Recurrent Neural Networks

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, GRU**
    - Try to capture the feature of the past

Sequential Data Processing

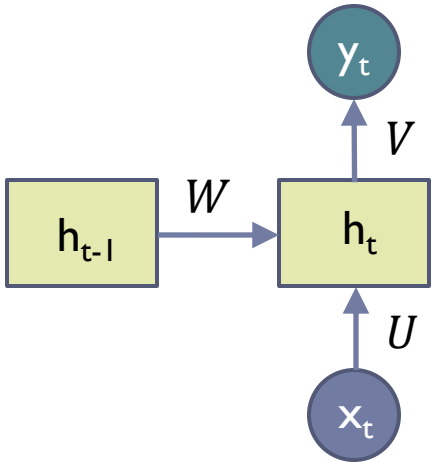
- What is sequential data?

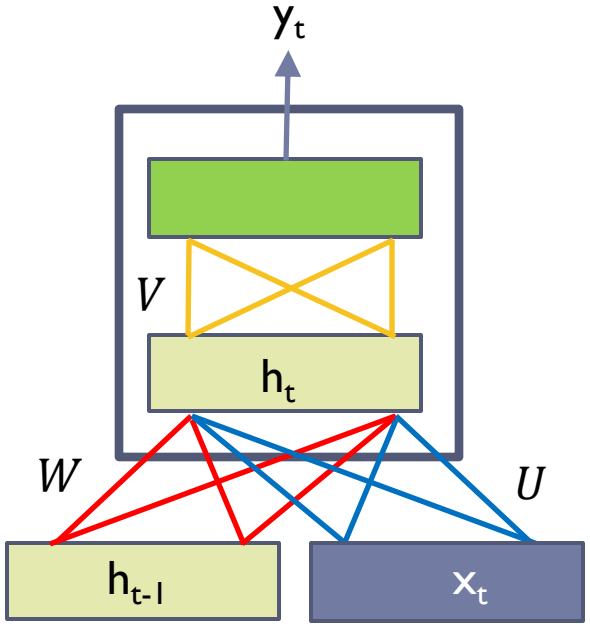
- What do we have to consider for sequential data processing?

Recurrent Neural Networks

- **Connections form cycles**

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





- 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**

Training Data

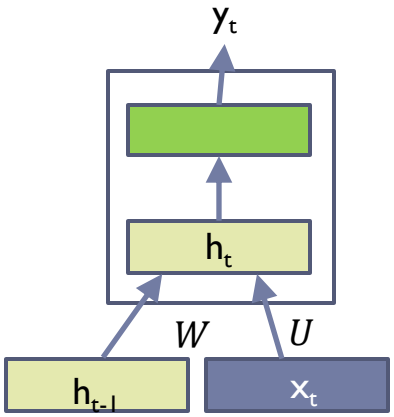
(x_1, y_1)

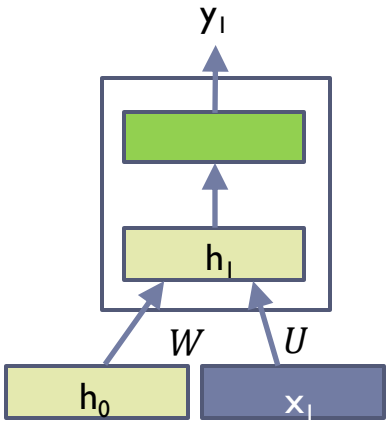
(x_2, y_2)

(x_3, y_3)

(x_4, y_4)

(x_5, y_5)





Recurrent Neural Networks

- Connections form cycles

Training Data

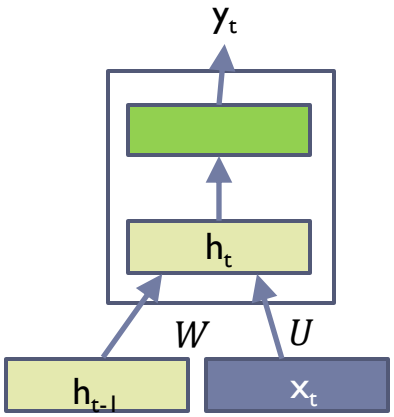
(x_1, y_1)

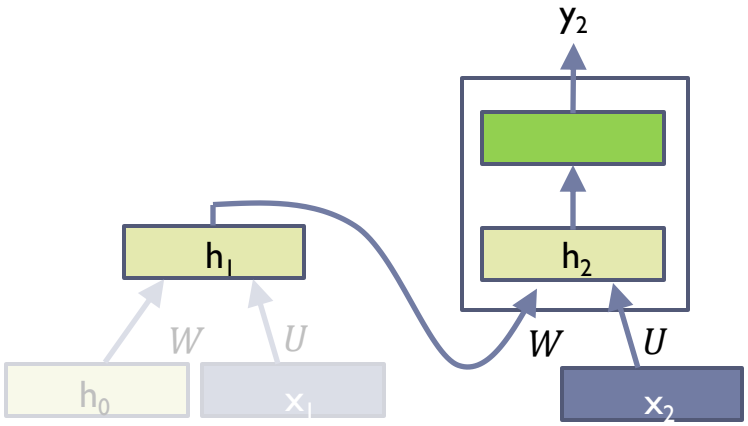
(x_2, y_2)

(x_3, y_3)

(x_4, y_4)

(x_5, y_5)



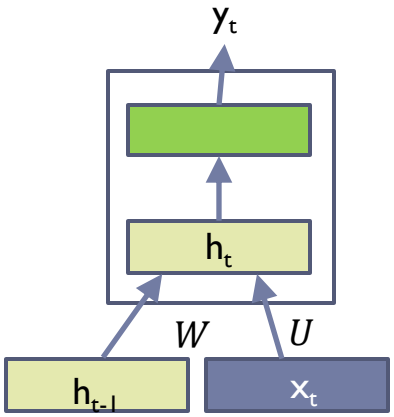


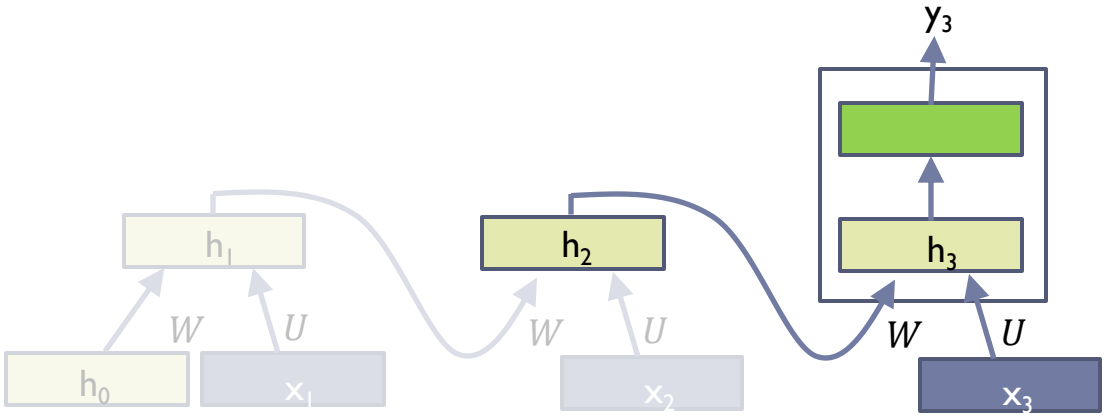
Recurrent Neural Networks

- **Connections form cycles**

Training Data

(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_5, y_5)



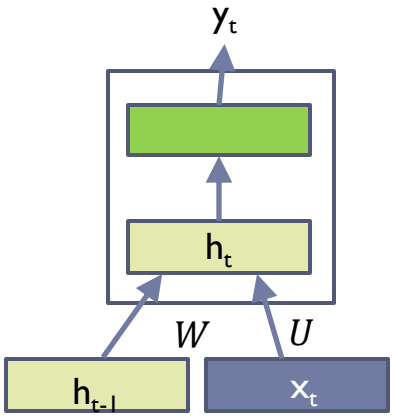


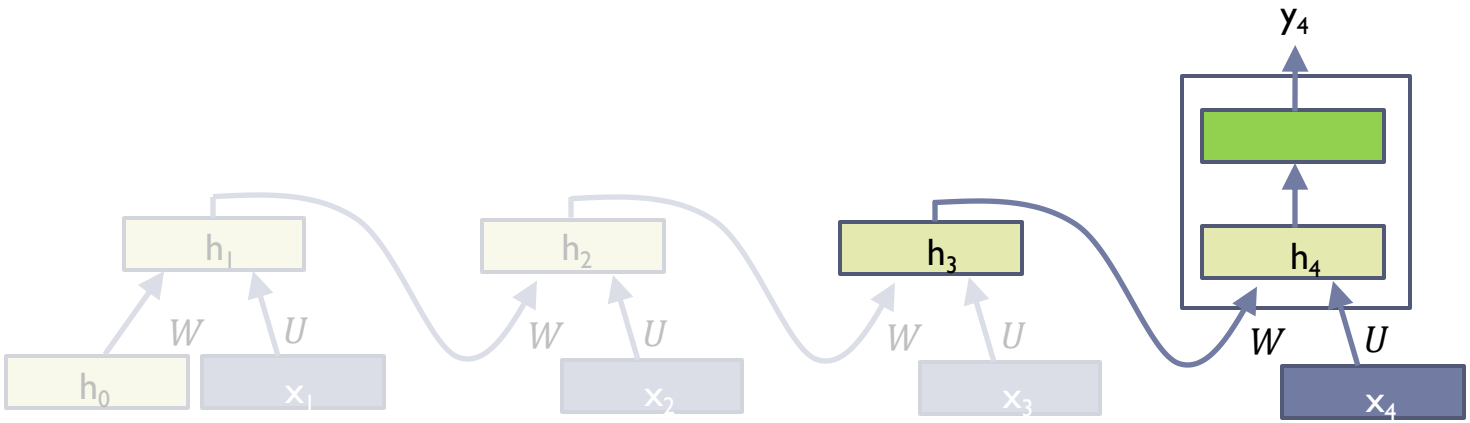
Recurrent Neural Networks

- **Connections form cycles**

Training Data

(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_5, y_5)



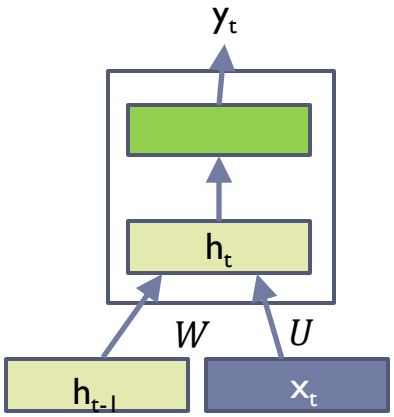


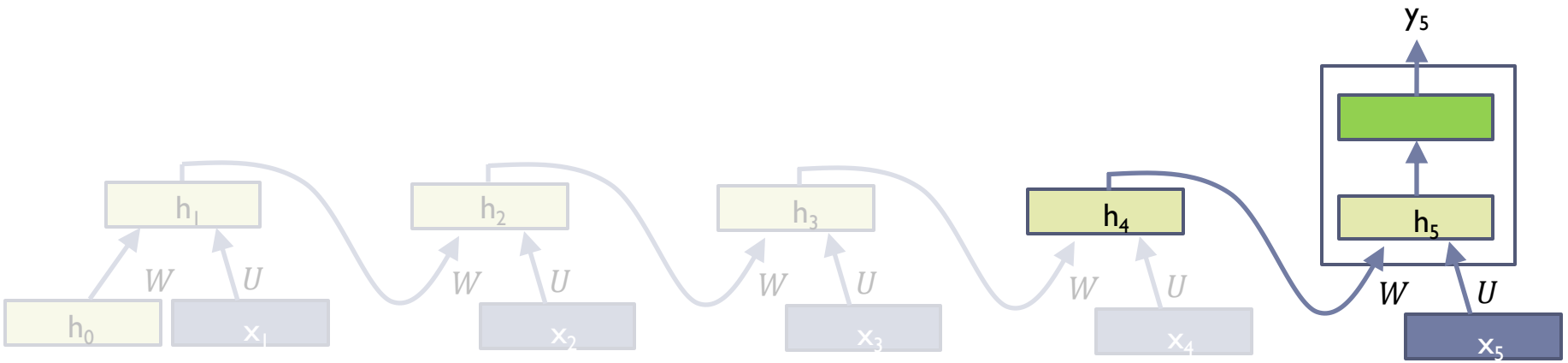
Recurrent Neural Networks

- **Connections form cycles**

Training Data

(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_5, y_5)



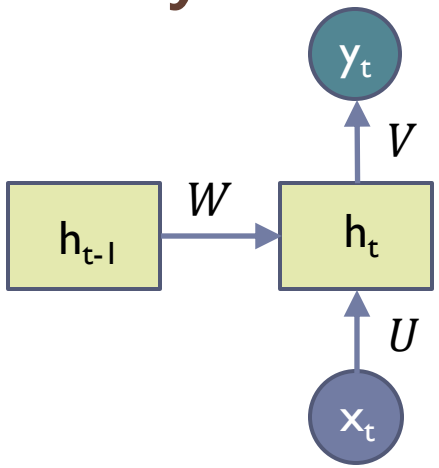


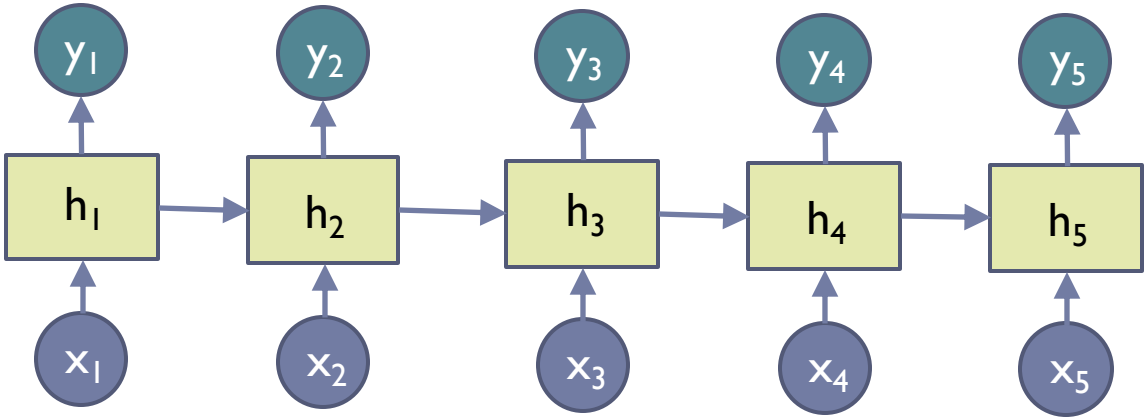
Recurrent Neural Networks

- **Connections form cycles**

Training Data

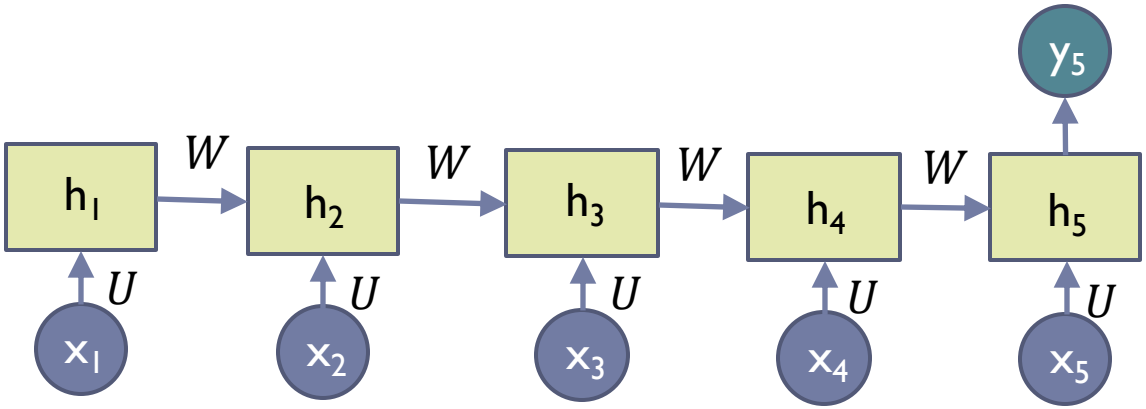
(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_5, y_5)





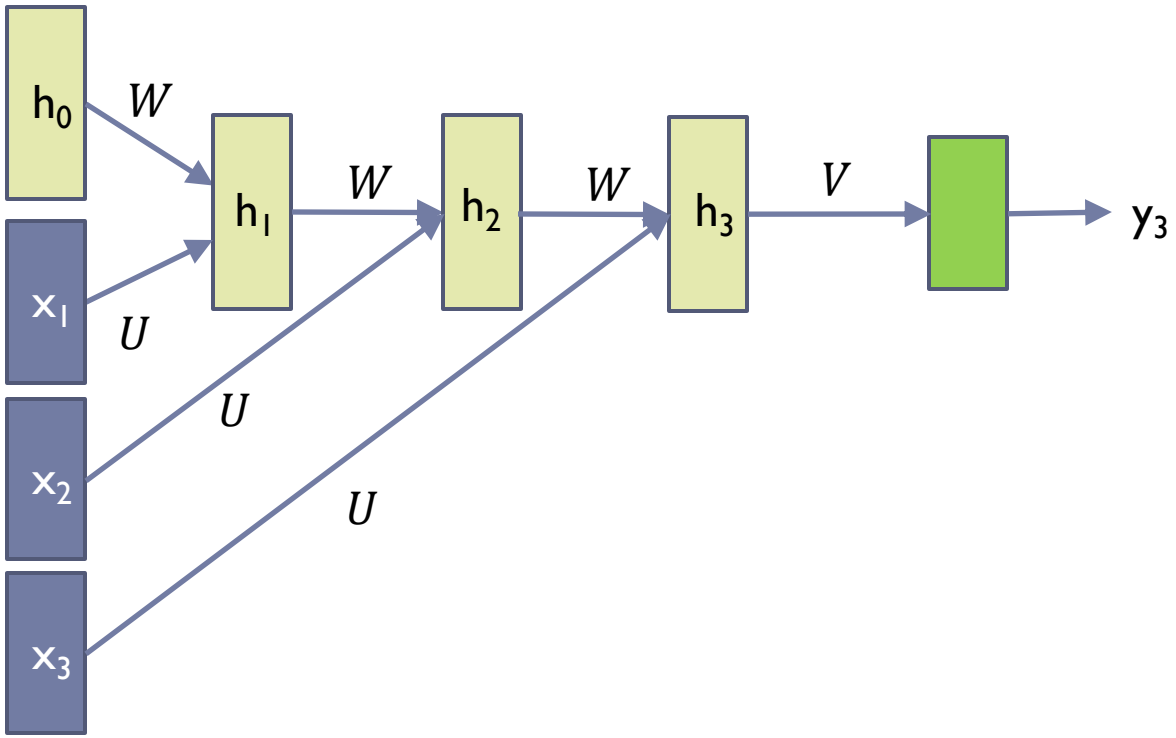
Recurrent Neural Networks

- **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



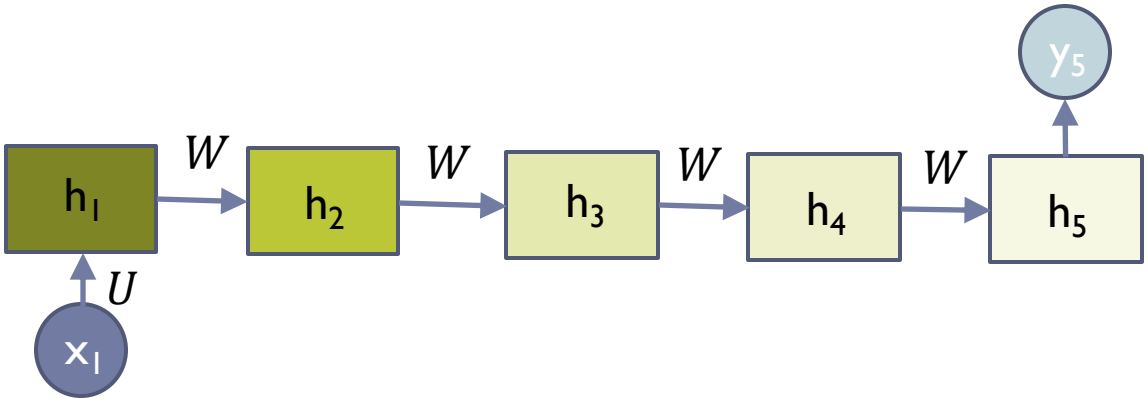
Recurrent Neural Networks

- **Fully-Connected NN**



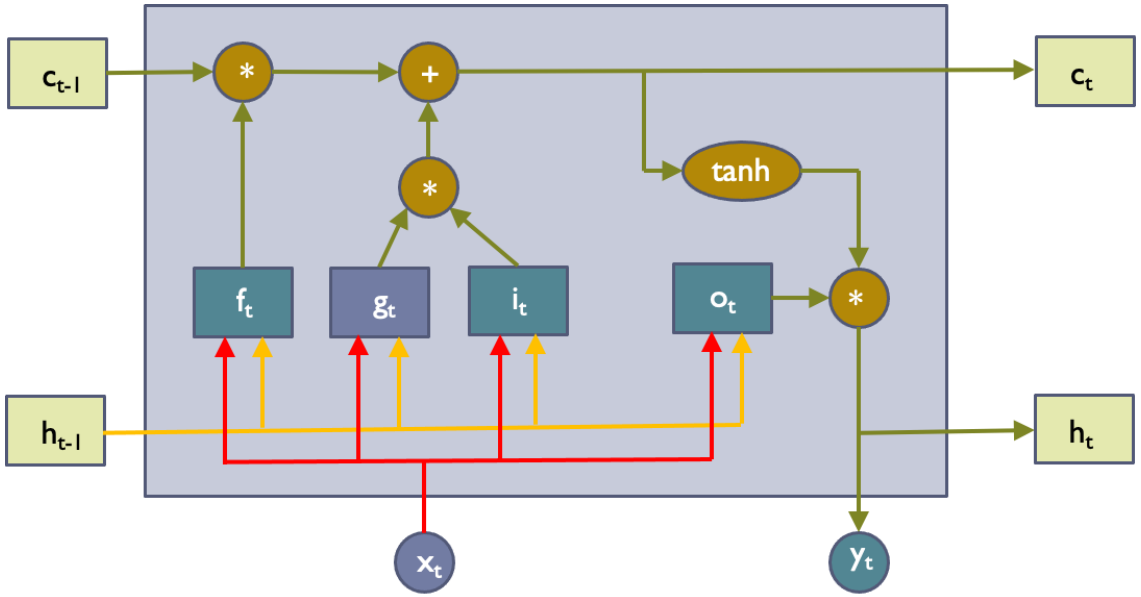
Recurrent Neural Networks

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



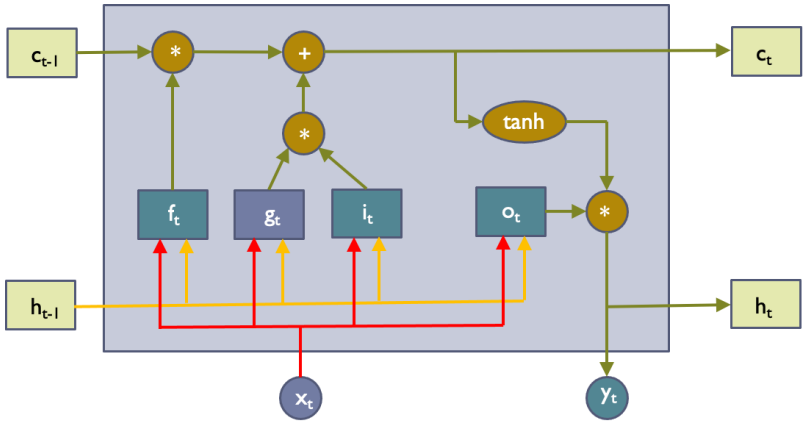
Long Short-Term Memory (LSTM)

- **Capable of learning long-term dependencies.**
  - An LSTM can learn to bridge time intervals of 1000 steps
  - Gate units that learn to open and close access to the past



Long Short-Term Memory (LSTM)

- **Equations**
  - i: input gate
  - f: forget gate
  - o: output gate
  - g: self-recurrent
  - c_t: internal memory
  - h_t: hidden state
  - y: final output



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

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

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

$$g = \tanh(x_t U^g + h_{t-1} W^g)$$

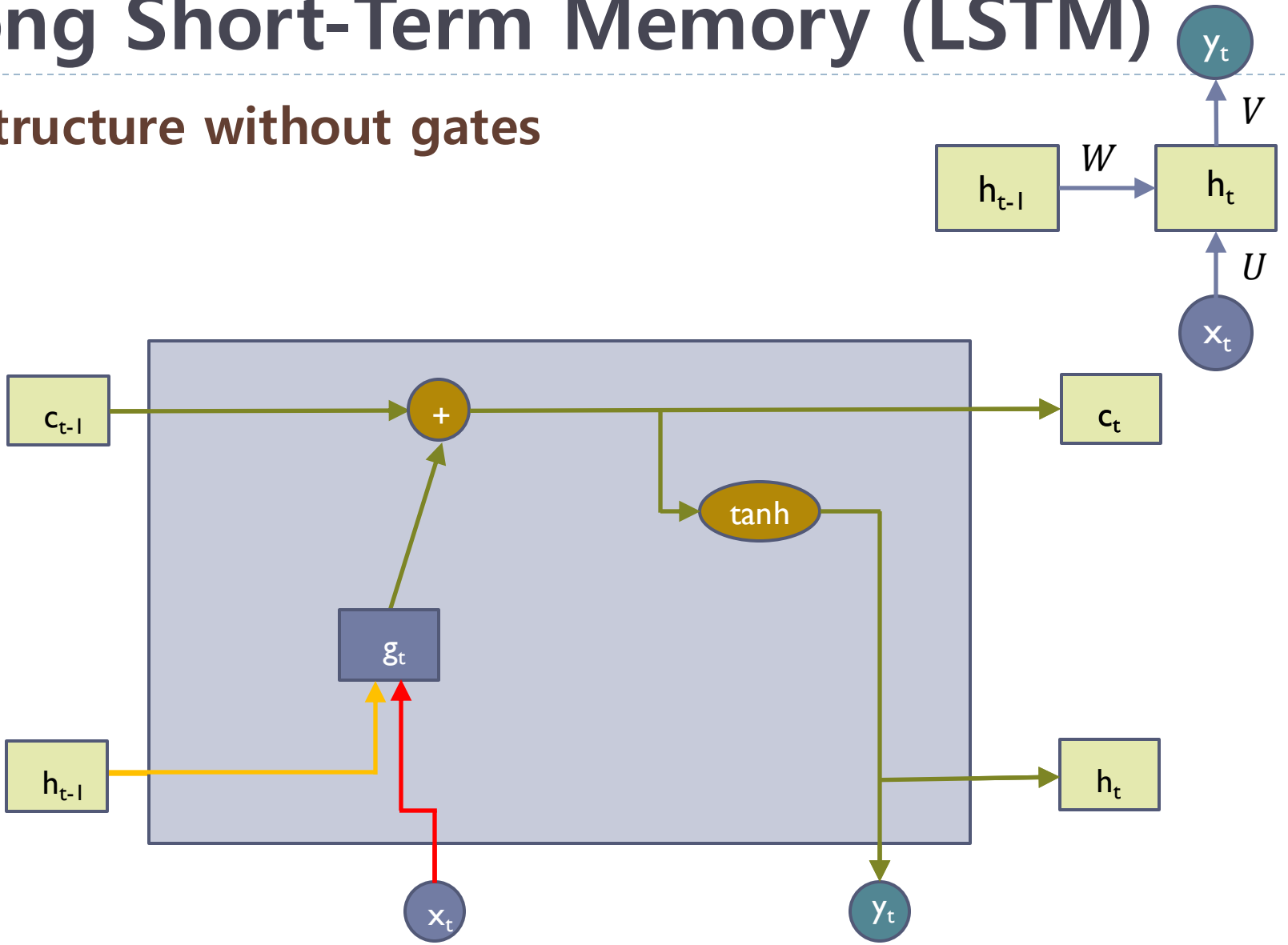
$$c_t = c_{t-1} \circ f + g \circ i$$

$$h_t = \tanh(c_t) \circ o$$

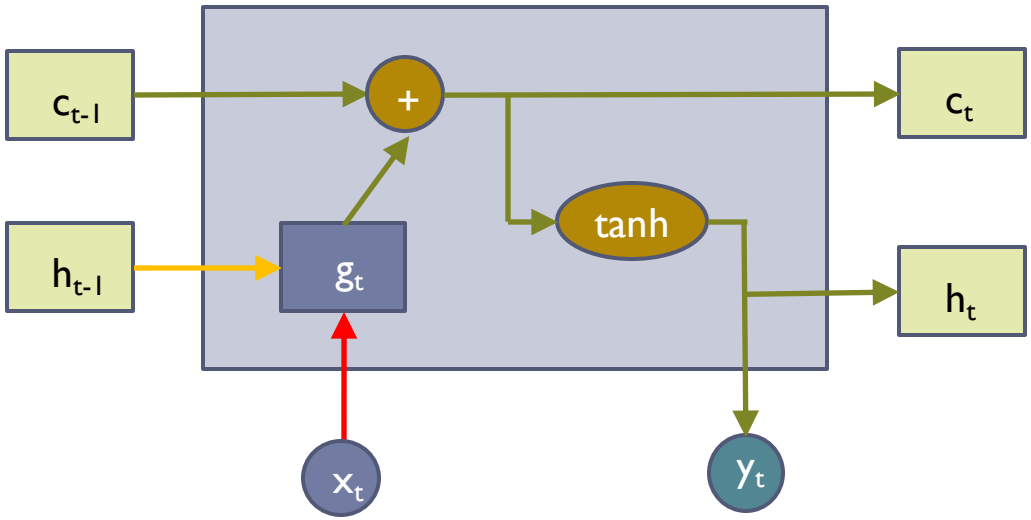
$$y = softmax(V h_t)$$

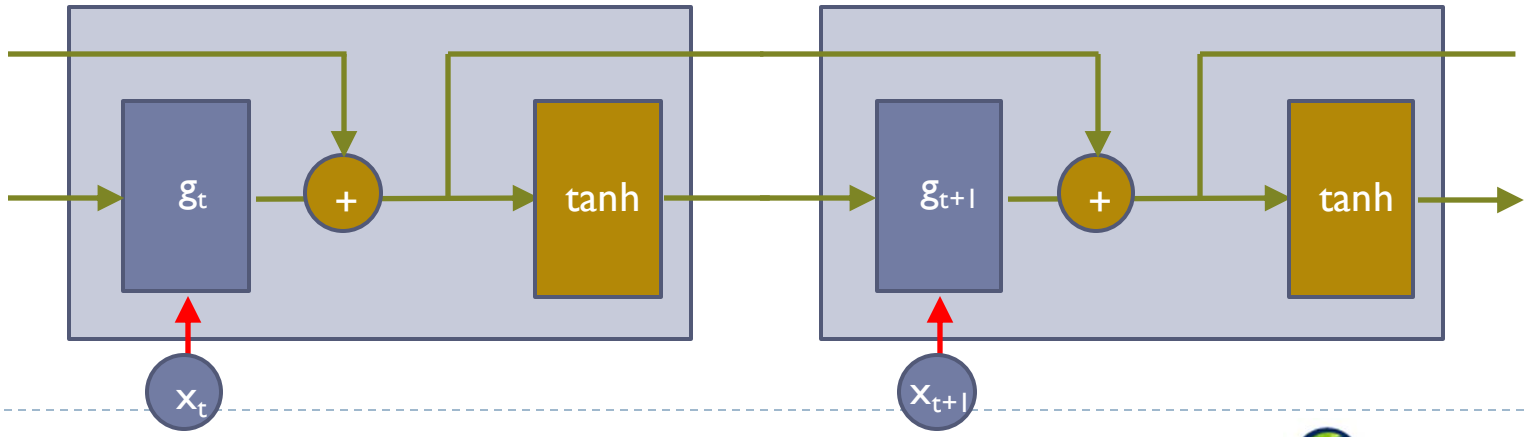
Long Short-Term Memory (LSTM)

- **Structure without gates**



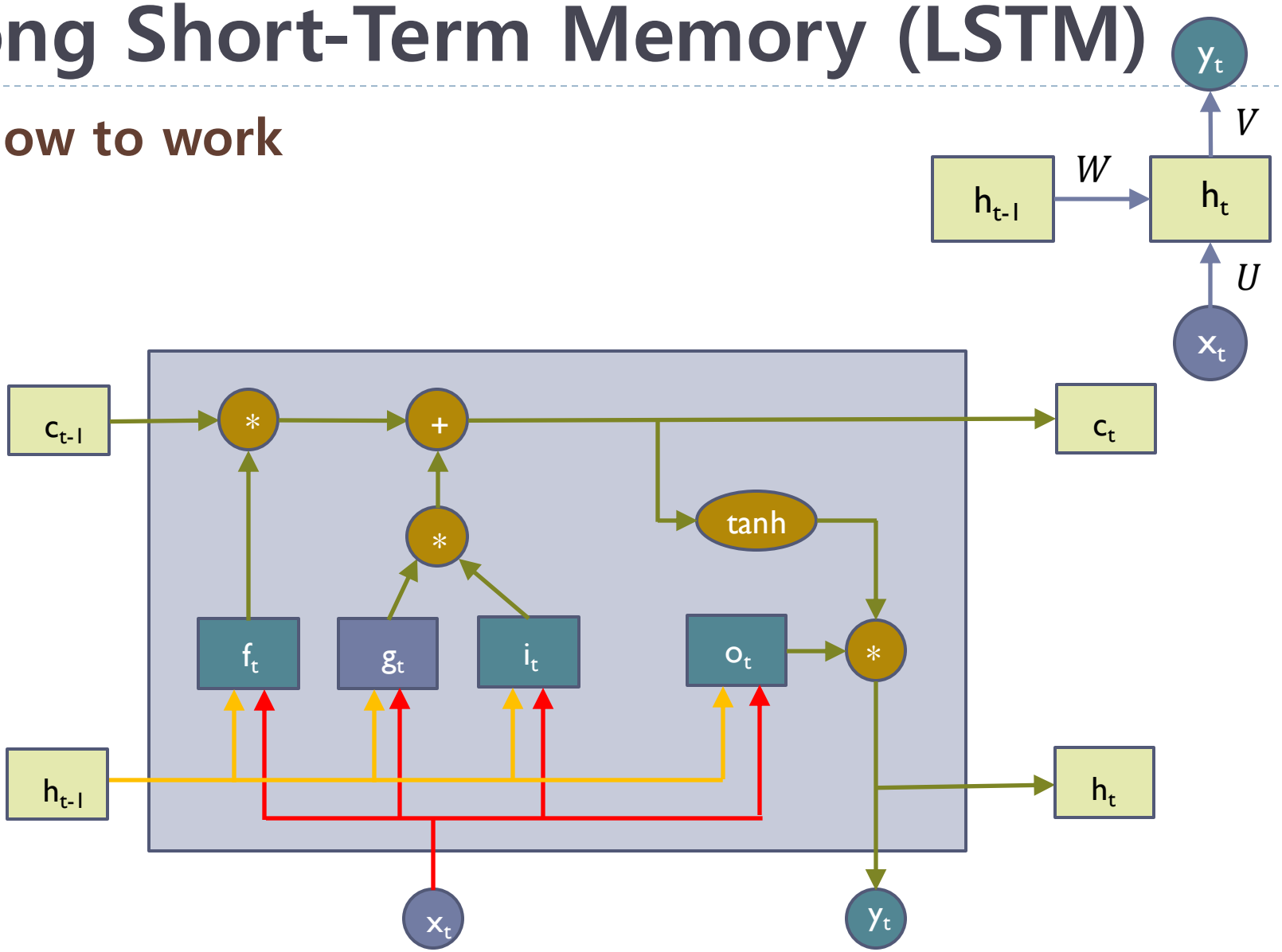
Long Short-Term Memory (LSTM)





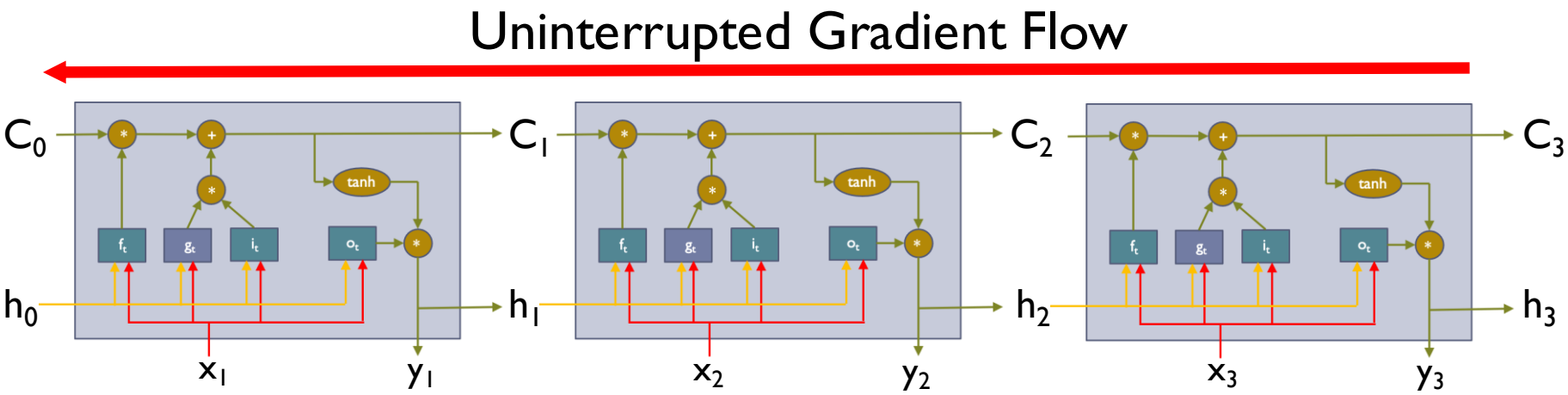
Long Short-Term Memory (LSTM)

- How to work

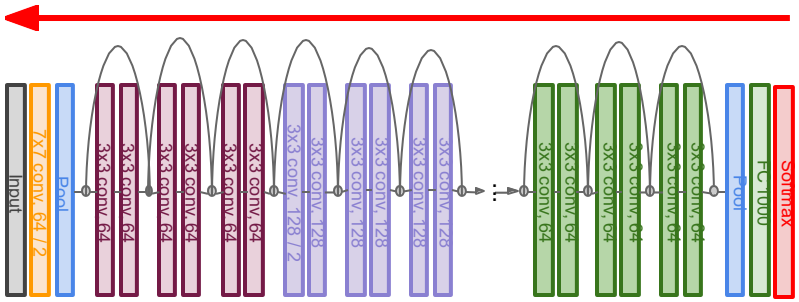


Long Short-Term Memory (LSTM)

- **Gradient Flow**



Similar to ResNet!



Sequence Processing

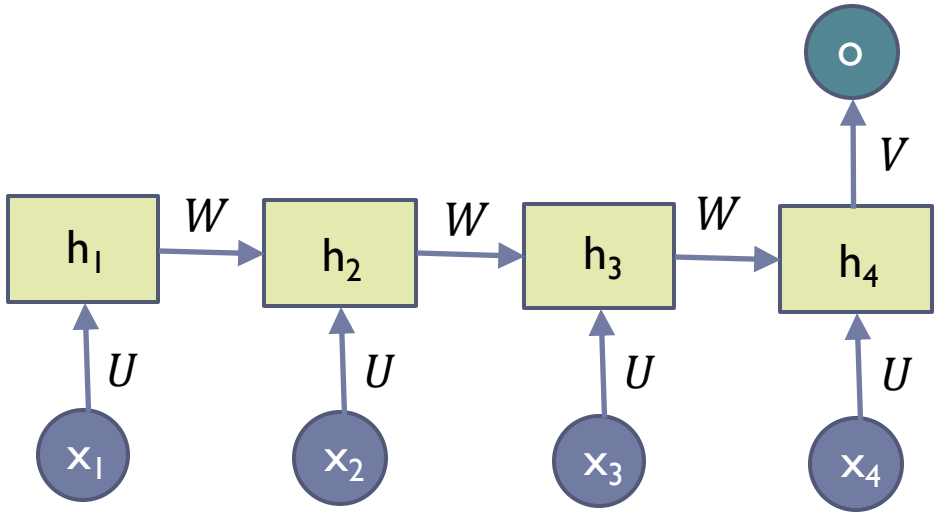
- **Training Data**
  - **Usually, samples are preprocessed in a fixed length**

$$
\begin{array}{l}
(x_1, y_1) \\
(x_2, y_2) \\
(x_3, y_3) \\
(x_4, y_4) \\
(x_5, y_5) \\
(x_6, y_6) \\
(x_7, y_7) \\
(x_8, y_8) \\
\dots
\end{array}
\quad \Rightarrow \quad
\begin{array}{l}
(x_1 x_2 x_3 x_4, y_4) \\
(x_2 x_3 x_4 x_5, y_5) \\
(x_3 x_4 x_5 x_6, y_6) \\
(x_4 x_5 x_6 x_7, y_7) \\
(x_5 x_6 x_7 x_8, y_8) \\
\dots
\end{array}
$$

Sequence Processing

- **Training**
  - Samples are trained with a fixed length of RNN

$(x_1 x_2 x_3 x_4, y_4)$
$(x_2 x_3 x_4 x_5, y_5)$
$(x_3 x_4 x_5 x_6, y_6)$
$(x_4 x_5 x_6 x_7, y_7)$
$(x_5 x_6 x_7 x_8, y_8)$
...



$$E = (y - o)^2$$

Question and Answer