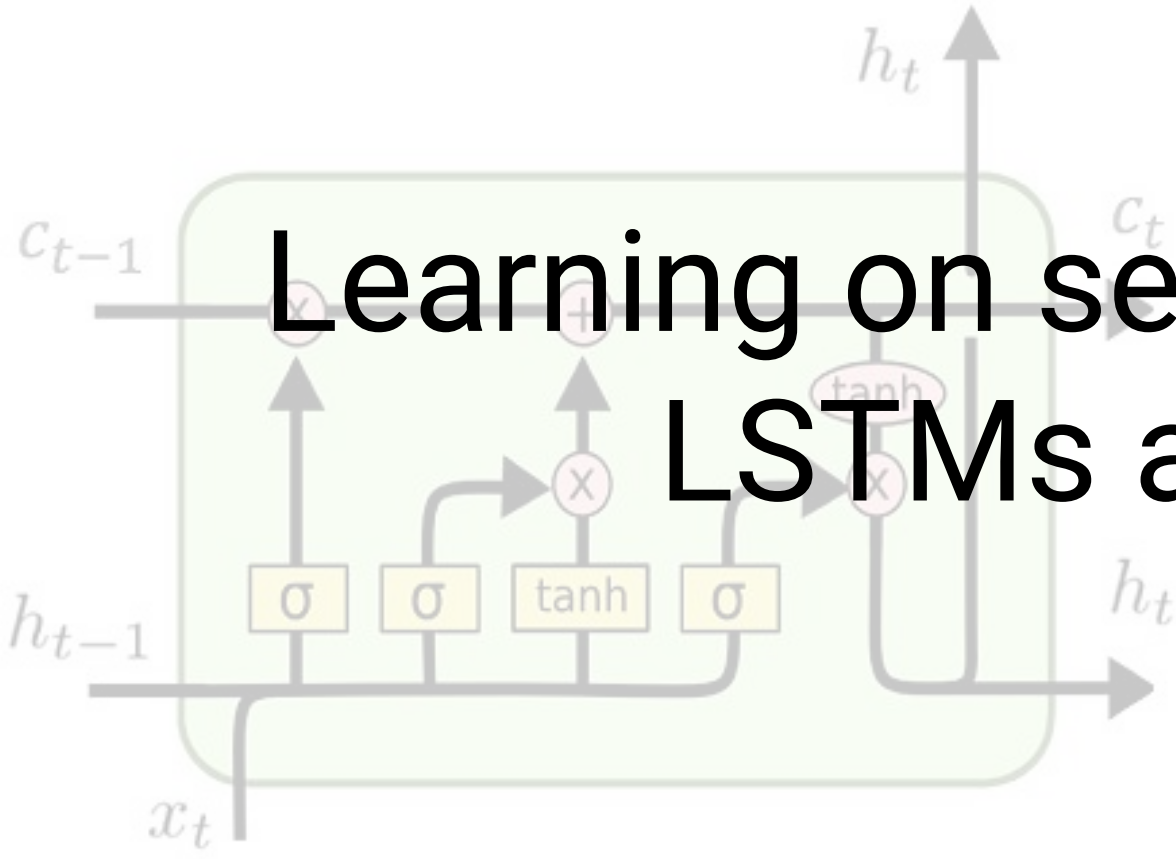
Class core values

1. Be **respect**ful to yourself and others
2. Be **confident** and believe in yourself
3. Always do your **best**
4. Be **cooperative**
5. Be **creative**
6. Have **fun**
7. Be **patient** with yourself while you learn
8. Don’t be shy to **ask “stupid” questions**
9. Be **inclusive** and **accepting**



Week 5, Lecture 2

Learning on sequences: LSTMs and GRUs

Learning Objectives

1. Describe the need for LSTMs and GRUs
2. Explain the basic architecture of LSTM unit
3. Explain the basic architecture of GRU
4. Implement a LSTM module using keras
5. Tune LSTM model

Parameter sharing in RNNs allow for adopting different lengths

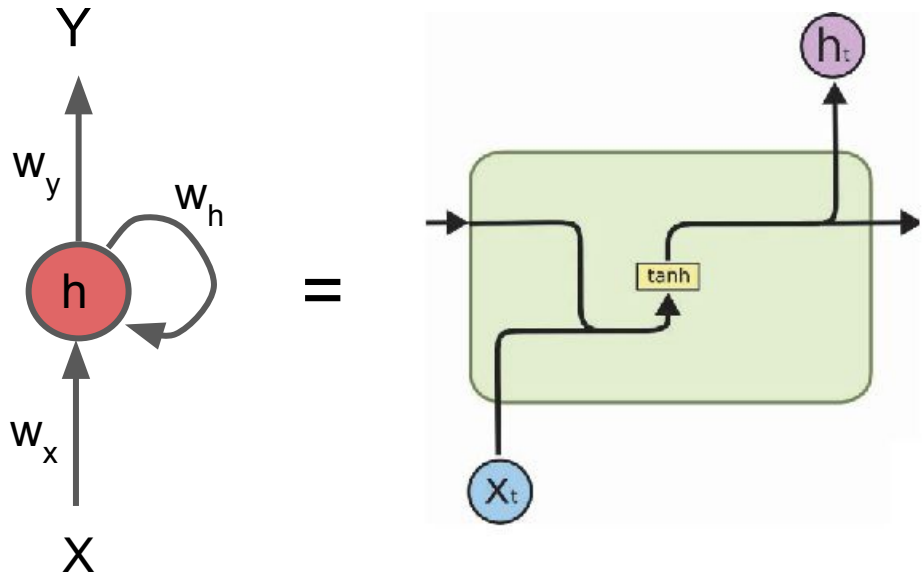
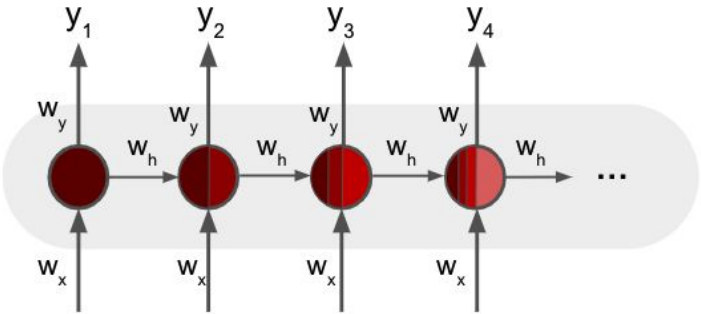


Image from Li et al,2016 International Joint Conference on Neural Networks (IJCNN), **2016**

Due to limitations of the architecture, remembering context from past is challenging

The sky is blue

I live in France. I love this city. I speak French



LSTMs are developed to help with the memory issue of RNNs

LSTM

Long **S**hort **T**erm **M**emory

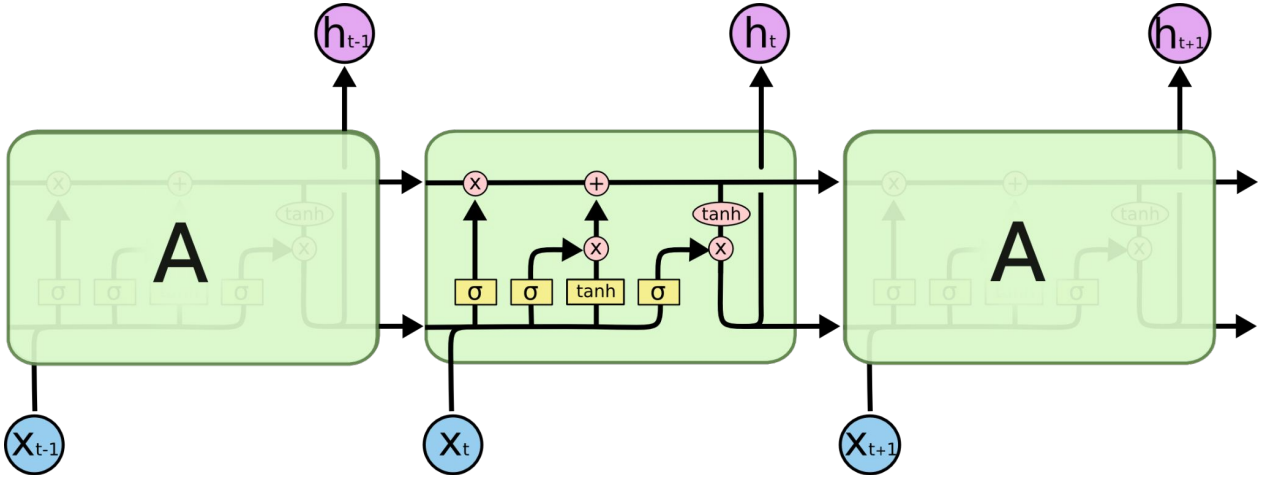


Image from Colah's blog: Understanding LSTMs

The key to LSTMs is the cell state

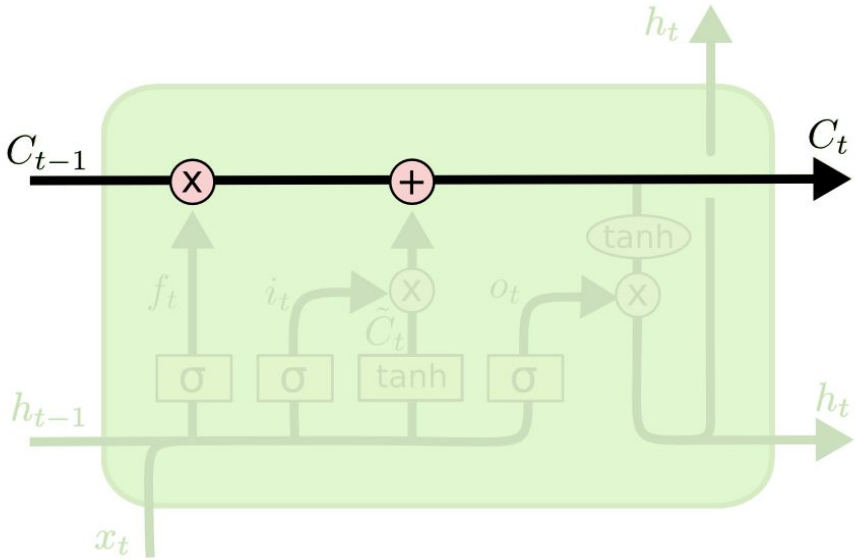


Image from Colah's blog: Understanding LSTMs

Cell states and output are controlled by a series of *gates*

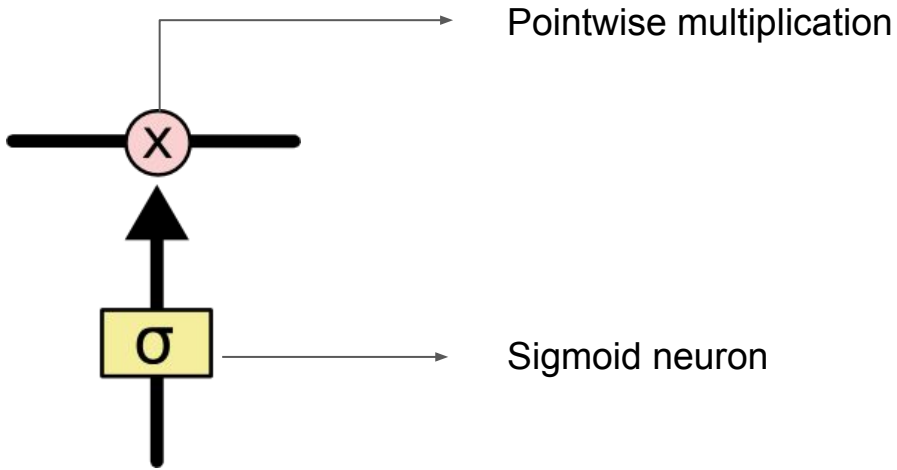
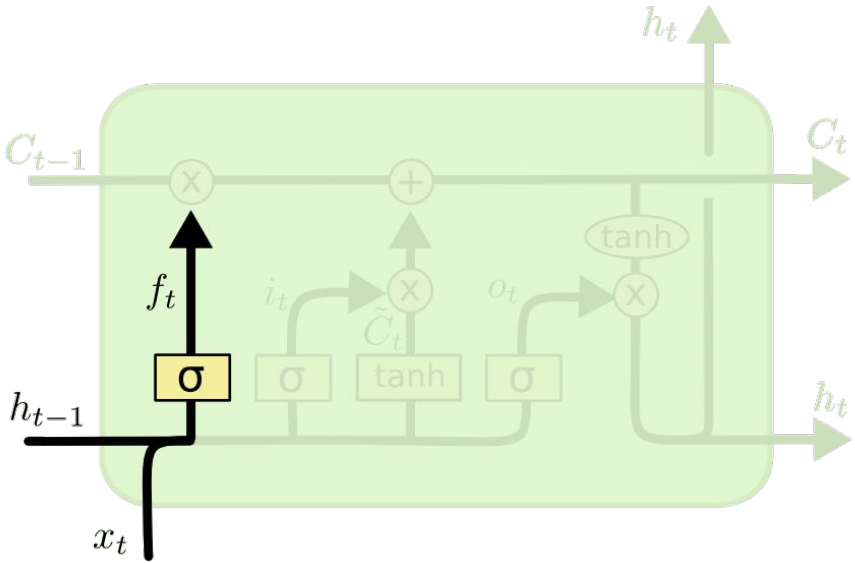


Image from Colah's blog: Understanding LSTMs

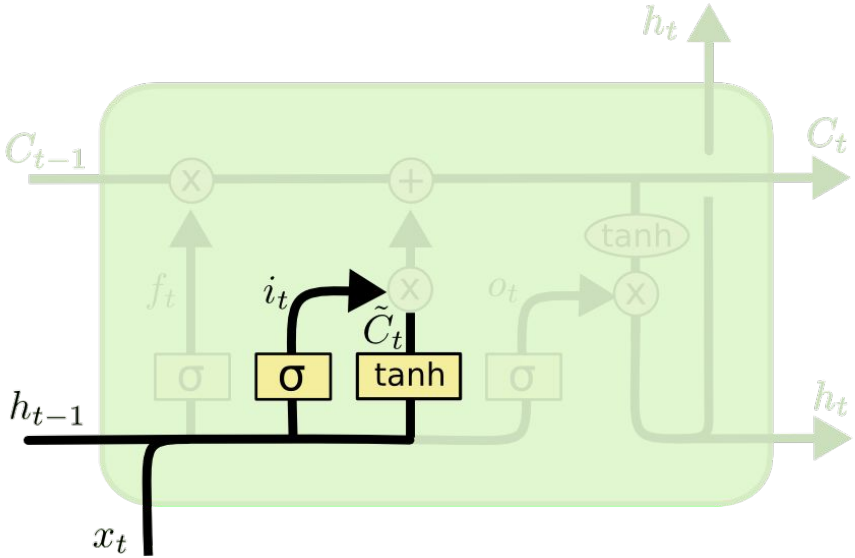
In the first step, we decide how much of the previous information to keep (forget gate)



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

Image from Colah's blog: Understanding LSTMs

The network then decides which values to update

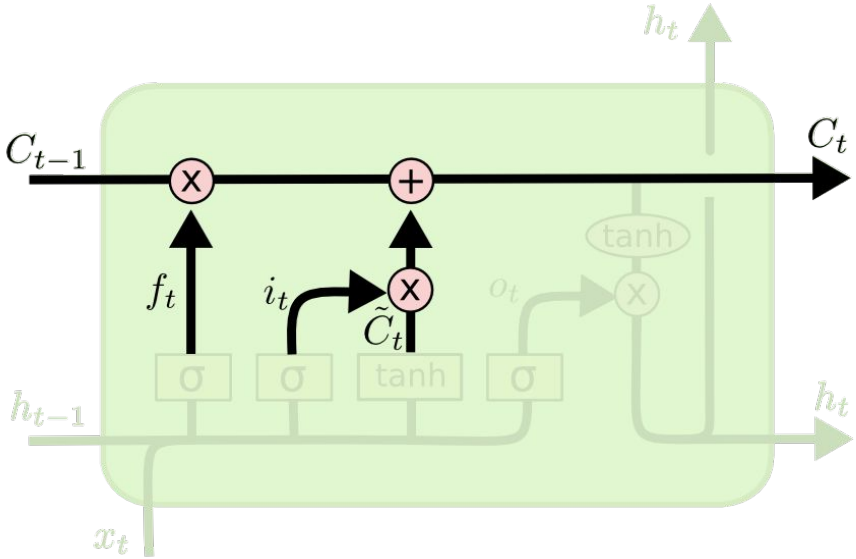


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

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

Image from Colah's blog: Understanding LSTMs

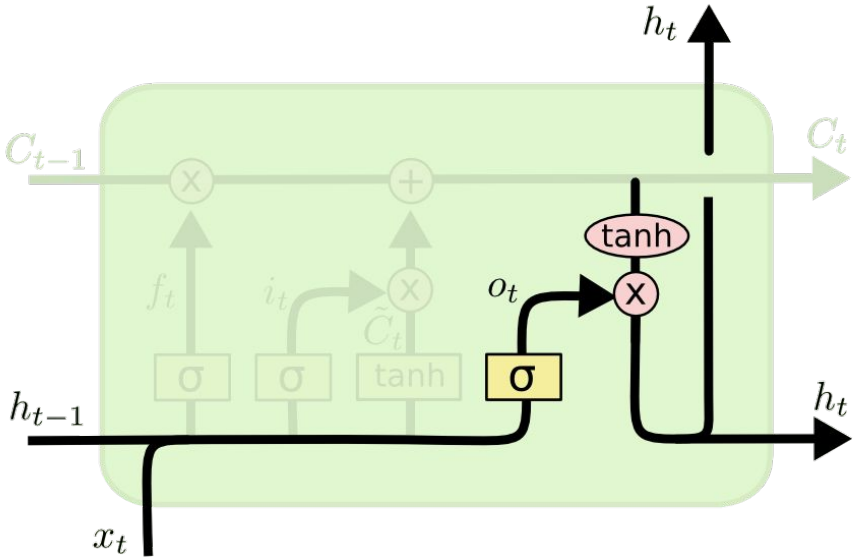
Cell state is updated by throwing out old information and adding new ones



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

Image from Colah's blog: Understanding LSTMs

The output is generated based on the input and the updated cell state



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

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

Image from Colah's blog: Understanding LSTMs

GRUs are a variation of LSTMs designed to deal with the vanishing gradient problem

GRU

Gated **R**ecurrent **U**nit

Update gate and reset gat

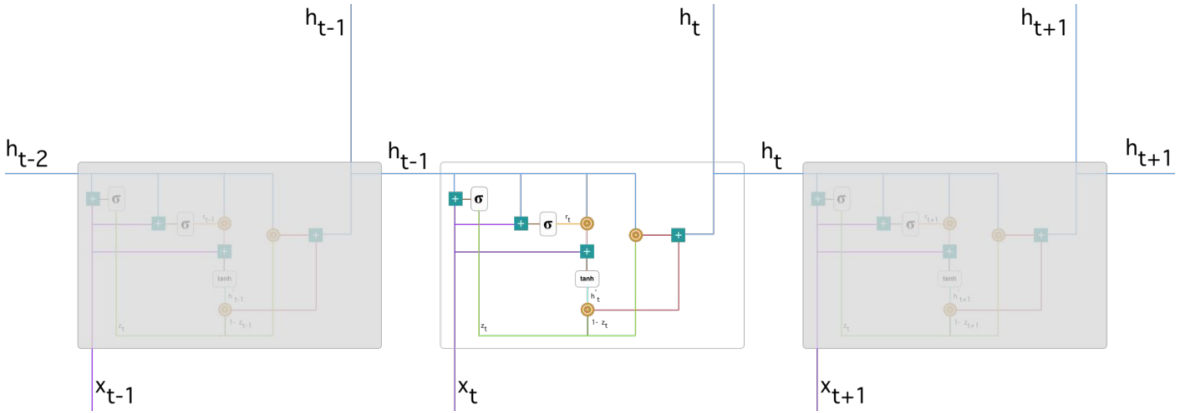
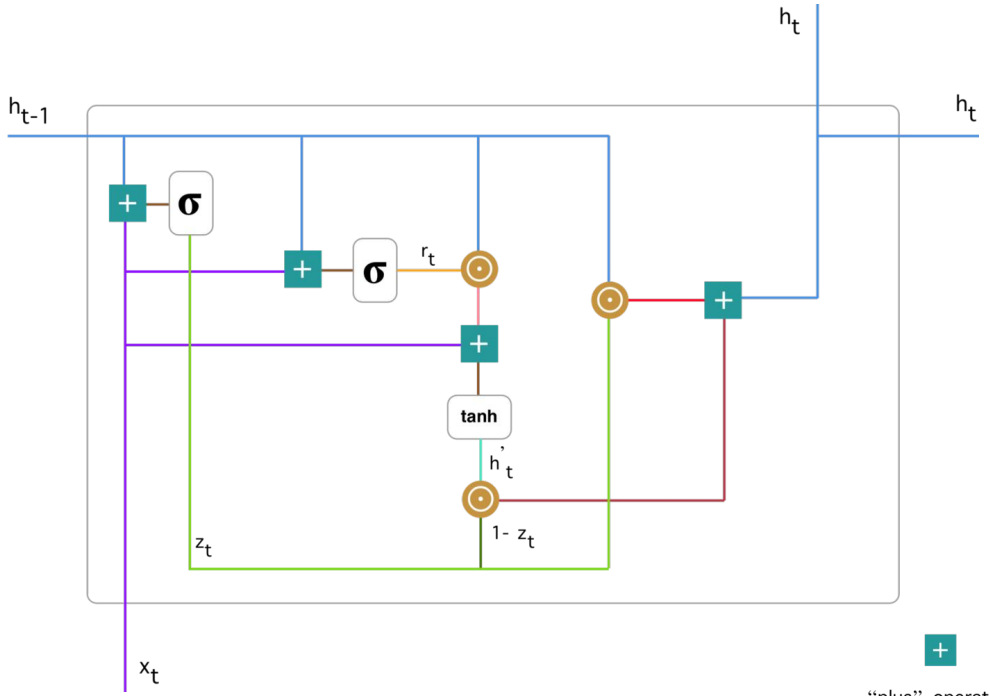


Image from TDS - understanding GRUs

The gates are designed to decide which information to keep/forget



"plus" operation

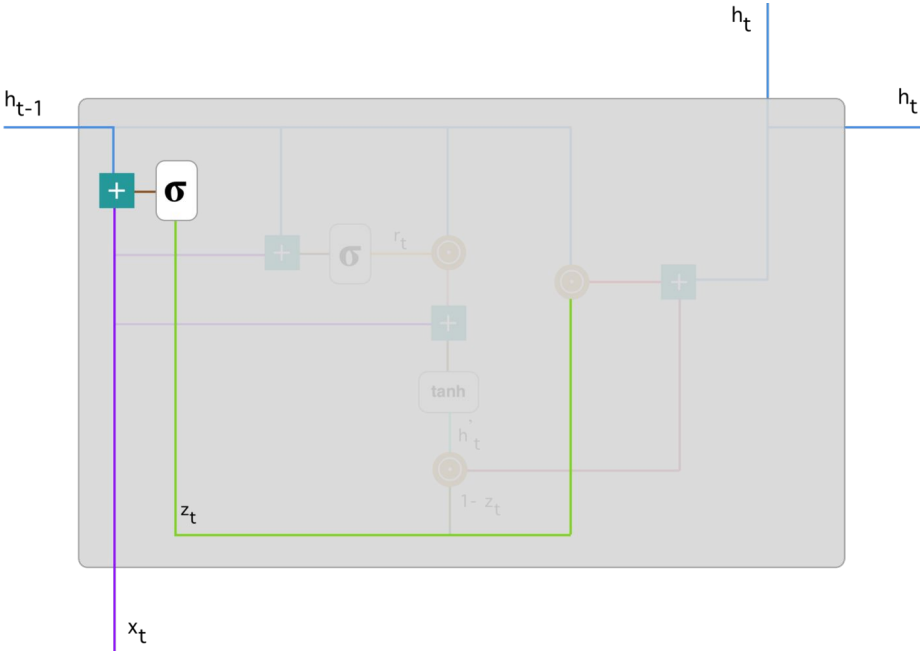
σ "sigmoid" function

"Hadamard product" operation

tanh "tanh" function

Image from TDS - understanding GRUs

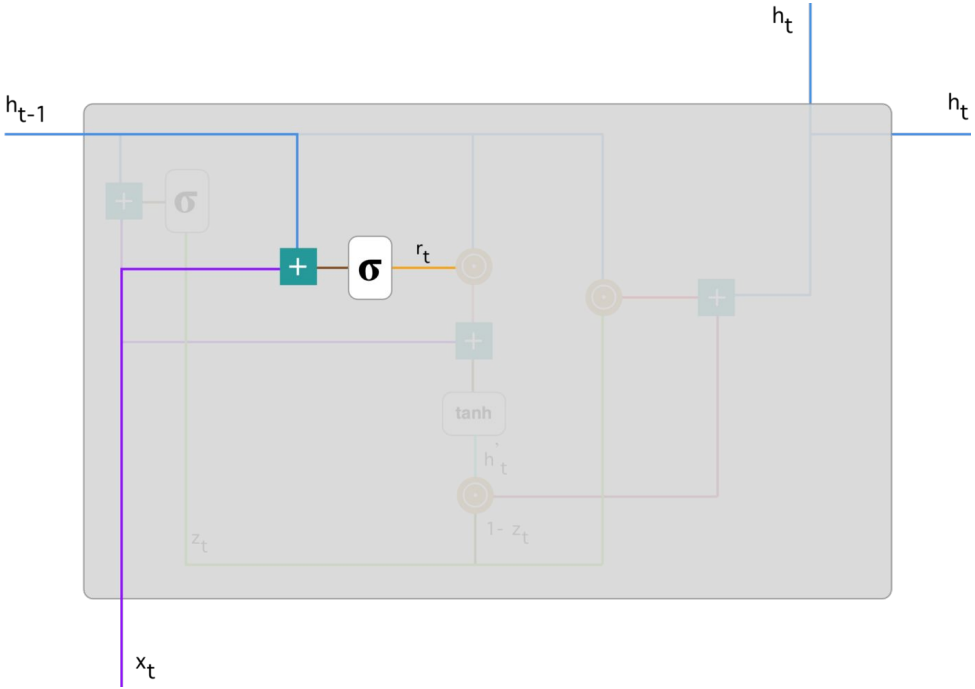
Update gate helps the network to learn how much of the past information to remember



$$z_t = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1})$$

Image from TDS - understanding GRUs

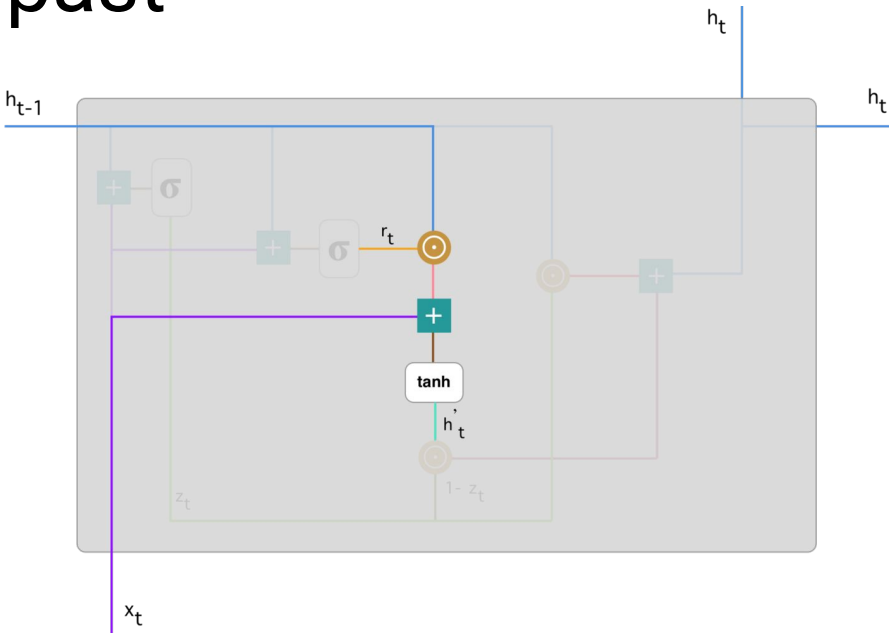
Reset gate helps the network to learn how much of the past information to forget



$$r_t = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1})$$

Image from TDS - understanding GRUs

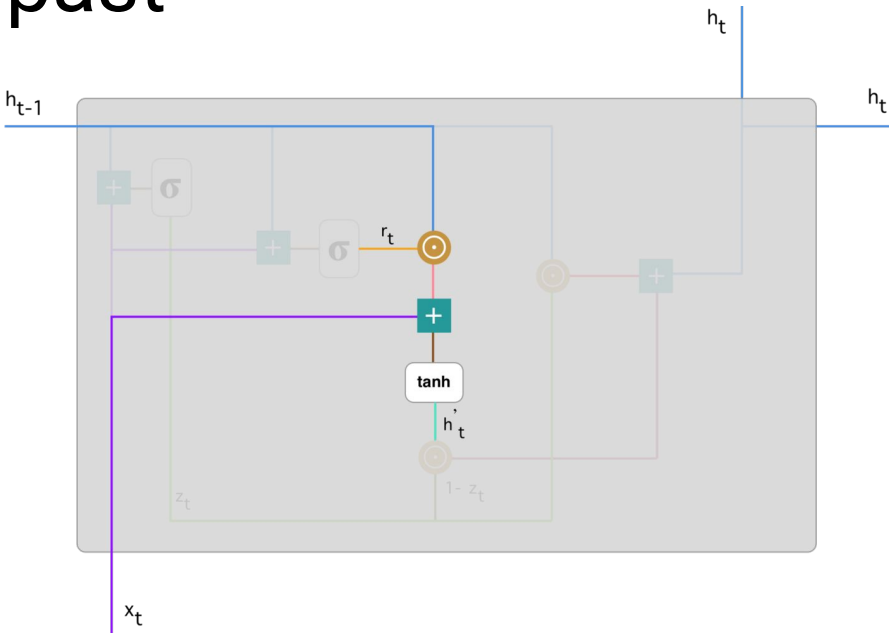
The information of reset gate is used to let the network keep only relevant information from past



$$h_t^{'} = \tanh(Wx_t + r_t \odot Uh_{t-1})$$

Image from TDS - understanding GRUs

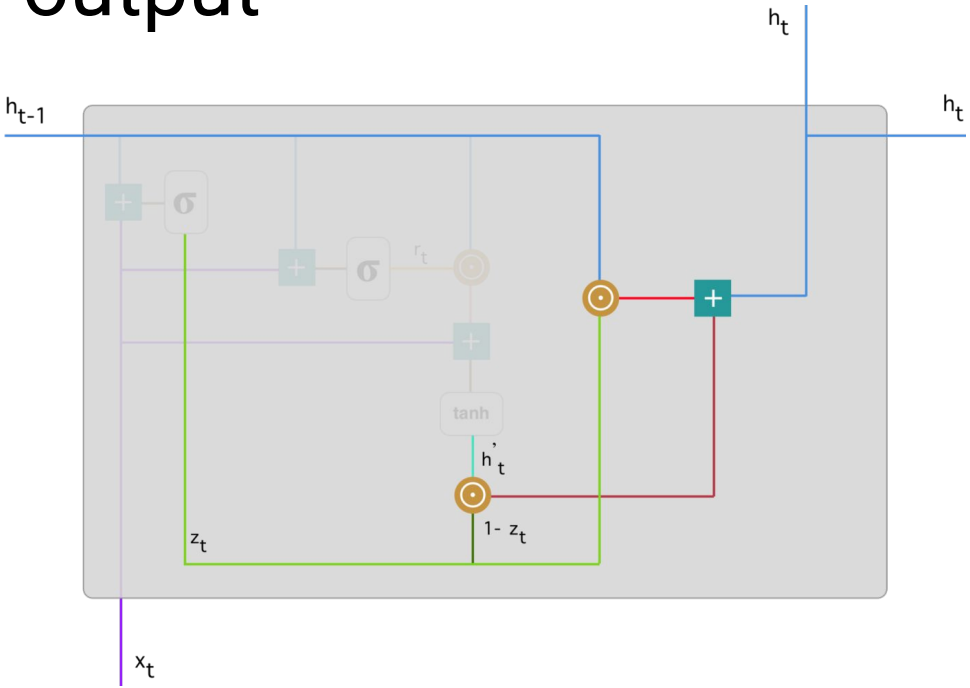
The information of reset gate is used to let the network keep only relevant information from past



$$h_t^{'} = \tanh(Wx_t + r_t \odot Uh_{t-1})$$

Image from TDS - understanding GRUs

The network then updates itself based on the knowledge on what to remember to generate output



$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h_t'$$

Image from TDS - understanding GRUs

Next lecture:

Transformers

Heads-up for next Wednesday

Kristine Deibler · 1st
Computational Design Scientist at Novo Nordisk

Layne Price · 1st
Sr Machine Learning Scientist at Amazon

Nikhil Naik · 2nd
Senior Research Manager | Machine learning, Computer Vision, NLP, AI for Biology

Jack Maguire · 1st
Senior Scientist at Genentech