Lecture 7.5.  NFN's to SCFG's

SCFG = (N, \Sigma, S, \mathcal{R})

- N - finite set of terminals \( \mathcal{V}_0 \) (Sam, thin, snow)
- \( \Sigma \) - finite set of non-terminals \( \mathcal{S}_0 = \{ S, NP, VP, V \} \)
- \( \mathcal{R} \) - finite set of production, \( A \rightarrow BC \in \mathcal{S}_0 \), or \( A \rightarrow x \in \mathcal{S}_0 \)
- \( s \in \mathcal{S} \) is the start symbol (\( S_0 = S \))

Production probability

\[ P(A \rightarrow \beta) = \frac{P(\beta|A)}{P(\beta)} \quad \text{for each } A \rightarrow \beta \in \mathcal{R} \]

\( r_1, \ldots, r_n \) is sequence of productions used to generate a tree \( \Psi \), then

\[ P_\Psi(\Psi) = \prod_{i=1}^n P(r_i) = \prod_{i=1}^n \frac{P(r_i|A)}{P(A)} \quad \text{for } r_i \text{-th move} \]

What do we want to compute?

1. What is the prob. \( P_\Psi(w) \) of string \( w \)

\[ P_\Psi(w) = \sum_{\Psi \in \mathcal{P}(w)} P_\Psi(\Psi) = \sum_{\Psi \in \mathcal{P}(w)} \frac{P_\Psi(\Psi)}{P(\Psi)} \]

2. What is the most probable parse \( \Psi(\Psi | w) \) of string \( w \)?

\[ \Psi(\Psi | w) = \arg \max_{\Psi \in \mathcal{P}(w)} P_\Psi(\Psi) \]

- \( \mathcal{P}(w) \) - set of parse trees for \( w \) generated by \( G \)
- \( P_\Psi(\Psi) \) - prob of tree \( \Psi \) w.r.t. grammar \( G \)

SCFG "inside" algorithm

Goal: Compute \( P(w) = \sum_{\Psi \in \mathcal{P}(w)} P(\Psi) = P(S \rightarrow^* w) \)

Data structure: table \( P(A \rightarrow^* w_i | \omega) \) for \( A \in \mathcal{S} \)

Base case: \( P(A \rightarrow^* w_i | \omega) = 0 \), \( i < 0 \)

Return:\( P(A \rightarrow^* w_i | \omega) = \sum_{j=0}^{n} P(A \rightarrow BC) P(BC \rightarrow^* w_i | \omega_j) P(C \rightarrow^* \omega) \)

Return: \( P(S \rightarrow^* w_i | \omega) \)

\( P_\Psi(A \rightarrow^* w_i | \omega) \) is called the "inside probability"
(2) Learning: Dataset of Examples

Assume the data is visible.

- Count how frequently a rule is applied

Rule: $S \rightarrow NP VP \quad \text{Count 3 \ Frequency 1}$
$NP \rightarrow Rice \quad \text{Count 2 \ "25"}$

This is also the MLE for the SCFG (i.e., $P(\text{data} | \text{parameters})$)

MLE is consistent
asymptotically optimal

But, this requires that the data is `visible` or has been labelled. (like first Hmm case)

Unsupervised Training: EM (again)

$Z = (x, y)$
- $x$ - observed data,
- $y$ - hidden variables (tags)
- $\theta$ - parameters

EM algorithm: given visible data $x$:

1. Guess initial value $\theta_0$ of parameters (rule pg)
2. Repeat:
   - $E$-Step: For all $y_1 \ldots y_n \in Y$ generate pseudo-data $(x, y_1) \ldots (x, y_n) \sim (x, y) | \text{frequency}$
   - $M$-Step: Set $\theta_{n+1}$ to MLE from the pseudo-data

sometimes can perform maximization directly from sufficient statistics:

(expected production frequency)

$$E_\theta \sum_{A \rightarrow BC} (w) = \sum_{A \rightarrow BC, \ z_1 \ldots z_n} \text{expected fraction of passes of } w \text{ in which } A \text{ is rewritten as } B_{ij} C_{\gamma n}$$

$$SE_\theta \sum_{A \rightarrow \gamma n} = \frac{P(S \rightarrow \omega w_i A \omega_i, n) P(A \rightarrow BC)}{P(S \rightarrow \gamma n) P(C \gamma \omega w_i, n) / P\_n}$$
(b) Calculate $P_\theta(S \Rightarrow^* \omega_{oi} A \omega_{kn})$

"outside" probabilities

Recursion from larger to smaller substrings in $\omega$

Base case: $P(\omega \Rightarrow^* \omega_{oi} A \omega_{kn})$

Recursion: $P(\omega \Rightarrow^* \omega_{ij} C A \omega_{kn}) =$

$$\sum_{i=1}^{n} \sum_{\omega_{ij} \in \text{rules}} P(\omega \Rightarrow^* \omega_{oi} A \omega_{kn}) P(A \Rightarrow BC) P(B \Rightarrow C)$$

$$+ \sum_{i=1}^{n} \sum_{\omega_{ij} \in \text{rules}} P(\omega \Rightarrow^* \omega_{oi} A \omega_{kn}) P(A \Rightarrow CD) P(D \Rightarrow \omega_{kn})$$

Recursion in $P_\theta(S \Rightarrow^* \omega_{oi} A \omega_{kn})$

$$P(\omega \Rightarrow^* \omega_{oi} C \omega_{kn}) =$$

$$\sum_{i=0}^{n} \sum_{\omega_{ij} \in \text{rules}} P(\omega \Rightarrow^* \omega_{oi} A \omega_{kn}) P(A \Rightarrow BC) P(B \Rightarrow C)$$

$$+ \sum_{i=1}^{n} \sum_{\omega_{ij} \in \text{rules}} P(\omega \Rightarrow^* \omega_{oi} A \omega_{kn}) P(A \Rightarrow CD) P(D \Rightarrow \omega_{kn})$$

EM for SCFG:

Infer 'hidden structure' by ML of visible data.
1. guess initial rule probs.
2. repeat:
   (a) parse a sample of sentences
   (b) weight each parse by its cond. prob
   (c) count rules used in each weighted parse, and estimate rule frequencies

- EM optimizes the likelihood of data $D$
- Each iteration is guaranteed not to decrease the likelihood of $D$
- But EM can get trapped in local minima
- The inside-outside algorithm can produce the expected counts without enumerating all parse trees
Important to realize that even follows from the free energy formula:

\[ - \log P(D|\theta) \quad D = \{x_1, \ldots, x_n\} \]
\[ P(D|\theta) = \prod_{i=1}^{m} P(x_i|\theta) \]

\[ P(D|\theta) = P(D, y|\theta) = P(D|y, \theta) P(y) \]

\[ Z[q, \theta] = \sum_{y} q(y) \log q(y) - \sum_{y} q(y) \log P(y, D|\theta) \]

E-step: compute \( q^{t+1}(y) = \frac{p(y|D, \theta^t)}{P(D|\theta^t)} \)

M-step: compute \( \theta^{t+1} = \text{arg max } \theta \left\{ \sum_{y} q^{t+1}(y) \log p(y|D, \theta) \right\} \).