

# *Word Window Classification - II*

Pawan Goyal

CSE, IIT Kharagpur

February 9th, 2017

# Training with backpropagation

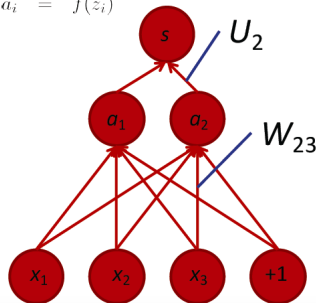
*Derivative of weight  $W_{ij}$ :*

# Training with backpropagation

## Derivative of weight $W_{ij}$ :

$$\begin{aligned}\frac{\partial}{\partial W_{ij}} U^T a &\rightarrow \frac{\partial}{\partial W_{ij}} U_i a_i \\ U_i \frac{\partial}{\partial W_{ij}} a_i &= U_i \frac{\partial a_i}{\partial z_i} \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i \frac{\partial f(z_i)}{\partial z_i} \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i f'(z_i) \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i f'(z_i) \frac{\partial W_i \cdot x + b_i}{\partial W_{ij}}\end{aligned}$$

$$\begin{aligned}z_i &= W_i \cdot x + b_i = \sum_{j=1}^3 W_{ij} x_j + b_i \\ a_i &= f(z_i)\end{aligned}$$



## Derivative continued ...

$$\begin{aligned} U_i \frac{\partial}{\partial W_{ij}} a_i &= U_i f'(z_i) \frac{\partial W_{i \cdot} x + b_i}{\partial W_{ij}} \\ &= U_i f'(z_i) \frac{\partial}{\partial W_{ij}} \sum_k W_{ik} x_k \\ &= \underbrace{U_i f'(z_i)}_{\delta_i} x_j \\ &= \underbrace{\delta_i}_{\text{Local error signal}} \underbrace{x_j}_{\text{Local input signal}} \end{aligned}$$

where  $f'(z) = f(z)(1 - f(z))$  for logistic  $f$

## *From single weight $W_{ij}$ to full $W$ :*

$$\frac{\partial s}{\partial W_{ij}} = \delta_i x_j$$

- We want all combinations of  $i = 1, 2, \dots$  and  $j = 1, 2, 3, \dots$

## *From single weight $W_{ij}$ to full $W$ :*

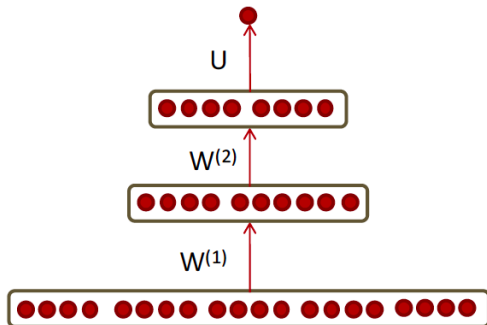
$$\frac{\partial s}{\partial W_{ij}} = \delta_i x_j$$

- We want all combinations of  $i = 1, 2, \dots$  and  $j = 1, 2, 3, \dots$
- Solution: Outer product

$$\frac{\partial s}{\partial W} = \delta x^T$$

# Two layer neural nets

$$\begin{aligned}x &= z^{(1)} = a^{(1)} \\ z^{(2)} &= W^{(1)}x + b^{(1)} \\ a^{(2)} &= f\left(z^{(2)}\right) \\ z^{(3)} &= W^{(2)}a^{(2)} + b^{(2)} \\ a^{(3)} &= f\left(z^{(3)}\right) \\ s &= U^T a^{(3)}\end{aligned}$$



# Weight Updates – Full Derivation

On Board