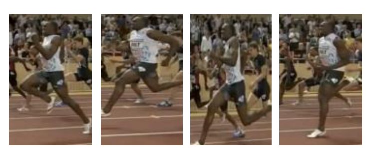
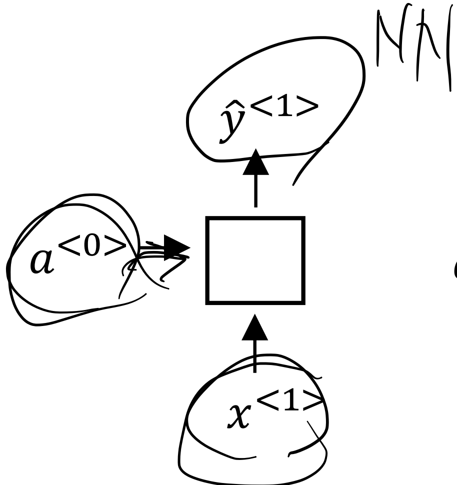
Recurrent Neural Networks (RNN)

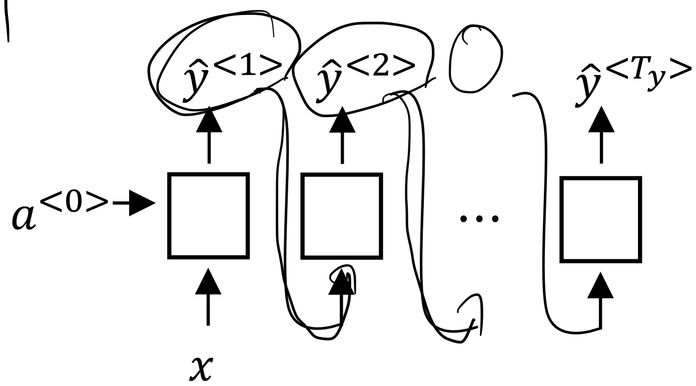
Model Selection and Regularization

Speech recognition		→	“The quick brown fox jumped over the lazy dog.”
Music generation	∅	→	
Sentiment classification	“There is nothing to like in this movie.”	→	★☆☆☆☆
DNA sequence analysis	AGCCCCTGTGAGGAACTAG	→	AGCCCCTGTGAGGAACTAG
Machine translation	Voulez-vous chanter avec moi?	→	Do you want to sing with me?
Video activity recognition		→	Running
Name entity recognition	Yesterday, Harry Potter met Hermione Granger.	→	Yesterday, Harry Potter met Hermione Granger.

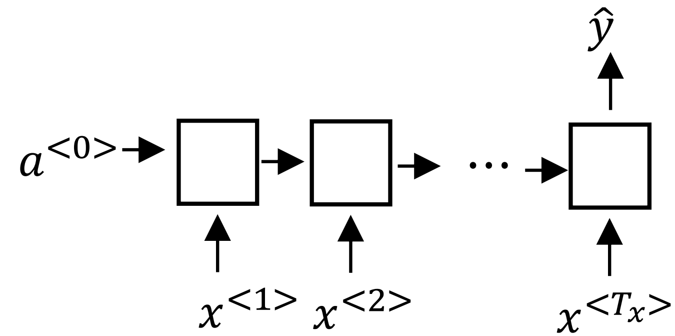
RNN types



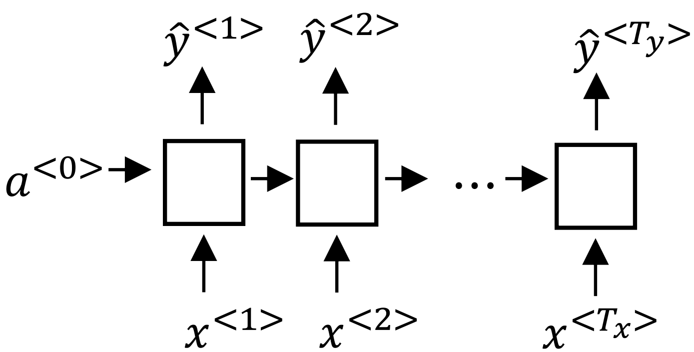
One to one



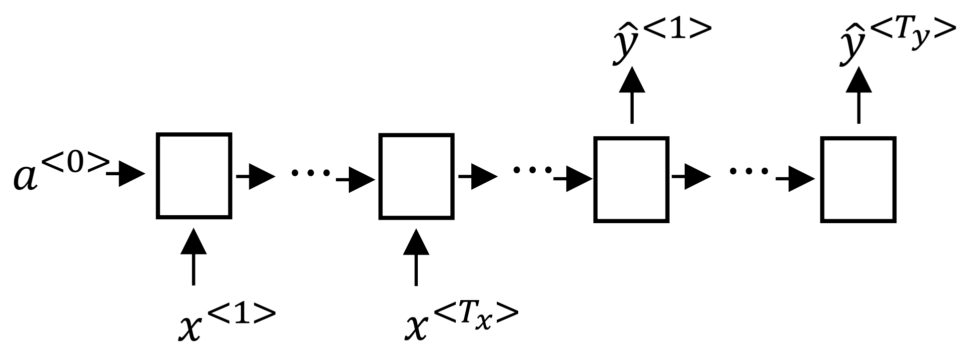
One to many



Many to one



Many to many



Many to many

Motivation

x: Harry Potter and Hermione Granger invented a new spell.

$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ … $x^{<9>}$

y: 1 1 0 1 1 0 0 0 0

a
b
c
d

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}$$

A = $\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix}$

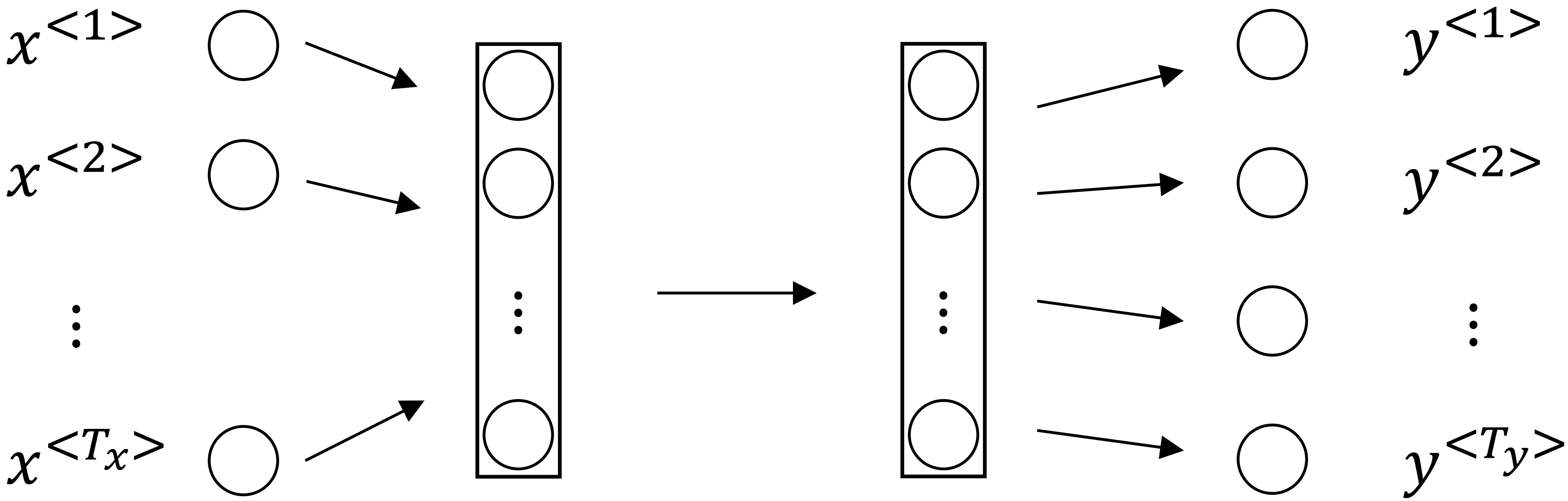
B

$\begin{bmatrix} \text{Harry} \\ \text{Potter} \\ \vdots \end{bmatrix}$

And = 367
Invented = 4700
A = 1
New = 5976
Spell = 8376
Harry = 4075
Potter = 6830
Hermione = 4200
Gran… = 4000

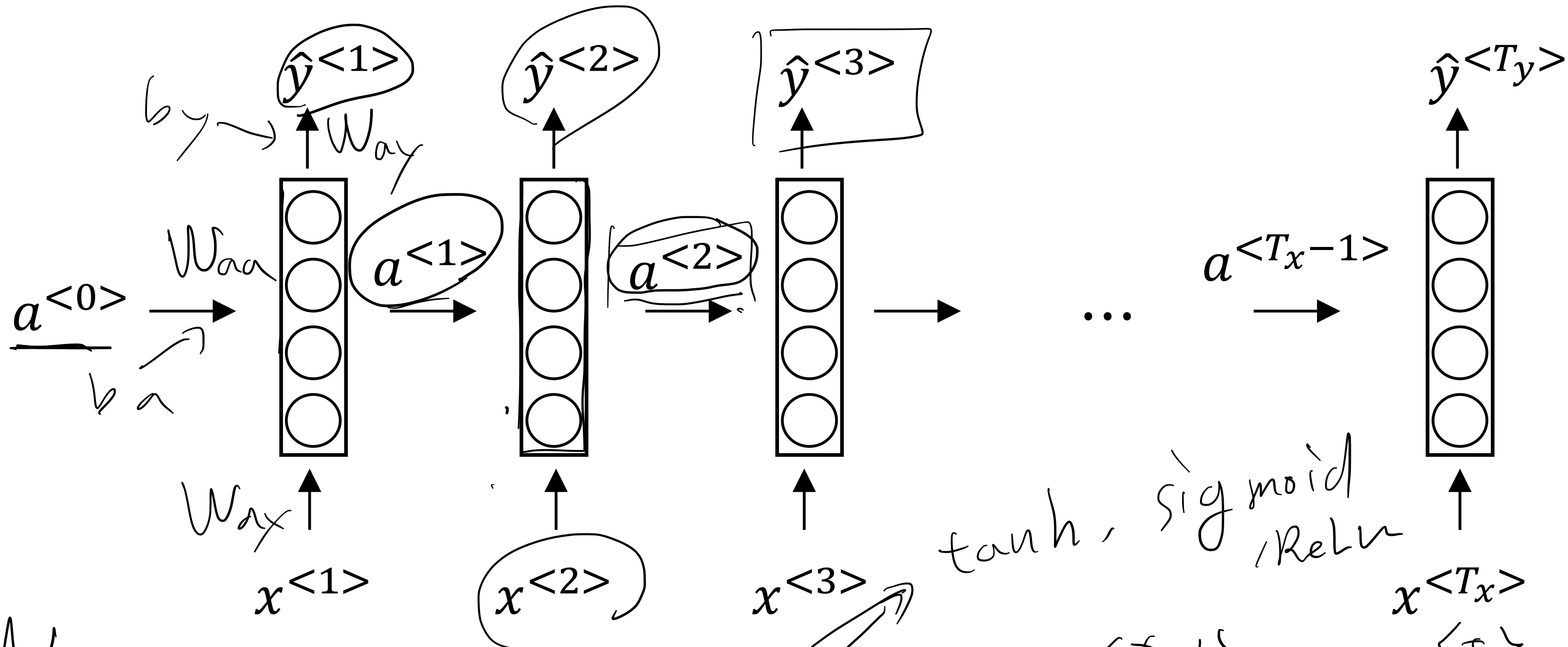
Word2Vec

Why not a standard neural net?



- Problems
  - Inputs, outputs can be different lengths in different examples
  - Inputs, outputs can be different lengths in different examples

RNN- Forward propagation



$$W = \left[W_{aa}, W_{ax}, W_{ay}, b_a, b_y \right]$$

$$a^{<t>} = g(W_{aa} a^{<t-1>} + W_{ax} x^{<t>} + b_a)$$

$$\hat{y}^{<t>} = g(W_{ay} a^{<t>} + b_y)$$

RNN- Forward propagation

$$a^{<t>} = g(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a)$$

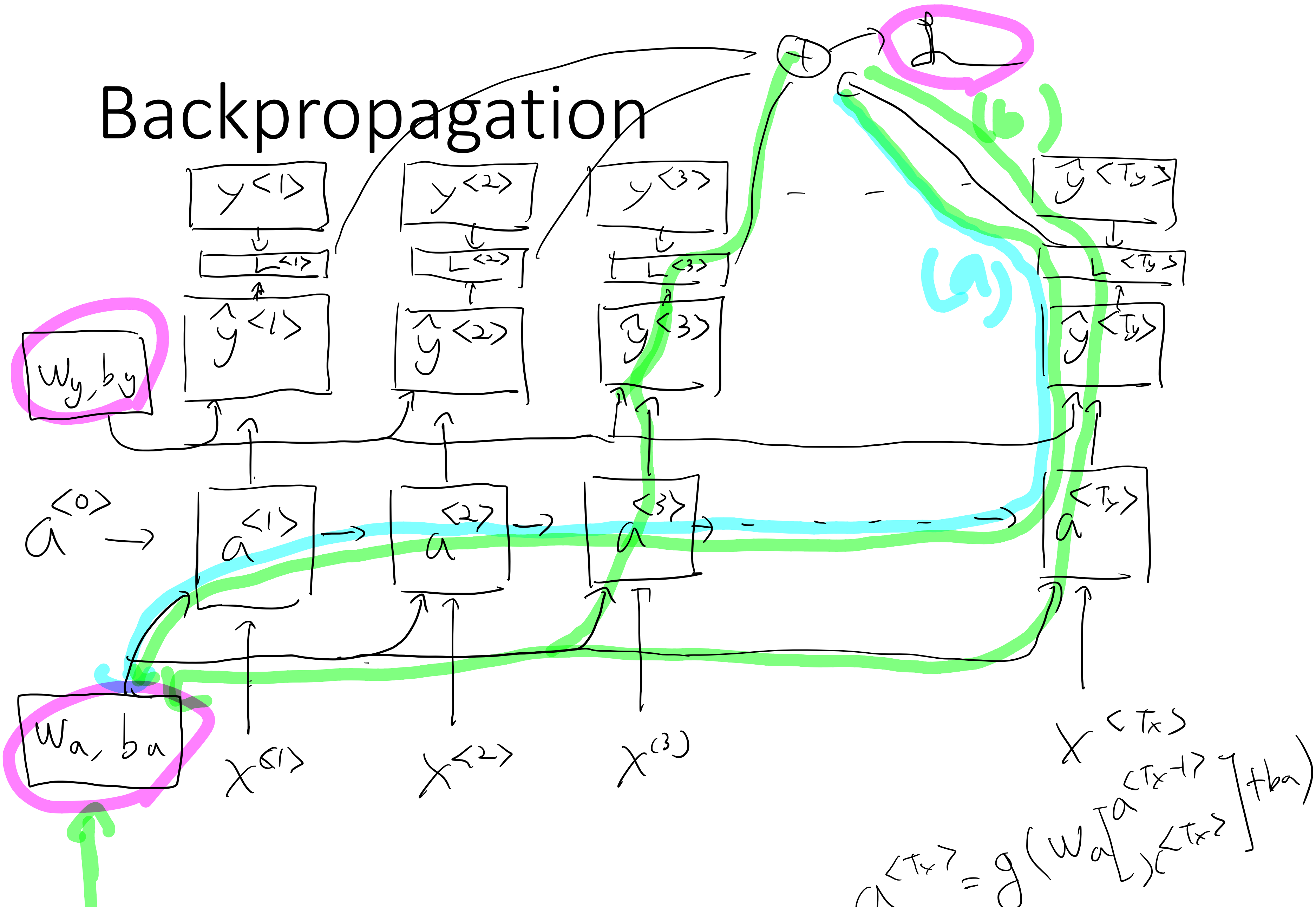
$$\hat{y}^{<t>} = g(W_{ya}a^{<t>} + b_y)$$

$$\frac{a^{<t>}}{h^{<t>}} = g\left(W_a \cdot \begin{bmatrix} a^{<t-1>} \\ x^{<t>} \end{bmatrix} + b_a\right)$$

$$\hat{y}^{<t>} = g(W_y a^{<t>} + b_y)$$

$$W_a = [W_{aa} ; W_{ax}]$$

Backpropagation



$$a^{<T_x>} = g\left(W_a \begin{bmatrix} a^{<T_x-1>} \\ x^{<T_x>} \end{bmatrix} + b_a\right)$$

Backpropagation

$$dW_a = \sum_{i=1}^{T_y} \nabla_{W_a} L^{<i>} = \sum_{i=1}^{T_y} \frac{\partial L^{<i>}}{\partial \hat{y}^{<i>}} \frac{\partial \hat{y}^{<i>}}{\partial W_a}$$

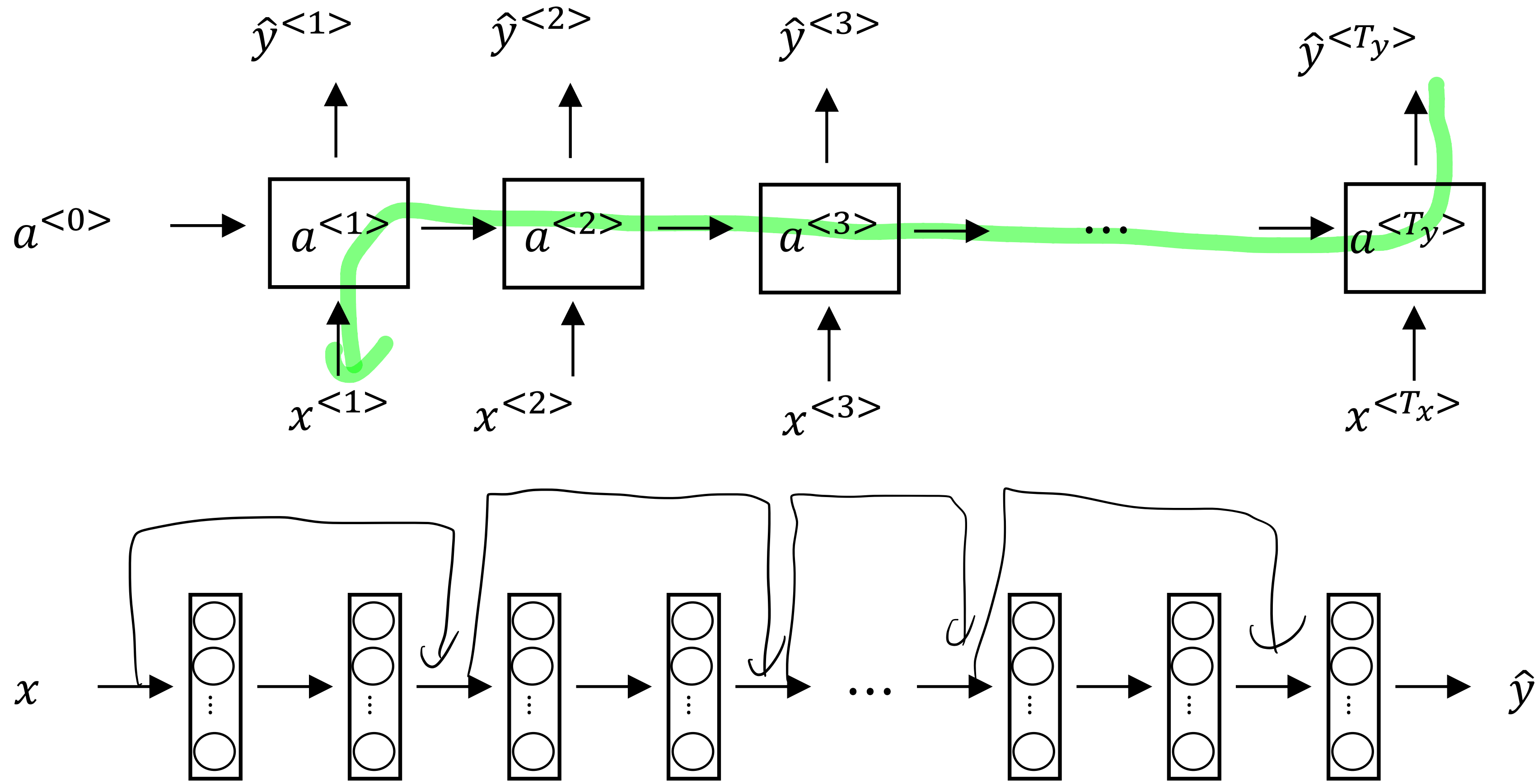
$$dW_a = \nabla_{W_a} \mathcal{L}$$

$$\nabla_{W_a} L^{<i>} = \frac{\partial L^{<i>}}{\partial W_a}$$

$$= \sum_{i=1}^{T_y} \frac{\partial L^{<i>}}{\partial \hat{y}^{<i>}} \sum_{j=1}^{i} \frac{\partial \hat{y}^{<i>}}{\partial a^{<j>}} \frac{\partial a^{<j>}}{\partial W_a}$$

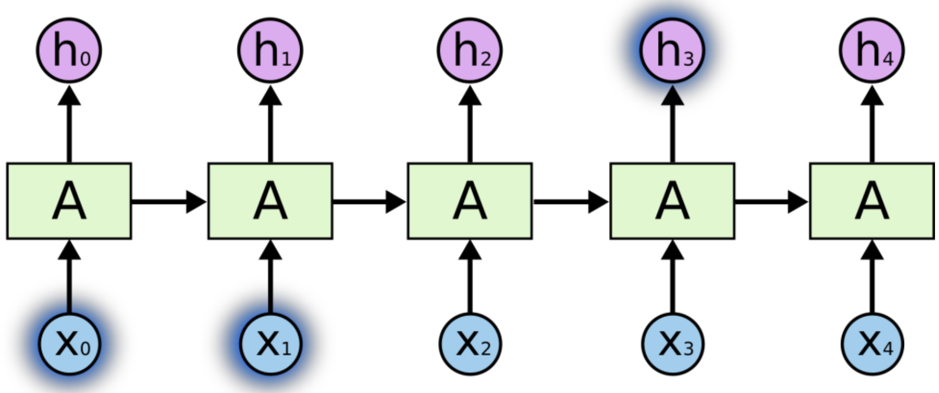
$$= \sum_{i=1}^{T_y} \frac{\partial L^{<i>}}{\partial \hat{y}^{<i>}} \sum_{j=1}^{i} \boxed{\frac{\partial \hat{y}^{<i>}}{\partial a^{<i>}} \cdot \frac{\partial a^{<i>}}{\partial a^{<i-1>}} \cdots \frac{\partial a^{<j+1>}}{\partial a^{<j>}}} \frac{\partial a^{<j>}}{\partial W_a}$$

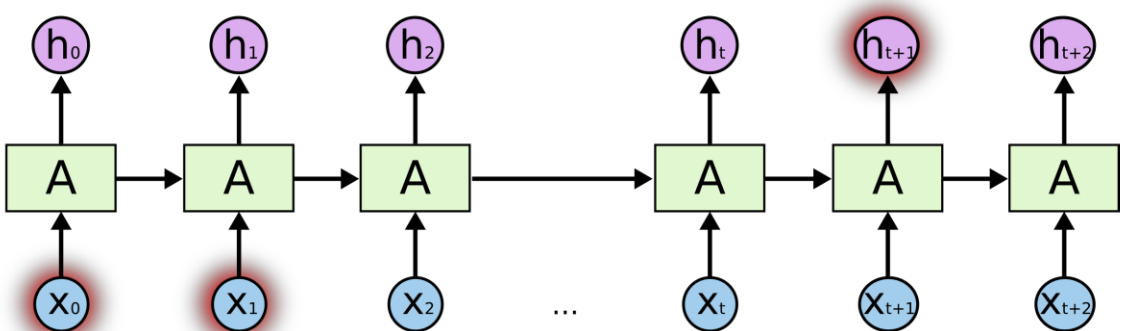
Vanishing Gradient



Long-term dependency

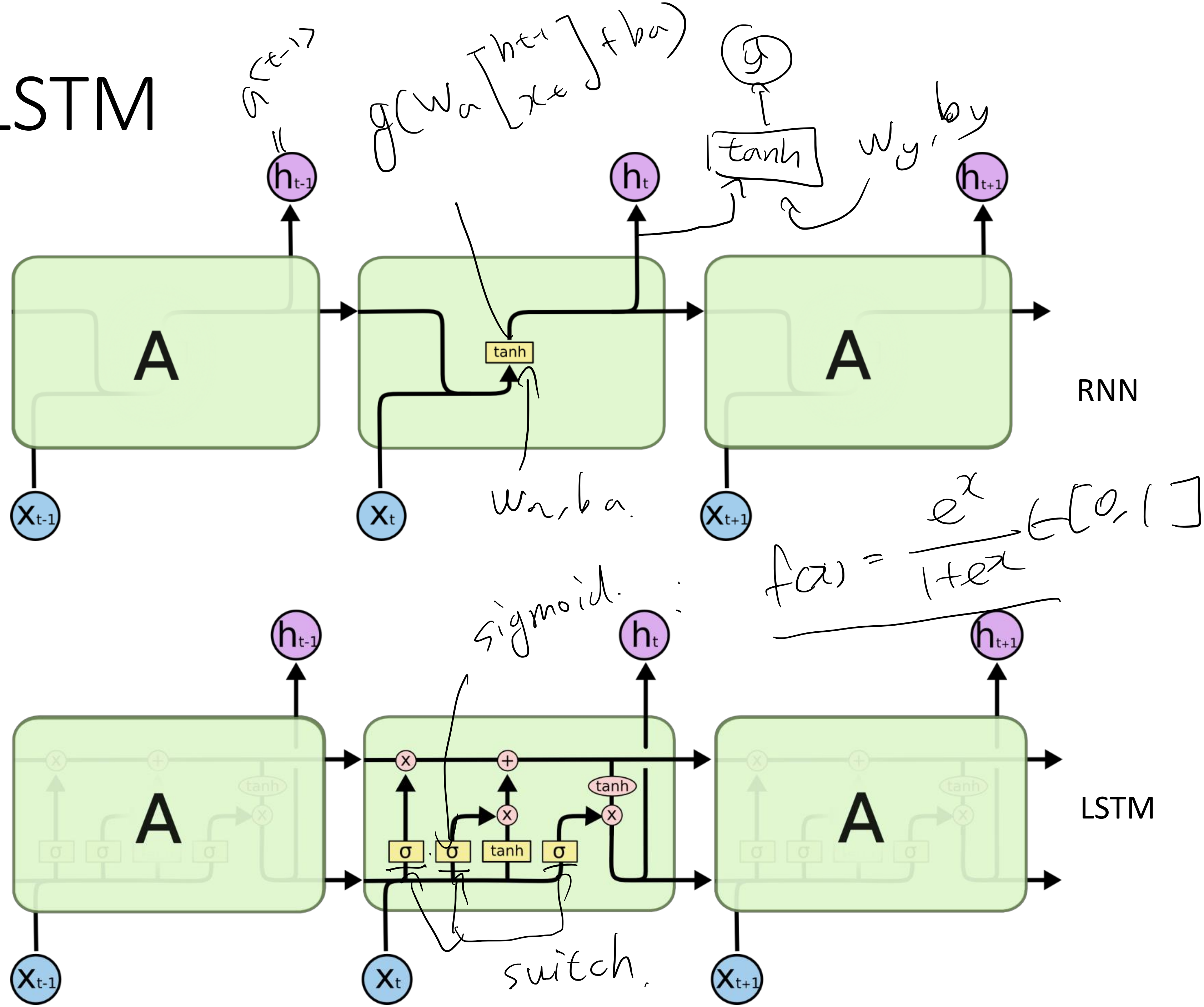
LSTM!
GRU!





- RNN cannot learn the long-term dependency in the bellow

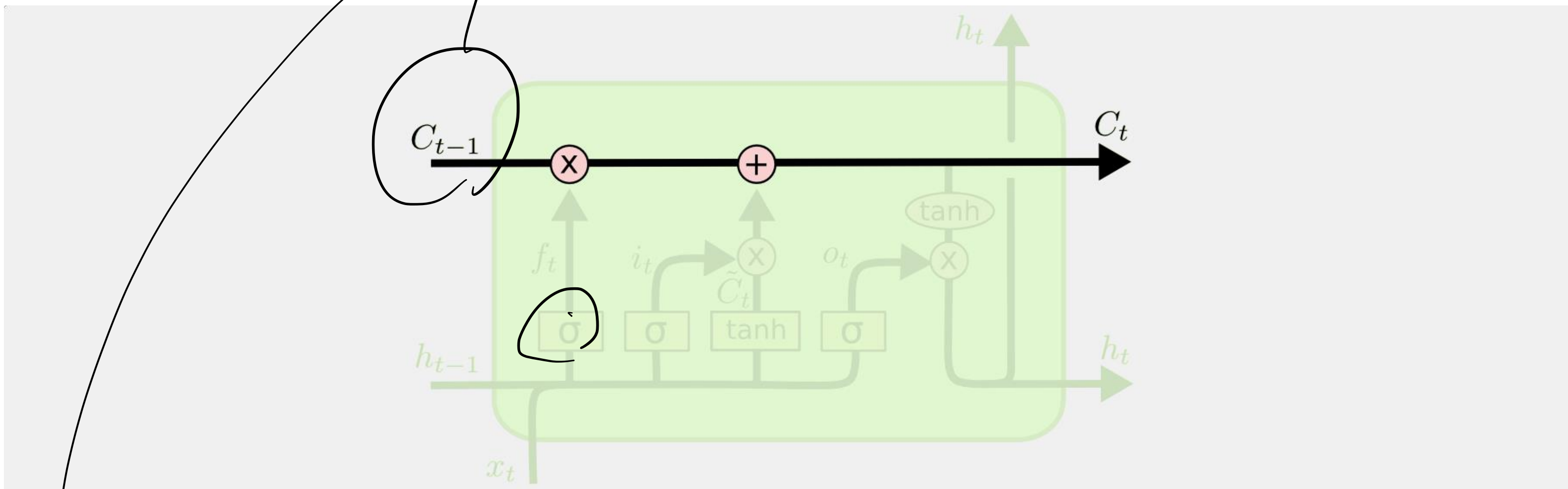
LSTM



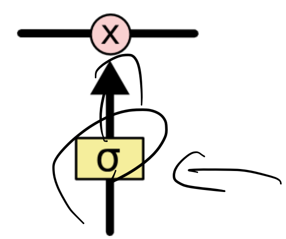
$a^{<t-1>}$
$g(W_a \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_a)$
$\hat{y}$
tanh
W_y, b_y
h_{t-1}
h_t
h_{t+1}
A
tanh
A
RNN
W_a, b_a
X_{t-1}
X_t
X_{t+1}
$f(x) = \frac{e^x}{1+e^x} \in [0,1]$
sigmoid.
h_{t-1}
h_t
h_{t+1}
A
σ
σ
tanh
σ
tanh
A
LSTM
switch.
X_{t-1}
X_t
X_{t+1}

LSTM

cell state : long term memory

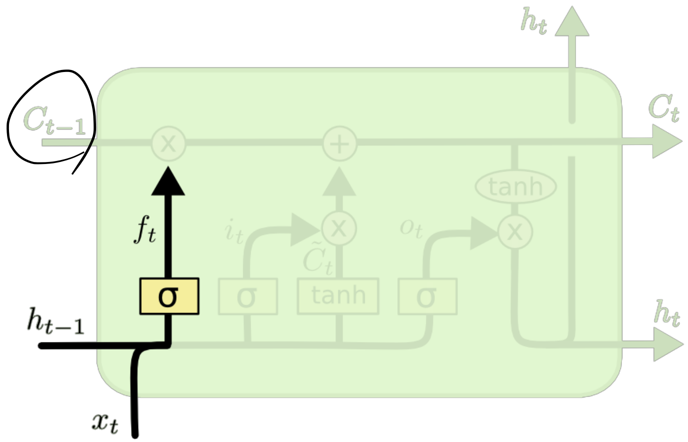


a conveyor belt of information

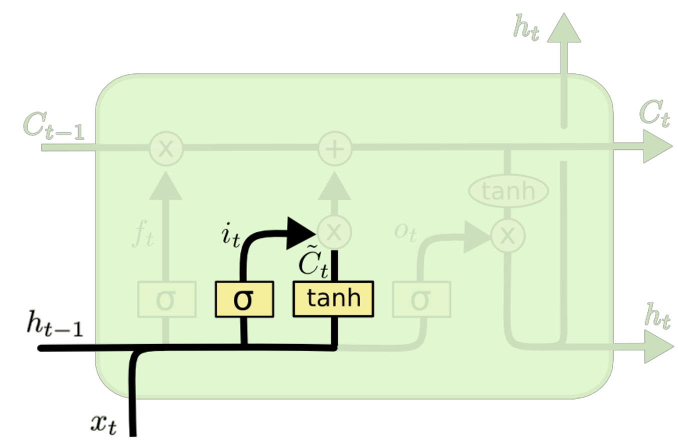


$$\sigma(x) = \frac{e^x}{1+e^x}$$

LSTM

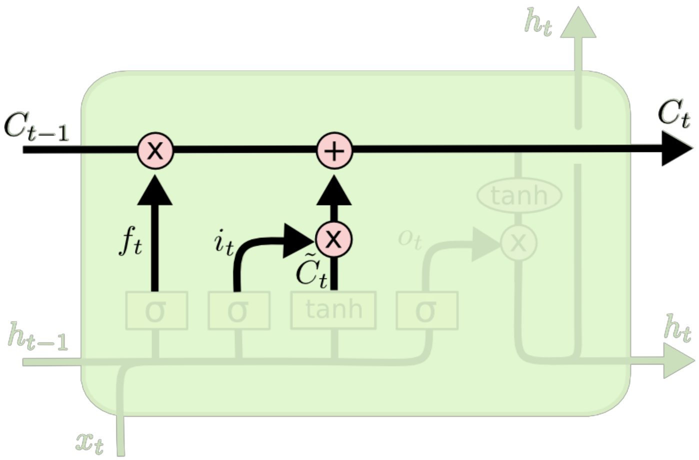


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$



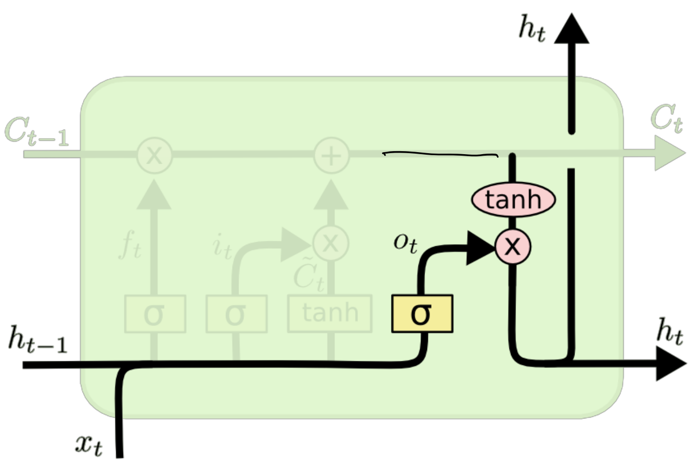
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

LSTM



(1-f_t)

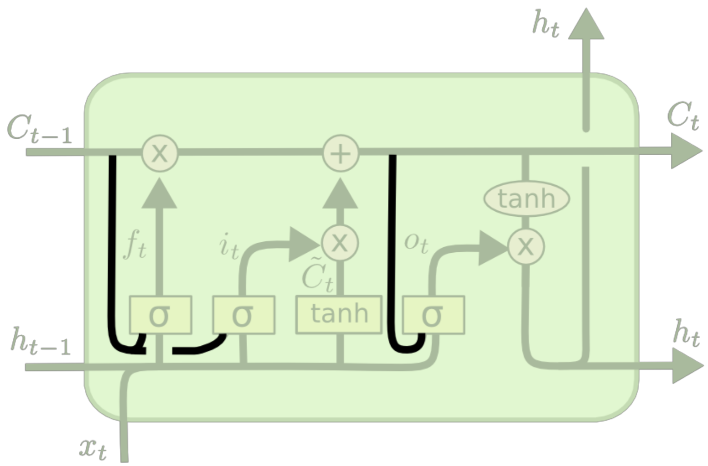
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$



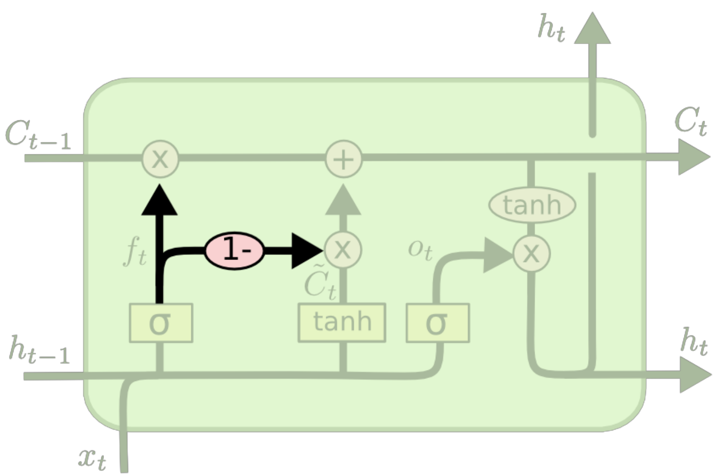
$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

Variants on LSTM

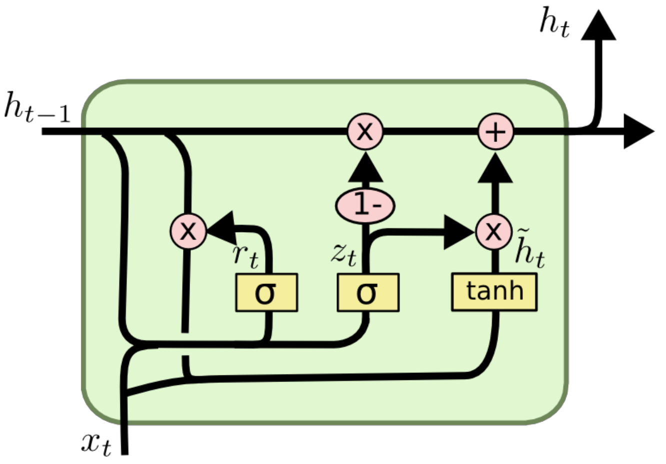


$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \; + \; b_o\right)$$



$$C_t = f_t * C_{t-1} + \boldsymbol{(1 - f_t)} * \tilde{C}_t$$

GRU



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$