_2)...b_{q_T}(O_T) a_{q_1 q_2} a_{q_2 q_3} ... a_{q_{T-1} q_T} = \\ = \sum_{q_1,..q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) a_{q_2 q_3} ... a_{q_{T-1} q_T} b_{q_T}(O_T)$$

The Evaluation problem

Two time steps:

$$
\begin{aligned}
P(O_1, O_2) = \; & \pi_1 b_1(O_1) a_{11} b_1(O_2) + \\
& \pi_1 b_1(O_1) a_{12} b_2(O_2) + \\
& \pi_2 b_2(O_1) a_{22} b_2(O_2) + \\
& \pi_2 b_2(O_1) a_{21} b_1(O_2) +
\end{aligned}
$$

Computational Complexity: $O(2T \cdot N^T)$


s_1
s_2
...
s_N
$t-2$
$t-1$
t
$t+1$

Forward Algorithm

Partial observation sequence $O_1 ... O_t$ that terminates at state S_i
$\alpha_t(i) = P(O_1, O_2 ... O_t, q_t = S_i | \lambda)$, $t \leq T$, $1 \leq i \leq N$

then

$$
\begin{aligned}
\alpha_1(j) &= P(O_1, q_1 = S_j|\lambda) = \pi_i b_j(O_1) \\
\alpha_2(j) &= P(O_1, O_2, q_2 = S_j|\lambda) = \sum_i \alpha_1(i) a_{ij} b_j(O_2) \\
\alpha_3(j) &= P(O_1, O_2, O_3, q_3 = S_j|\lambda) = \sum_i \alpha_2(i) a_{ij} b_j(O_3) \\
\alpha_{t+1}(j) &= \sum_i \alpha_t(i) a_{ij} b_j(O_{t+1})
\end{aligned}
$$

$P(O|\lambda) = \sum_i P(O, q_T = S_i) = \sum_i \alpha_T(i)$

Forward-backward algorithm

Forward procedure

Let $\alpha_t(i) = P(O_1, O_2 ... O_t, q_t = S_i | \lambda)$, $t \leq T$

1. $\alpha_1(i) = \pi_i b_i(O_1)$
2. $\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_i \alpha_t(i) a_{ij}$
3. $P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$

Backward procedure

Let $\beta_t(i) = P(O_{t+1} ... O_T | q_t = S_i, \lambda)$, $t \leq T$

1. $\beta_T(i) = 1$
2. $\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$
3. $P(O|\lambda) = \sum_{i=1}^N \beta_1(i) \pi_i b_i(0_1)$

Complexity ?

The Decoding problem

Find: $Q^* = q_1 q_2 q_3..q_T$: $Q^* = \arg\max_Q P(Q|O,\lambda)$
The highest probability that partial observation and state sequences up to time t can have, when terminates at S_i.
$\delta_t(i) = max_{q_1,q_2,..q_{t-1}} P(O_1,..O_{t-1},q_1,..,q_t = S_i|\lambda)$

$$
\begin{aligned}
\delta_1(j) &= \max_{q_1} P(O_1, q_1 = S_j|\lambda) = \pi_j b_j(O_1) \\
\delta_2(j) &= \max_{q_1,q_2} P(O_1, O_2, q_2 = S_j|\lambda) = \max_i \{\delta_1(i) a_{ij}\} b_j(O_2) \\
\delta_3(j) &= \max_{q_1,q_2,q_3} P(O_1, O_2, O_3, q_3 = S_j|\lambda) = \max_i \{\delta_2(i) a_{ij}\} b_j(O_3) \\
\delta_{t+1}(j) &= \max_i \{\delta_t(i) a_{ij}\} b_j(O_{t+1})
\end{aligned}
$$

$q^* = \arg\max_j$

Viterbi Algorithm

$\delta_t(i) = max_{q_1,q_2,..q_{t-1}} P(O_1, ..O_{t-1}, q_1, .., q_t = S_i|\lambda)$

1. $\delta_1(i) = \pi_i b_i(O_1)$ $\psi_1(i) = 0$
2. $\delta_{t+1}(j) = \max_i\{\delta_t(i)a_{ij}\}b_j(O_{t+1}), \quad 1 \le i \le N, 1 \le t \le T-1$
   $\psi_{t+1}(j) = \arg\max\{\delta_t(i)a_{ij}\}b_j(O_{t+1})$
3. $q^*_T = \arg\max_i\{\delta_T(i)\}$

backtracking

$q^*_t = \psi_{t+1}(q^*_{t+1}), \quad t = T-1, T-2, ...1$

The Learning problem

HMM parameter estimation $\lambda = (A, B, \pi)$
Find: $\lambda^* = \arg\max_\lambda P(O|\lambda)$

For Markov chain parameter estimation (not HMM!)

$$P(X_1, ..X_n|\theta) = P(X_1) \prod_i \prod_j p_{ij}^{n_{ij}} = L(p)$$

Maximum likelihood MLE estimate:

$$p_{ij} = \frac{n_{ij}}{\sum_j n_{ij}}$$

n_{ij} - number of times in sequence X_J followed X_i, i.e. transition $i \to j$

In HMM can find expected number of transitions $\langle n_{ij} \rangle$

Forward-backward variables

forward variable
$\alpha_t(i) = P(O_1, O_2 ... O_t, q_t = S_i|\lambda)$
backward variable
$\beta_t(i) = P(O_{t+1} ... O_T|q_t = S_i, \lambda)$

Then:
$\alpha_t(i)\beta_t(i) = P(O_1, O_2 ... O_t, O_{t+1} ... O_T, q_t = S_i|\lambda) = P(O, q_t = S_i|\lambda)$

and
$P(O|\lambda) = \sum_i P(O, q_t = S_i|\lambda) = \sum_i \alpha_t(i)\beta_t(i)$

Baum-Welch Algorithm

$$P(O, q_t = S_i|\lambda) = P(q_t = S_i|O, \lambda)P(O|\lambda)$$

probability to visit state i at t

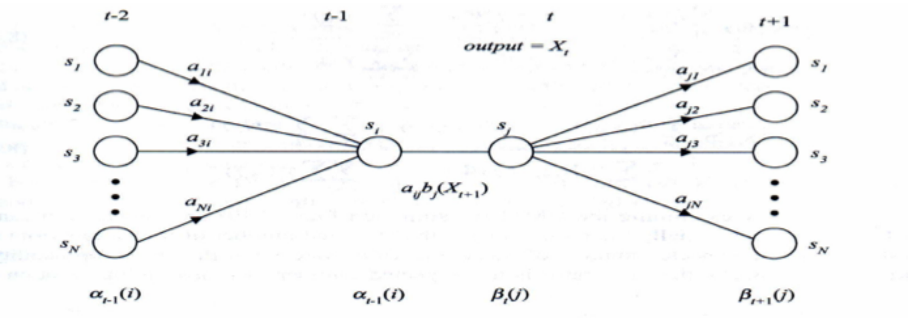
$$\gamma_t(i) = P(q_t = S_i|O, \lambda) = \frac{P(O, q_t = S_i|\lambda)}{P(O|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_i \alpha_t(i)\beta_t(i)}$$

probability to visit state i at t and j at $t+1$, $i \to j$ transition

$$\zeta_t(i,j) = P(q_t = S_i, q_{t+1} = S_j|O, \lambda) = \frac{P(O, q_t = S_i, q_{t+1} = S_j|\lambda)}{P(O|\lambda)} =$$

$$= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_i \sum_j \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}$$

$[\gamma_t(i) = \sum_j \zeta_t(i,j)]$


t-2
t-1
t
t+1
output = X_t
s_1
s_2
s_3
s_N
a_1i
a_2i
a_3i
a_Ni
s_i
s_j
a_ij b_j(X_t+1)
a_j1
a_j2
a_j3
a_jN
α_t-1(i)
α_t-1(i)
β_t(j)
β_t+1(j)

Baum-Welch Algorithm

- $\sum_{t=1}^{T} \gamma_t(i)$ - expected number of visits state i
- $\sum_{t=1}^{T-1} \gamma_t(i)$ - expected number of transitions from i
- $\sum_{t=1}^{T-1} \zeta_t(i,j)$ - expected number of transitions from i to j

Baum-Welch Algorithm

EM type maximization of $P(O|\lambda)$

- Set initial random values for A, B, π
- compute $\alpha_t(i), \beta_t(i), \gamma_t(i), \zeta_t(i,j)$
- estimate

$$\pi_i = \gamma_1(i)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \zeta_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\bar{b}_j(k) = \frac{\sum_{t=1, o_t=k}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(j)}$$

- repeat while $P(O|\lambda) = \sum_i \alpha_t(i)\beta_t(i)$ is growing