COS 484: Natural Language Processing

Expectation Maximization

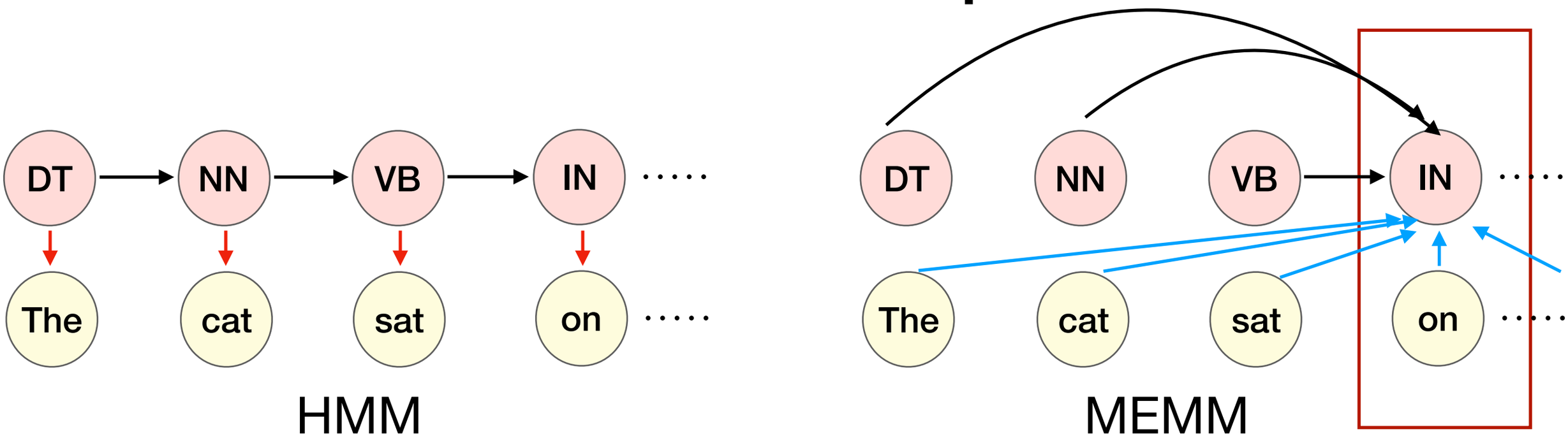
Fall 2019

(some slides adapted from Regina Barzilay and Michael Collins)

Announcements

- Oct 22: Midterm review
- Oct 24: Midterm (in-class)
- Nov 5: Project details + PyTorch tutorial
- Nov 11: Project proposal due
    - Start forming teams now! (2-3 members)
        - Can use Piazza
- Nov 18: Assignment 4 due date changed

MEMM recap



- In general, we can use all observations and all previous states:

$$\hat{S} = \arg\max_S P(S \mid O) = \arg\max_S \prod_i P(s_i \mid o_n, o_{n-1}, \ldots, o_1, s_{i-1}, \ldots, s_1)$$

$$P(s_i \mid s_{i-1}, \ldots, s_1, O) \propto \exp(w \cdot f(s_i, s_{i-1}, \ldots, s_1, O)$$

Features in an MEMM

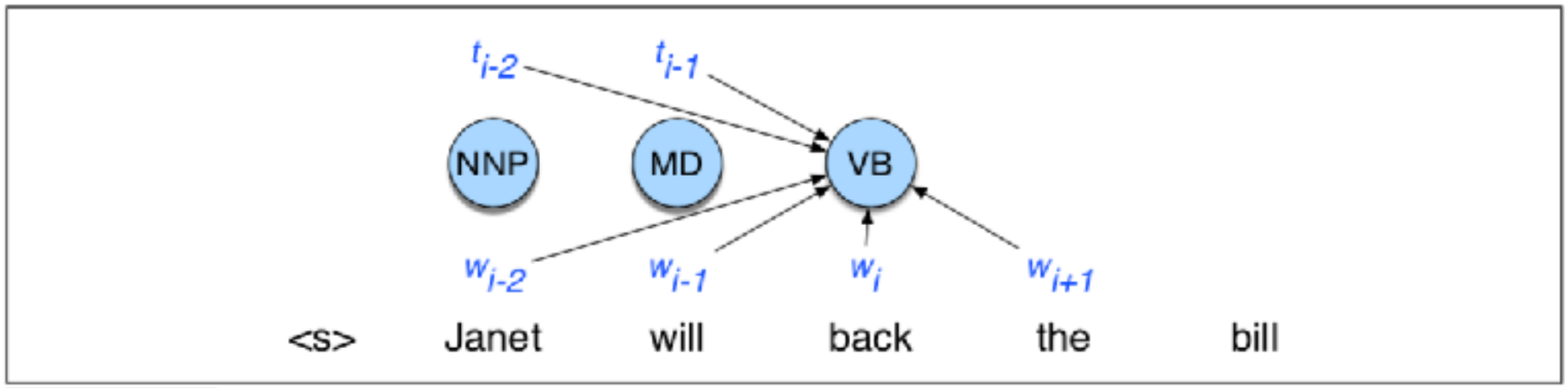


Figure 8.13 An MEMM for part-of-speech tagging showing the ability to condition on more features.

$$\langle t_i, w_{i-2}\rangle, \langle t_i, w_{i-1}\rangle, \langle t_i, w_i\rangle, \langle t_i, w_{i+1}\rangle, \langle t_i, w_{i+2}\rangle$$
$$\langle t_i, t_{i-1}\rangle, \langle t_i, t_{i-2}, t_{i-1}\rangle,$$
$$\langle t_i, t_{i-1}, w_i\rangle, \langle t_i, w_{i-1}, w_i\rangle \langle t_i, w_i, w_{i+1}\rangle,$$

Feature templates

t_i = VB and w_{i-2} = Janet
t_i = VB and w_{i-1} = will
t_i = VB and w_i = back
t_i = VB and w_{i+1} = the
t_i = VB and w_{i+2} = bill
t_i = VB and t_{i-1} = MD
t_i = VB and t_{i-1} = MD and t_{i-2} = NNP
t_i = VB and w_i = back and w_{i+1} = the

Features

MEMM: Learning

- Gradient descent: similar to logistic regression!

$$P(s_i \mid s_1, \ldots, s_{i-1}, O) \propto \exp(w \cdot f(s_1, \ldots, s_i, O))$$

- Given: pairs of (S, O) where each $S = \langle s_1, s_2, \ldots, s_n \rangle$

Loss for one sequence, $L = -\sum_i \log P(s_i \mid s_1, \ldots, s_{i-1}, O)$

- Compute gradients with respect to weights w and update

EM: Some intuition

- Let's say I have 3 coins in my pocket,
  - Coin 0 has probability λ of heads
    Coin 1 has probability p_1 of heads
    Coin 2 has probability p_2 of heads
- **For each trial:**
  - First I toss Coin 0
    If coin 0 turns up **heads**, I toss coin 1 three times
    If coin 0 turns up **tails**, I toss coin 2 three times

    I don't tell you the results of the coin 0 toss, or whether coin 1 or coin 2 was tossed, but I tell you how many heads/tails are seen after each trial
- You see the following sequence:
  $\langle H, H, H \rangle, \langle T, T, T \rangle, \langle H, H, H \rangle, \langle T, T, T \rangle, \langle H, H, H \rangle$

What would you estimate as values for λ, p_1, p_2 ?

Maximum Likelihood Estimate

- Data points $x_1, x_2, \ldots, x_n$ from (finite or countable) set $\mathcal{X}$
- Parameter vector θ
- Parameter space Ω
- We have a distribution $P(x \mid \theta)$ for any $\theta \in \Omega$, such that

$$\sum_{x \in \mathcal{X}} P(x \mid \theta) = 1 \text{ and } P(x \mid \theta) \geq 0 \quad \forall x$$

- Assume data points are drawn independently and identically distributed from a distribution $P(x \mid \theta^*)$ for some $\theta^* \in \Omega$

Log Likelihood

- Data points $x_1, x_2, \ldots, x_n$ from (finite or countable) set $\mathcal{X}$
- Parameter vector θ and a parameter space Ω
- Probability distribution $P(x \mid \theta)$ for any $\theta \in \Omega$
- Likelihood(θ) = $P(x_1, x_2, \ldots, x_n \mid \theta) = \prod_{i=1}^{n} P(x_i \mid \theta)$
- Log-likelihood, $L(\theta) = \sum_{i=1}^{n} \log P(x_i \mid \theta)$

Example 1: Coin Tossing

- $\mathcal{X} = \{H, T\}$. Our data points $x_1, x_2, \ldots, x_n$ are a sequence of heads and tails, e.g.
    - HTHTHHHHTTT
- Parameter vector θ is a single parameter, i.e probability of coin coming up heads
- Parameter space $\Omega = [0,1]$
- Distribution $P(x \mid \theta) = \begin{cases} \theta \text{ if } x = H \\ 1 - \theta \text{ if } x = T \end{cases}$

Example 2: Markov chains

- $\mathcal{X}$ is the set of all possible state (e.g tag) sequences generated by the underlying generative process. Our sample is n sequences $X_1, \ldots, X_n$ such that each $X_i \in \mathcal{X}$, consists of a sequence of states.
- θ_T is the vector of all transition $(s_i \rightarrow s_j)$ parameters. Without loss of generality, assume a dummy start state ϕ and initial transition $\phi \rightarrow s_1$ (how many parameters?)
- Let $T(\alpha) \subset T$ be all the transitions of the form $\alpha \rightarrow \beta$
- Parameter space Ω is the set of $\theta \in [0,1]^{|S+1||S|}$ where S is set of all states (tags), such that:

$$\text{for all } \alpha \in S, \sum_{t \in T(\alpha)} \theta_t = 1$$

Example 2: Markov chains

- θ_T is the vector of all transition parameters
- We have:

$$P(X \mid \theta) = \prod_{t \in T} \theta_t^{Count(X,t)}$$

where $Count(X, t)$ is the number of times transition t is seen in sequence X

$$\implies \log P(X \mid \theta) = \sum_{t \in T} Count(X, t) \ \log \theta_t$$

MLE for Markov chains

- We have

$$\log P(X \mid \theta) = \sum_{t \in T} Count(X, t) \ \log \theta_t$$

  where $Count(X, t)$ is the number of times transition t is seen in sequence X

- And,

$$L(\theta) = \sum_i \log P(X_i \mid \theta) = \sum_i \sum_{t \in T} Count(X_i, t) \ \log \theta_t$$

MLE for Markov chains

- $L(\theta) = \sum_i \log P(X_i \mid \theta) = \sum_i \sum_{t \in T} Count(X_i, t) \ \log \theta_t$

- Solve $\theta_{MLE} = \arg\max_{\theta \in \Omega} L(\theta)$

  $\Longrightarrow$ find θ s. t. $\frac{\partial L(\theta)}{\partial \theta} = 0$ with appropriate probability constraints

- This gives: $\theta_t = \frac{\sum_i Count(X_i, t)}{\sum_i \sum_{t' \in T(\alpha)} Count(X_i, t')}$

  where t is of the form $\alpha \rightarrow \beta$ for some β

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 observable data**, (x_i, y_i) pairs, then

$$L(\theta) = \sum_i \log P(x_i, y_i \mid \theta)$$

- If we have **partially observable data**, x_i examples only, then

$$L(\theta) = \sum_i \log P(x_i \mid \theta)$$

$$= \sum_i \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \theta)$$

Unsupervised Learning

Expectation Maximization

- If we have **partially observable data**, x_i examples only, then

$$L(\theta) = \sum_i \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \theta)$$

- The EM (Expectation Maximization) algorithm is a method for finding

$$\theta_{MLE} = \arg\max_\theta L(\theta) = \arg\max_\theta \sum_i \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \theta)$$

The three coins example

- In the three coins example,

  $\mathcal{Y} = \{H, T\}$ (possible outcomes of coin 0)

  $\mathcal{X} = \{HHH, TTT, HTT, THH, HHT, TTH, HTH, THT\}$

  $\theta = \{\lambda, p_1, p_2\}$

- and $P(x, y \,|\, \theta) = P(y \,|\, \theta)\ P(x \,|\, y, \theta)$

  where

$$P(y \,|\, \theta) = \begin{cases} \lambda \text{ if } y = H \\ 1 - \lambda \text{ if } y = T \end{cases}$$

  and

$$P(x \,|\, y, \theta) = \begin{cases} p_1^h\ (1 - p_1)^t \text{ if } y = H \\ p_2^h\ (1 - p_2)^t \text{ if } y = T \end{cases}$$

The three coins example

- Calculating various probabilities:

$$P(x = THT, y = H \mid \theta) = \lambda p_1 (1 - p_1)^2$$

$$P(x = THT, y = T \mid \theta) = (1 - \lambda) p_2 (1 - p_2)^2$$

- $$P(x = THT \mid \theta) = P(x = THT, y = H \mid \theta) + P(x = THT, y = T \mid \theta)$$
$$= \lambda p_1 (1 - p_1)^2 + (1 - \lambda) p_2 (1 - p_2)^2$$

$$P(y = H \mid x = THT, \theta) = \frac{P(x = THT, y = H \mid \theta)}{P(x = THT \mid \theta)}$$

$$= \frac{\lambda p_1 (1 - p_1)^2}{\lambda p_1 (1 - p_1)^2 + (1 - \lambda) p_2 (1 - p_2)^2}$$

The three coins example

- Fully observed data might look like:
  $(\langle HHH \rangle, H), (\langle TTT \rangle, T), (\langle HHH \rangle, H), (\langle TTT \rangle, T), (\langle HHH \rangle, H)$
- In this case, maximum likelihood estimates are:

$$\lambda = \frac{3}{5}$$

$$p_1 = \frac{9}{9}$$

$$p_2 = \frac{0}{6}$$

The three coins example

- Partially observed data might look like:

$$\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle$$

- How do we find the MLE parameters?

The three coins example

- Partially observed data might look like:

$$\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle$$

- If the current parameters are λ, p_1, p_2

$$P(y = H \,|\, x = \langle HHH\rangle) = \frac{P(\langle HHH\rangle, H)}{P(\langle HHH\rangle, H) + P(\langle HHH\rangle, T)}$$

$$= \frac{\lambda p_1^3}{\lambda p_1^3 + (1-\lambda) p_2^3}$$

$$P(y = H \,|\, x = \langle TTT\rangle) = \frac{P(\langle TTT\rangle, H)}{P(\langle TTT\rangle, H) + P(\langle TTT\rangle, T)}$$

$$= \frac{\lambda (1-p_1)^3}{\lambda (1-p_1)^3 + (1-\lambda)(1-p_2)^3}$$

The three coins example

- If the current paramters are λ, p_1, p_2

$$P(y = H \,|\, x = \langle HHH \rangle) = \frac{P(\langle HHH \rangle, H)}{P(\langle HHH \rangle, H) + P(\langle HHH \rangle, T)}$$

$$= \frac{\lambda p_1^3}{\lambda p_1^3 + (1 - \lambda) p_2^3}$$

$$P(y = H \,|\, x = \langle TTT \rangle) = \frac{P(\langle HHH \rangle, H)}{P(\langle TTT \rangle, H) + P(\langle TTT \rangle, T)}$$

$$= \frac{\lambda (1 - p_1)^3}{\lambda (1 - p_1)^3 + (1 - \lambda)(1 - p_2)^3}$$

- If $\lambda = 0.3,\ p_1 = 0.3,\ p_2 = 0.6$:

$$P(y = H \,|\, x = \langle HHH \rangle) = 0.0508$$

$$P(y = H \,|\, x = \langle TTT \rangle) = 0.6967$$

The three coins example

- After filling in hidden variables for each example, partially observed data might look like:

$$
\begin{array}{ll}
(\langle \texttt{HHH} \rangle, H) & P(y = \texttt{H} \mid \texttt{HHH}) = 0.0508 \\
(\langle \texttt{HHH} \rangle, T) & P(y = \texttt{T} \mid \texttt{HHH}) = 0.9492 \\
(\langle \texttt{TTT} \rangle, H) & P(y = \texttt{H} \mid \texttt{TTT}) = 0.6967 \\
(\langle \texttt{TTT} \rangle, T) & P(y = \texttt{T} \mid \texttt{TTT}) = 0.3033 \\
(\langle \texttt{HHH} \rangle, H) & P(y = \texttt{H} \mid \texttt{HHH}) = 0.0508 \\
(\langle \texttt{HHH} \rangle, T) & P(y = \texttt{T} \mid \texttt{HHH}) = 0.9492 \\
(\langle \texttt{TTT} \rangle, H) & P(y = \texttt{H} \mid \texttt{TTT}) = 0.6967 \\
(\langle \texttt{TTT} \rangle, T) & P(y = \texttt{T} \mid \texttt{TTT}) = 0.3033 \\
(\langle \texttt{HHH} \rangle, H) & P(y = \texttt{H} \mid \texttt{HHH}) = 0.0508 \\
(\langle \texttt{HHH} \rangle, T) & P(y = \texttt{T} \mid \texttt{HHH}) = 0.9492
\end{array}
$$

The three coins example

- New estimates:

$$
\begin{array}{ll}
(\langle \mathtt{HHH} \rangle, H) & P(y = \mathtt{H} \mid \mathtt{HHH}) = 0.0508 \\
(\langle \mathtt{HHH} \rangle, T) & P(y = \mathtt{T} \mid \mathtt{HHH}) = 0.9492 \\
(\langle \mathtt{TTT} \rangle, H) & P(y = \mathtt{H} \mid \mathtt{TTT}) = 0.6967 \\
(\langle \mathtt{TTT} \rangle, T) & P(y = \mathtt{T} \mid \mathtt{TTT}) = 0.3033 \\
\cdots &
\end{array}
$$

$$\lambda = \frac{3 \times 0.0508 + 2 \times 0.6967}{5} = 0.3092$$

$$p_1 = \frac{3 \times 3 \times 0.0508 + 0 \times 2 \times 0.6967}{3 \times 3 \times 0.0508 + 3 \times 2 \times 0.6967} = 0.0987$$

$$p_2 = \frac{3 \times 3 \times 0.9492 + 0 \times 2 \times 0.3033}{3 \times 3 \times 0.9492 + 3 \times 2 \times 0.3033} = 0.8244$$

Summary

- Begin with parameters: $\lambda = 0.3,\ p_1 = 0.3,\ p_2 = 0.6$
- Fill in hidden variables, using
  $P(y = H \mid x = \langle HHH \rangle) = 0.0508$
  $P(y = H \mid x = \langle TTT \rangle) = 0.6967$
- Re-estimate parameters to be
  $\lambda = 0.3092,\ p_1 = 0.0987,\ p_2 = 0.8244$

EM iterations

Iteration	λ	p_1	p_2	$\bar{p}_1$	$\bar{p}_2$	$\bar{p}_3$	$\bar{p}_4$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967
1	0.3738	0.0680	0.7578	0.0004	0.9714	0.0004	0.9714
2	0.4859	0.0004	0.9722	0.0000	1.0000	0.0000	1.0000
3	0.5000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000

The coin example for $x = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. The solution that EM reaches is intuitively correct: the coin tosser has two coins, one which always shows heads, and another which always shows tails, and is picking between them with equal probability ($\lambda = 0.5$).

Posterior probabilities $\bar{p}_i$ show that we are certain that coin 1 (tail-biased) generate x_2 **and** x_4, whereas coin 2 generated x_1 **and** x_3

EM iterations

Iteration	λ	p_1	p_2	$\bar{p}_1$	$\bar{p}_2$	$\bar{p}_3$	$\bar{p}_4$	$\bar{p}_5$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967	0.0508
1	0.3092	0.0987	0.8244	0.0008	0.9837	0.0008	0.9837	0.0008
2	0.3940	0.0012	0.9893	0.0000	1.0000	0.0000	1.0000	0.0000
3	0.4000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000

Coin example for $\{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle\}$

λ is now 0.4, indicating that coin 0 has a probability 0.4 of selecting the tail-biased coin

EM iterations

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.3000	0.6000	0.1579	0.6967	0.0508	0.6967
1	0.4005	0.0974	0.6300	0.0375	0.9065	0.0025	0.9065
2	0.4632	0.0148	0.7635	0.0014	0.9842	0.0000	0.9842
3	0.4924	0.0005	0.8205	0.0000	0.9941	0.0000	0.9941
4	0.4970	0.0000	0.8284	0.0000	0.9949	0.0000	0.9949

Coin example for $x = \{\langle HHT\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$.

EM selects a tails-only coin, and a coin which is heavily heads-biased ($p_2 = 0.8284$). It's certain that x_1 and x_3 were generated by coin 2 since they contain heads. x_2 and x_4 could have been generated by either coin but coin 1 (tail-biased) is far more likely.

Initialization matters

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7000	0.7000	0.3000	0.3000	0.3000	0.3000
1	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
2	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
3	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
4	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
5	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
6	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000

Coin example for $x = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$.

In this case, EM is stuck at a **saddle point**.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7001	0.7000	0.3001	0.2998	0.3001	0.2998
1	0.2999	0.5003	0.4999	0.3004	0.2995	0.3004	0.2995
2	0.2999	0.5008	0.4997	0.3013	0.2986	0.3013	0.2986
3	0.2999	0.5023	0.4990	0.3040	0.2959	0.3040	0.2959
4	0.3000	0.5068	0.4971	0.3122	0.2879	0.3122	0.2879
5	0.3000	0.5202	0.4913	0.3373	0.2645	0.3373	0.2645
6	0.3009	0.5605	0.4740	0.4157	0.2007	0.4157	0.2007
7	0.3082	0.6744	0.4223	0.6447	0.0739	0.6447	0.0739
8	0.3593	0.8972	0.2773	0.9500	0.0016	0.9500	0.0016
9	0.4758	0.9983	0.0477	0.9999	0.0000	0.9999	0.0000
10	0.4999	1.0000	0.0001	1.0000	0.0000	1.0000	0.0000
11	0.5000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000

Coin example for $x = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$.

If we initialize p_1 and p_2 even a small amount away from the saddle point $p_1 = p_2$, EM diverges and eventually reaches the global maximum

The EM algorithm

- θ^t is the parameter vector at the t^{th} iteration
- Choose θ^0 at random (or using smart heuristics)
- Iterative procedure defined as:

$$\theta^t = \arg\max_{\theta} Q(\theta, \theta^{t-1})$$

where

$$Q(\theta, \theta^{t-1}) = \sum_i \sum_{y \in \mathcal{Y}} P(y \mid x_i, \theta^{t-1}) \ \log P(x_i, y \mid \theta)$$

The EM algorithm

- θ^t is the parameter vector at the t^{th} iteration
- Choose θ^0 at random (or using smart heuristics)
- (E step): Compute *expected* counts

$$\overline{Count}(r) = \sum_{i=1}^{n} \sum_{y} P(y \mid x_i, \theta^{t-1}) \; Count(x_i, y, r)$$

  for every parameter θ_r

- e.g.

$$\overline{Count}(DT \rightarrow NN) = \sum_{i} \sum_{y} P(S \mid O_i, \theta^{t-1}) \; Count(O_i, S, \theta_{DT \rightarrow NN})$$

The EM algorithm

- θ^t is the parameter vector at the t^{th} iteration
- Choose θ^0 at random (or using smart heuristics)
- (E step): Compute *expected* counts

$$\overline{Count}(r) = \sum_{i=1}^{n} \sum_{y} P(y \mid x_i, \theta^{t-1})\ Count(x_i, y, r)$$

  for every parameter θ_r

- (M step): Re-estimate parameters using expected counts to maximize likelihood

$$\text{e.g. } \theta_{DT \to NN} = \frac{\overline{Count}(DT \to NN)}{\sum_{\beta} \overline{Count}(DT \to \beta)}$$

The EM algorithm

- Iterative procedure defined as $\theta^t = \arg\max_\theta Q(\theta, \theta^{t-1})$ where

  $$Q(\theta, \theta^{t-1}) = \sum_i \sum_{y \in \mathcal{Y}} P(y \mid x_i, \theta^{t-1}) \ \log P(x_i, y \mid \theta)$$

- Key points:
  - Intuition: Fill in hidden variables y according to $P(y \mid x_i, \theta)$
  - EM is guaranteed to converge to a local maximum, or saddle-point, of the likelihood function
  - In general, if $\arg\max_\theta \sum_i \log P(x_i, y_i \mid \theta)$ has a simple analytic solution, then $\arg\max_\theta \sum_i \sum_y P(y \mid x_i, \theta) \log P(x_i, y \mid \theta)$ also has a simple solution.

Example: EM for HMM

- We observe only word sequences $X_1, X_2, \ldots, X_n$ (no tags Y)
- θ is the vector of all transition parameters (include initial state distribution as a special case, $\varnothing \rightarrow s$
- ϕ is the vector of all emission parameters
- Initialize parameters θ^0 and ϕ^0

Example: EM for HMM

- Initialize parameters θ^0 and ϕ^0
- **(E-Step)**

$$\overline{Count}(\theta_k) = \sum_{i=1}^{n} \sum_{Y} P(Y|X_i, \theta^{t-1}, \phi^{t-1})\ Count(X_i, Y, \theta_k)$$

$$= \sum_{i=1}^{n} \sum_{Y} P(Y|X_i, \theta^{t-1}, \phi^{t-1})\ Count(Y, \theta_k)$$

$$\overline{Count}(\phi_k) = \sum_{i=1}^{n} \sum_{Y} P(Y|X_i, \theta^{t-1}, \phi^{t-1})\ Count(X_i, Y, \phi_k)$$

Example: EM for HMM

- Initialize parameters θ^0 and ϕ^0
- **(M-Step)**

$$\theta_k^t = \frac{\overline{Count}(\theta_k)}{\sum_{\theta' \in M(\theta_k)} \overline{Count}(\theta')}$$ where $M(\theta_k)$ is the set of all transitions $(a \to b, \text{ all } b)$ that share the same previous state as the k^{th} transition $(a \to c \text{ for some } c)$.

$$\phi_k^t = \frac{\overline{Count}(\phi_k)}{\sum_{\phi' \in M'(\phi_k)} \overline{Count}(\phi')}$$ where $M'(\phi_k)$ is the set of all emissions $(a \to x, \text{ all } x)$ that share the same hidden state as the k^{th} emission $(a \to x', \text{ for some } x')$.

Efficient EM?

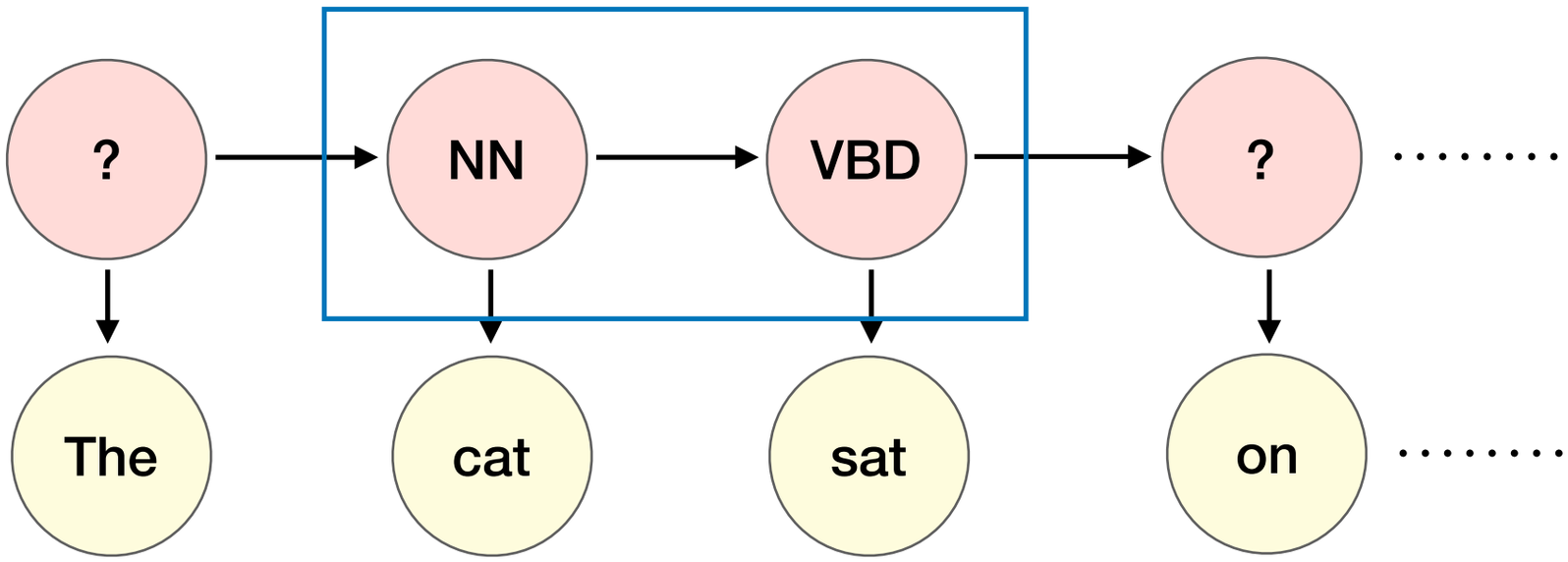
- **(E-Step)**

$$\overline{Count}(\theta_k) = \sum_{i=1}^{n} \sum_{Y} P(Y \mid X_i, \theta^{t-1}, \phi^{t-1})\ Count(Y, \theta_k)$$

$$\overline{Count}(\phi_k) = \sum_{i=1}^{n} \sum_{Y} P(Y \mid X_i, \theta^{t-1}, \phi^{t-1})\ Count(X_i, Y, \phi_k)$$

Cannot enumerate all possible Y!

Efficient EM?



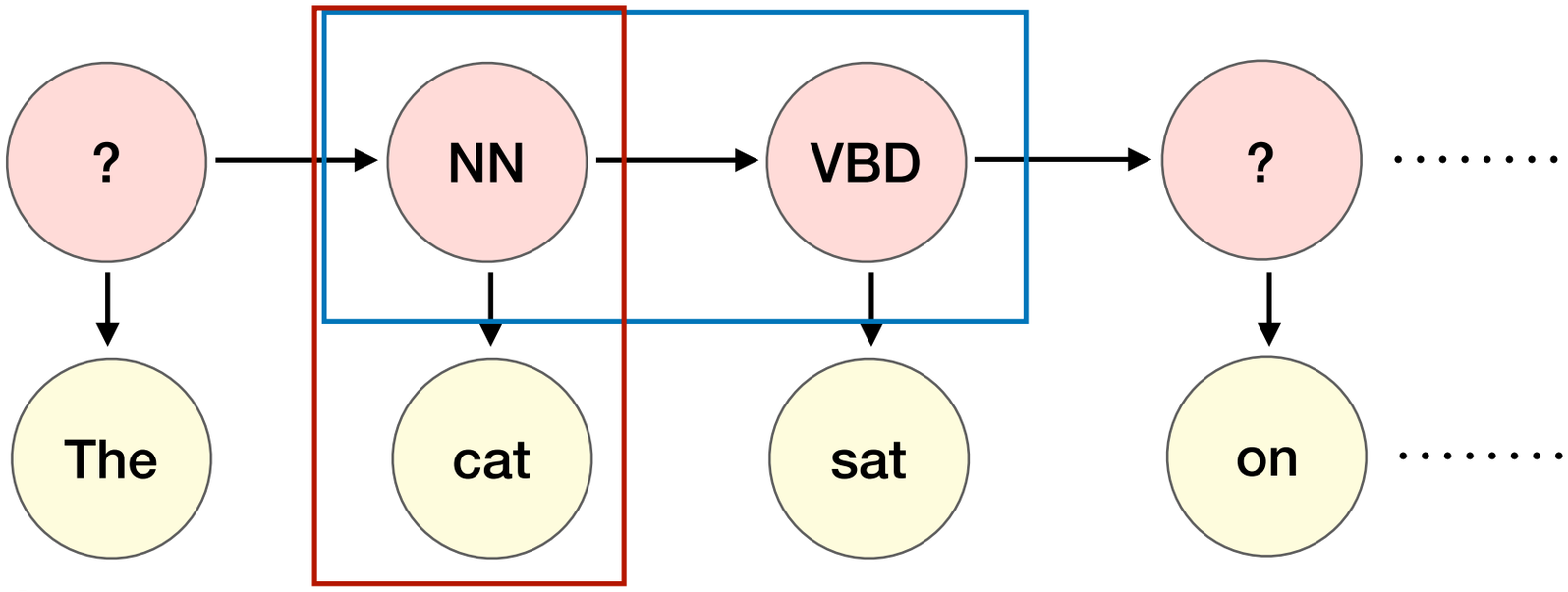
- **(E-Step)**

$$\overline{Count}(\theta_{NN \to VBD}) = \sum_{i=1}^{n} \sum_{Y} P(Y \mid X_i, \theta^{t-1}, \phi^{t-1}) \; Count(Y, \theta_k)$$

$$= \sum_{i} \sum_{j=1}^{m} P(y_j = NN, y_{j+1} = VBD \mid X_i, \theta^{t-1}, \phi^{t-1})$$

where m is the length of the sequence X_i

Efficient EM?



- **(E-Step)**

$$\overline{Count}(\theta_{NN \to VBD}) = \sum_{i=1}^{n} \sum_{Y} P(Y \mid X_i, \theta^{t-1}, \phi^{t-1}) \; Count(Y, \theta_k)$$

$$= \sum_{i} \sum_{j=1}^{m} P(y_j = NN, y_{j+1} = VBD \mid X_i, \theta^{t-1}, \phi^{t-1})$$

where m is the length of the sequence X_i

Similarly, $\overline{Count}(\phi_{NN \to cat}) = \sum_{i} \sum_{j:X_{ij} = cat} P(y_j = NN \mid X_i, \theta^{t-1}, \phi^{t-1})$

Forward-backward algorithm

- Define:

  $\alpha_s(j) = P(x_1, \ldots, x_{j-1}, y_j = s \,|\, \theta, \phi)$ (forward probability)

  $\beta_s(j) = P(x_j, \ldots, x_m \,|\, y_j = s, \theta, \phi)$ (backward probability)

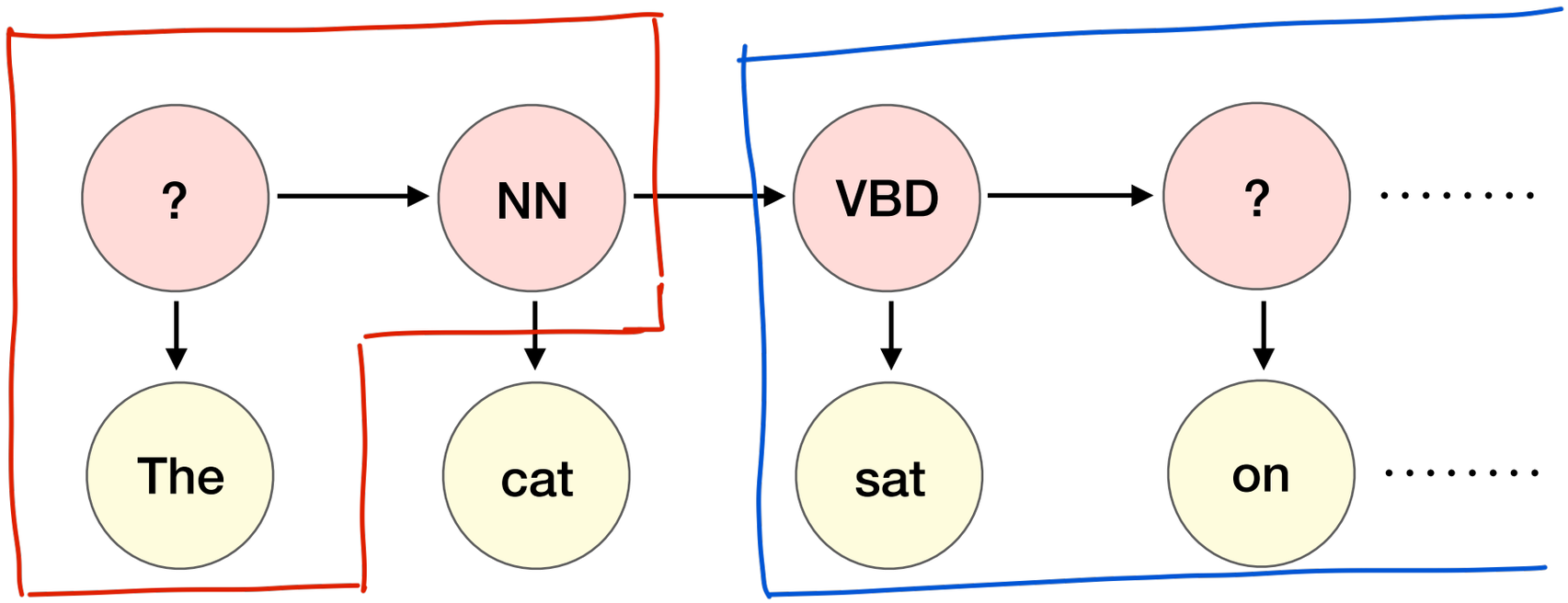
- Observation likelihood,

  $Z = P(x_1, x_2, \ldots, x_m \,|\, \theta, \phi) = \sum_s \alpha_s(j)\beta_s(j)$ for any $j \in 1,\ldots,m$

- $P(y_j = s \,|\, X, \theta, \phi) = \dfrac{\alpha_s(j)\beta_s(j)}{Z}$

  $P(y_j = s, y_{j+1} = s' \,|\, X, \theta, \phi) = \dfrac{\alpha_s(j)\ \theta_{s \to s'}\ \phi_{s \to x_j}\ \beta_{s'}\ (j+1)}{Z}$

Forward-backward algorithm



$\alpha_{NN}(2)$ $\beta_{VBD}(3)$

- $P(y_j = s \mid X, \theta, \phi) = \dfrac{\alpha_s(j)\beta_s(j)}{Z}$

$$P(y_j = s, y_{j+1} = s' \mid X, \theta, \phi) = \frac{\alpha_s(j)\ \theta_{s \to s'}\ \phi_{s \to x_j}\ \beta_{s'}\ (j+1)}{Z}$$

Forward-backward algorithm

- $P(y_j = s \,|\, X, \theta, \phi) = \dfrac{\alpha_s(j)\beta_s(j)}{Z}$

$$P(y_j = s, y_{j+1} = s' \,|\, X, \theta, \phi) = \frac{\alpha_s(j)\ \theta_{s \to s'}\ \phi_{s \to x_j}\ \beta_{s'}\ (j+1)}{Z}$$

- Given these, we can now estimate:

$$\overline{Count}(\theta_{s \to s'}) = \sum_i \sum_{j=1}^{m} P(y_j = s, y_{j+1} = s' \,|\, X_i, \theta, \phi)$$

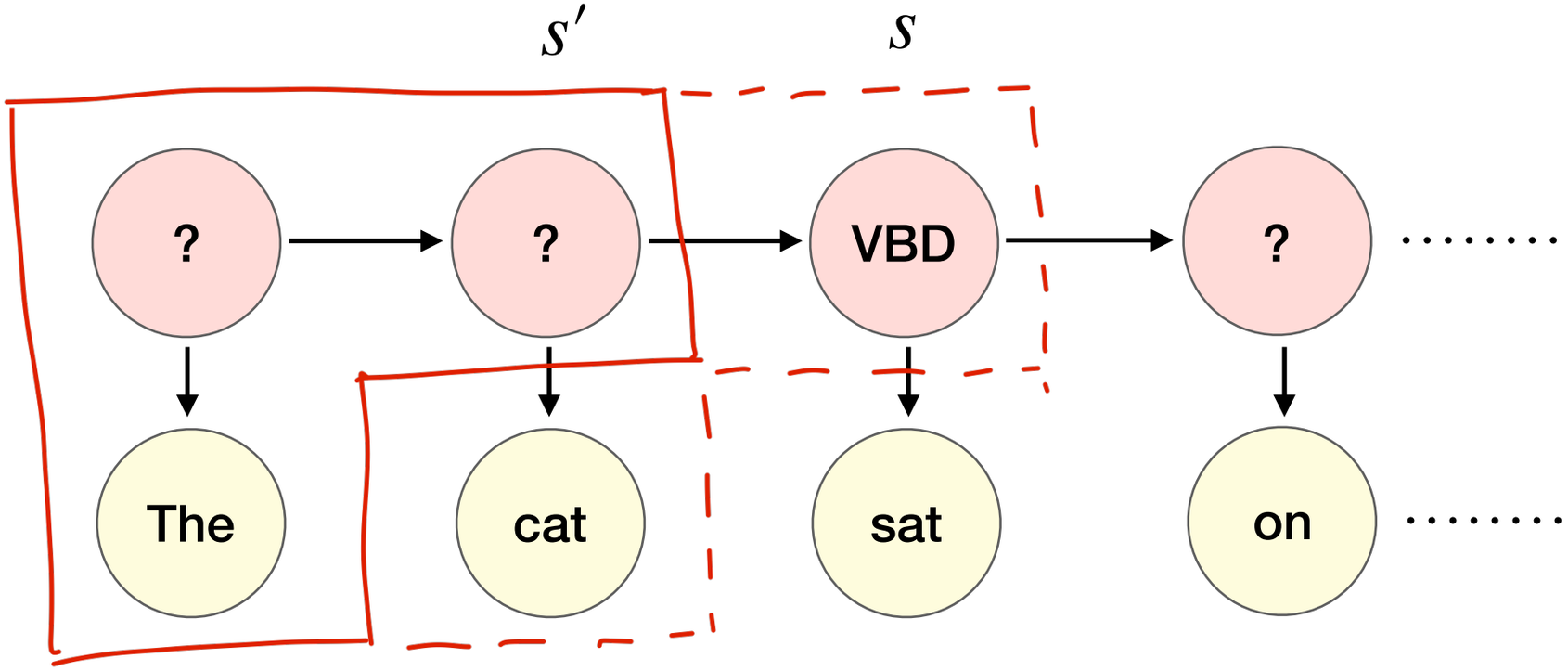
$$\overline{Count}(\phi_{s \to o}) = \sum_i \sum_{j : X_{ij} = o} P(y_j = s \,|\, X_i, \theta, \phi)$$

Dynamic programming

$$\alpha_s(j) = P(y_j = s, x_1, \ldots, x_{j-1})$$

$$= \sum_{s'} P(y_{j-1} = s', x_1, \ldots, x_{j-2})\ P(x_{j-1} \,|\, y_{j-1} = s')\ P(y_j = s \,|\, y_{j-1} = s')$$

$$= \sum_{s'} \alpha_{s'}\ (j-1)\ \phi_{s' \to x_{j-1}}\ \theta_{s' \to s}$$



Dynamic programming

$$\alpha_s(j) = P(y_j = s, x_1, \ldots, x_{j-1})$$
$$= \sum_{s'} P(y_{j-1} = s', x_1, \ldots, x_{j-2})\ P(x_{j-1} \mid y_{j-1} = s')\ P(y_j = s \mid y_{j-1} = s')$$
$$= \sum_{s'} \alpha_{s'}\ (j-1)\ \phi_{s' \to x_{j-1}}\ \theta_{s' \to s}$$

- Similarly,

$$\beta_s(j) = \phi_{s \to x_j} \sum_{s'} \beta_{s'}\ (j+1) \quad \theta_{s \to s'}$$

- Runtime: $O(|S|^2 \cdot m)$