Conditional Random Fields

11-711 recitation

Maria Ryskina

Example task: NER

PER PER O
Michael Bloomberg ,

O O ORG ORG O
founder of Bloomberg L.P. ,

O O LOC LOC
lives in New York

Recap: Logistic Regression

x	input	New York
$\mathcal{Y}$	candidate set	{O PER, O ORG, …}
$y \in \mathcal{Y}$	candidate label	O PER
$f(x, y)$	feature function	[1 0 0 0 1 … 0]
y^*	true ("gold") label	LOC LOC

Model form:

$$P(y|x,w) = \frac{\exp(w^\top f(x,y))}{\sum_{y' \in \mathcal{Y}} \exp(w^\top f(x,y'))} \propto \exp(w^\top f(x,y))$$

Recap: Logistic Regression

$\{x^{(1)}, \dots x^{(N)}\}$ training instances

$\{y^{*(1)}, \dots y^{*(N)}\}$ gold outputs

Learning: maximizing likelihood for gold outputs

$$L(w) = \log \prod_{k=1}^{N} P(y^{*(k)}|x^{(k)}, w) = \sum_{k=1}^{N} \log \left(\frac{\exp(w^\top f(x^{(k)}, y^{*(k)}))}{\sum_{y'} \exp(w^\top f(x^{(k)}, y'))} \right)$$

Recap: Logistic Regression

$\{x^{(1)}, \dots x^{(N)}\}$ training instances

$\{y^{*(1)}, \dots y^{*(N)}\}$ gold outputs

Learning: maximizing likelihood for gold outputs

$$L(w) = \log \prod_{k=1}^{N} P(y^{*(k)} | x^{(k)}, w) = \sum_{k=1}^{N} \log \left(\frac{\exp(w^\top f(x^{(k)}, y^{*(k)}))}{\sum_{y'} \exp(w^\top f(x^{(k)}, y'))} \right)$$

$$L(w) = \sum_{k=1}^{N} \left(w^\top f(x^{(k)}, y^{*(k)}) - \log \sum_{y'} \exp(w^\top f(x^{(k)}, y')) \right)$$

Recap: Logistic Regression

$$L(w) = \sum_{k=1}^{N} \left(w^\top f(x^{(k)}, y^{*(k)}) - \log \sum_{y' \in \mathcal{Y}} \exp(w^\top f(x^{(k)}, y')) \right)$$

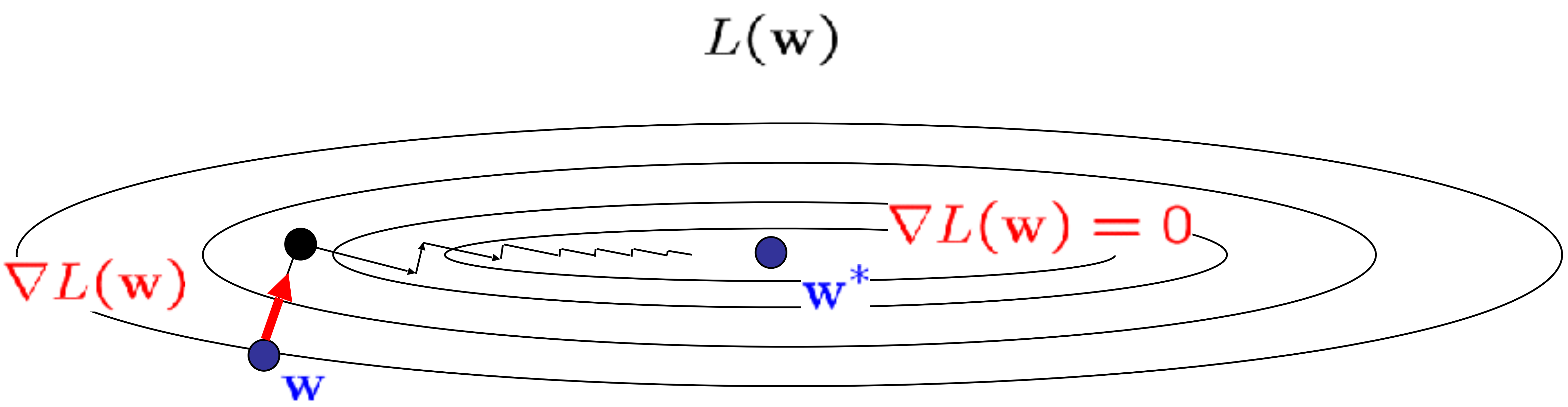
$$\frac{\partial L(w)}{\partial w} = \sum_{k=1}^{N} \left(f(x^{(k)}, y^{*(k)}) - \sum_{y' \in \mathcal{Y}} P(y' | x^{(k)}, w) f(x^{(k)}, y') \right)$$

Count of features under target labels

Expected count of features under model predicted label distribution

Gradient Ascent

- The maxent objective is an unconstrained optimization problem



- Gradient Ascent
  - Basic idea: move uphill from current guess
  - Gradient ascent / descent follows the gradient incrementally
  - At local optimum, derivative vector is zero
  - Will converge if step sizes are small enough, but not efficient
  - All we need is to be able to evaluate the function and its derivative

Source: Lecture 5

Recap: Logistic Regression

Prediction:

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} P(y|x, w)$$

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} (w^\top f(x, y))$$

If y is a sequence, $\mathcal{Y}$ grows exponentially!

Structured output space

$x = \{x_1, \dots x_n\}$ input (sequential)

$y = \{y_1, \dots y_n\}$ candidates (sequential)

Computation issues arise:

- sum over output space in training
- argmax over output space in decoding

Structured output space

$x = \{x_1, \ldots x_n\}$ input (sequential)

$y = \{y_1, \ldots y_n\}$ candidates (sequential)

Computation issues arise:

- sum over output space in training $\leftarrow$ Forward-backward
- argmax over output space in decoding $\leftarrow$ Viterbi

Solution: dynamic programming

Factorization assumption

To be able to use DP, we have to assume:

$$f(x,y) = \sum_{t=1}^{n+1} f(x, y_t, y_{t-1})$$

$$y_0 = \text{START}$$
$$y_{n+1} = \text{STOP}$$

Abuse of notation! $f(x, y_t, y_{t-1}) \triangleq f(x, y_t, y_{t-1}, t)$

CRF model form:

$$P(y|x,w) = \frac{1}{Z(w)} \exp\left(w^\top \sum_{t=1}^{n+1} f(x, y_t, y_{t-1}) \right)$$

CRF vs. HMM

CRF: $$P(y|x,w) \propto \exp\left(w^\top \sum_{t=1}^{n+1} f(x, y_t, y_{t-1})\right)$$

Compare to HMM:

$$P(x,y) = P(x_1|y_1)P(y_1) \cdot \prod_{t=2}^{n} P(y_t|y_{t-1})P(x_t|y_t)$$

- CRF is discriminative
  Conditional objective means no modeling $P(x)$
- CRF is more expressive
  Models dependence between each state and full sequence

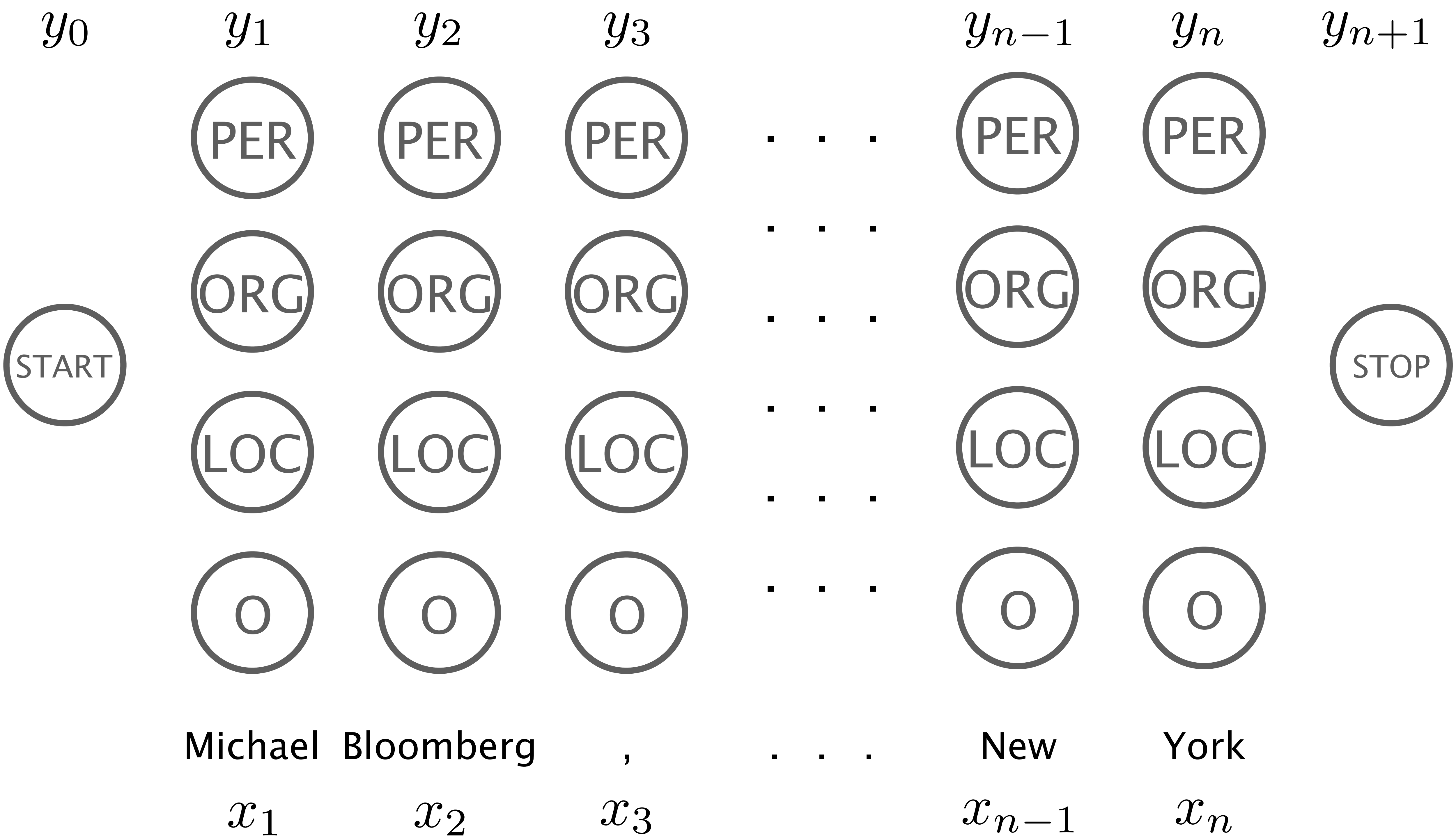
Computing gradients

$$\frac{\partial L(w)}{\partial w} = \sum_{k=1}^{N} \left(\sum_{t=1}^{n+1} f(x^{(k)}, y_t^{*(k)}, y_{t-1}^{*(k)}) - \sum_{y' \in \mathcal{Y}} P(y' | x^{(k)}, w) \sum_{t=1}^{n+1} f(x^{(k)}, y'_t, y'_{t-1}) \right)$$

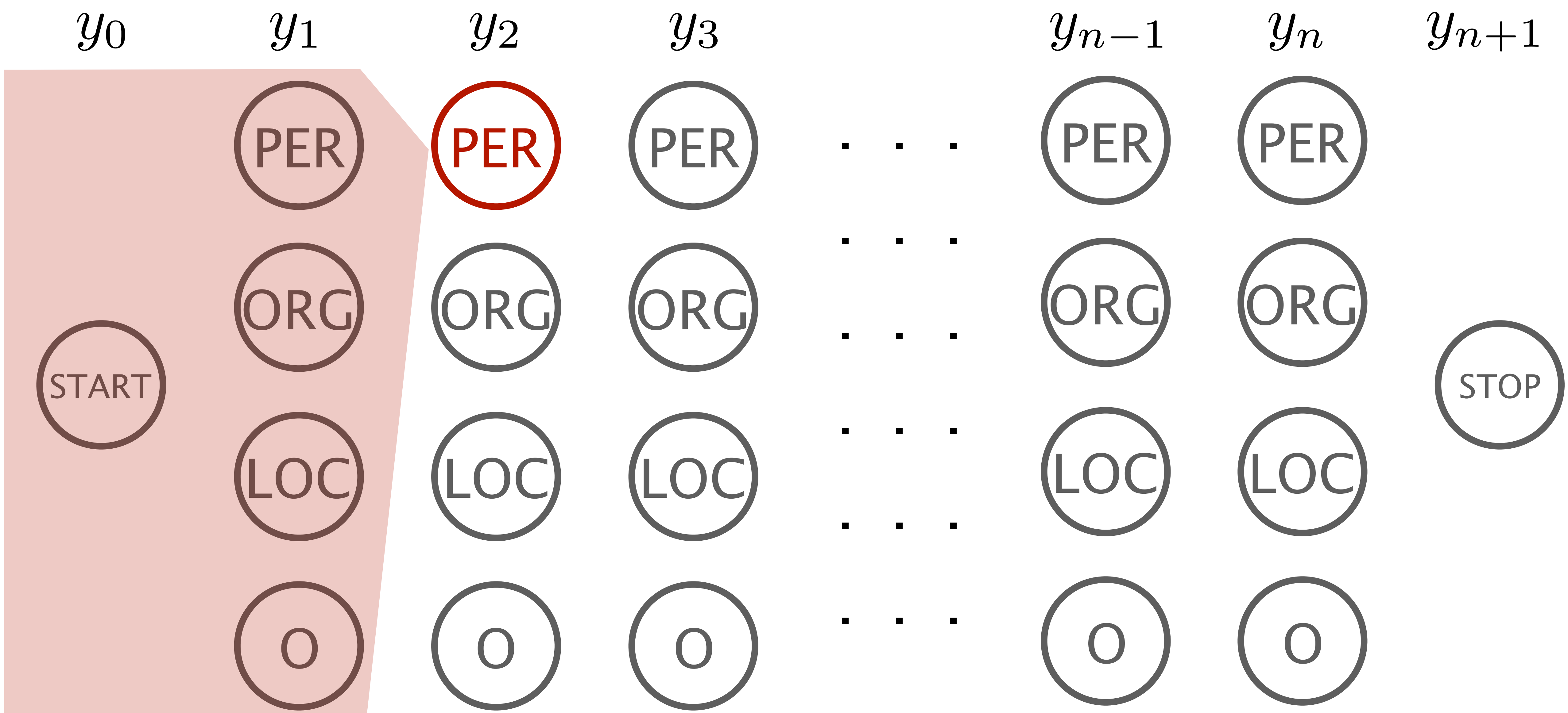
$$\Downarrow$$

$$\frac{\partial L(w)}{\partial w} = \sum_{k=1}^{N} \left(\sum_{t=1}^{n+1} f(x^{(k)}, y_t^{*(k)}, y_{t-1}^{*(k)}) - \sum_{t=1}^{n+1} \sum_{s,s'} P(y_t = s, y_{t-1} = s' | x^{(k)}, w) f(x^{(k)}, y_t = s, y_{t-1} = s') \right)$$

Dynamic programming



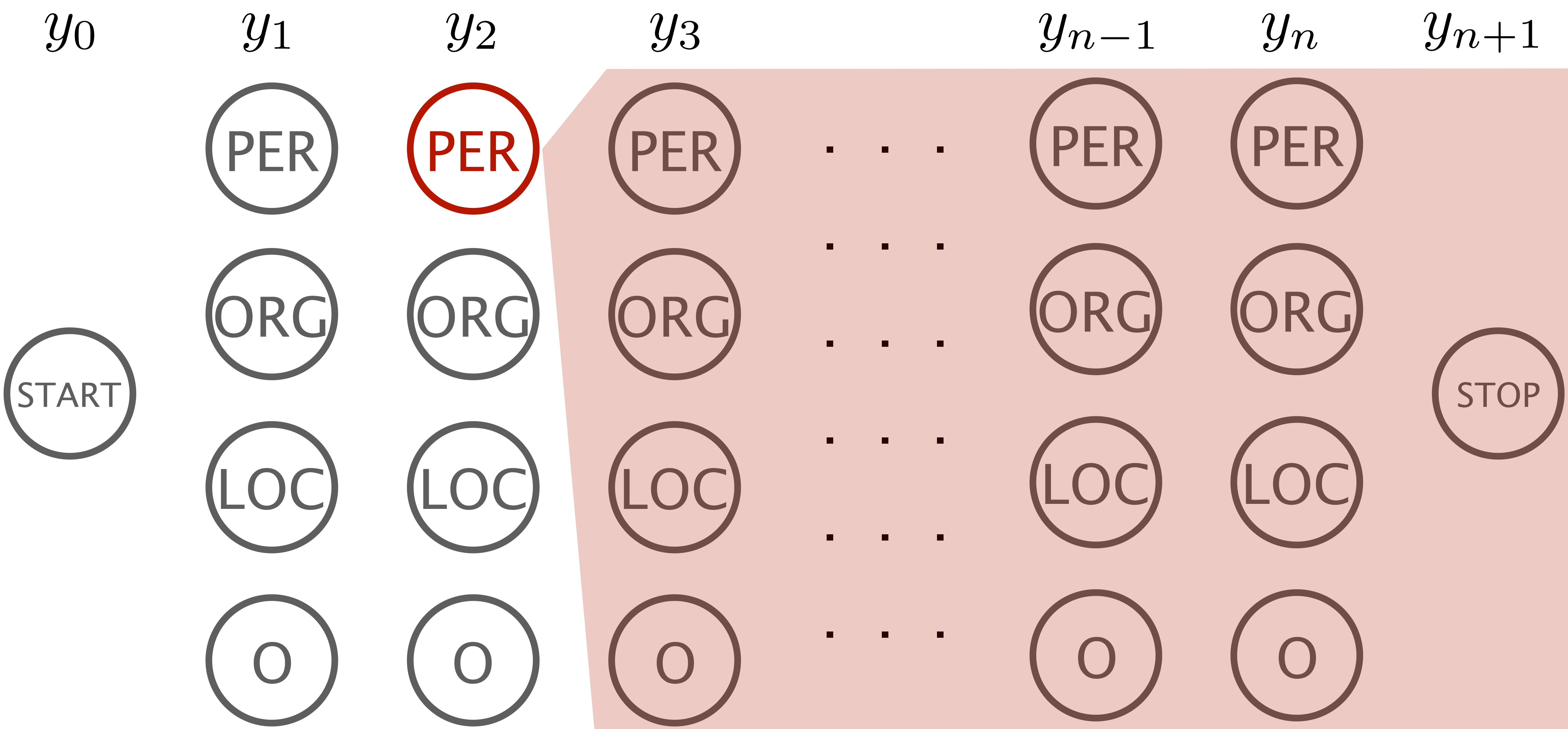
Forward pass



$$\alpha_t(s) = \sum_{s'} \exp(w^\top f(x, y_t = s, y_{t-1} = s'))\alpha_{t-1}(s')$$

$$\alpha_0(s) = \mathbb{1}[s = \text{START}]$$

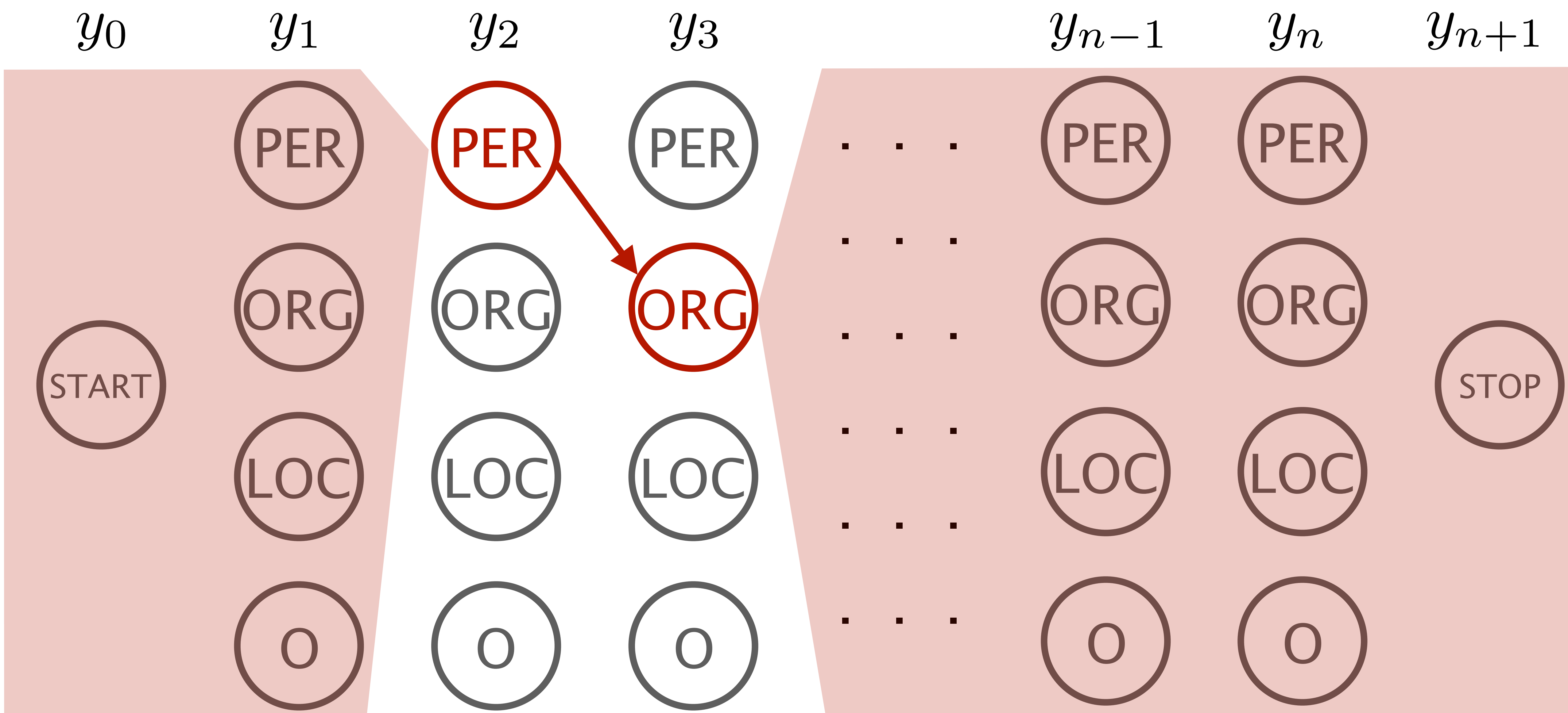
Backward pass



$$\beta_t(s) = \sum_{s'} \exp(w^\top f(x, y_t = s, y_{t+1} = s'))\beta_{t+1}(s')$$

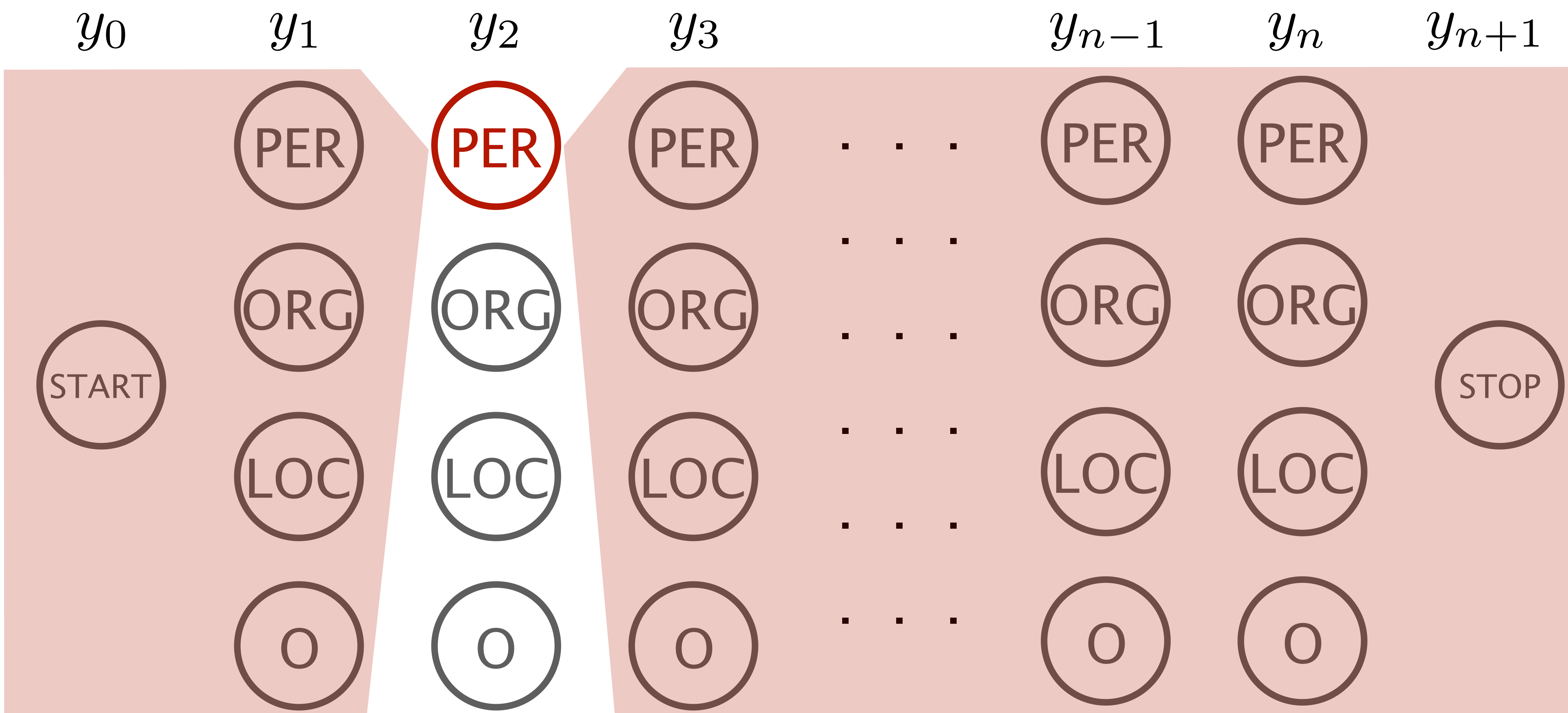
$$\beta_{n+1}(s) = \mathbb{1}[s = \text{STOP}]$$

Computing marginals



$$P(y_t = s, y_{t-1} = s' | x, w) = \frac{\alpha_{t-1}(s') \exp(w^\top f(x, y_t = s, y_{t-1} = s'))\beta_t(s)}{\alpha_{n+1}(\text{STOP})}$$

Computing marginals



$$P(y_t = s|x, w) = \frac{\alpha_t(s)\beta_t(s)}{\alpha_{n+1}(\text{STOP})}$$

Questions?