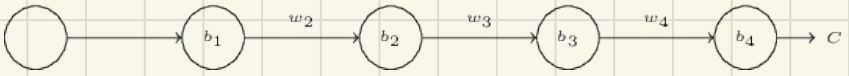
Deep learing

$$\frac{\partial C}{\partial b_1} = \sigma'(z_1) \times w_2 \times \sigma'(z_2) \times w_3 \times \sigma'(z_3) \times w_4 \times \sigma'(z_4) \times \frac{\partial C}{\partial a_4}$$

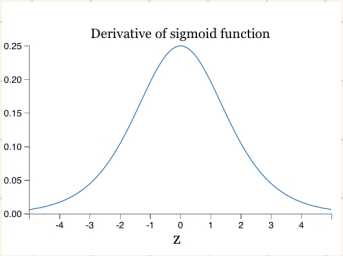


$$\frac{\partial C}{\partial b_1} \approx \frac{\Delta C}{\Delta b_1}$$

$$\Delta a_1 \approx \frac{\partial a_1}{\partial z} \frac{\partial z}{\partial b} \Delta b_1$$
$$= \sigma'(z_1)\, \Delta b_1$$

$$\Delta a_2 \approx \frac{\partial a_2}{\partial z_2} \frac{\partial z_2}{\partial a_1} \Delta a_1$$
$$= \sigma'(z_2) \cdot w_2\, \sigma'(z_1)\, \Delta b_1$$

$$\frac{\Delta C}{\Delta b_1} = \sigma'(z_4)\, w_4\, \sigma'(z_3)\, w_3\, \sigma'(z_2)\, w_2\, \sigma'(z_1)$$
$$< \frac{1}{4} \quad < \frac{1}{4} \quad < \frac{1}{4}$$



$\max \sigma'(z) = \frac{1}{4}$

$w \sim N(0, 1)$

avoid 1) $w_j = 100 \quad z_j = 0$

$b_j = -w_j x$

P

1) $|w\sigma'(wa+b)| \geq 1$

$|\sigma'| \leq \frac{1}{4} \quad \therefore |w| \geq 4$

2) $|w| \frac{e^{wa+b}}{(e^{wa+b}+1)^2} \geq 1$

$$\frac{e^{wa+b}}{(\quad)^2} \geq \frac{1}{|w|}$$

$$(e^{wa+b})^2 + (2-|w|)\, e^{wa+b} + 1 \leq 0$$

$$\frac{-}{2} \leq e^{wa+b} \leq \frac{|w|-2+\sqrt{|w|(|w|-4)}}{2}$$

$$- \qquad a \leq \frac{1}{|w|} \ln\left(\frac{|w|+\sqrt{w^2-4|w|}}{2} - 1\right) + \frac{b}{|w|}$$

so band $\leq \frac{2}{|w|} \ln\left(\frac{|w|+\sqrt{w^2-4|w|}}{2} - 1\right)$

3) greatest value

band' $= -\frac{2}{|w|^2} \ln(\quad) + \frac{2}{|w|} \frac{1}{(\quad)} \frac{1}{2}\left(1 + \frac{1}{2} \frac{1}{\sqrt{\quad}}\right) = 0$

$$-2(\quad)\ln(\quad) \cdot \sqrt{\quad} + |w|\left(\sqrt{\quad} + \tfrac{1}{2}\right) = 0$$

$$\sqrt{\quad}\,|w| - 2(\quad)\ln(\quad) + \tfrac{1}{2}|w| = 0$$

Deep Learning

Supervised Learning

Input(x)	Output (y)	Application	
Home features	Price	Real Estate	Standard NN
Ad, user info	Click on ad? (0/1)	Online Advertising	Standard NN
Image	Object (1,...,1000)	Photo tagging	CNN
Audio	Text transcript	Speech recognition	RNN
English	Chinese	Machine translation	RNN
Image, Radar info	Position of other cars	Autonomous driving	Custom/Hyber

Structured Data

Size	#bedrooms	...	Price (1000$s)
2104	3		400
1600	3		330
2400	3		369
⋮	⋮		⋮
3000	4		540

User Age	Ad Id	...	Click
41	93242		1
80	93287		0
18	87312		1
⋮	⋮		⋮
27	71244		1

Unstructured Data

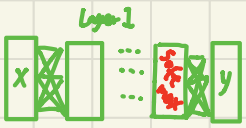
Audio Image

Four scores and seven years ago...

Text

Good Results on Training Data

(0.) Softmax : $a_j^L = \frac{e^{z_j^L}}{\sum e^{z_j^L}}$



1. choosing proper loss : cross entropy

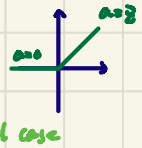
$-y_i \ln y_i$

✗ square error $(\hat{y}_i - y_i)^2$

2. Mini batch

3. New activation func

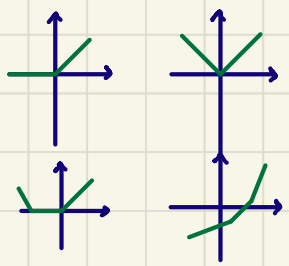
1) ReLU



- fast
- infinite Sigmoid with biases
- vanishing gradient problem

special case

2) Maxout



4. learning Rate

1) small → y ← large

fast → result ← converge

2) Adagrad $g_w(y, \frac{\partial L}{\partial w})$

$w \leftarrow w - g_w \frac{\partial L}{\partial w}$

$g_w = \frac{y}{\sqrt{\sum_{past} (\frac{\partial L}{\partial w})^2}}$

- g_w 越來越小 always
- if $\frac{\partial L}{\partial w}$ small, g_w big

5. Momentum

1) Movement = $-\frac{\partial L}{\partial w}$ + Momentum

2) Adam RMSProp (Advanced Adagrad) + Momentum

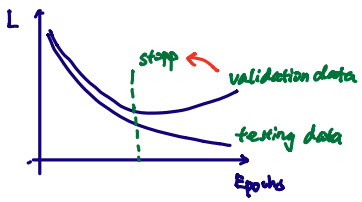
Good Results on Testing Data

(Over-fitting)

more training ← get / create

1. Early Stopping

testing data → validation data

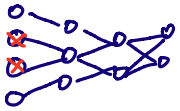


2. Weight Decay (Regularization)

$J = \sum (\hat{y} - y)^2 + W_d \cdot \sum w^2$

3. Dropout

Train :



- p% drop out
- after ... restore

Test:

- No drop out
- $w \leftarrow w \cdot (1-p)\%$

4. NN Structure

CNN Convolutional : convolution, max pooling } × n → Flatten → Fully connected Feedforward Network

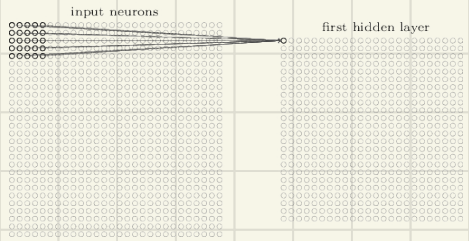
RNN Recurrent

Convolutional Neural Network

local receptive fields

shared weights

pooling



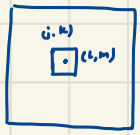
input → hidden 1st

5×5 → 1×1

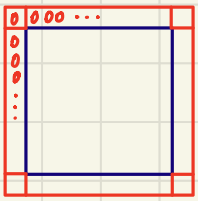
28×28 → 24×24

$\sigma\left(b + \sum_{l=0}^{4}\sum_{m=0}^{4} w_{l,m}\, a_{j+l,\,k+m}\right)$

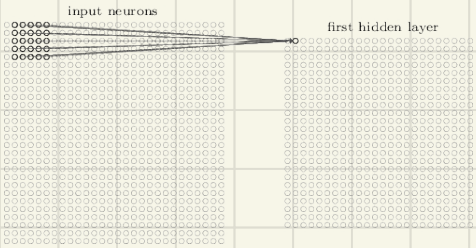
shared bias — shared weights ($n = 25$)

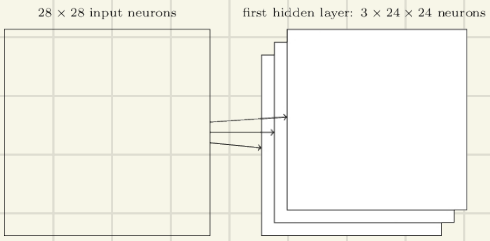


→ Zero Padding



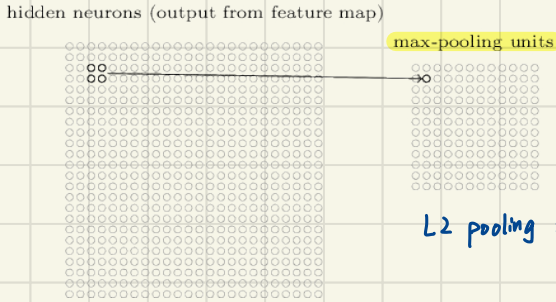
① Convolutional layer





3 Filter

② pooling layer



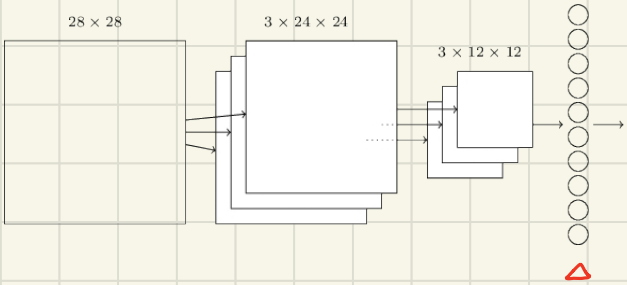
= max (ai)

condense

L2 pooling = $\sqrt{\sum a_i}$

full connected

③ Flatten



① Initial $w \sim N(0,1)$
$b \sim N(0,1)$

② foreward

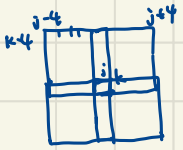
1) L_{1i} : $a_{jk} = \sigma\left(\sum_{\cdot}^{i}\sum_{\cdot}^{m} w_{i,m} \underbrace{a_{j-i,k+m}}_{=z_{jk}} + b\right)$

2) L_{2i} : $a_{jk} = \max\left(a_{2j-1,2k-1}, a_{2j-1,2k}, a_{2j,2k-1}, a_{2j,2k}\right)$

3) L_{3i} : $a_j = \sum^{\text{pool}} w\, a_{k,h}$

③ Back propagation

$L_0 \to L_1$: $\Delta a^0 + a^0_{jk}$:



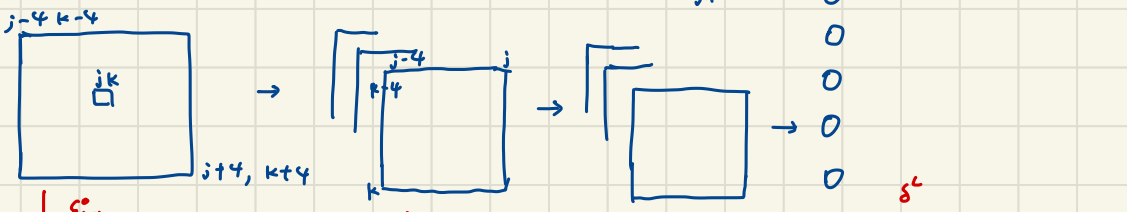
$a'_{j-4,k-4} = \sigma\left[b + \sum^{(x;K)}\sum \Delta a + w_{4,4}(a^0 + \Delta a^0)\right] \to \frac{\partial a'_{j-4,k-4}}{\partial a^0_{jk}} = \sigma' \, w_{4,4}$

$a'_{j-3,k-4}$
$a'_{j-2,k-4}$
$a'_{j-1,k-4}$
$\vdots$
$a'_{j,k}$

$\downarrow$

$\frac{\partial a'_{j+l,k+m}}{\partial a^0_{jk}} = \sigma' \cdot w^{1,i}_{l,m}$ layer 1 convolution i

$l \in [-4,0]$
$m \in [-4,0]$



$\frac{\partial C}{\partial z^0_{j,k}} = \sum_{i}^{3}\sum_{l,m}^{4,4} \frac{\partial C}{\partial z^{1,i}_{j+l,k+m}} \cdot \frac{\partial z^{1,i}_{j+l,k+m}}{\partial z^0_{j,k}}$

$= \sum_{i}^{3}\sum_{l,m}^{4,4} \delta^{1,i}_{j+l,k+m} \cdot w^{1,i}_{l,m}$

$= \sum_{i}^{3} \underset{1\times 4}{[\,\delta^1\,]} \underset{4\times 1}{\left[w^{1,i}_{l,m}\right]}$

$\delta^{1,i}_{j,k}$

$\frac{\partial C}{\partial z^{1,i}_{j,k}} = \frac{\partial C}{\partial z^{2,i}_{l,m}} \cdot \frac{\partial z^{2,i}}{\partial z^{1,i}_{j,k}}$

$= \delta^2 \cdot 1$: if (forward)
$\cdot 0$

$\frac{\partial C}{\partial w^{1,i}_{l,m}} = \sum_{j,k}^{all} \frac{\partial C}{\partial z^{1,i}_{j,k}} \cdot \frac{\partial z^{1,i}_{j,k}}{\partial w^{1,i}_{l,m}}$

$= \sum_{j,k}^{4,4} \delta^{1,i}_{j,k}\, a^0_{j-l,k-m}$

$\delta^{2,i}_{j,k}$

$\frac{\partial C}{\partial z^{2,i}_{j,k}} = \sum_{m}^{10} \frac{\partial C}{\partial z^L_m} \cdot \frac{\partial z^L_m}{\partial z^{2,i}_{j,k}}$

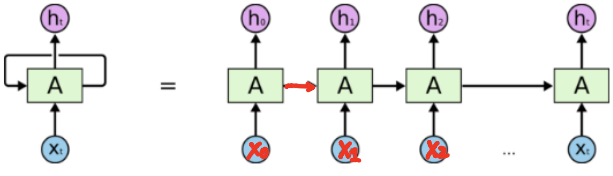
$= \underset{1\times 10}{[\,\delta^L\,]} \cdot \underset{10\times 1}{\left[w^{2,i}_{j,k,m}\right]}$

δ^L

$\frac{\partial C}{\partial z^L_j} = \frac{\partial C}{\partial a}\frac{\partial a}{\partial z^L_j}$

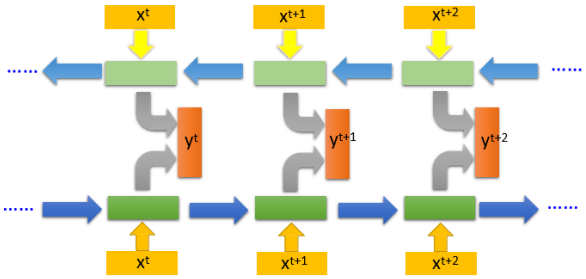
$= \nabla_a C \cdot \sigma'(z^L_j)$

RNN



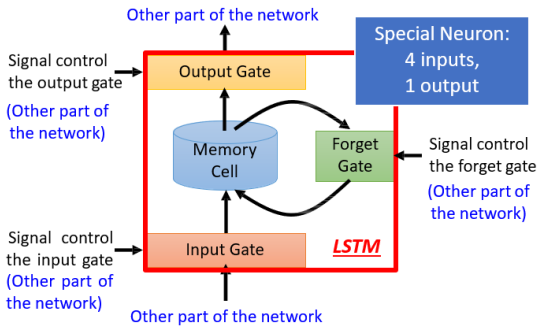
An unrolled recurrent neural network.

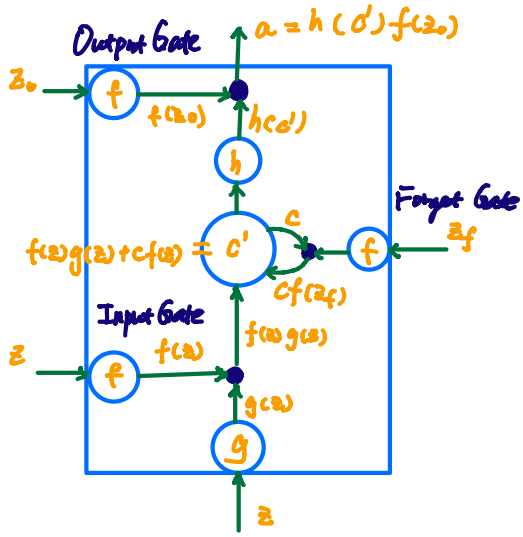
1. Bidirectional RNN

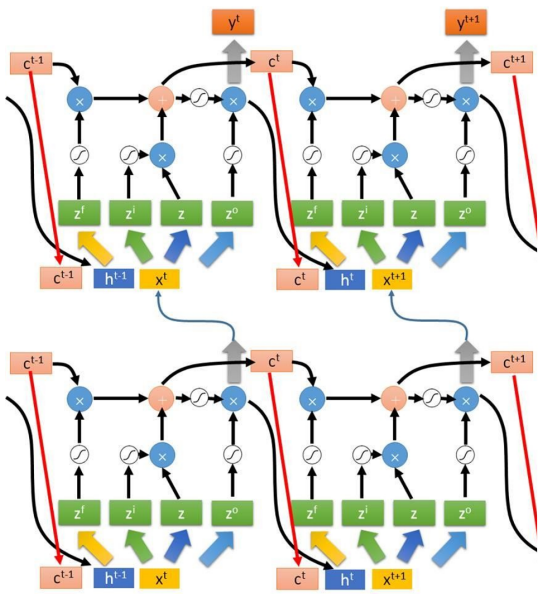


2. Long Short-Term Memory

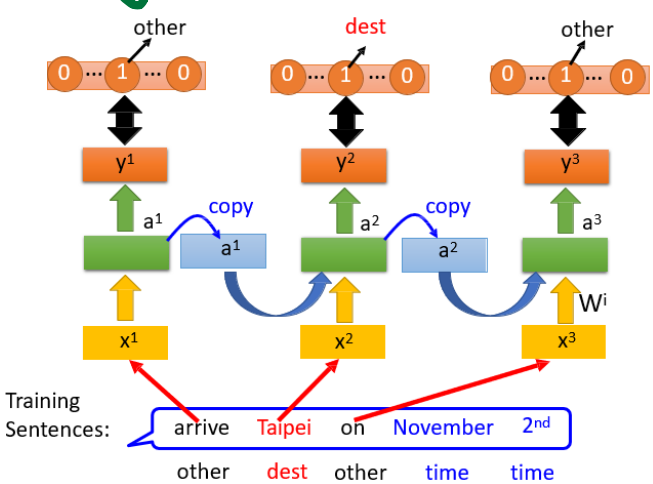
LSTM







Training



Learning