Recurrent Neural Networks

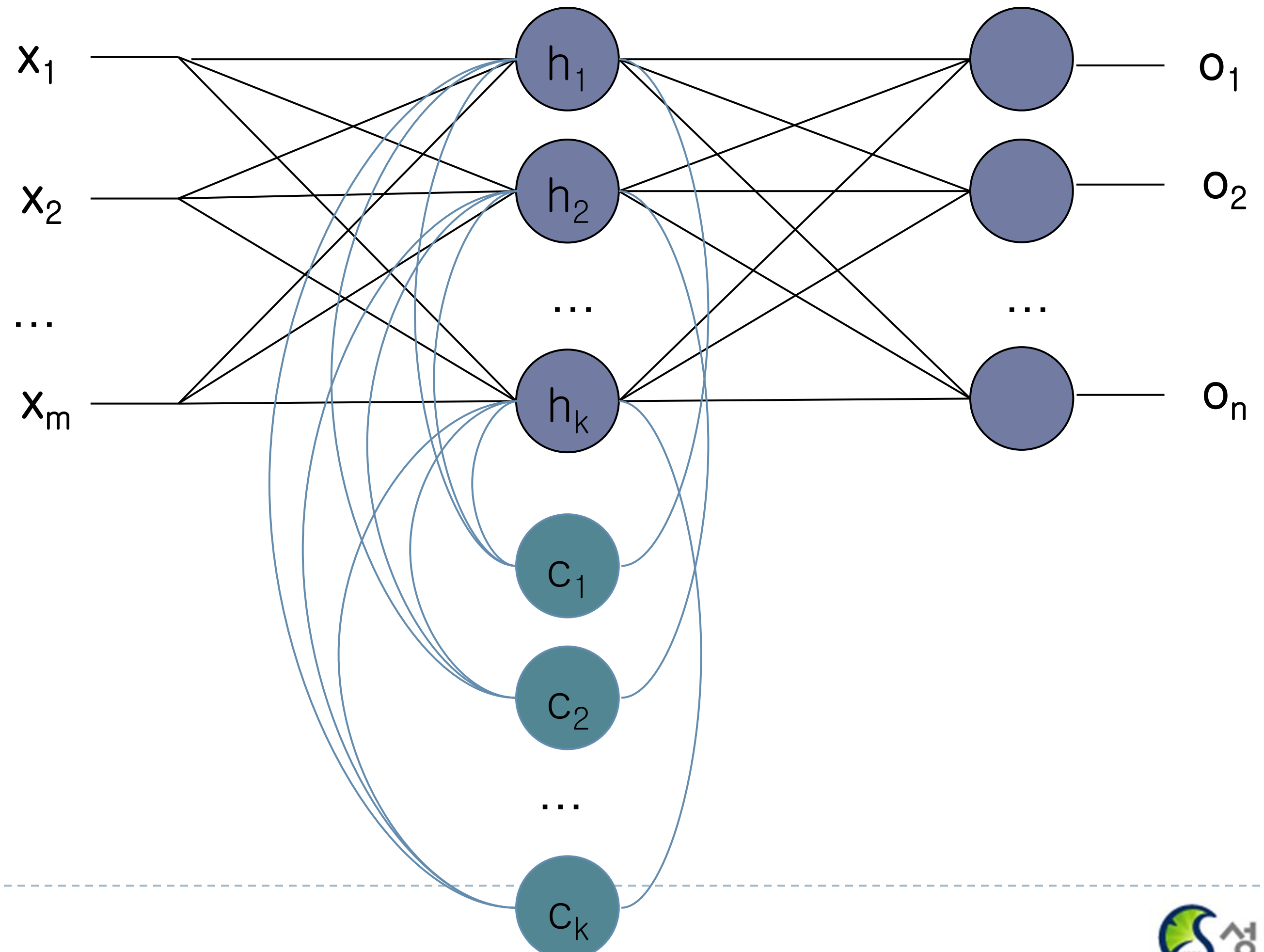
성균관대학교 소프트웨어학과
이 지 형

Sequential Data Modeling

- **Sequential Data**
  - Most of data are sequential
  - Speech, Text, Image, …
- **Deep Learnings for Sequential Data**
  - Convolutional Neural Networks (CNN)
    - Try to find local features from a sequence
  - Recurrent Neural Networks: LSTM, GLU
    - Try to capture the feature of the past

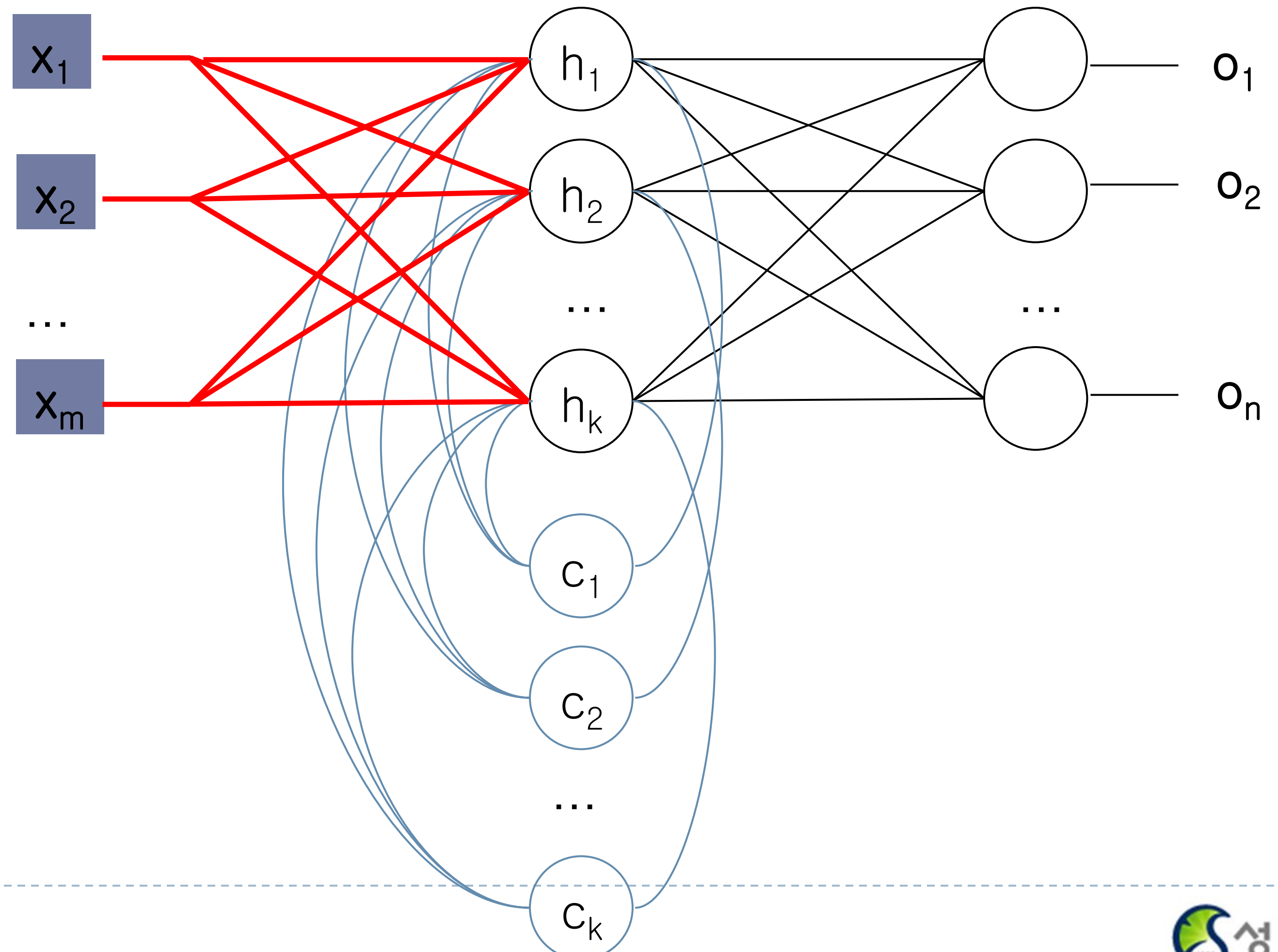
Recurrent Neural Networks

- **Connections form cycles**



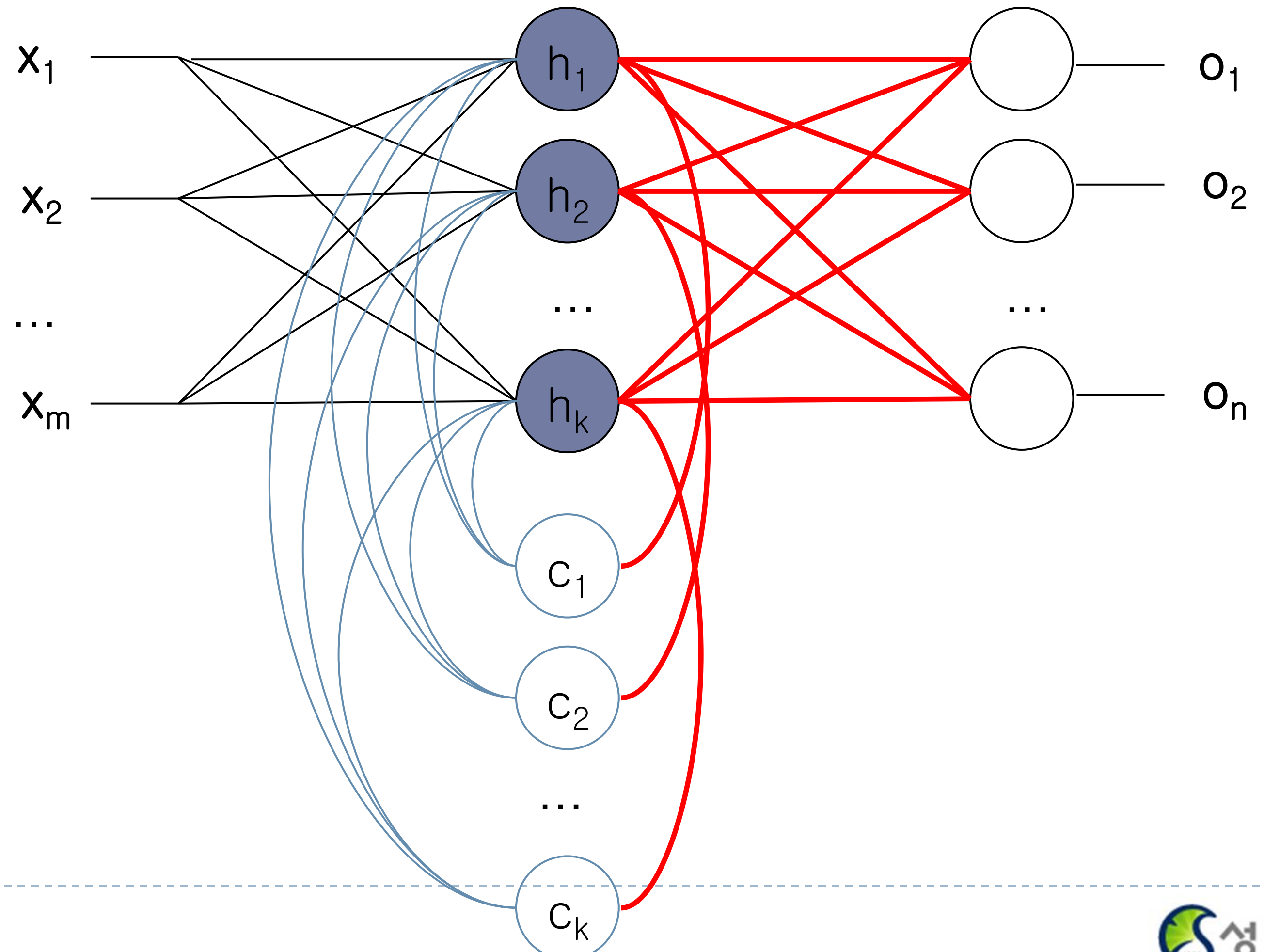
Recurrent Neural Networks

- **Connections form cycles**



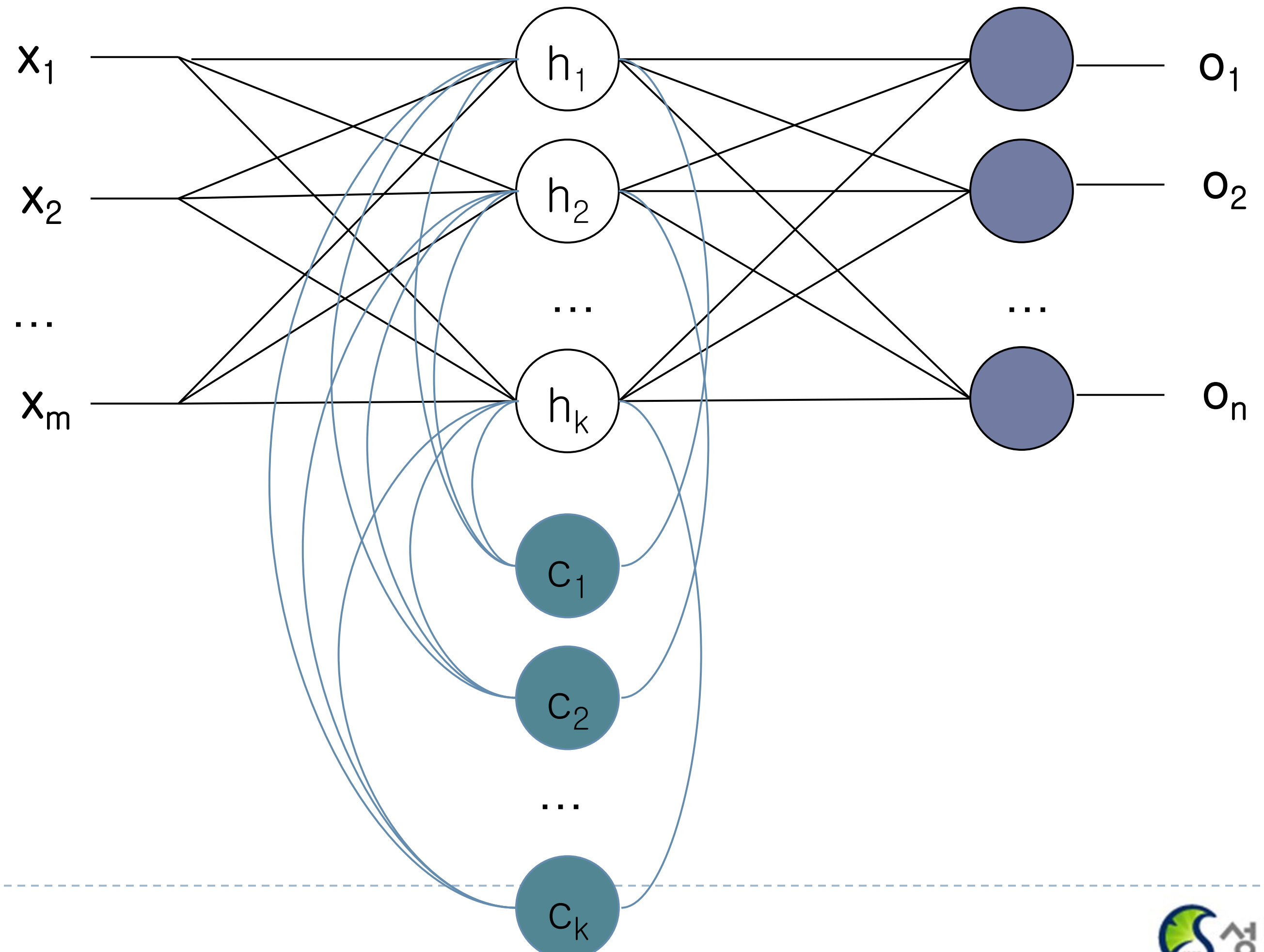
Recurrent Neural Networks

- **Connections form cycles**



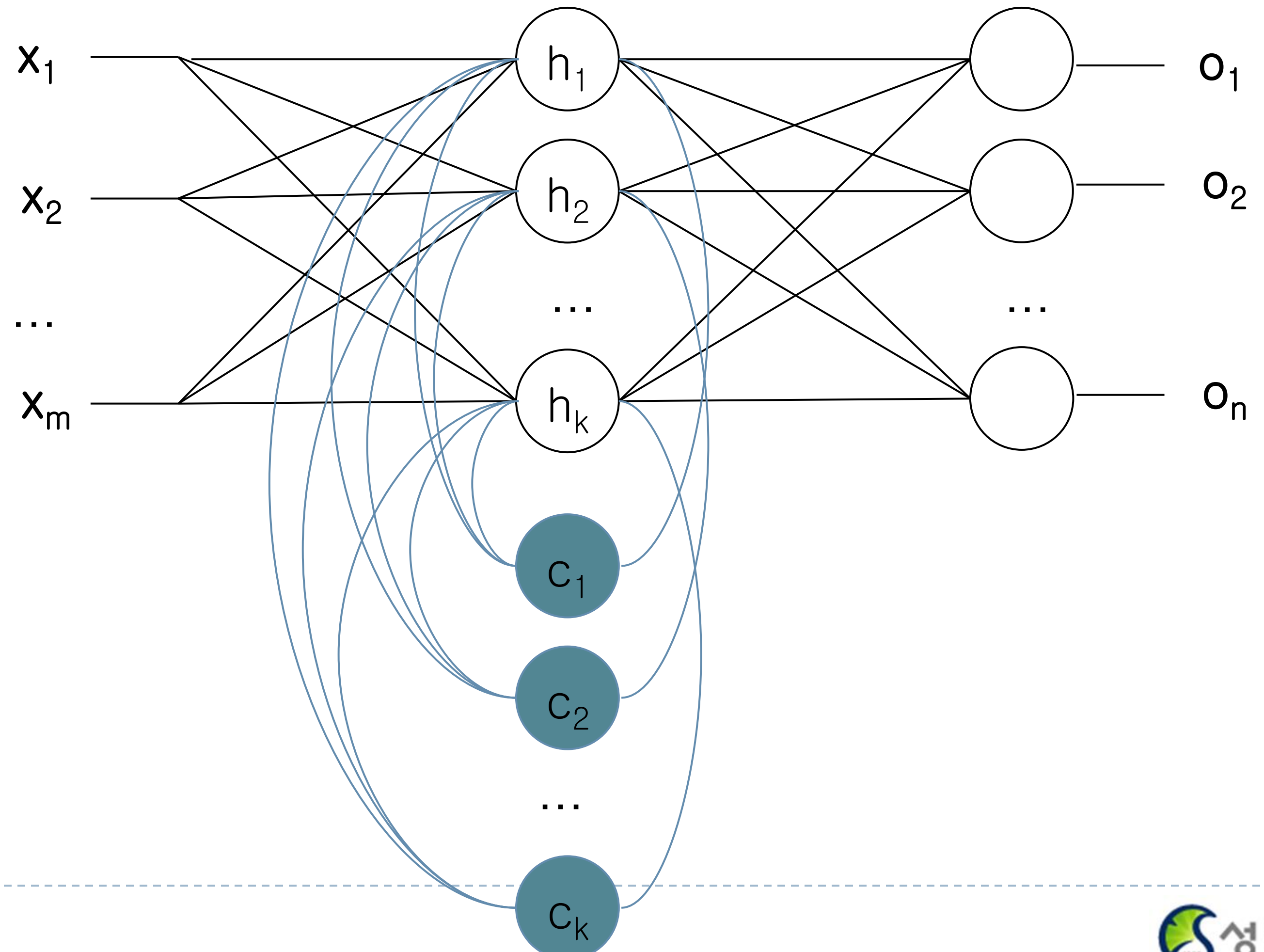
Recurrent Neural Networks

- **Connections form cycles**



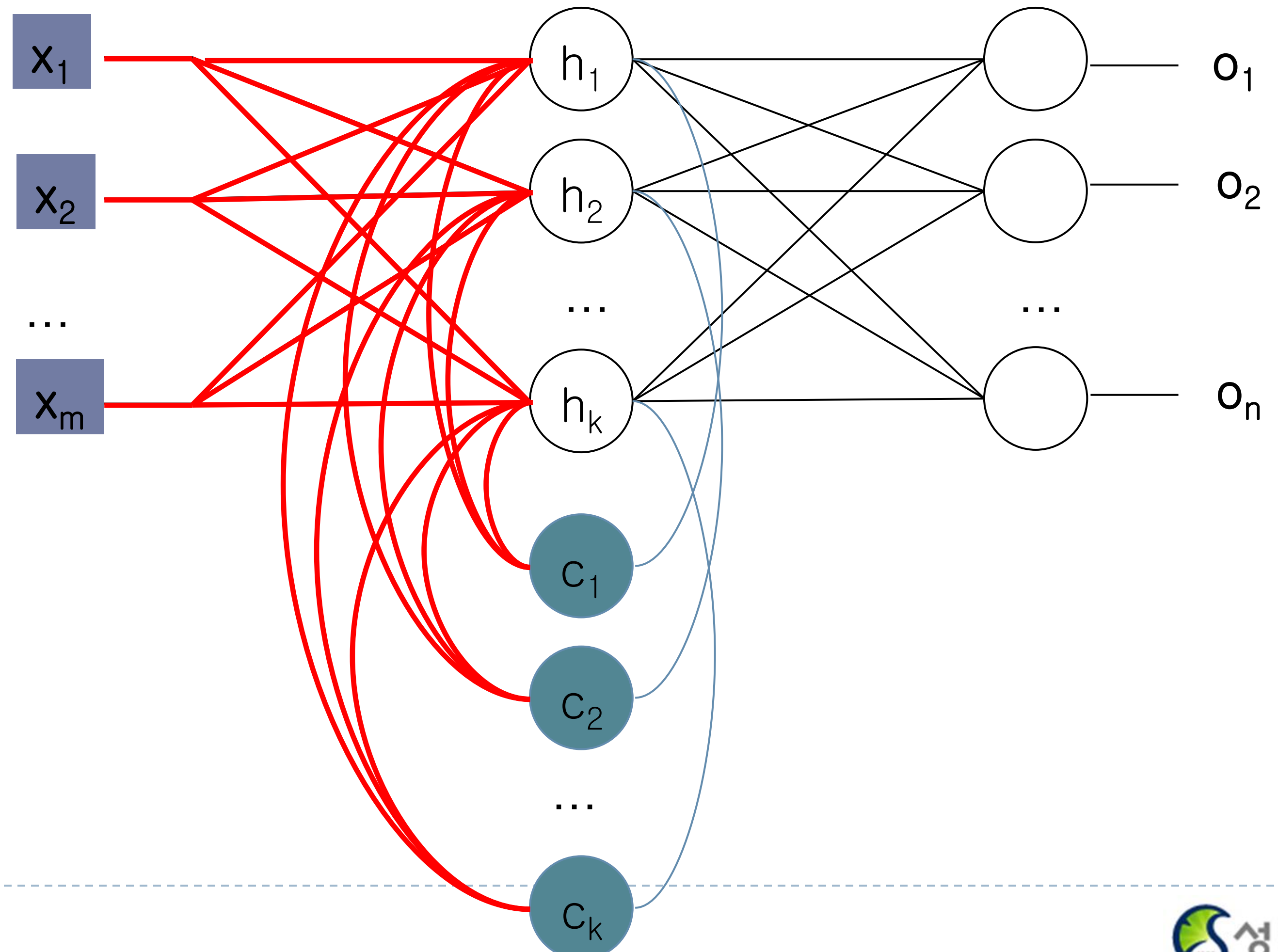
Recurrent Neural Networks

- Connections form cycles



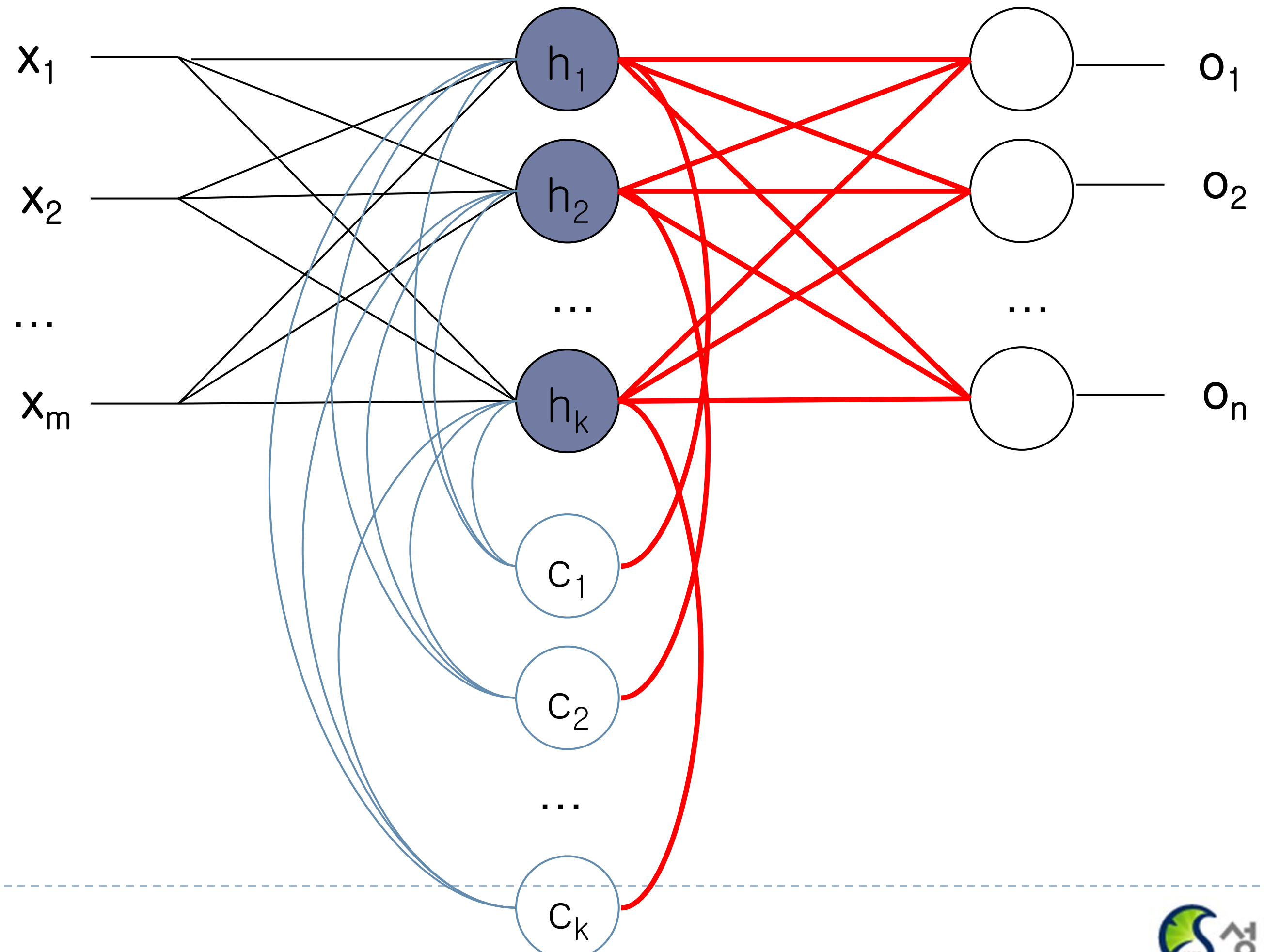
Recurrent Neural Networks

- **Connections form cycles**



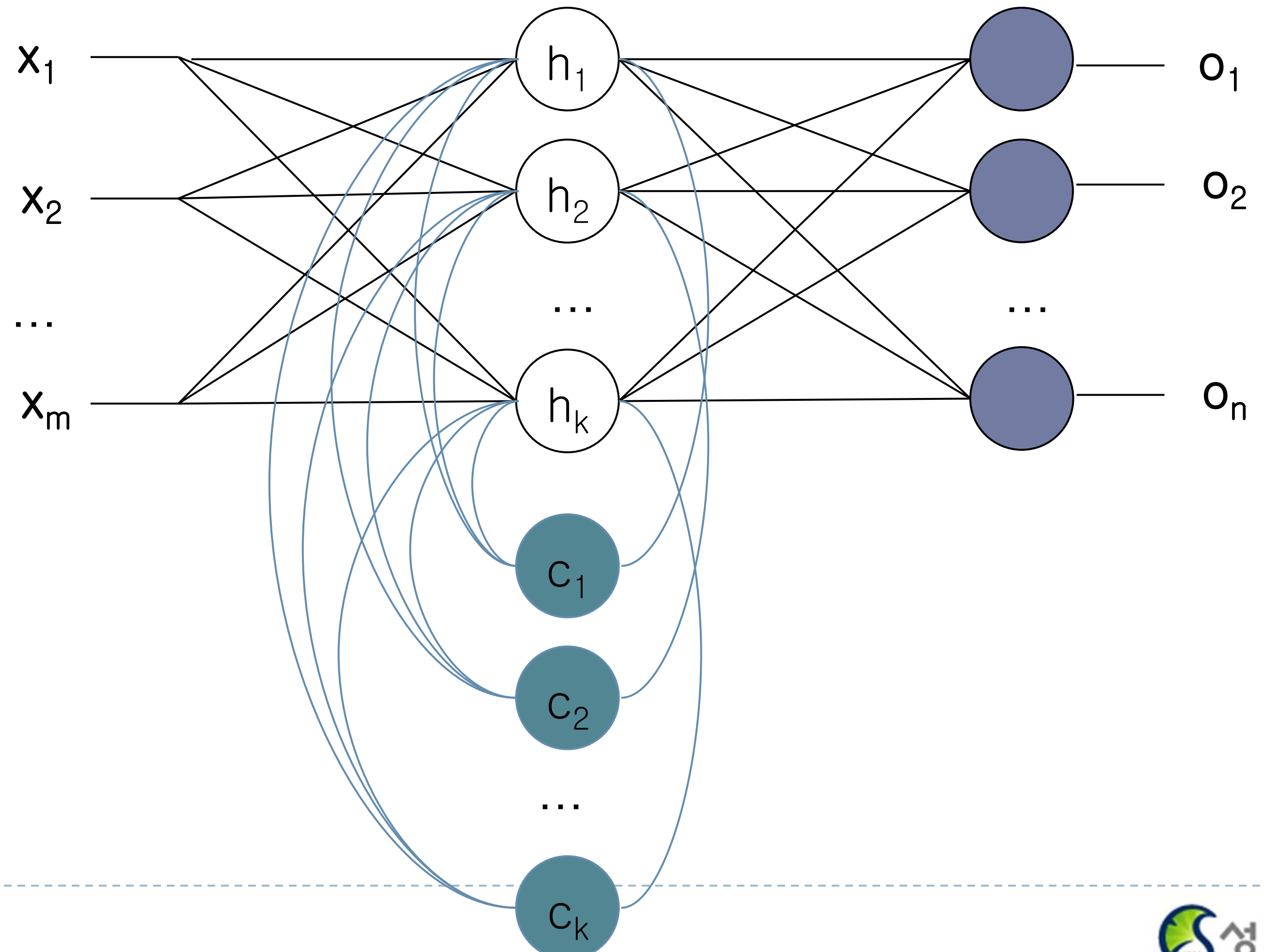
Recurrent Neural Networks

- **Connections form cycles**



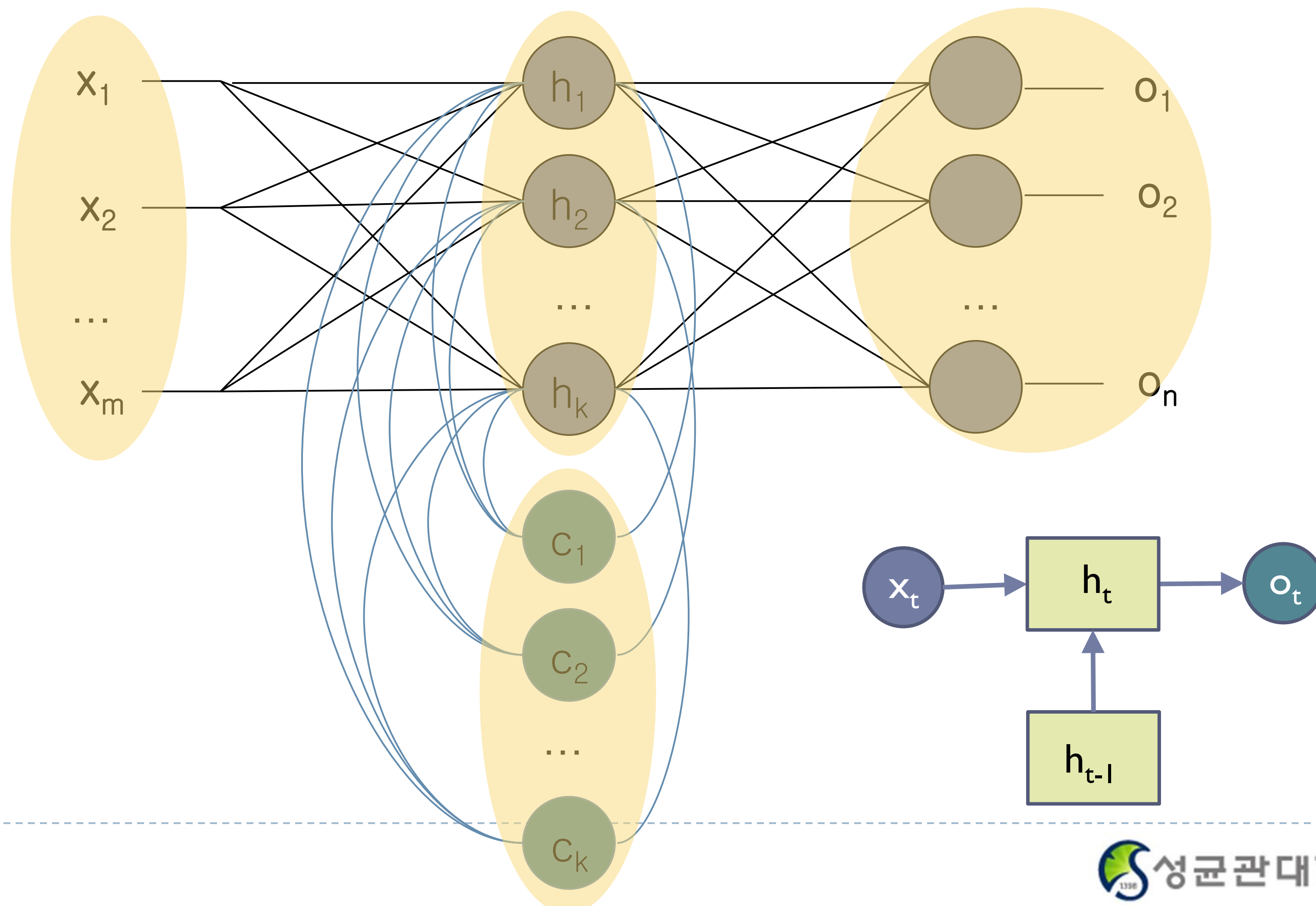
Recurrent Neural Networks

- **Connections form cycles**



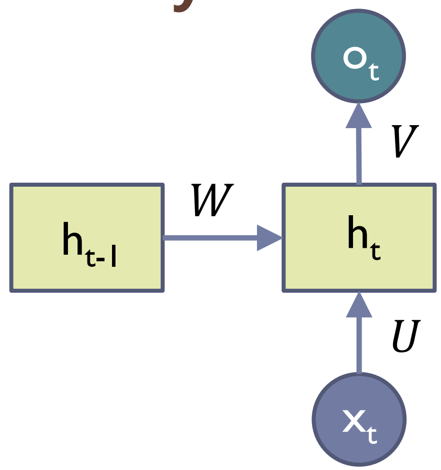
Recurrent Neural Networks

- Connections form cycles



Recurrent Neural Networks

- Connections form cycles

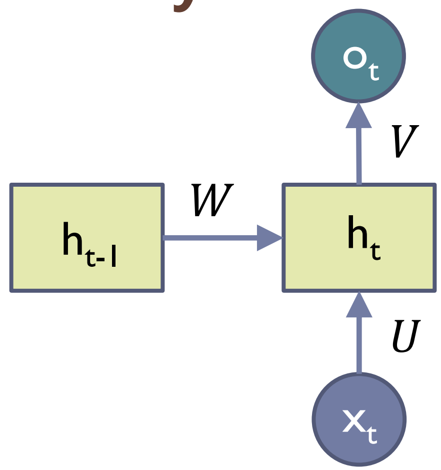


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

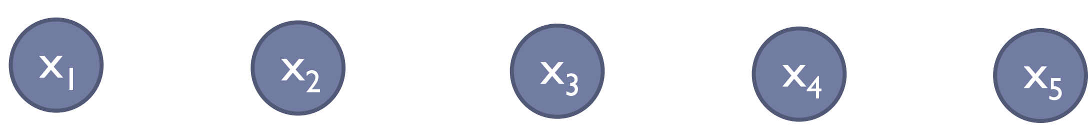
- x_t: input at time t
- h_t: hidden state at time t
- f: is an activation function
- U, V, W: network parameters
  - RNN shares the same parameters across all time steps
- g: activation function for the output layer

Recurrent Neural Networks

- **Connections form cycles**

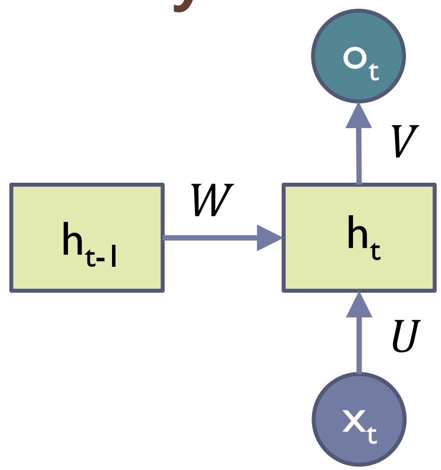


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$



Recurrent Neural Networks

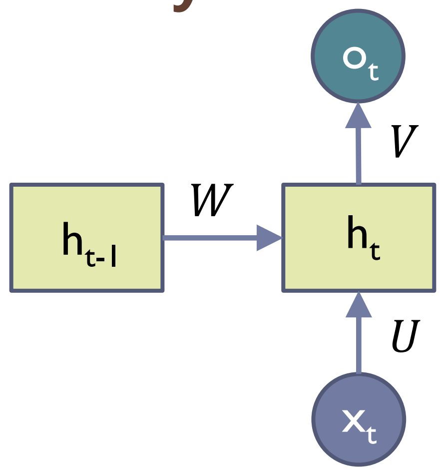
- **Connections form cycles**



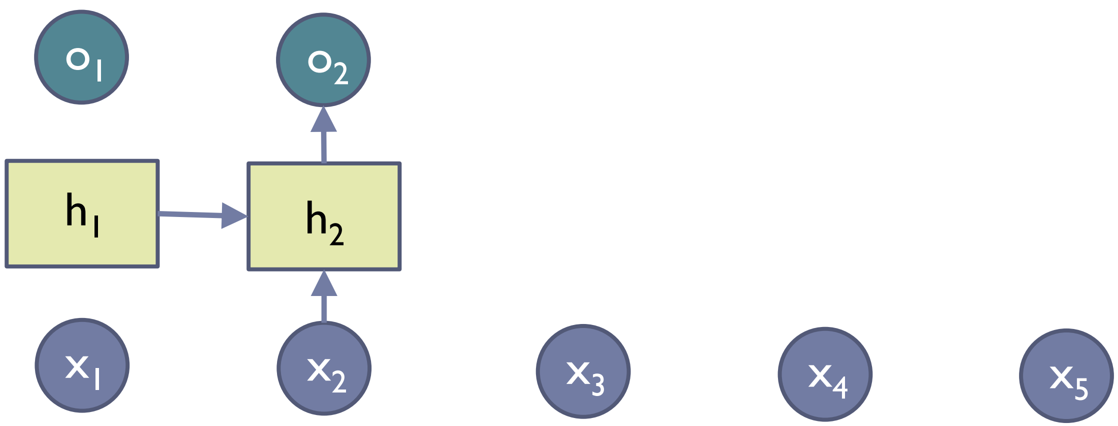
$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

Recurrent Neural Networks

- **Connections form cycles**

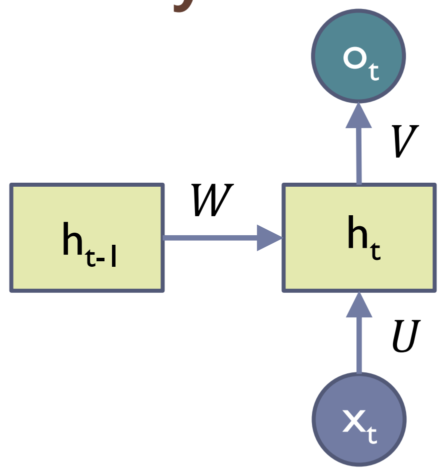


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

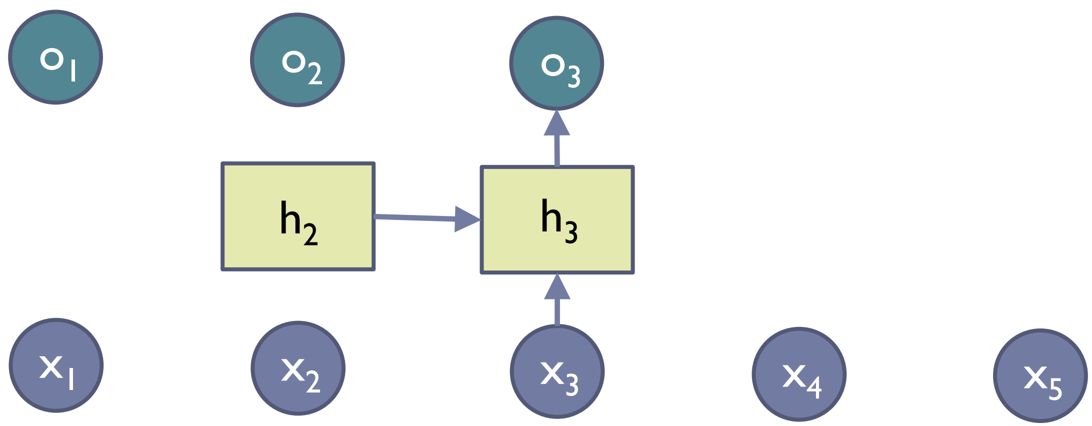


Recurrent Neural Networks

- **Connections form cycles**

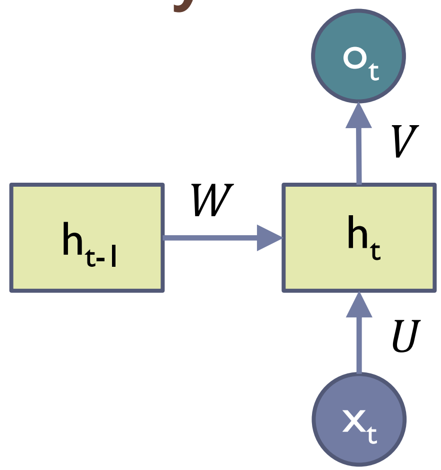


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

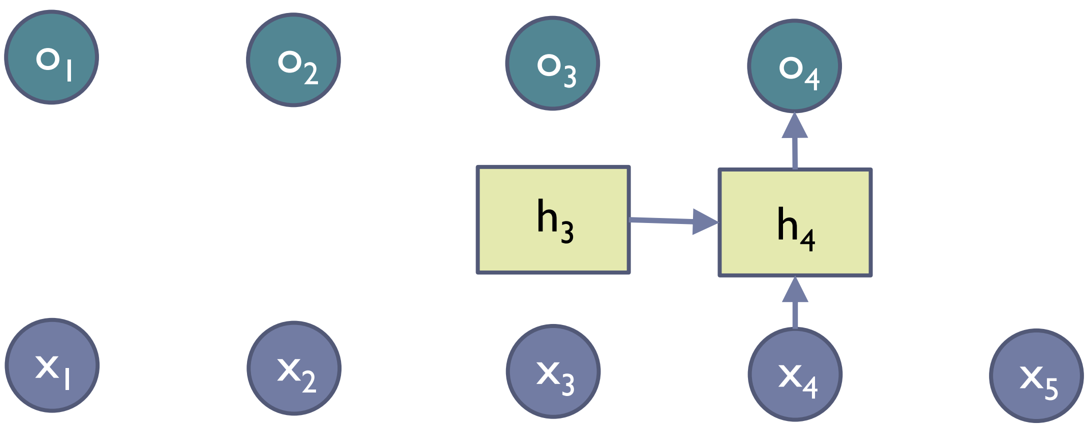


Recurrent Neural Networks

- **Connections form cycles**

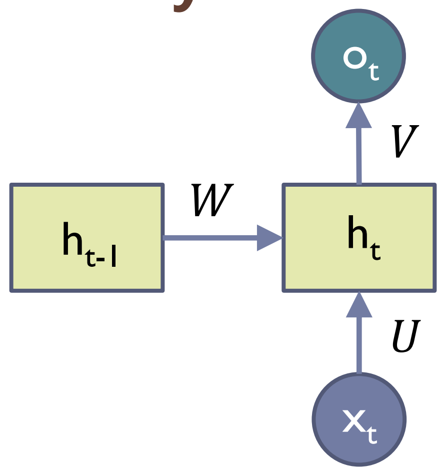


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

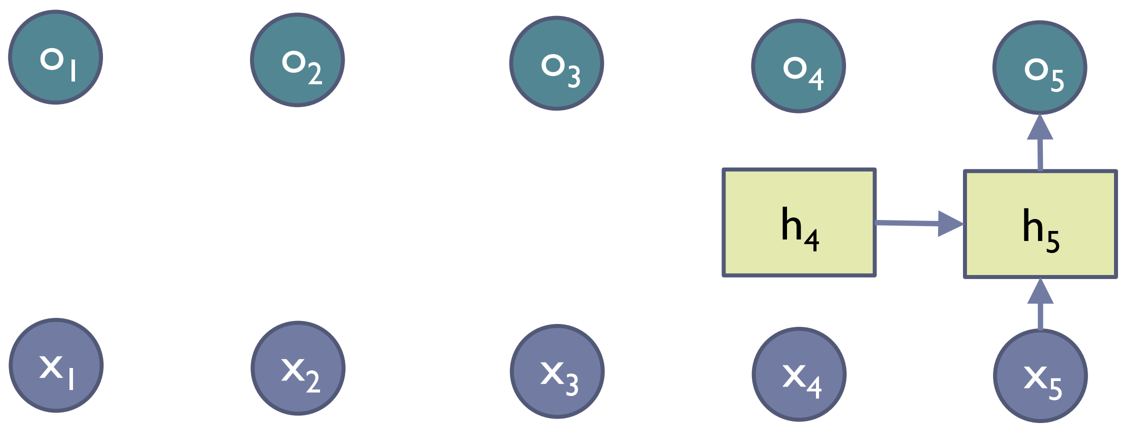


Recurrent Neural Networks

- **Connections form cycles**

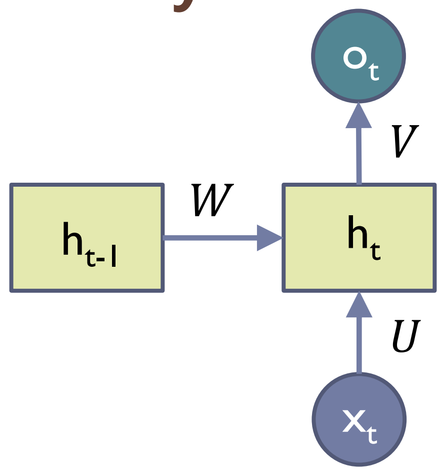


$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$

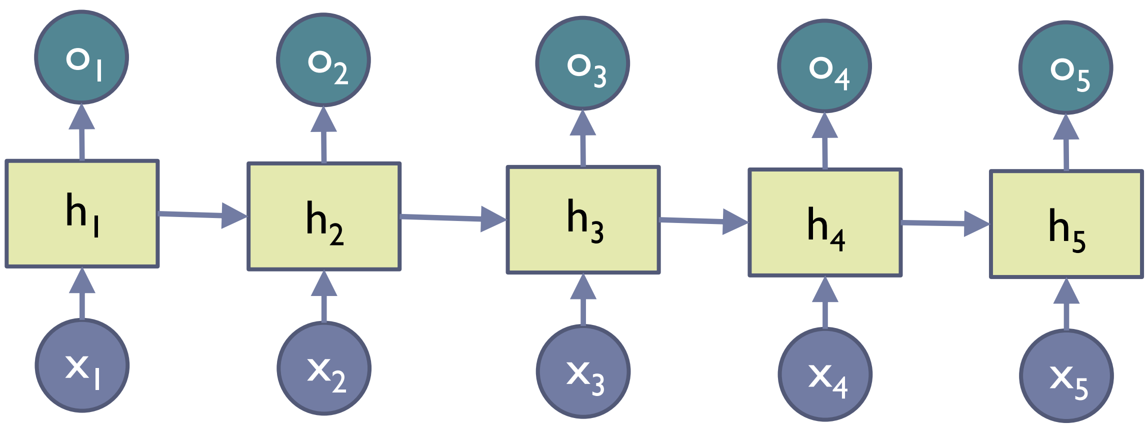


Recurrent Neural Networks

- **Connections form cycles**



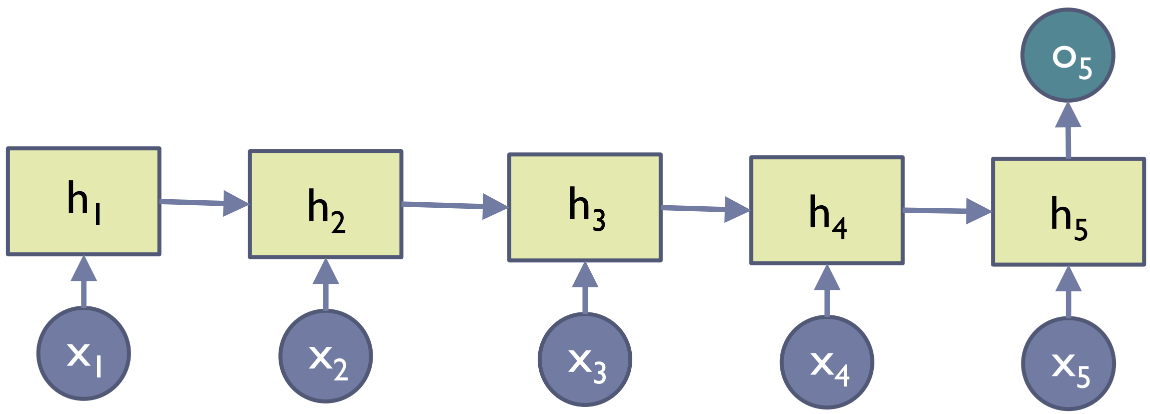
$$h_t = f(Ux_t + Wh_{t-1})$$
$$o_t = g(Vh_t)$$



Recurrent Neural Networks

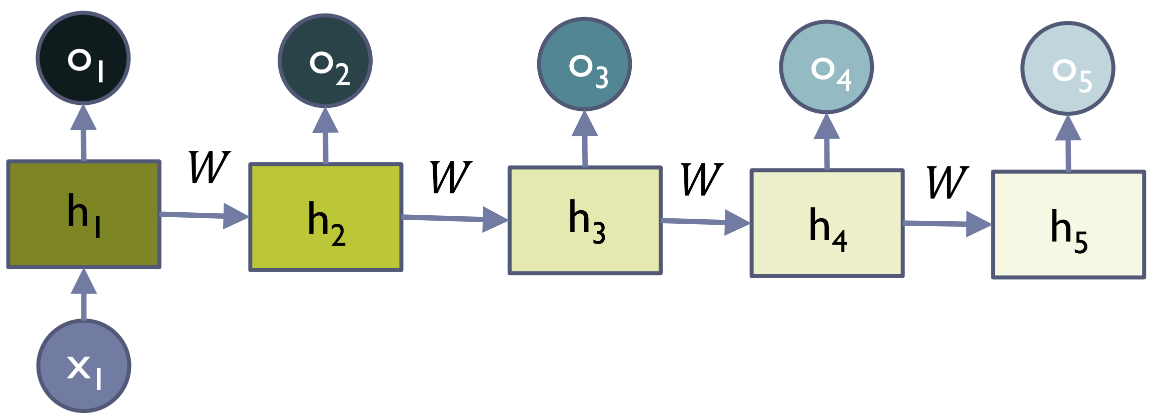
Long Term Dependency

- $x_1 \sim x_{t-1}$ are encoded into h_{t-1}
- h_{t-1} has the information on the past
- It is a context to process x_t



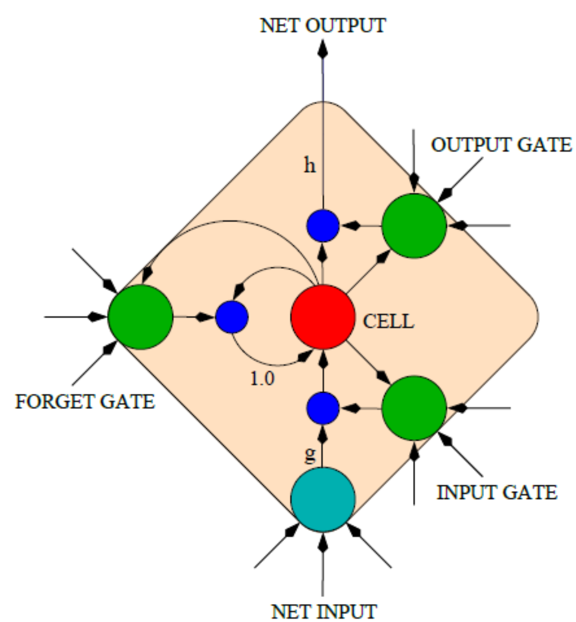
Recurrent Neural Networks

- **Long Term Dependency of Standard RNN**
  - However, it may exponentially decade or grow
  - Usually it is limited to 10 steps



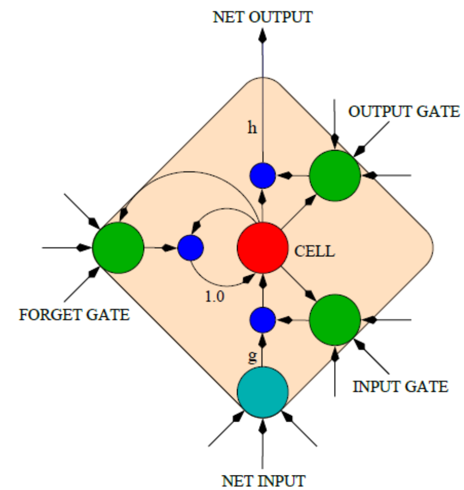
Long Short-Term Memory (LSTM)

- **Capable of learning long-term dependencies.**
  - **LSTM networks introduce a new structure called a memory cell**
    - An LSTM can learn to bridge time intervals in excess of 1000 steps
  - **Gate units that learn to open and close access to the past**
    - Input gate
    - Forget gate
    - Output gate
    - Neuron with a self-recurrent



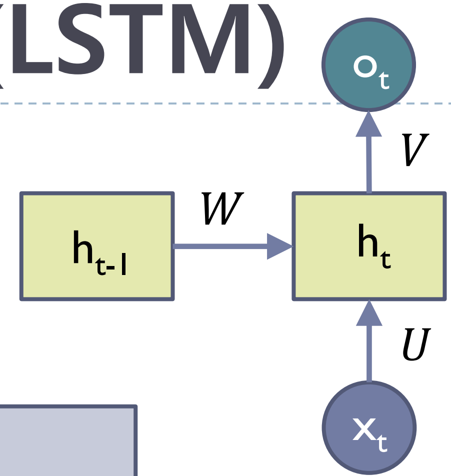
Long Short-Term Memory (LSTM)

- **Equations**
  - i: input gate to accept the new
  - f: forget gate to forget the past
  - o: output gate, how much of the information will be passed to expose to the next time step.
  - g: self-recurrent which is equal to standard RNN
  - c_t: internal memory
  - s_t: hidden state
  - y: final output

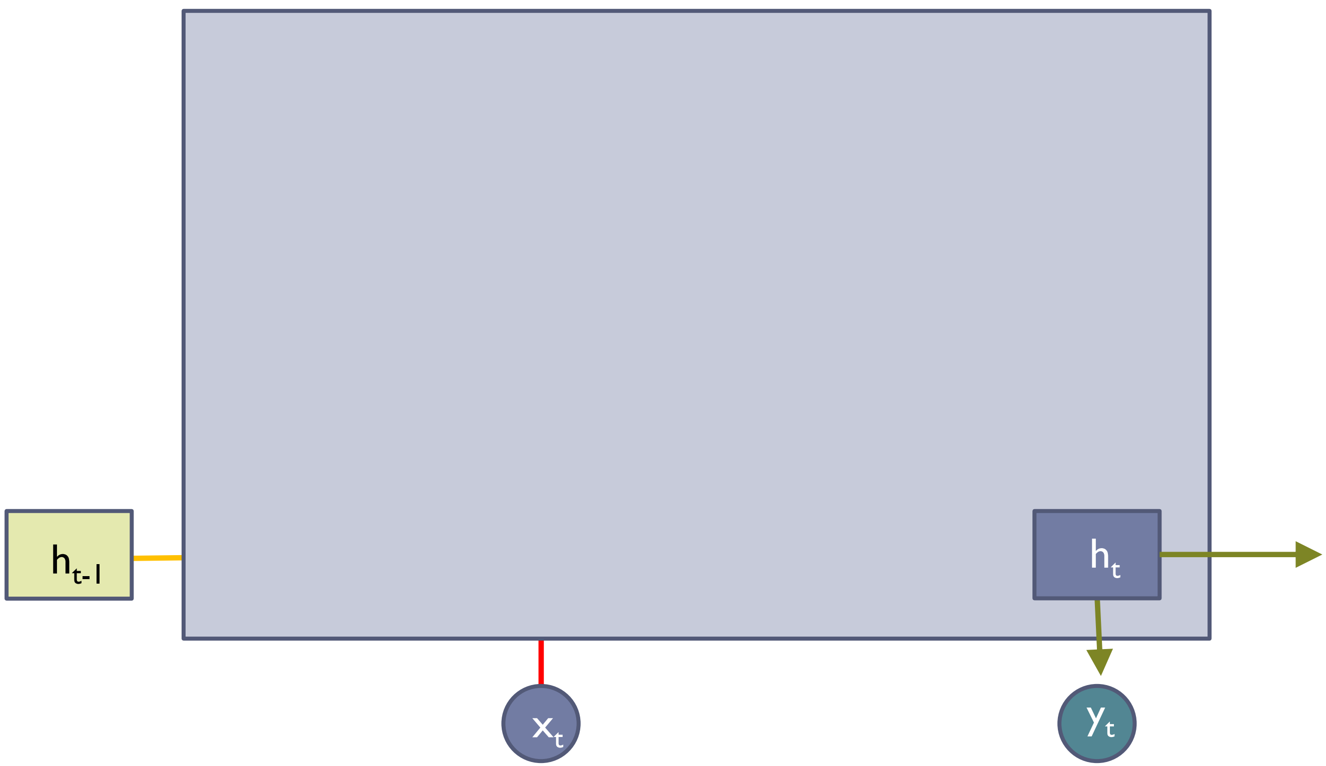


$$i = \sigma(x_t U^i + s_{t-1} W^i)$$
$$f = \sigma(x_t U^f + s_{t-1} W^f)$$
$$o = \sigma(x_t U^o + s_{t-1} W^o)$$
$$g = \tanh(x_t U^g + s_{t-1} W^g)$$
$$c_t = c_{t-1} \circ f + g \circ i$$
$$s_t = \tanh(c_t) \circ o$$
$$y = softmax(V s_t)$$

Long Short-Term Memory (LSTM)

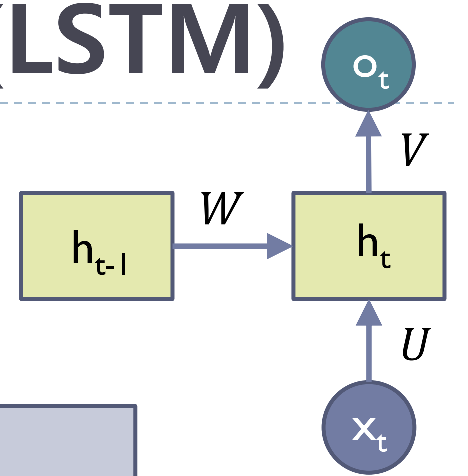


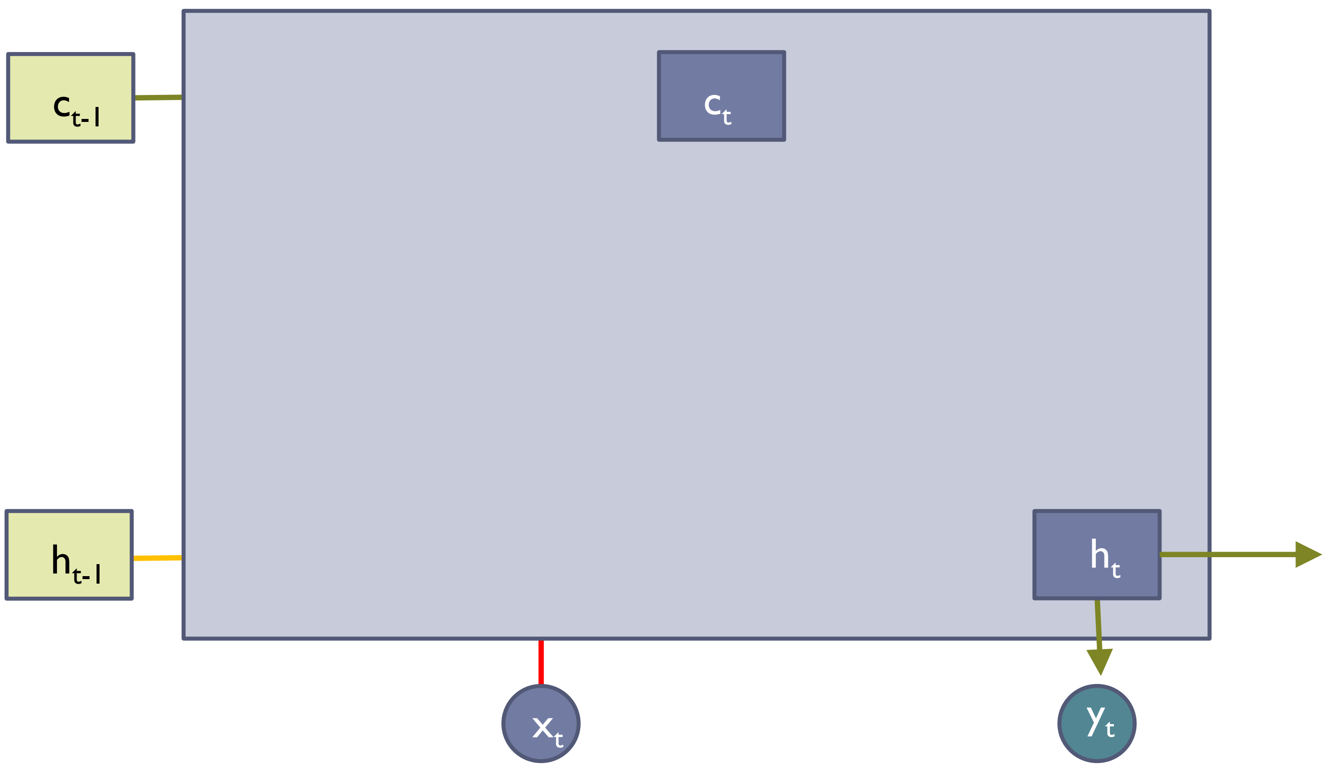
- **How to work**



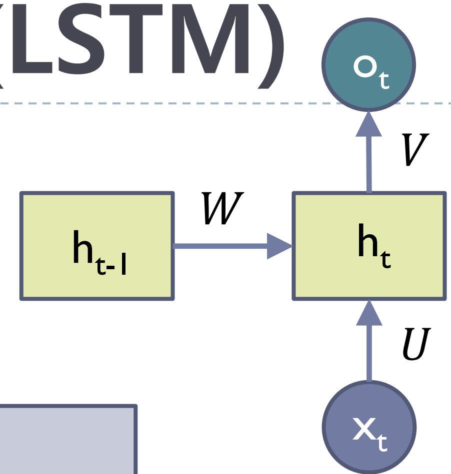
Long Short-Term Memory (LSTM)

- ## How to work

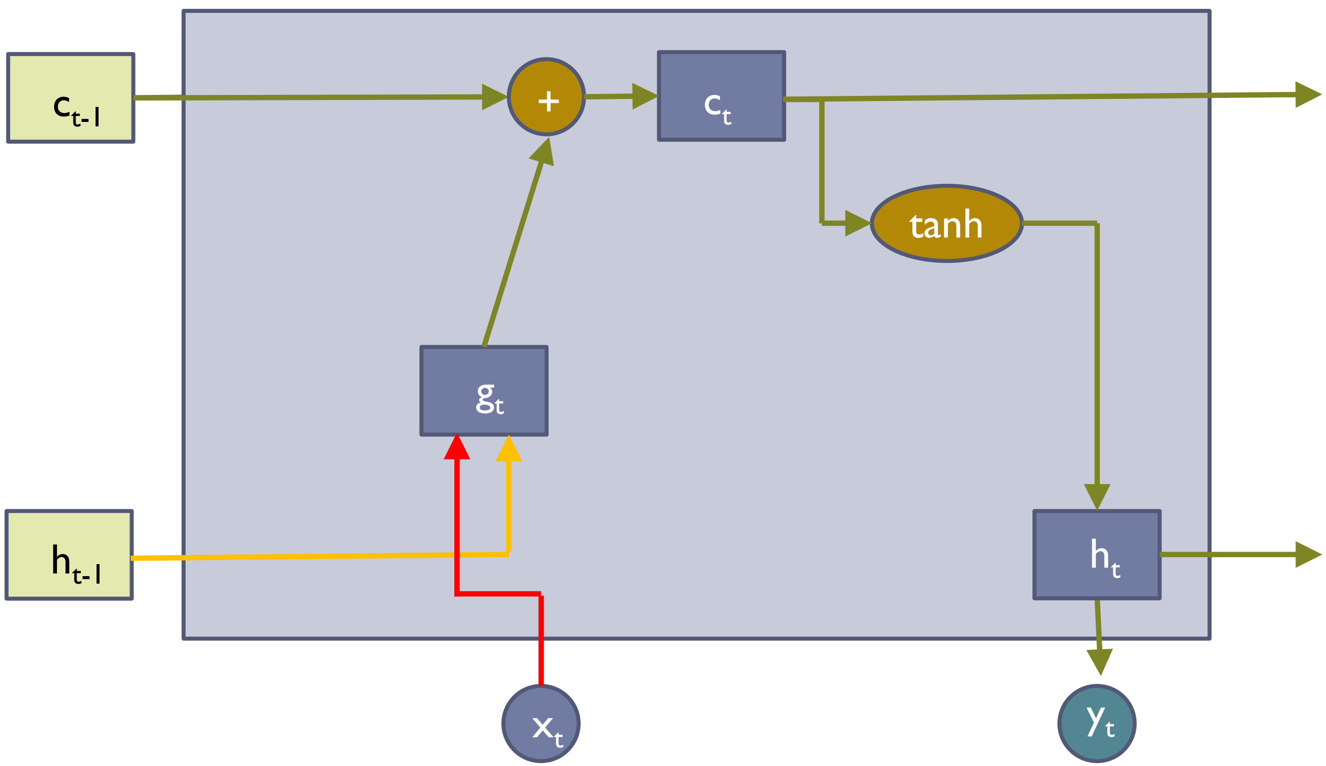




Long Short-Term Memory (LSTM)

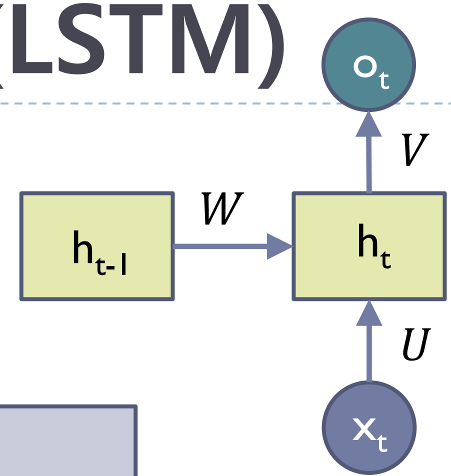


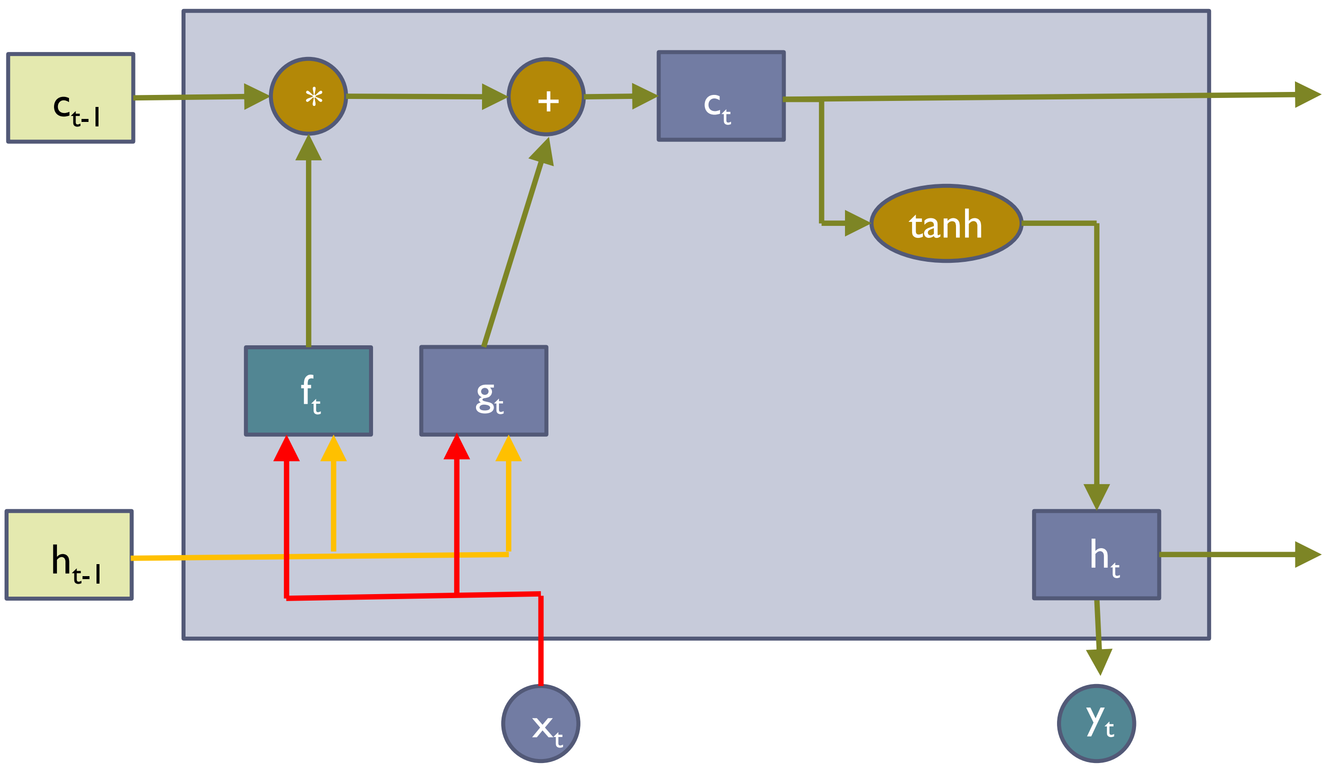
- How to work



Long Short-Term Memory (LSTM)

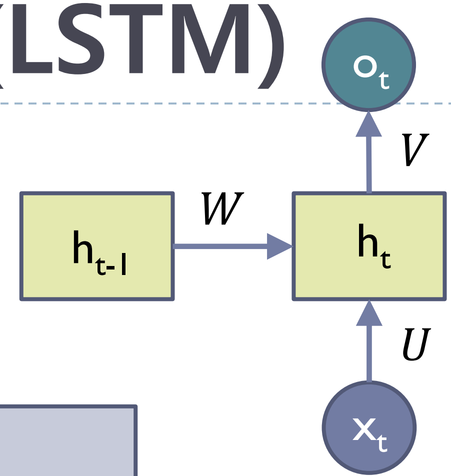
- How to work

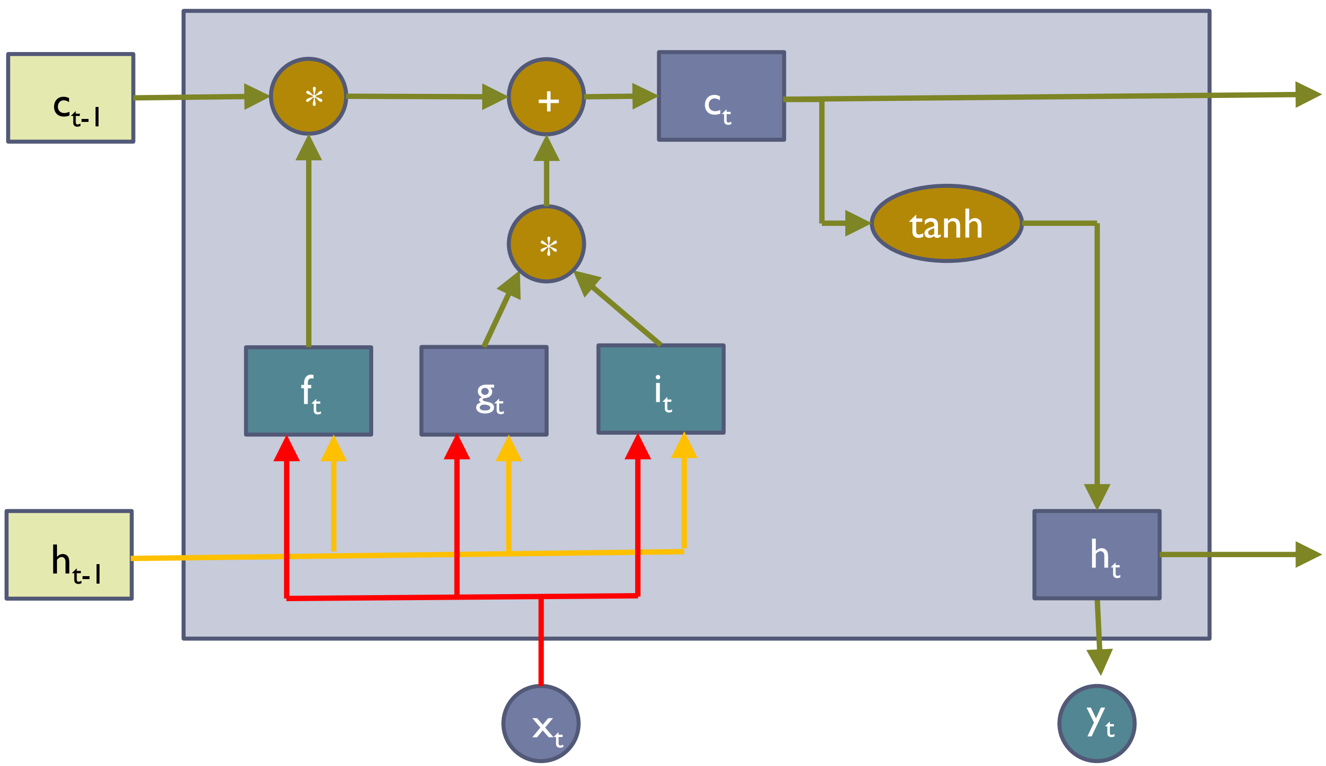




Long Short-Term Memory (LSTM)

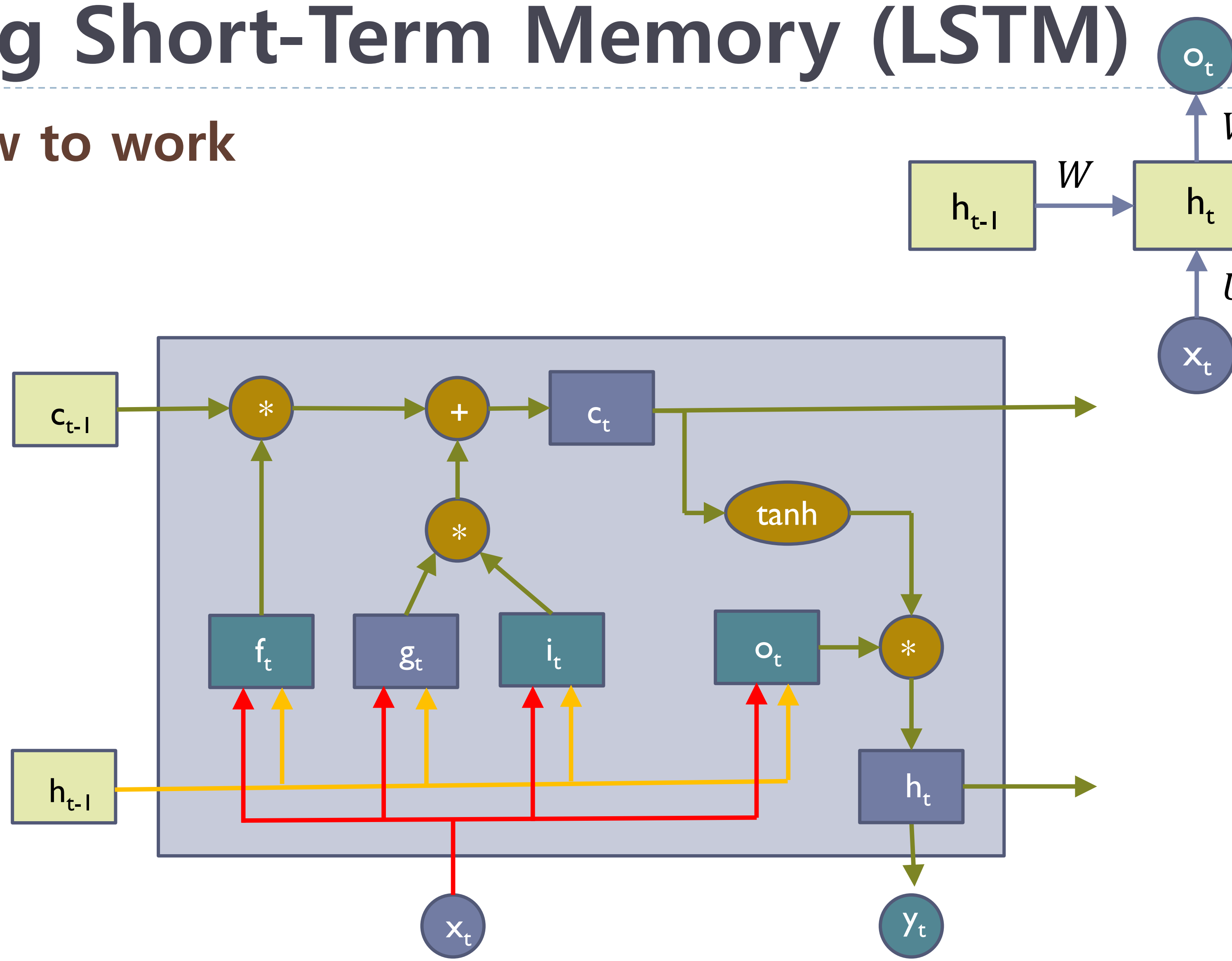
- **How to work**





Long Short-Term Memory (LSTM)

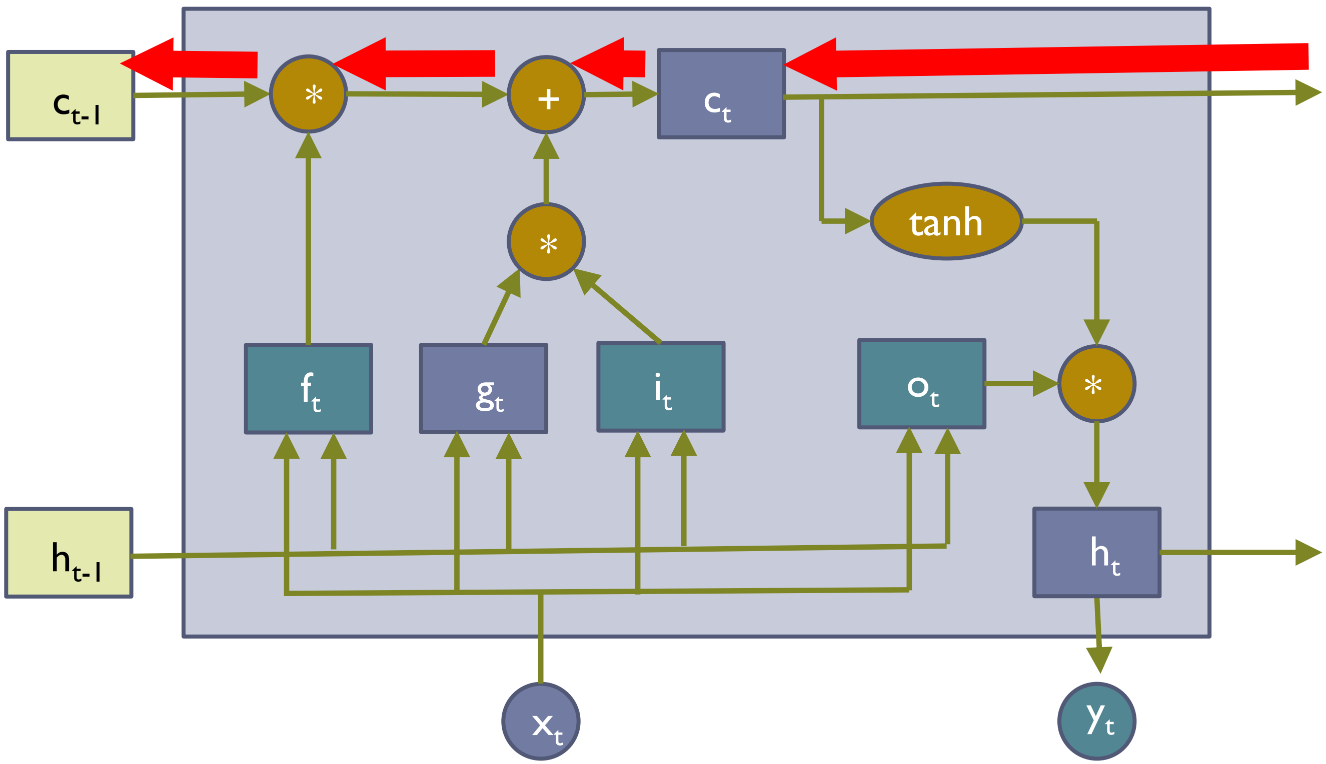
- **How to work**



Long Short-Term Memory (LSTM)

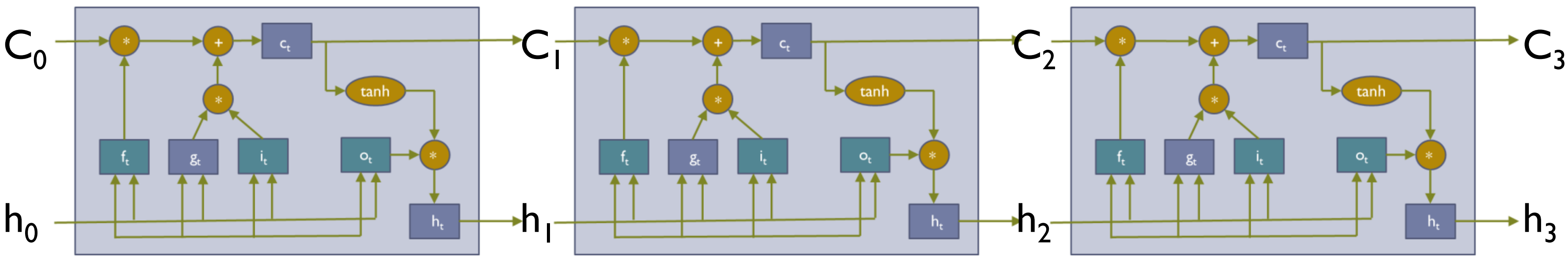
Gradient Flow

Backpropagation from c_t to c_{t-1} only elementwise multiplication by f, no matrix multiply by W



Long Short-Term Memory (LSTM)

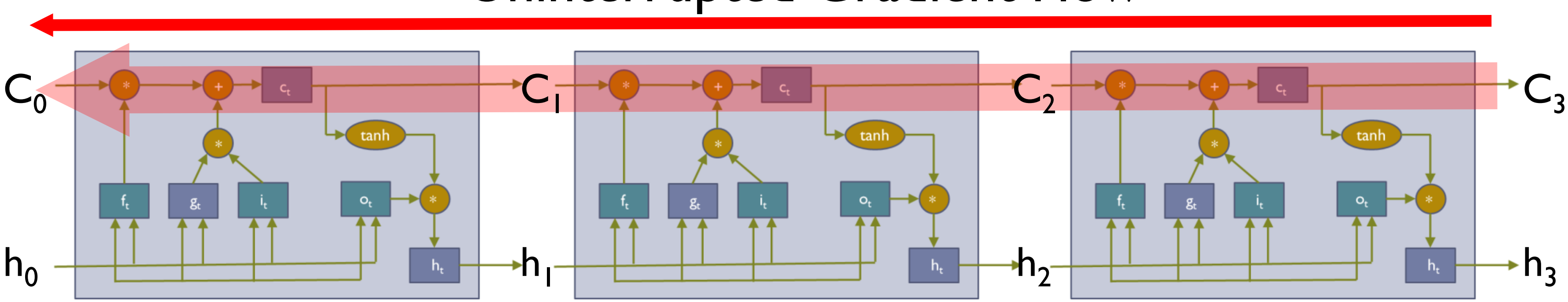
- **Gradient Flow**



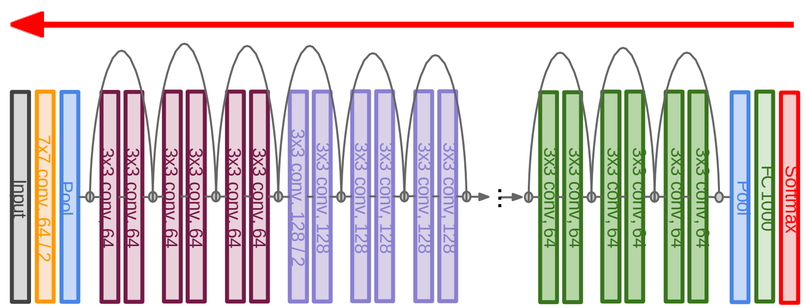
Long Short-Term Memory (LSTM)

- **Gradient Flow**

Uninterrupted Gradient Flow



Similar to ResNet!



In between:
Highway Networks

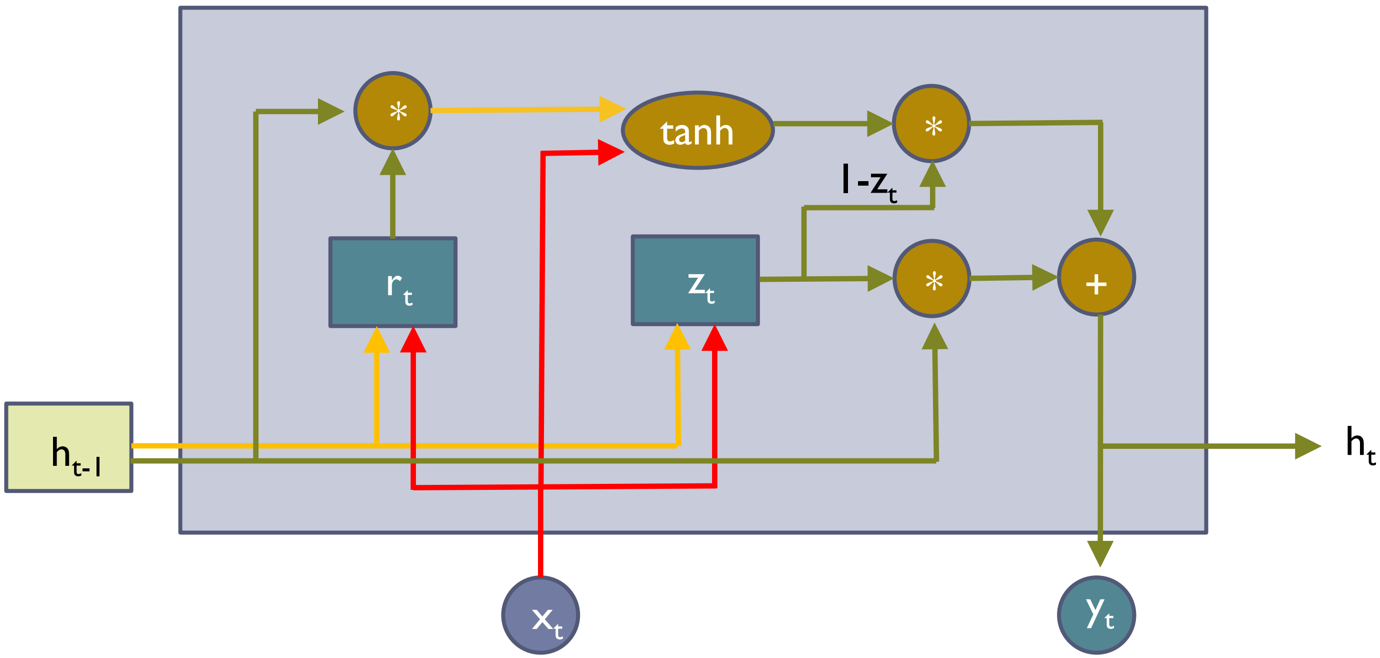
$$g = T(x, W_T)$$
$$y = g \odot H(x, W_H) + (1 - g) \odot x$$

Srivastava et al, "Highway Networks", ICML DL Workshop 2015

Other Variants

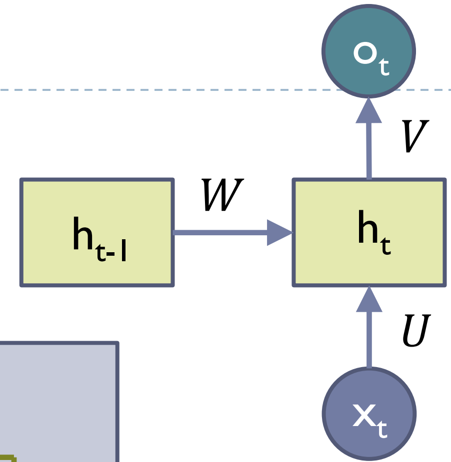
- **GRU**

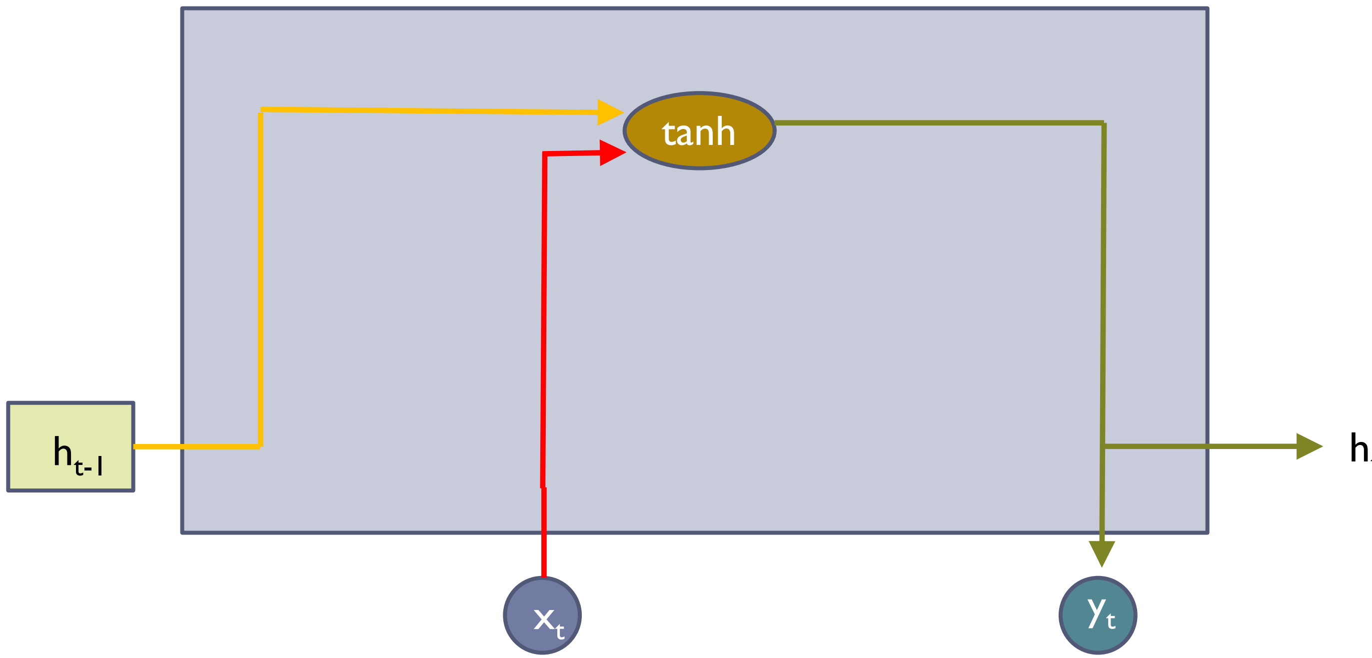
$$r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r)$$
$$z_t = \sigma(W_{xz}x_t + W_{hz}h_{t-1} + b_z)$$
$$\tilde{h}_t = \tanh(W_{xh}x_t + W_{hh}(r_t \odot h_{t-1}) + b_h)$$
$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t$$



Other Variants

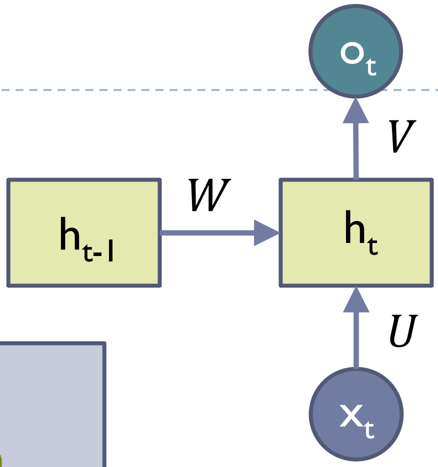
- **GRU**

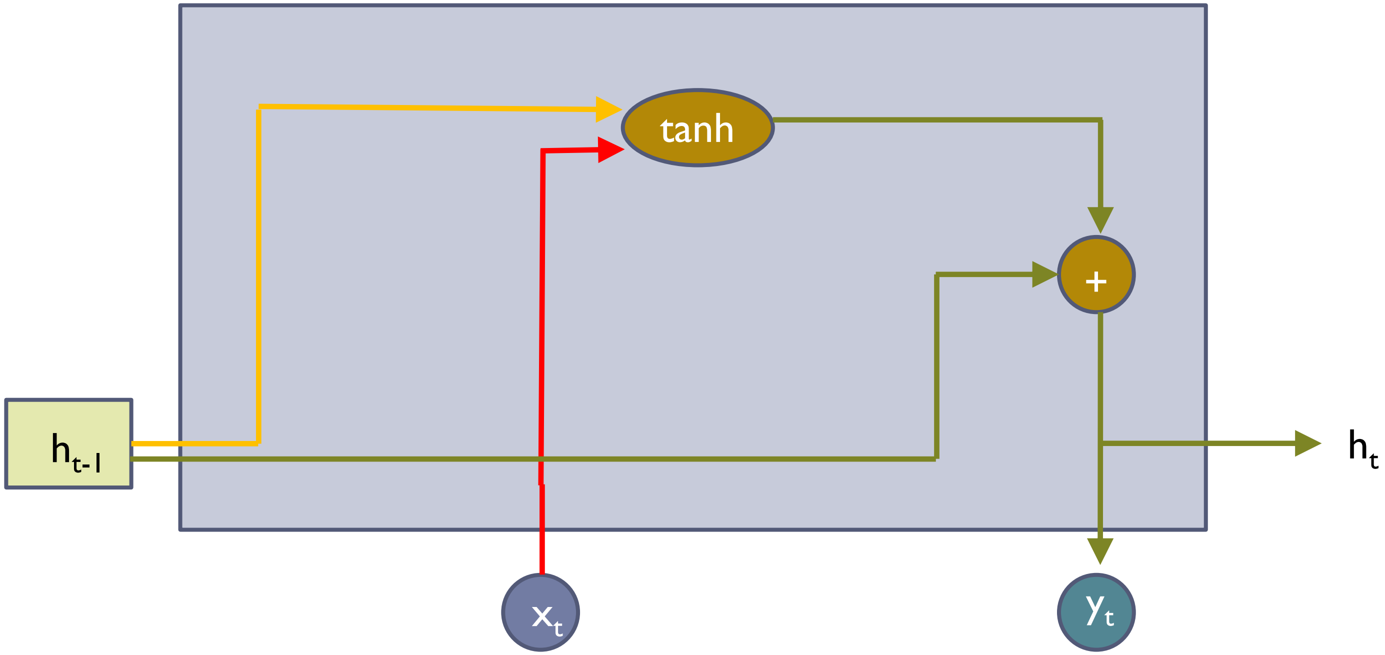




Other Variants

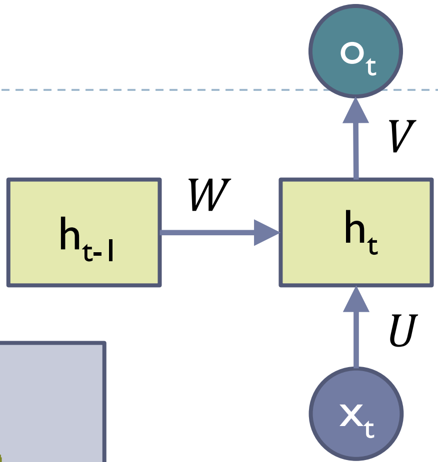
- **GRU**

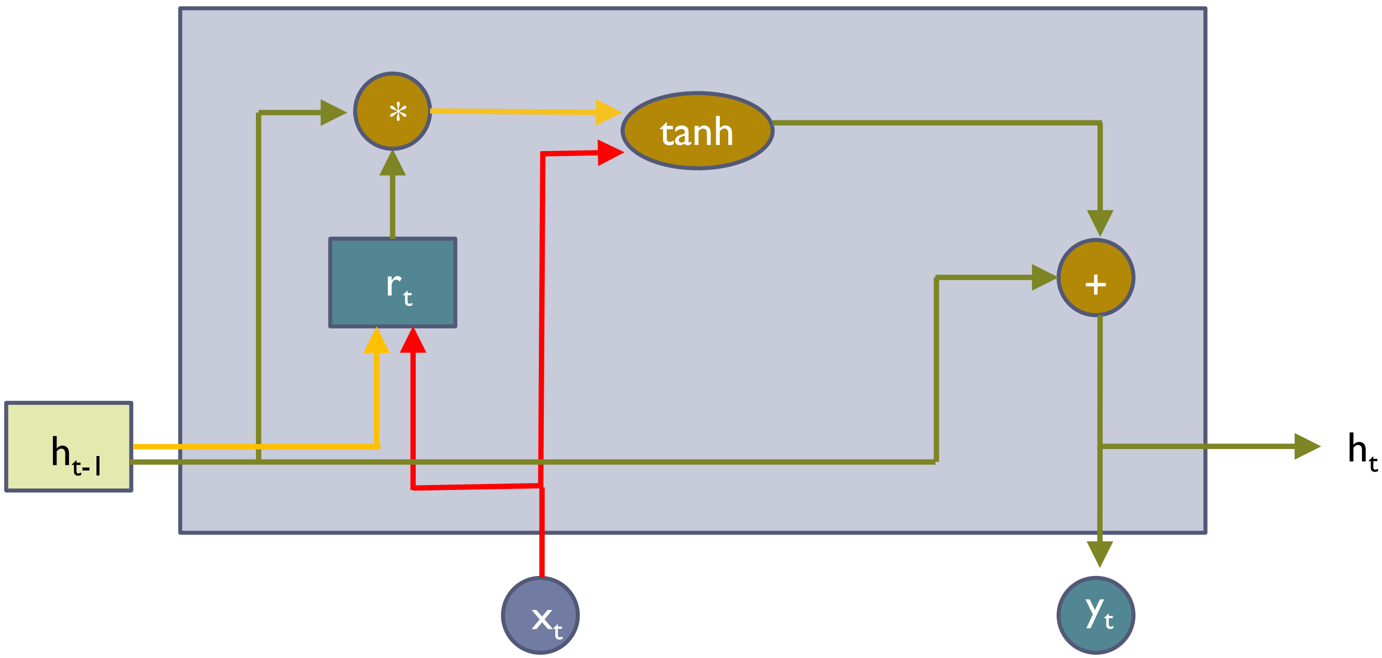




Other Variants

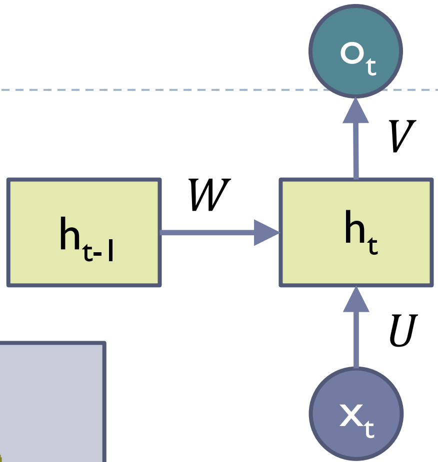
- **GRU**

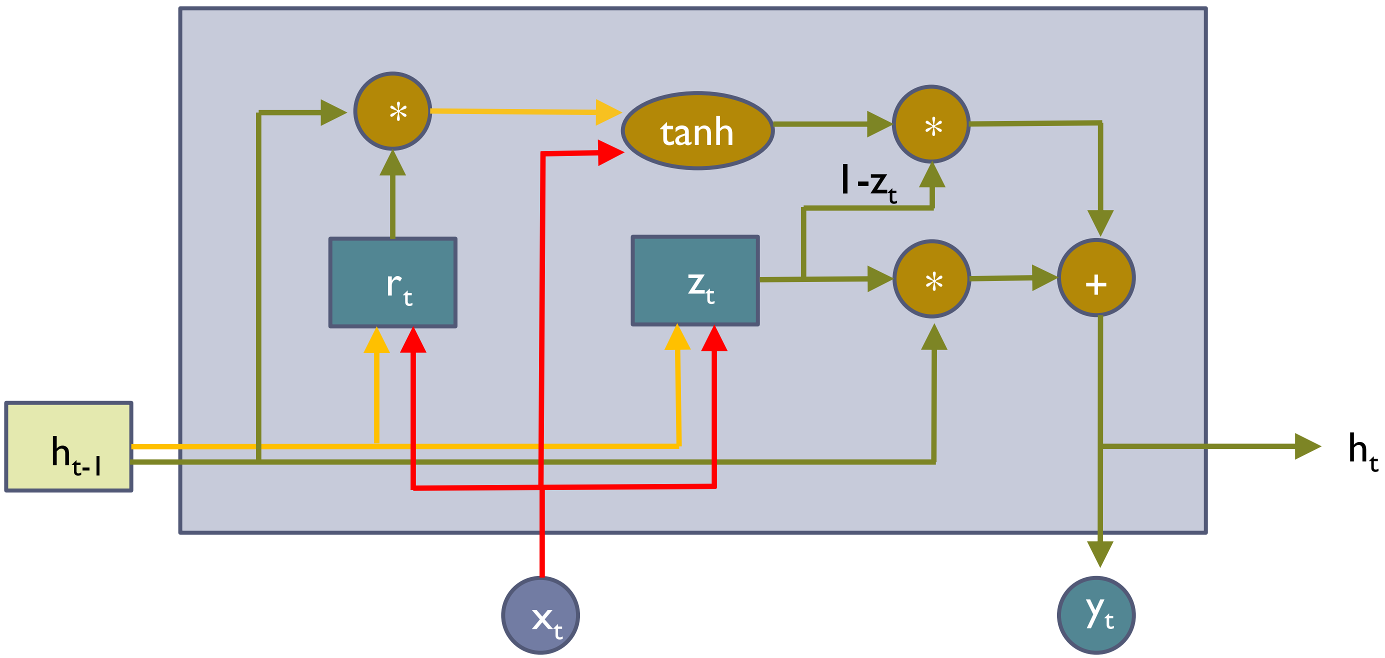




Other Variants

- GRU





Question and Answer