HMM Inference: The Viterbi Algorithm

HMMs: model of $P(\bar{y}, \bar{x}) = P(y_1)\,P(x_1 | y_1)\,P(y_2 | y_1) \ldots$

Inference: $\operatorname*{argmax}_{\bar{y}} P(\bar{y} | \bar{x}) = \operatorname*{argmax}_{\bar{y}} \dfrac{P(\bar{y}, \bar{x})}{P(\bar{x})}$ constant w.r.t. $\bar{y}$

$$= \operatorname*{argmax}_{\bar{y}} P(\bar{y}, \bar{x}) = \operatorname*{argmax}_{\bar{y}} \log P(\bar{y}, \bar{x})$$

$$= \operatorname*{argmax}_{\tilde{y}_1, \ldots, \tilde{y}_n} \log P(\tilde{y}_1) + \log P(x_1 | \tilde{y}_1) + \log P(\tilde{y}_2 | \tilde{y}_1) + \ldots$$

Viterbi Dynamic Program

Define $v_i(\tilde{y}) = n \times |\mathcal{T}|$ — n sent len; $|\mathcal{T}|$ number of tags

score of the best path ending in $\tilde{y}$ at time i

Base: $v_1(\tilde{y}) = \log P(x_1 | \tilde{y}) + \log P(\tilde{y})$

Recurrence: $v_i(\tilde{y}) = \log P(x_i | \tilde{y}) + \max_{\tilde{y}_{prev}} \log P(\tilde{y} | \tilde{y}_{prev}) + v_{i-1}(\tilde{y}_{prev})$

<u>Viterbi</u>

for $i = 1 \ldots n$
  for $\tilde{y}$ in $\mathcal{T}$:
    compute $v_i(\tilde{y})$

Compute $v_{n+1}(\text{STOP})$, this $= \max_{\bar{y}} \log P(\bar{x}, \bar{y})$

Track "backpointers"

Example   log probs

$$S = \begin{array}{c} N \\ V \end{array} \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$T = \begin{array}{c|ccc} & N & V & STOP \\ \hline N & -2 & -1 & -1 \\ V & -1 & -1 & -2 \end{array}$$

$$E = \begin{array}{c|ccc} & \text{they} & \text{can} & \text{fish} \\ \hline N & -1 & -3 & -1 \\ V & -3 & -1 & -1 \end{array}$$

$V_i(\tilde{y})$

	they	can	can	fish	STOP
N	-2	★ $-2-2-3=\boxed{-7}$ prev tr em $\cancel{-4-1-3=-8}$			
V	-4	$-2-1-1=\boxed{-4}$			

STOP