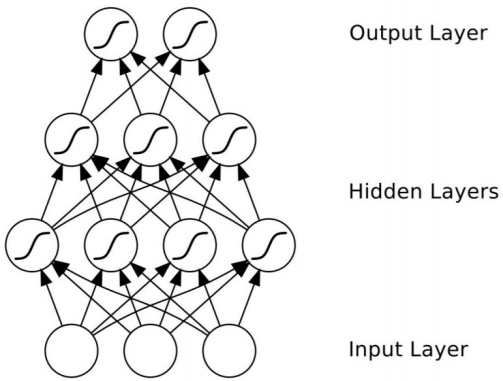
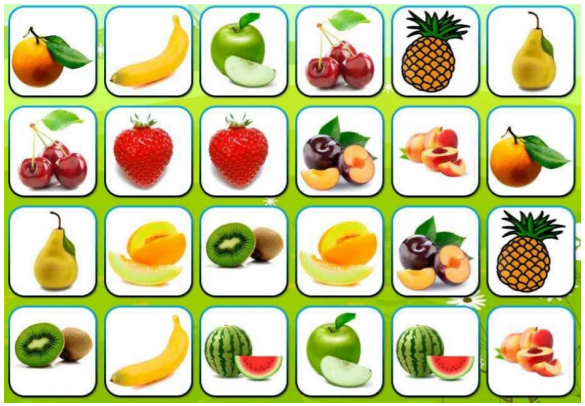
Recurrent Neural Networks

James T. Kwok

Feedforward Neural Network

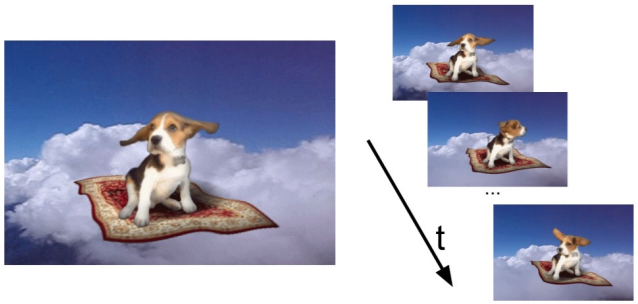


- output depends only on the current input

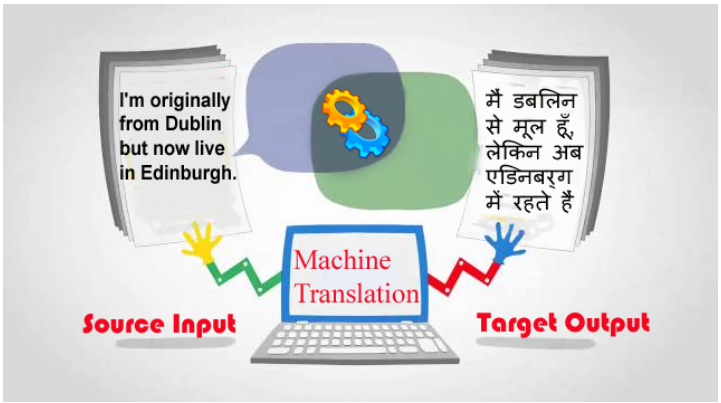


Learning with Sequences

video processing



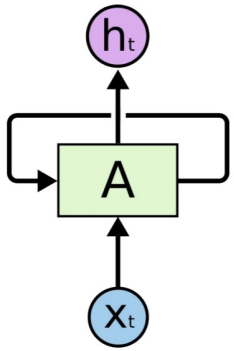
machine translation



speech recognition

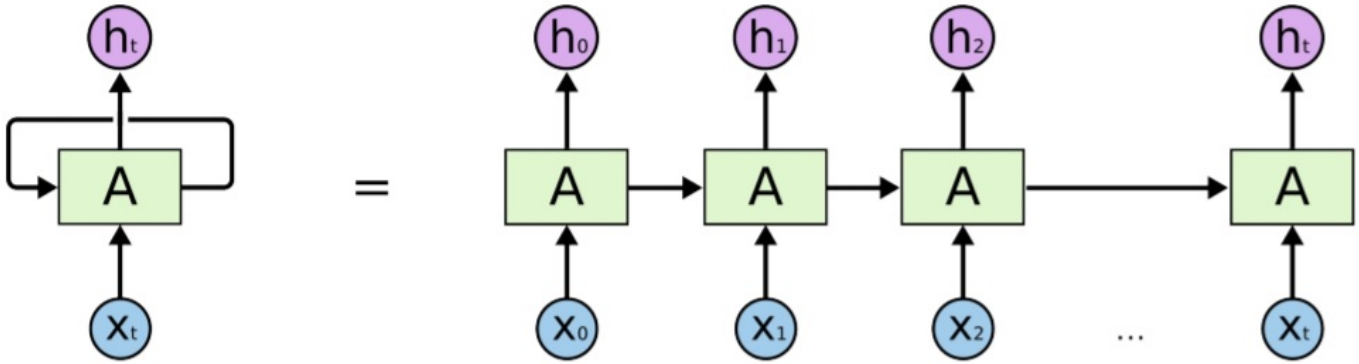
- acoustic features at successive time frames

Recurrent Neural Network (RNN)

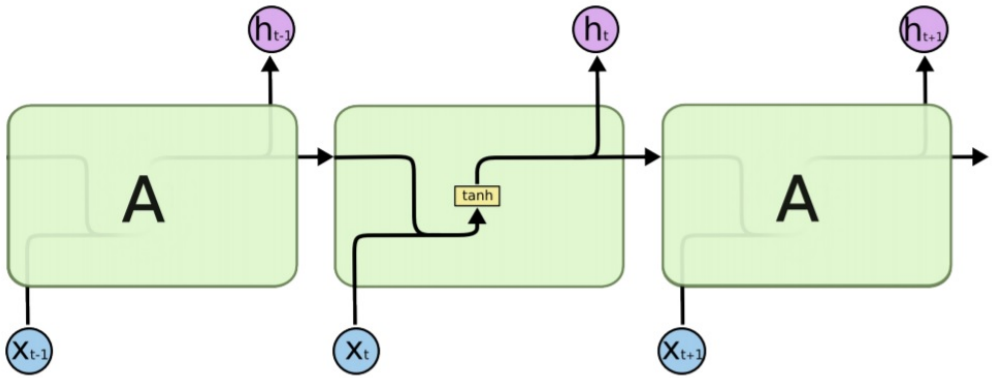


- hidden layers and output depend from previous states of the hidden layers
- new state = function of (old state, input x_t at some time step)

Unfolding a RNN



- the same weights are used for different instances of the artificial neurons at different time steps
- multiple copies of the same network, each passing the old state to a successor
  - state: keeps some important aspects of the past sequence



Vanishing and Exploding Gradients

problem

vanishing gradients

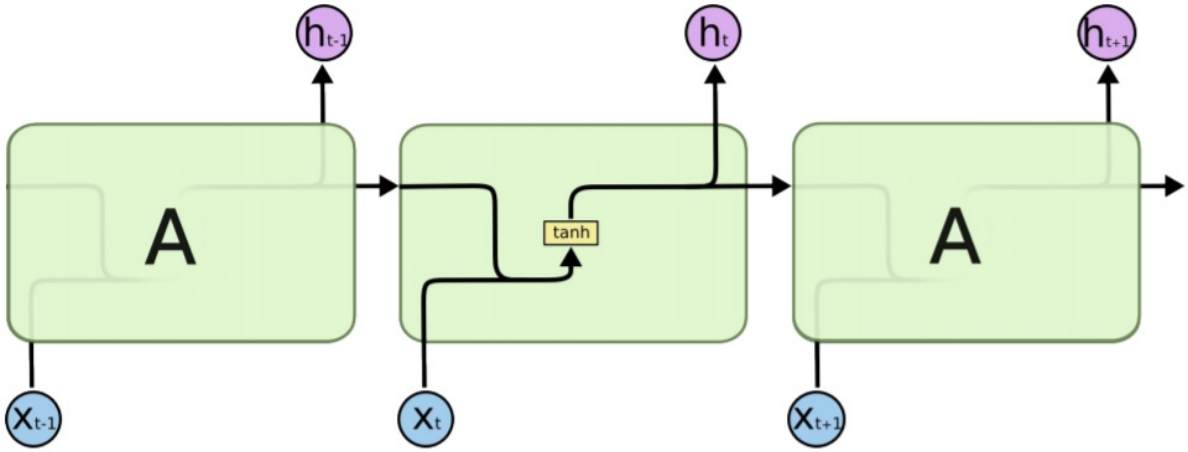
- gradient signal gets so small that learning either becomes very slow or stops working altogether

exploding gradients

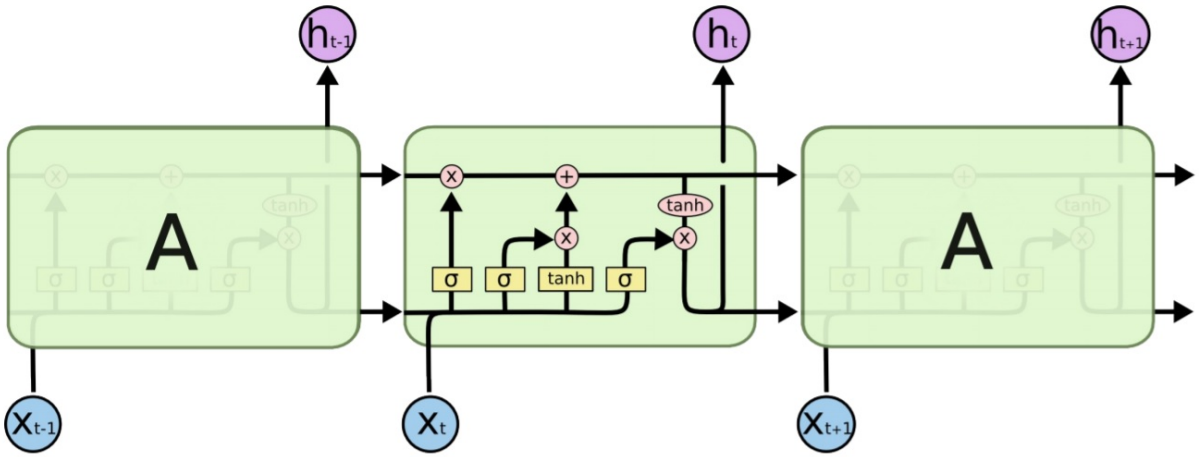
- gradient signal is so large that it can cause learning to diverge

Long Short-Term Memory (LSTM)

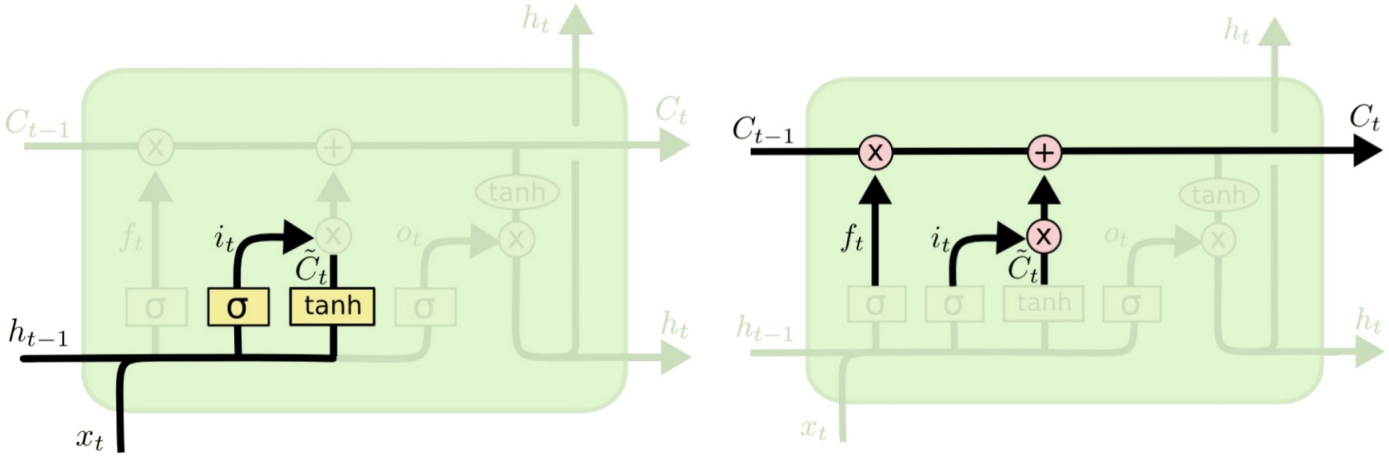
RNN



LSTM



Input Gate



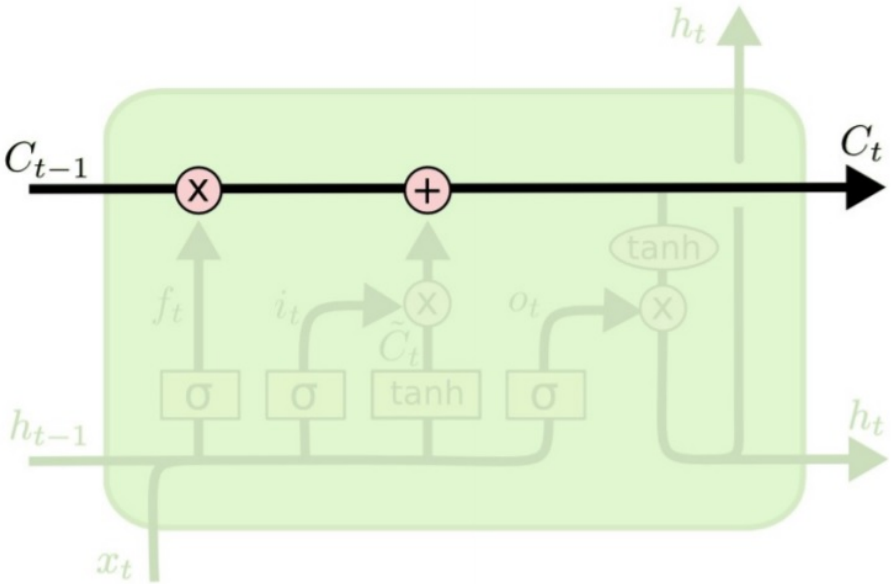
$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$$

- creates a vector that can be added to the state

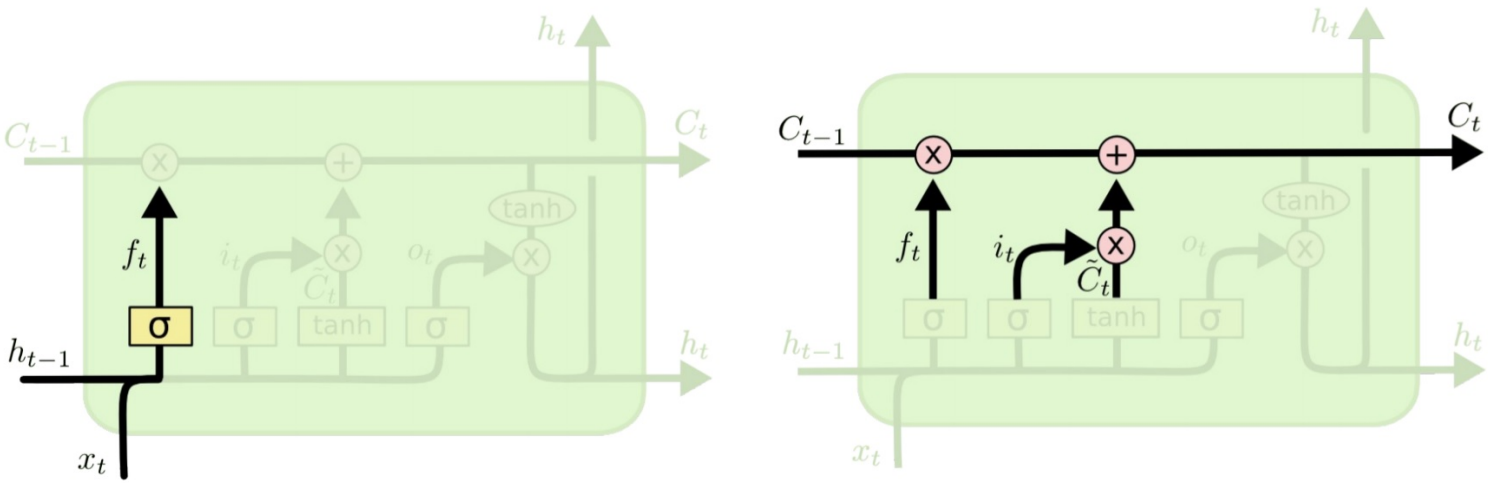
$$\text{(input gate)} \quad i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

- allows incoming signal to alter the state of the memory cell or block it

Cell State



Forget Gate



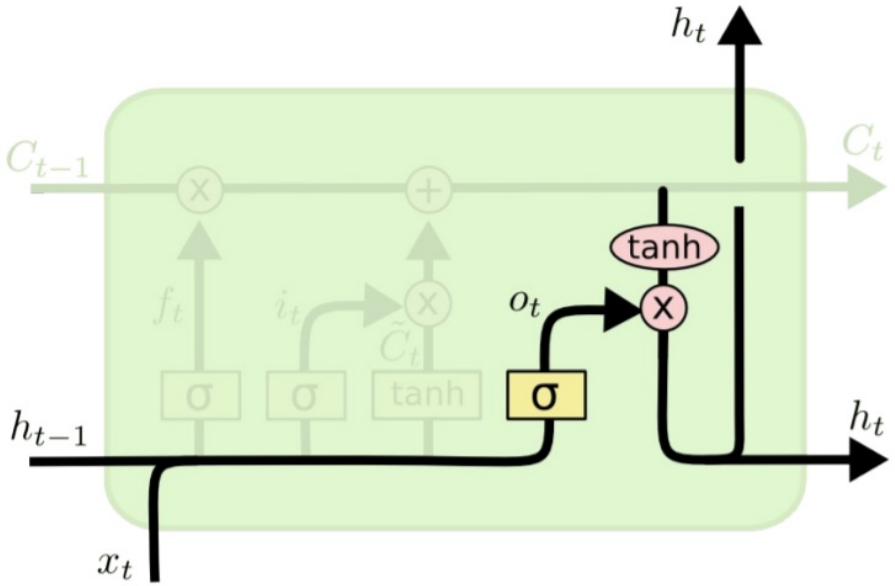
$$\text{(forget gate)} \quad f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

- allow the cell to remember or forget its previous state
- information stays in the cell so long as its forget gate is on

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

- $\odot$: elementwise multiplication

Output Gate



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

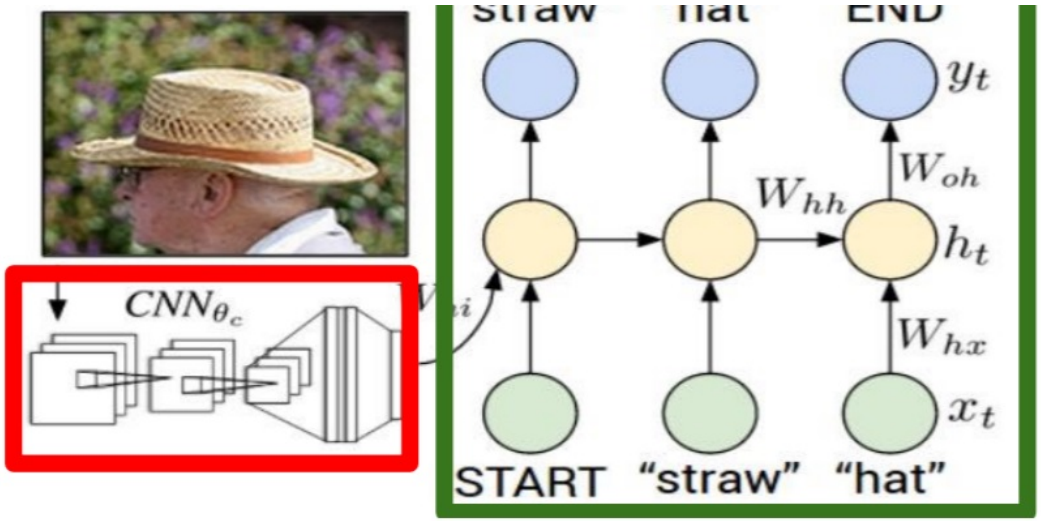
- output gate can allow/prevent the state of the memory cell to have an effect on other neurons

$$h_t = o_t \odot \tanh(C_t)$$

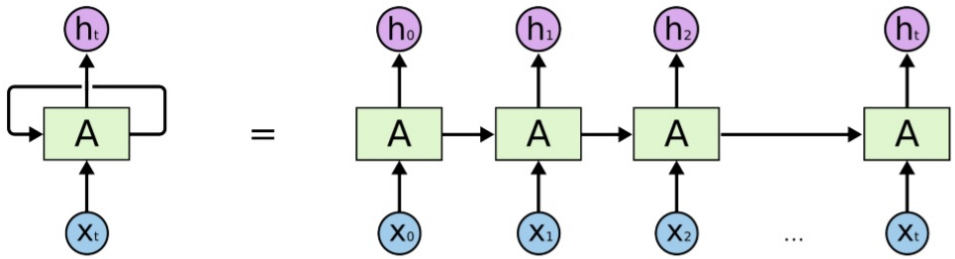
Combination with Other Deep Networks

Example (image captioning)

image $\rightarrow$ sequence of words



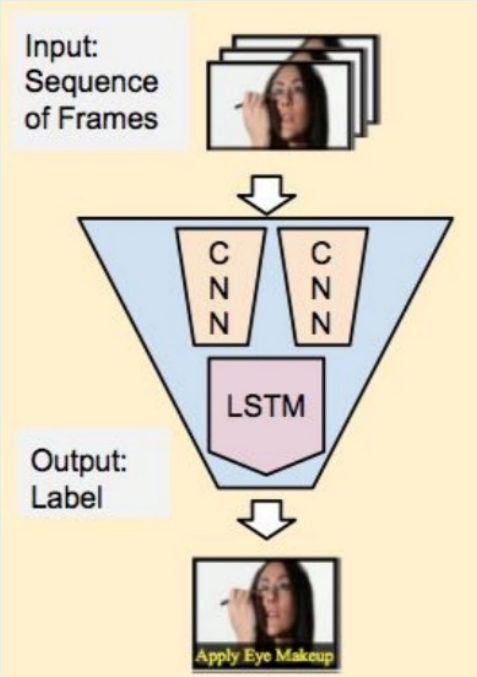
- use CNN to extract image features
- the LSTM is placed after the convolution layer of the CNN



Example: Activity Recognition

Example

sequence of images $\rightarrow$ activity



Example: Machine Translation

Example

sequence of words $\rightarrow$ sequence of words