# TTIC 31230, Fundamentals of Deep Learning

David McAllester, Autumn 2020

# Loopy Belief Propagation (Loopy BP)

## Loopy Belief Propagation (Loopy BP)

We design an algorithm that is correct for tree graphs and use it on non-tree (loopy) graphs.

# Belief Propagation on Trees

Belief Propagation is a message passing procedure (actually dynamic programming).

For each edge $\{n, m\}$ and possible value $y$ for node $n$ we define the message $Z_{m \to n}[y]$ from $m$ to $n$ to be the partition function for the subtree attached to $n$ through $m$ and with $\hat{\mathcal{Y}}[n]$ restricted to $y$.

# Dynamic Programming Computes the Messages

$$Z_{m \to n}[y] = \sum_{y'} e^{s^N[m,y'] + s^E[\langle m, n \rangle, y', y]} \left( \prod_{k \in N(m),\ k \neq n} Z_{k \to m}[y'] \right)$$

## Loopy BP

In a Loopy Graph we can initializing all message $Z_{n \to m}[y] = 1$ and then repeating (until convergence) the updates

$$\tilde{Z}_{m \to n}[y] \;=\; \frac{1}{Z_{m \to n}}\, Z_{m \to n}[y] \qquad Z_{m \to n} = \sum_y Z_{m \to n}[y]$$

$$Z_{m \to n}[y] \;=\; \sum_{y'} e^{s^N[m,y'] + s^E[m,n,y',y]} \left( \prod_{k \in N(m),\; k \neq n} \tilde{Z}_{k \to m}[y'] \right)$$

# Computing Node Marginals from Messages

$$
\begin{aligned}
Z^N(y) &= \sum_{\hat{\mathcal{Y}}:\, \hat{\mathcal{Y}}[n]=y} e^{s(\hat{\mathcal{Y}})} \\
&= e^{s^N[y]} \left( \prod_{m \in N(n)} Z_{m \to n}[y] \right) \\
{\color{red} P^N(y)} &= Z^N(y)/Z, \qquad Z = \sum_y Z^N(y)
\end{aligned}
$$

## Computing Edge Marginals from Messages

$$
\begin{aligned}
Z_{n,m}(y, y') &\doteq \sum_{\hat{\mathcal{Y}}:\ \hat{\mathcal{Y}}[n]=c,\ \hat{\mathcal{Y}}[m]=y'} e^{s(\hat{\mathcal{Y}})} \\
&= e^{s^N[n,y]+s^N[m,y']+s^E[n,m,y,y']} \\
&\quad \prod_{m\in N(n),\ k\neq m} Z_{m\to n}[y] \\
&\quad \prod_{m\in N(m),\ k\neq n} Z_{m\to n}[y'] \\
P_{n,m}(y, y') &= Z_{n,m}(y, y')/Z \quad Z = \sum_{y,y'} Z_{n,m}(y, y')
\end{aligned}
$$

**END**