

# The Structure of Hidden Markov Models

- Have  $N$  states, states  $1 \dots N$
- Without loss of generality, take  $N$  to be the final or stop state
- Have an alphabet  $K$ . For example  $K = \{a, b\}$
- Parameter  $\pi_i$  for  $i = 1 \dots N$  is probability of starting in state  $i$
- Parameter  $a_{i,j}$  for  $i = 1 \dots (N - 1)$ , and  $j = 1 \dots N$  is probability of state  $j$  following state  $i$
- Parameter  $b_i(o)$  for  $i = 1 \dots (N - 1)$ , and  $o \in K$  is probability of state  $i$  emitting symbol  $o$

## An Example

- Take  $N = 3$  states. States are  $\{1, 2, 3\}$ . Final state is state 3.
- Alphabet  $K = \{the, dog\}$ .
- Distribution over initial state is  $\pi_1 = 1.0, \pi_2 = 0, \pi_3 = 0$ .
- Parameters  $a_{i,j}$  are

	j=1	j=2	j=3
i=1	0.5	0.5	0
i=2	0	0.5	0.5

- Parameters  $b_i(o)$  are

	o=the	o=dog
i=1	0.9	0.1
i=2	0.1	0.9

# A Generative Process

- Pick the start state  $s_1$  to be state  $i$  for  $i = 1 \dots N$  with probability  $\pi_i$ .
- Set  $t = 1$
- Repeat while current state  $s_t$  is not the stop state ( $N$ ):
  - Emit a symbol  $o_t \in K$  with probability  $b_{s_t}(o)$
  - Pick the next state  $s_{t+1}$  as state  $j$  with probability  $a_{s_t,j}$ .
  - $t = t + 1$

# Probabilities Over Sequences

- An **output sequence** is a sequence of observations  $o_1 \dots o_T$  where each  $o_i \in K$   
e.g. **the dog the dog dog the**
- An **state sequence** is a sequence of states  $s_1 \dots s_T$  where each  $s_i \in \{1 \dots N\}$   
e.g. **1 2 1 2 2 1**
- HMM defines a probability for each state/output sequence pair

e.g. **the/1 dog/2 the/1 dog/2 the/2 dog/1** has probability

$$\pi_1 b_1(\text{the}) a_{1,2} b_2(\text{dog}) a_{2,1} b_1(\text{the}) a_{1,2} b_2(\text{dog}) a_{2,2} b_2(\text{the}) a_{2,1} b_1(\text{dog}) a_{1,3}$$

Formally:

$$P(s_1 \dots s_T, o_1 \dots o_T) = \pi_1 \times \left( \prod_{i=2}^{T-1} P(s_i | s_{i-1}) \right) \times \left( \prod_{i=1}^T P(o_i | s_i) \right) \times P(N | s_T)$$

## A Hidden Variable Problem

- We have an HMM with  $N = 3$ ,  $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

e g  
e h  
f h  
f g

- How would you choose the parameter values for  $\pi_i$ ,  $a_{i,j}$ , and  $b_i(o)$ ?

## Another Hidden Variable Problem

- We have an HMM with  $N = 3$ ,  $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

e g h  
e h  
f h g  
f g g  
e h

- How would you choose the parameter values for  $\pi_i$ ,  $a_{i,j}$ , and  $b_i(o)$ ?

# A Reminder: Models with Hidden Variables

- Now say we have two sets  $\mathcal{X}$  and  $\mathcal{Y}$ , and a joint distribution  $P(X, Y \mid \Theta)$

- If we had **fully observed data**,  $(X_i, Y_i)$  pairs, then

$$L(\Theta) = \sum_i \log P(X_i, Y_i \mid \Theta)$$

- If we have **partially observed data**,  $X_i$  examples, then

$$\begin{aligned} L(\Theta) &= \sum_i \log P(X_i \mid \Theta) \\ &= \sum_i \log \sum_{Y \in \mathcal{Y}} P(X_i, Y \mid \Theta) \end{aligned}$$

# Hidden Markov Models as a Hidden Variable Problem

- We have two sets  $\mathcal{X}$  and  $\mathcal{Y}$ , and a joint distribution  $P(X, Y | \Theta)$
- In Hidden Markov Models:
  - each  $x \in \mathcal{X}$  is an output sequence  $o_1 \dots o_T$
  - each  $y \in \mathcal{Y}$  is an state sequence  $s_1 \dots s_T$



## Maximum Likelihood Estimates

- We have an HMM with  $N = 3$ ,  $K = \{e, f, g, h\}$   
We see the following **paired sequences** in training data

e/1    g/2

e/1    h/2

f/1    h/2

f/1    g/2

- Maximum likelihood estimates:

$$\pi_1 = 1.0, \quad \pi_2 = 0.0, \quad \pi_3 = 0.0$$

		j=1	j=2	j=3
for parameters $a_{i,j}$ :	i=1	0	1	0
	i=2	0	0	1

		o=e	o=f	o=g	o=h
for parameters $b_i(o)$ :	i=1	0.5	0.5	0	0
	i=2	0	0	0.5	0.5

# The Likelihood Function for HMMs: Fully Observed Data

- Say  $(x, y) = \{o_1 \dots o_T, s_1 \dots s_T\}$ , and

$f(i, j, x, y) =$  Number of times state  $j$  follows state  $i$  in  $(x, y)$

$f(i, x, y) =$  Number of times state  $i$  is the initial state in  $(x, y)$  (1 or 0)

$f(i, o, x, y) =$  Number of times state  $i$  is paired with observation  $o$

- Then

$$P(x, y) = \prod_{i \in \{1 \dots N-1\}} \pi_i^{f(i, x, y)} \prod_{\substack{i \in \{1 \dots N-1\}, \\ j \in \{1 \dots N\}}} a_{i,j}^{f(i,j,x,y)} \prod_{\substack{i \in \{1 \dots N-1\}, \\ o \in K}} b_i(o)^{f(i,o,x,y)}$$

# The Likelihood Function for HMMs: Fully Observed Data

- If we have training examples  $(x_l, y_l)$  for  $l = 1 \dots m$ ,

$$\begin{aligned} L(\Theta) &= \sum_{l=1}^m \log P(x_l, y_l) \\ &= \sum_{l=1}^m \left( \sum_{i \in \{1 \dots N-1\}} f(i, x_l, y_l) \log \pi_i + \right. \\ &\quad \sum_{\substack{i \in \{1 \dots N-1\}, \\ j \in \{1 \dots N\}}} f(i, j, x_l, y_l) \log a_{i,j} + \\ &\quad \left. \sum_{\substack{i \in \{1 \dots N-1\}, \\ o \in K}} f(i, o, x_l, y_l) \log b_i(o) \right) \end{aligned}$$

- Maximizing this function gives maximum-likelihood estimates:

$$\pi_i = \frac{\sum_l f(i, x_l, y_l)}{\sum_l \sum_k f(k, x_l, y_l)}$$

$$a_{i,j} = \frac{\sum_l f(i, j, x_l, y_l)}{\sum_l \sum_k f(i, k, x_l, y_l)}$$

$$b_i(o) = \frac{\sum_l f(i, o, x_l, y_l)}{\sum_l \sum_{o' \in K} f(i, o', x_l, y_l)}$$

# The Likelihood Function for HMMs: Partially Observed Data

- If we have training examples  $(x_l)$  for  $l = 1 \dots m$ ,

$$L(\Theta) = \sum_{l=1}^m \log \sum_y P(x_l, y)$$

$$Q(\Theta, \Theta^{t-1}) = \sum_{l=1}^m \sum_y P(y | x_l, \Theta^{t-1}) \log P(x_l, y | \Theta)$$

$$\begin{aligned}
Q(\Theta, \Theta^{t-1}) &= \sum_{l=1}^m \sum_y P(y | x_l, \Theta^{t-1}) \left( \sum_{i \in \{1 \dots N-1\}} f(i, x_l, y) \log \pi_i + \right. \\
&\quad \left. \sum_{\substack{i \in \{1 \dots N-1\}, \\ j \in \{1 \dots N\}}} f(i, j, x_l, y) \log a_{i,j} + \sum_{\substack{i \in \{1 \dots N-1\}, \\ o \in K}} f(i, o, x_l, y) \log b_i(o) \right) \\
&= \sum_{l=1}^m \left( \sum_{i \in \{1 \dots N-1\}} g(i, x_l) \log \pi_i + \sum_{\substack{i \in \{1 \dots N-1\}, \\ j \in \{1 \dots N\}}} g(i, j, x_l) \log a_{i,j} + \sum_{\substack{i \in \{1 \dots N-1\}, \\ o \in K}} g(i, o, x_l) \log b_i(o) \right)
\end{aligned}$$

where each  $g$  is an **expected count**:

$$\begin{aligned}
g(i, x_l) &= \sum_y P(y | x_l, \Theta^{t-1}) f(i, x_l, y) \\
g(i, j, x_l) &= \sum_y P(y | x_l, \Theta^{t-1}) f(i, j, x_l, y) \\
g(i, o, x_l) &= \sum_y P(y | x_l, \Theta^{t-1}) f(i, o, x_l, y)
\end{aligned}$$

- Maximizing this function gives EM updates:

$$\pi_i = \frac{\sum_l g(i, x_l)}{\sum_l \sum_k g(k, x_l)} \quad a_{i,j} = \frac{\sum_l g(i, j, x_l)}{\sum_l \sum_k g(i, k, x_l)} \quad b_i(o) = \frac{\sum_l g(i, o, x_l)}{\sum_l \sum_{o' \in K} g(i, o', x_l)}$$

- Compare this to maximum likelihood estimates in fully observed case:

$$\pi_i = \frac{\sum_l f(i, x_l, y_l)}{\sum_l \sum_k f(k, x_l, y_l)} \quad a_{i,j} = \frac{\sum_l f(i, j, x_l, y_l)}{\sum_l \sum_k f(i, k, x_l, y_l)} \quad b_i(o) = \frac{\sum_l f(i, o, x_l, y_l)}{\sum_l \sum_{o' \in K} f(i, o', x_l, y_l)}$$

## A Hidden Variable Problem

- We have an HMM with  $N = 3$ ,  $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

e g  
e h  
f h  
f g

- How would you choose the parameter values for  $\pi_i$ ,  $a_{i,j}$ , and  $b_i(o)$ ?



- Four possible state sequences for the first example:

e/1 g/1

e/1 g/2

e/2 g/1

e/2 g/2

- Four possible state sequences for the first example:

e/1    g/1

e/1    g/2

e/2    g/1

e/2    g/2

- Each state sequence has a different probability:

e/1    g/1                     $\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g)$

e/1    g/2                     $\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g)$

e/2    g/1                     $\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g)$

e/2    g/2                     $\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g)$

## A Hidden Variable Problem

- Say we have initial parameter values:

$$\pi_1 = 0.35, \quad \pi_2 = 0.3, \quad \pi_3 = 0.35$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.2	0.3	0.5
i=2	0.3	0.2	0.5

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.2	0.25	0.3	0.25
i=2	0.1	0.2	0.3	0.4

- Each state sequence has a different probability:

e/1	g/1	$\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g) = 0.0021$
e/1	g/2	$\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g) = 0.00315$
e/2	g/1	$\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g) = 0.00135$
e/2	g/2	$\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g) = 0.0009$

# A Hidden Variable Problem

- Each state sequence has a different probability:

e/1	g/1	$\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g) = 0.0021$
e/1	g/2	$\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g) = 0.00315$
e/2	g/1	$\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g) = 0.00135$
e/2	g/2	$\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g) = 0.0009$

- Each state sequence has a different **conditional** probability, e.g.:

$$P(1\ 1 \mid e\ g, \Theta) = \frac{0.0021}{0.0021 + 0.00315 + 0.00135 + 0.0009} = 0.28$$

e/1	g/1	$P(1\ 1 \mid e\ g, \Theta) = 0.28$
e/1	g/2	$P(1\ 2 \mid e\ g, \Theta) = 0.42$
e/2	g/1	$P(2\ 1 \mid e\ g, \Theta) = 0.18$
e/2	g/2	$P(2\ 2 \mid e\ g, \Theta) = 0.12$

## fill in hidden values for (e g), (e h), (f h), (f g)

$$e/1 \quad g/1 \quad P(1 \ 1 \mid e \ g, \Theta) = 0.28$$

$$e/1 \quad g/2 \quad P(1 \ 2 \mid e \ g, \Theta) = 0.42$$

$$e/2 \quad g/1 \quad P(2 \ 1 \mid e \ g, \Theta) = 0.18$$

$$e/2 \quad g/2 \quad P(2 \ 2 \mid e \ g, \Theta) = 0.12$$

$$e/1 \quad h/1 \quad P(1 \ 1 \mid e \ h, \Theta) = 0.211$$

$$e/1 \quad h/2 \quad P(1 \ 2 \mid e \ h, \Theta) = 0.508$$

$$e/2 \quad h/1 \quad P(2 \ 1 \mid e \ h, \Theta) = 0.136$$

$$e/2 \quad h/2 \quad P(2 \ 2 \mid e \ h, \Theta) = 0.145$$

$$f/1 \quad h/1 \quad P(1 \ 1 \mid f \ h, \Theta) = 0.181$$

$$f/1 \quad h/2 \quad P(1 \ 2 \mid f \ h, \Theta) = 0.434$$

$$f/2 \quad h/1 \quad P(2 \ 1 \mid f \ h, \Theta) = 0.186$$

$$f/2 \quad h/2 \quad P(2 \ 2 \mid f \ h, \Theta) = 0.198$$

$$f/1 \quad g/1 \quad P(1 \ 1 \mid f \ g, \Theta) = 0.237$$

$$f/1 \quad g/2 \quad P(1 \ 2 \mid f \ g, \Theta) = 0.356$$

$$f/2 \quad g/1 \quad P(2 \ 1 \mid f \ g, \Theta) = 0.244$$

$$f/2 \quad g/2 \quad P(2 \ 2 \mid f \ g, \Theta) = 0.162$$

**Calculate the expected counts:**

$$\sum_l g(1, x_l) = 0.28 + 0.42 + 0.211 + 0.508 + 0.181 + 0.434 + 0.237 + 0.356 = 2.628$$

$$\sum_l g(2, x_l) = 1.372$$

$$\sum_l g(3, x_l) = 0.0$$

$$\sum_l g(1, 1, x_l) = 0.28 + 0.211 + 0.181 + 0.237 = 0.910$$

$$\sum_l g(1, 2, x_l) = 1.72$$

$$\sum_l g(2, 1, x_l) = 0.746$$

$$\sum_l g(2, 2, x_l) = 0.626$$

$$\sum_l g(1, 3, x_l) = 1.656$$

$$\sum_l g(2, 3, x_l) = 2.344$$



**Calculate the expected counts:**

$$\sum_l g(1, e, x_l) = 0.28 + 0.42 + 0.211 + 0.508 = 1.4$$

$$\sum_l g(1, f, x_l) = 1.209$$

$$\sum_l g(1, g, x_l) = 0.941$$

$$\sum_l g(1, h, x_l) = 0.827$$

$$\sum_l g(2, e, x_l) = 0.6$$

$$\sum_l g(2, f, x_l) = 0.385$$

$$\sum_l g(2, g, x_l) = 1.465$$

$$\sum_l g(2, h, x_l) = 1.173$$



**Calculate the new estimates:**

$$\pi_1 = \frac{\sum_l g(1, x_l)}{\sum_l g(1, x_l) + \sum_l g(2, x_l) + \sum_l g(3, x_l)} = \frac{2.628}{2.628 + 1.372 + 0} = 0.657$$

$$\pi_2 = 0.343 \quad \pi_3 = 0$$

$$a_{1,1} = \frac{\sum_l g(1, 1, x_l)}{\sum_l g(1, 1, x_l) + \sum_l g(1, 2, x_l) + \sum_l g(1, 3, x_l)} = \frac{0.91}{0.91 + 1.72 + 1.656} = 0.212$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.212	0.401	0.387
i=2	0.201	0.169	0.631

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.320	0.276	0.215	0.189
i=2	0.166	0.106	0.404	0.324

**Iterate this 3 times:**

$$\pi_1 = 0.9986, \quad \pi_2 = 0.00138 \quad \pi_3 = 0$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.0054	0.9896	0.00543
i=2	0.0	0.0013627	0.9986

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.497	0.497	0.00258	0.00272
i=2	0.001	0.000189	0.4996	0.4992

# Overview

- The EM algorithm in general form  
(more about the 3 coin example)
- The EM algorithm for hidden markov models (brute force)
- The EM algorithm for hidden markov models (dynamic programming)

# The Forward-Backward or Baum-Welch Algorithm

- Aim is to (efficiently!) calculate the expected counts:

$$g(i, x_l) = \sum_y P(y | x_l, \Theta^{t-1}) f(i, x_l, y)$$

$$g(i, j, x_l) = \sum_y P(y | x_l, \Theta^{t-1}) f(i, j, x_l, y)$$

$$g(i, o, x_l) = \sum_y P(y | x_l, \Theta^{t-1}) f(i, o, x_l, y)$$

# The Forward-Backward or Baum-Welch Algorithm

---

- Suppose we could calculate the following quantities, given an input sequence  $o_1 \dots o_T$ :

$$\alpha_i(t) = P(o_1 \dots o_{t-1}, s_t = i \mid \Theta) \quad \text{forward probabilities}$$

$$\beta_i(t) = P(o_t \dots o_T \mid s_t = i, \Theta) \quad \text{backward probabilities}$$

- The probability of being in state  $i$  at time  $t$ , is

$$\begin{aligned} p_t(i) &= P(s_t = i \mid o_1 \dots o_T, \Theta) \\ &= \frac{P(s_t = i, o_1 \dots o_T \mid \Theta)}{P(o_1 \dots o_T \mid \Theta)} \\ &= \frac{\alpha_t(i)\beta_t(i)}{P(o_1 \dots o_T \mid \Theta)} \end{aligned}$$

also,

$$P(o_1 \dots o_T \mid \Theta) = \sum_i \alpha_t(i)\beta_t(i) \text{ for any } t$$

## Expected Initial Counts

- As before,

$g(i, o_1 \dots o_T) =$  expected number of times state  $i$  is state 1

- We can calculate this as

$$g(i, o_1 \dots o_T) = p_1(i)$$

## Expected Emission Counts

- As before,

$g(i, o, o_1 \dots o_T)$  = expected number of times state  $i$  emits the symbol  $o$

- We can calculate this as

$$g(i, o, o_1 \dots o_T) = \sum_{t:o_t=o} p_t(i)$$

# The Forward-Backward or Baum-Welch Algorithm

---

- Suppose we could calculate the following quantities, given an input sequence  $o_1 \dots o_T$ :

$$\alpha_i(t) = P(o_1 \dots o_{t-1}, s_t = i \mid \Theta) \quad \text{forward probabilities}$$

$$\beta_i(t) = P(o_t \dots o_T \mid s_t = i, \Theta) \quad \text{backward probabilities}$$

- The probability of being in state  $i$  at time  $t$ , and in state  $j$  at time  $t + 1$ , is

$$\begin{aligned} p_t(i, j) &= P(s_t = i, s_{t+1} = j \mid o_1 \dots o_T, \Theta) \\ &= \frac{P(s_t = i, s_{t+1} = j, o_1 \dots o_T \mid \Theta)}{P(o_1 \dots o_T \mid \Theta)} \\ &= \frac{\alpha_t(i) a_{i,j} b_i(o_t) \beta_{t+1}(j)}{P(o_1 \dots o_T \mid \Theta)} \end{aligned}$$

also,

$$P(o_1 \dots o_T \mid \Theta) = \sum_i \alpha_t(i) \beta_t(i) \text{ for any } t$$



## Expected Transition Counts

- As before,

$g(i, j, o_1 \dots o_T)$  = expected number of times state  $j$  follows state  $i$

- We can calculate this as

$$g(i, j, o_1 \dots o_T) = \sum_t p_t(i, j)$$

# Recursive Definitions for Forward Probabilities

- Given an input sequence  $o_1 \dots o_T$ :

$$\alpha_i(t) = P(o_1 \dots o_{t-1}, s_t = i \mid \Theta) \quad \text{forward probabilities}$$

- **Base case:**

$$\alpha_i(1) = \pi_i \quad \text{for all } i$$

- **Recursive case:**

$$\alpha_j(t+1) = \sum_i \alpha_i(t) a_{i,j} b_i(o_t) \quad \text{for all } j = 1 \dots N \text{ and } t = 2 \dots T$$

# Recursive Definitions for Backward Probabilities

- Given an input sequence  $o_1 \dots o_T$ :

$$\beta_i(t) = P(o_t \dots o_T \mid s_t = i, \Theta) \quad \text{backward probabilities}$$

- **Base case:**

$$\beta_i(T + 1) = 1 \quad \text{for } i = N$$

$$\beta_i(T + 1) = 0 \quad \text{for } i \neq N$$

- **Recursive case:**

$$\beta_i(t) = \sum_j a_{i,j} b_i(o_t) \beta_j(t+1) \quad \text{for all } j = 1 \dots N \text{ and } t = 1 \dots T$$

# Overview

- The EM algorithm in general form  
(more about the 3 coin example)
- The EM algorithm for hidden markov models (brute force)
- The EM algorithm for hidden markov models (dynamic programming)
- Briefly: The EM algorithm for PCFGs

# EM for Probabilistic Context-Free Grammars

- A PCFG defines a distribution  $P(S, T \mid \Theta)$  over tree/sentence pairs  $(S, T)$
- If we had tree/sentence pairs (**fully observed data**) then

$$L(\Theta) = \sum_i \log P(S_i, T_i \mid \Theta)$$

- Say we have sentences only,  $S_1 \dots S_n$   
 $\Rightarrow$  trees are hidden variables

$$L(\Theta) = \sum_i \log \sum_T P(S_i, T \mid \Theta)$$

# EM for Probabilistic Context-Free Grammars

- Say we have sentences only,  $S_1 \dots S_n$   
⇒ trees are hidden variables

$$L(\Theta) = \sum_i \log \sum_T P(S_i, T \mid \Theta)$$

- EM algorithm is then  $\Theta^t = \operatorname{argmax}_{\Theta} Q(\Theta, \Theta^{t-1})$ , where

$$Q(\Theta, \Theta^{t-1}) = \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \log P(S_i, T \mid \Theta)$$

- Remember:

$$\log P(S_i, T \mid \Theta) = \sum_{r \in R} \text{Count}(S_i, T, r) \log \Theta_r$$

where  $\text{Count}(S, T, r)$  is the number of times rule  $r$  is seen in the sentence/tree pair  $(S, T)$

$$\begin{aligned} \Rightarrow Q(\Theta, \Theta^{t-1}) &= \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \log P(S_i, T \mid \Theta) \\ &= \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \sum_{r \in R} \text{Count}(S_i, T, r) \log \Theta_r \\ &= \sum_i \sum_{r \in R} \text{Count}(S_i, r) \log \Theta_r \end{aligned}$$

where  $\text{Count}(S_i, r) = \sum_T P(T \mid S_i, \Theta^{t-1}) \text{Count}(S_i, T, r)$   
**the expected counts**

- Solving  $\Theta_{ML} = \operatorname{argmax}_{\Theta \in \Omega} L(\Theta)$  gives

$$\Theta_{\alpha \rightarrow \beta} = \frac{\sum_i \text{Count}(S_i, \alpha \rightarrow \beta)}{\sum_i \sum_{s \in R(\alpha)} \text{Count}(S_i, s)}$$

- There are efficient algorithms for calculating

$$\text{Count}(S_i, r) = \sum_T P(T \mid S_i, \Theta^{t-1}) \text{Count}(S_i, T, r)$$

for a PCFG. See (Baker 1979), called “The Inside Outside Algorithm”. See also Manning and Schuetze section 11.3.4.