递归神经网络

Recurrent Neural Network

CONTENT

1 递归神经网络

Recurrent Neural Network: Sequence Problem

递归神经网络: 序列问题

- 翻译
- 语音识别
- 时序预测
- 语言生成

Why Recurrent?

为什么递归?

- 传统模型通常假设样本是iid
- 时序依赖数据需要一个机制来记住之前的信息
    - “我是中国人，我说?(中文?英文?)
- HMM模型训练复杂度过高, 参数数量过高

Recurrent Neural Network

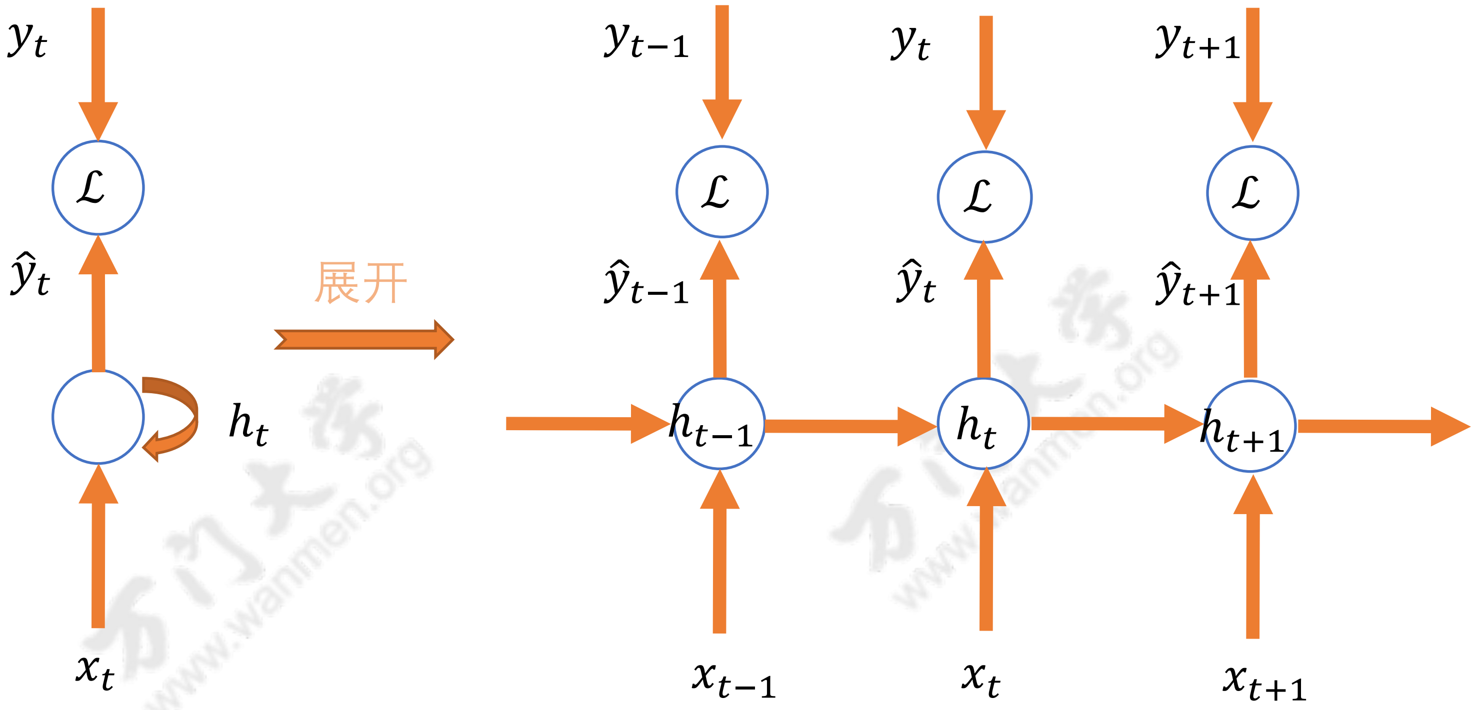
递归神经网络

$$h_t = f(Ux_t + Wh_{t-1} + b)$$

$$o_t = Vh_t + c$$

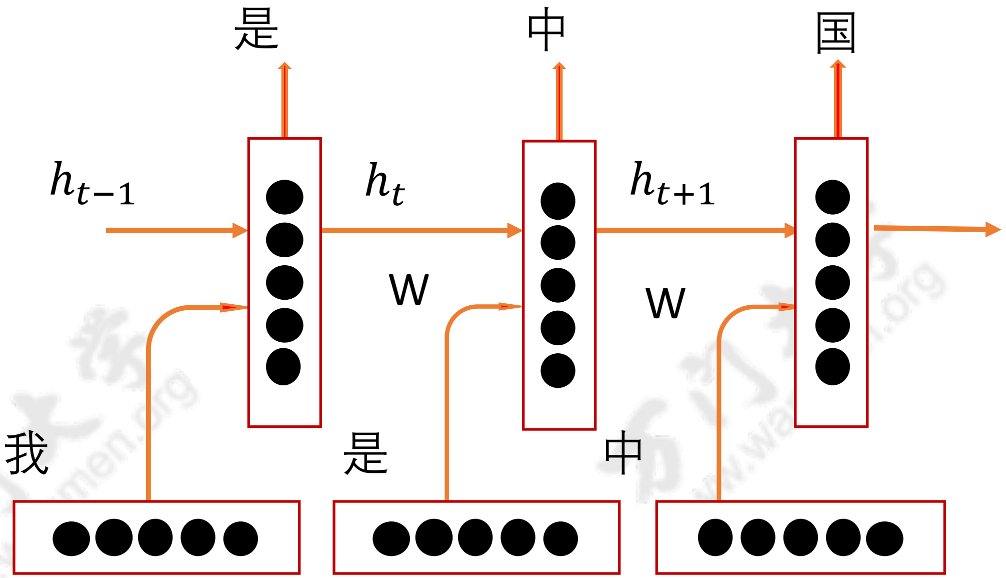
Recurrent Neural Network

递归神经网络



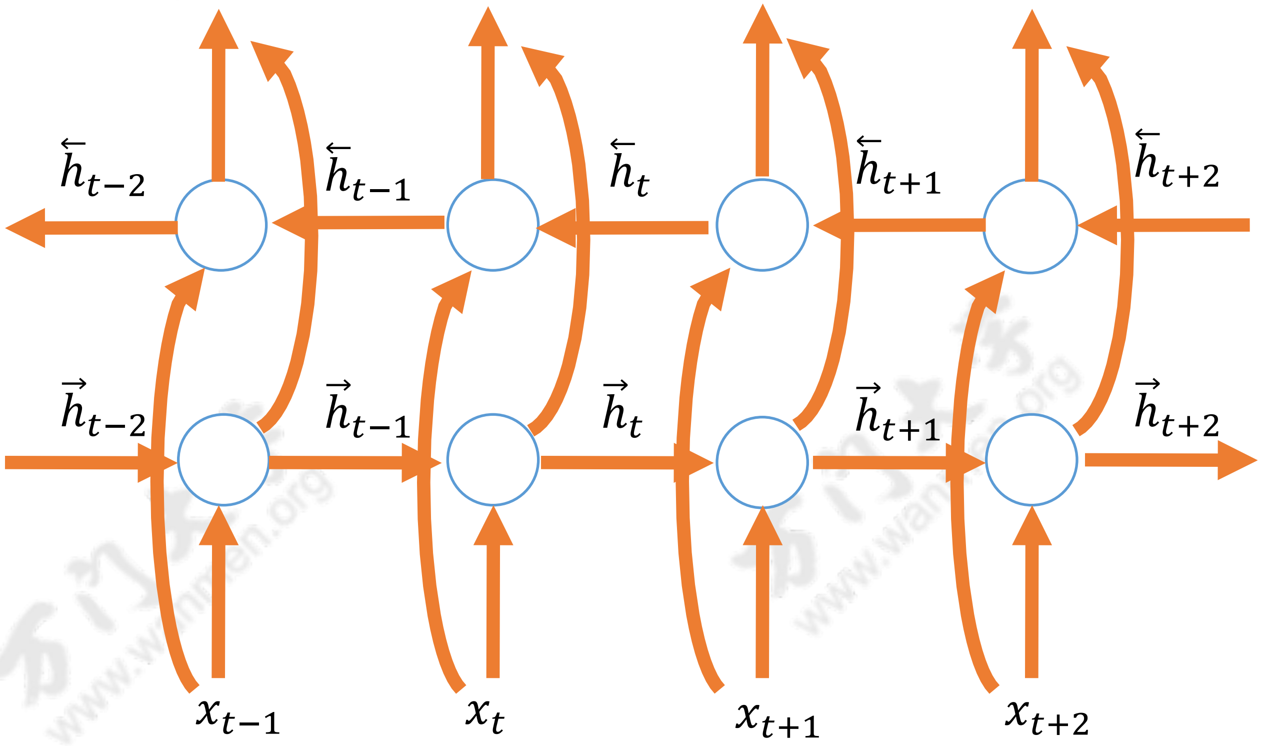
Recurrent Neural Network

递归神经网络



Bidirectional Recurrent Neural Network

双向递归神经网络

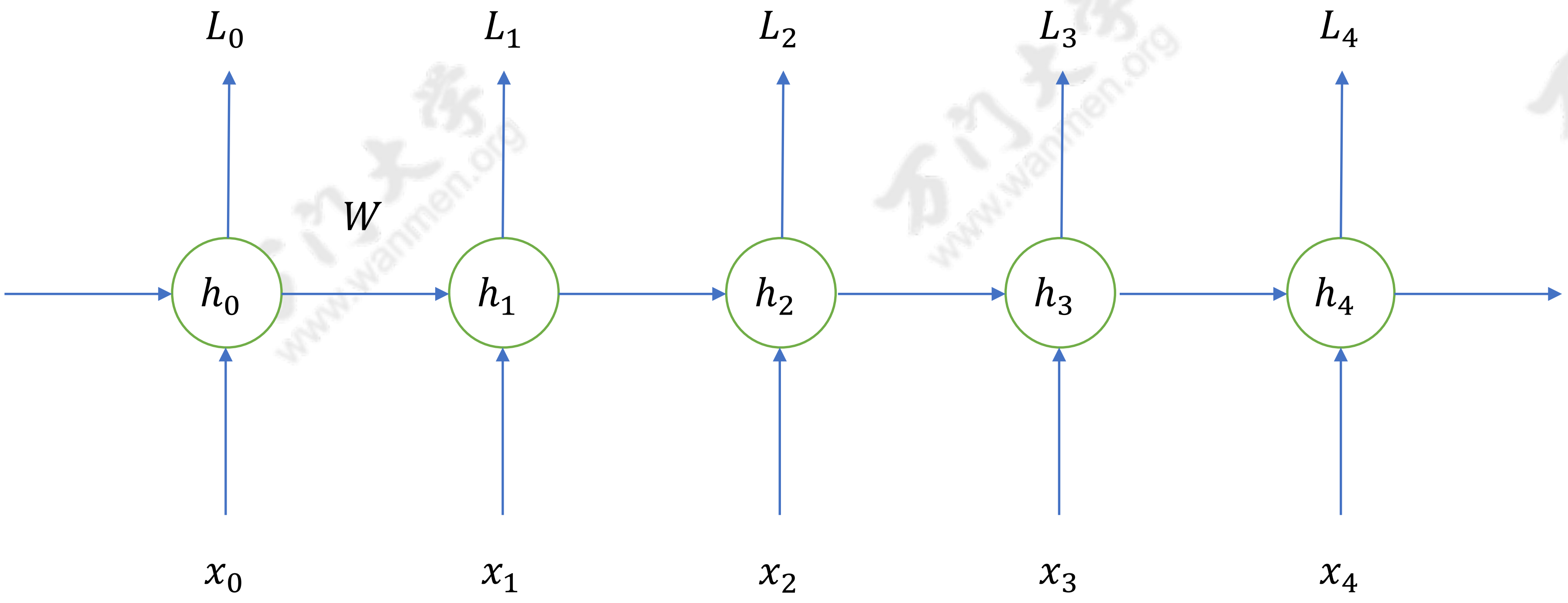


Backpropagation Through Time

沿时间反向传播

$$L_t(y_t, \hat{y}_t) = -y_t log \hat{y}_t$$

$$L(y, \hat{y}) = \sum_t L_t$$



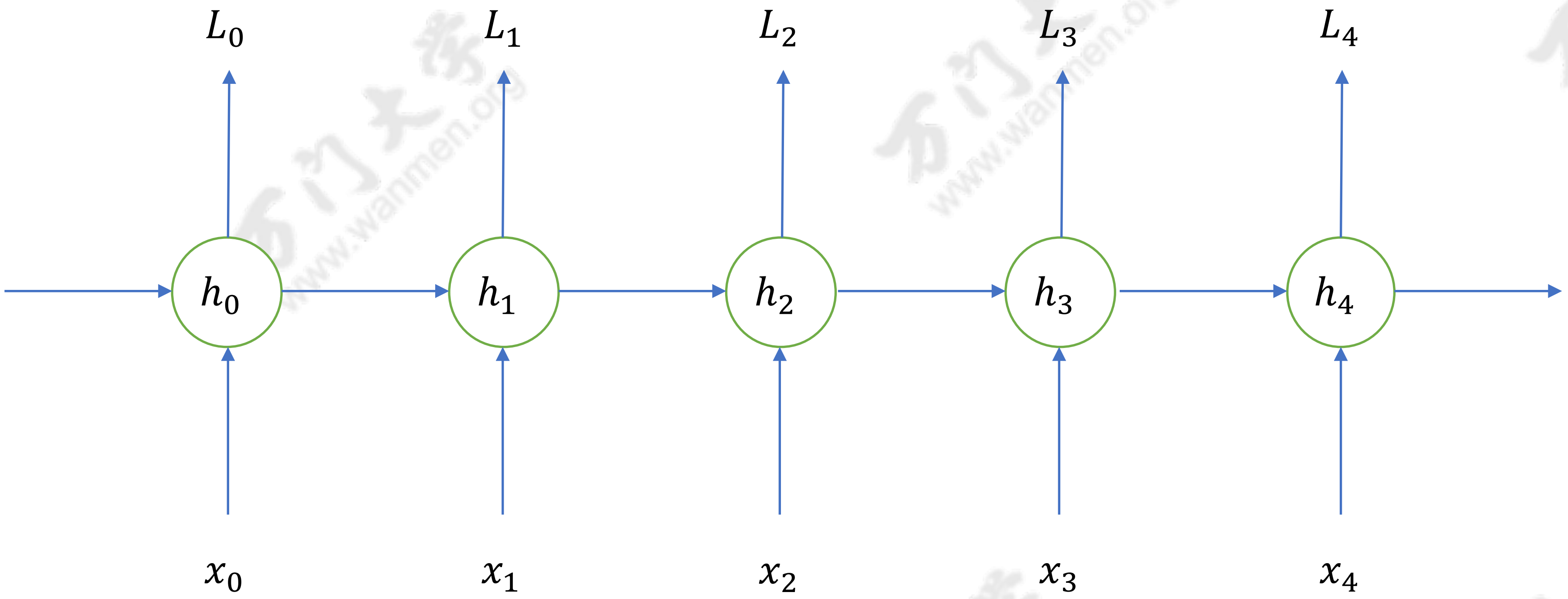
Backpropagation Through Time

沿时间反向传播

$$\frac{\partial L}{\partial W} = \sum_t \frac{\partial L_t}{\partial W}$$

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

$$\frac{\partial L_3}{\partial W} = \frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial W} = \sum_{t=0}^{3}\frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial h_t}\frac{\partial h_t}{\partial W}$$



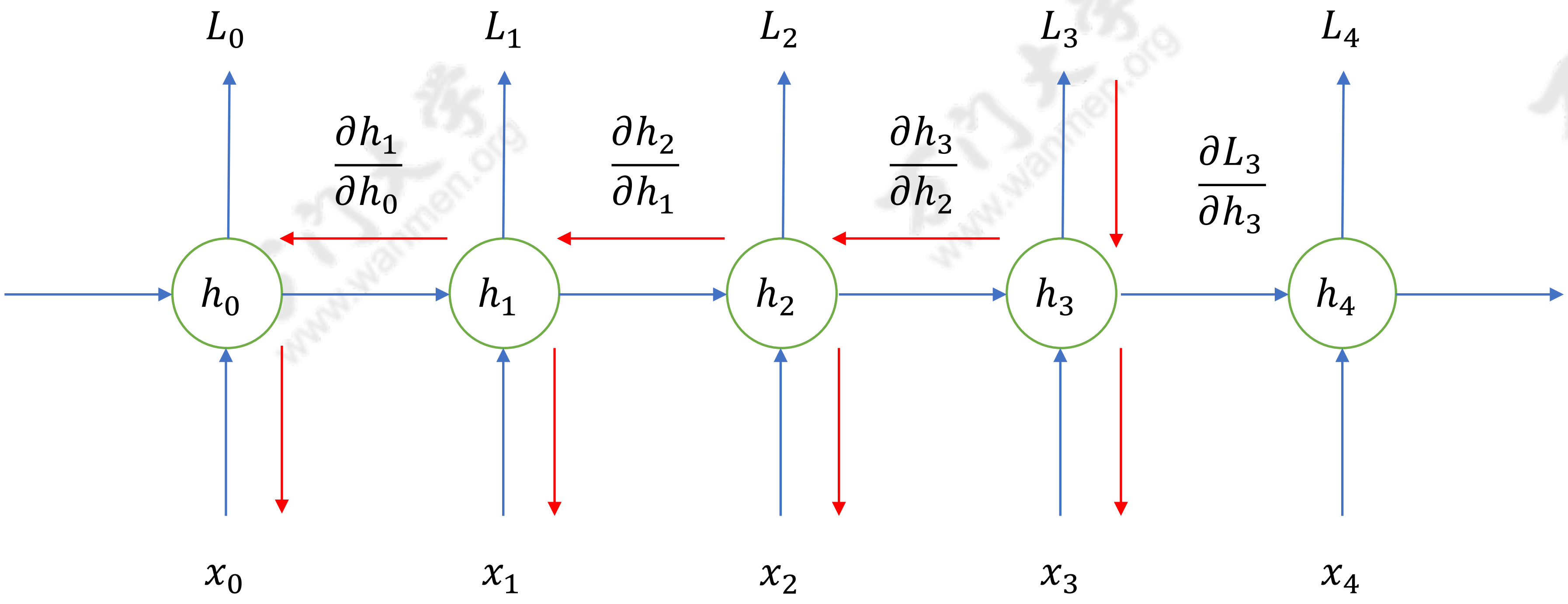
Backpropagation Through Time

沿时间反向传播

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

$$\frac{\partial L}{\partial W} = \sum_t \frac{\partial L_t}{\partial W}$$

$$\frac{\partial L_3}{\partial W} = \frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial W} = \sum_{t=0}^{3}\frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial h_t}\frac{\partial h_t}{\partial W}$$



Gradient Vanish

梯度消失

$$\frac{\partial L_3}{\partial W}=\frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial W}=\sum_{t=0}^{3}\frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\frac{\partial h_3}{\partial h_t}\frac{\partial h_t}{\partial W}$$

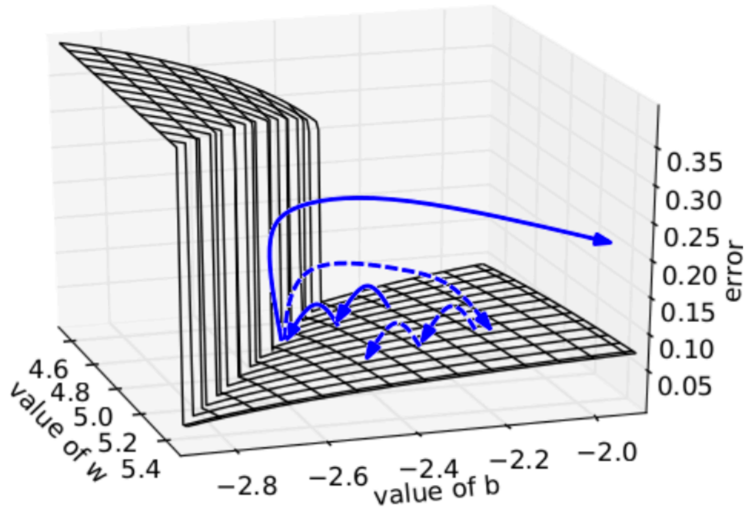
对$\frac{\partial h_3}{\partial h_k}$, 再次使用链式法则

$$\frac{\partial h_3}{\partial h_1}=\frac{\partial h_3}{\partial h_2}\frac{\partial h_2}{\partial h_1}$$

$$\frac{\partial L_3}{\partial W}=\sum_{t=0}^{3}\frac{\partial L_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial h_3}\left(\prod_{j=t+1}^{3}\frac{\partial h_j}{\partial h_{j-1}}\right)\frac{\partial h_t}{\partial W}$$

Gradient Vanish

梯度消失



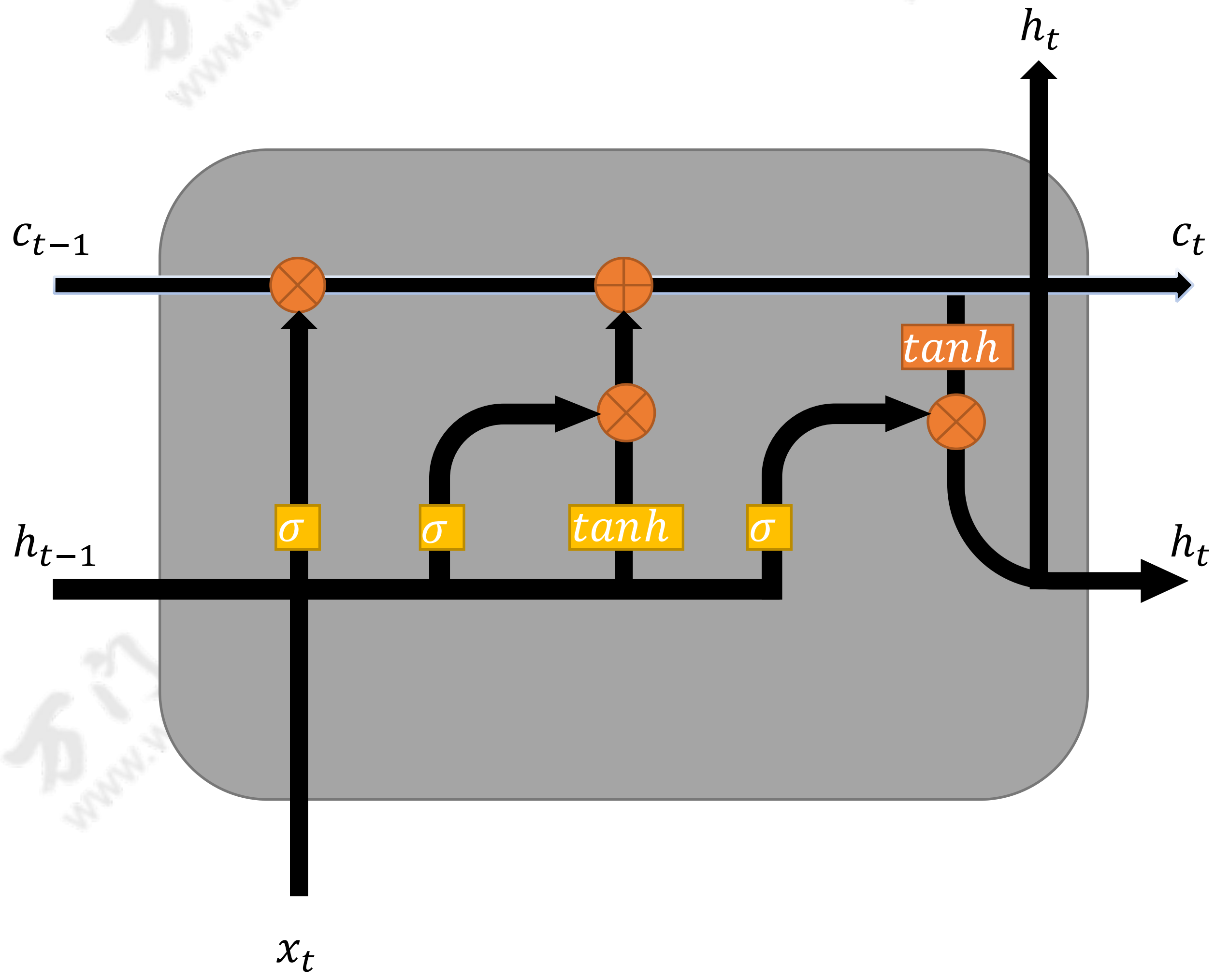
On the difficulty of training recurrent neural networks

Learning Long-Term Dependencies with Gradient Descent is Difficult

2 长短记忆网络

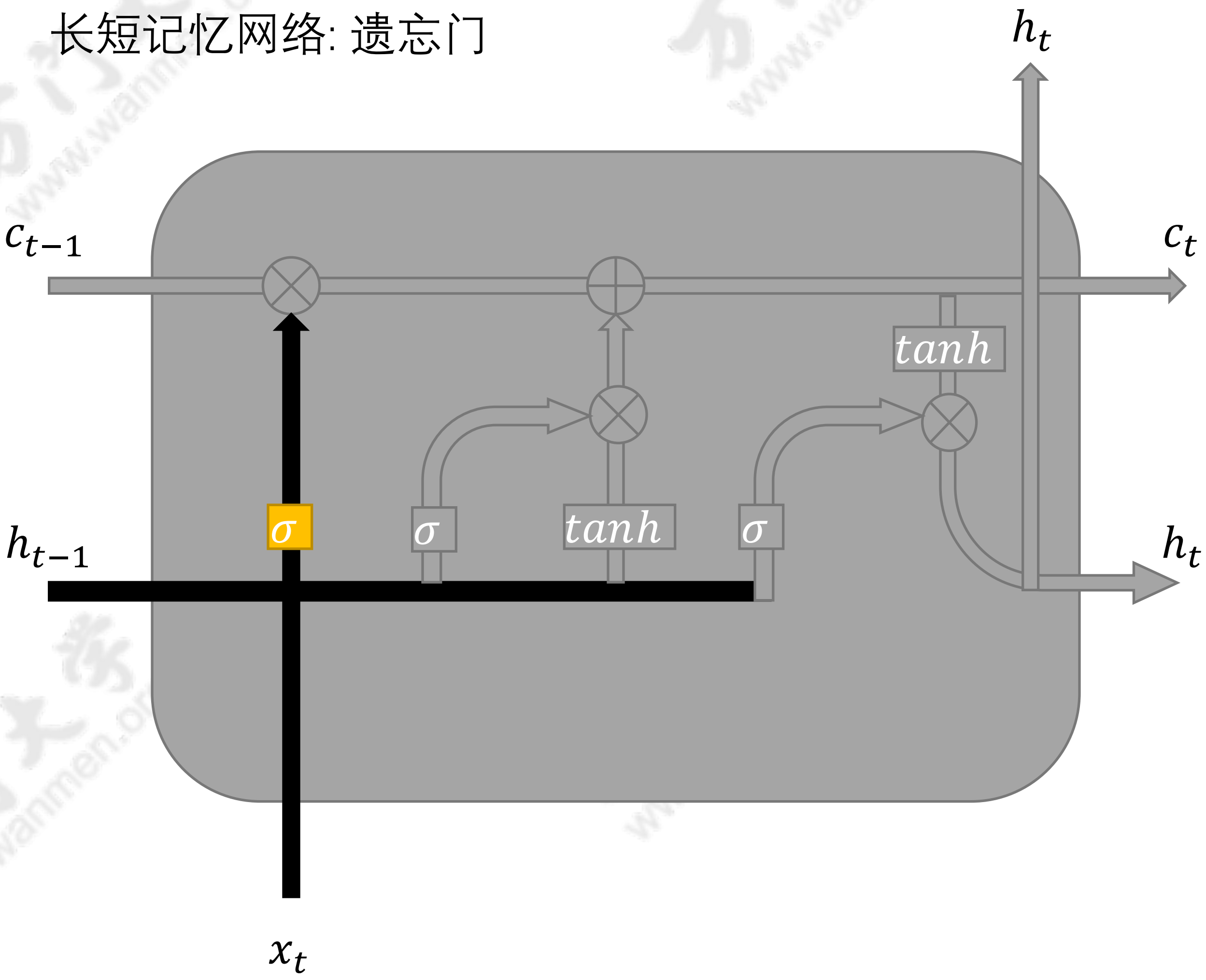
Long-short Term Memory Network

长短记忆网络



Long-short Term Memory Network: Forget Gate

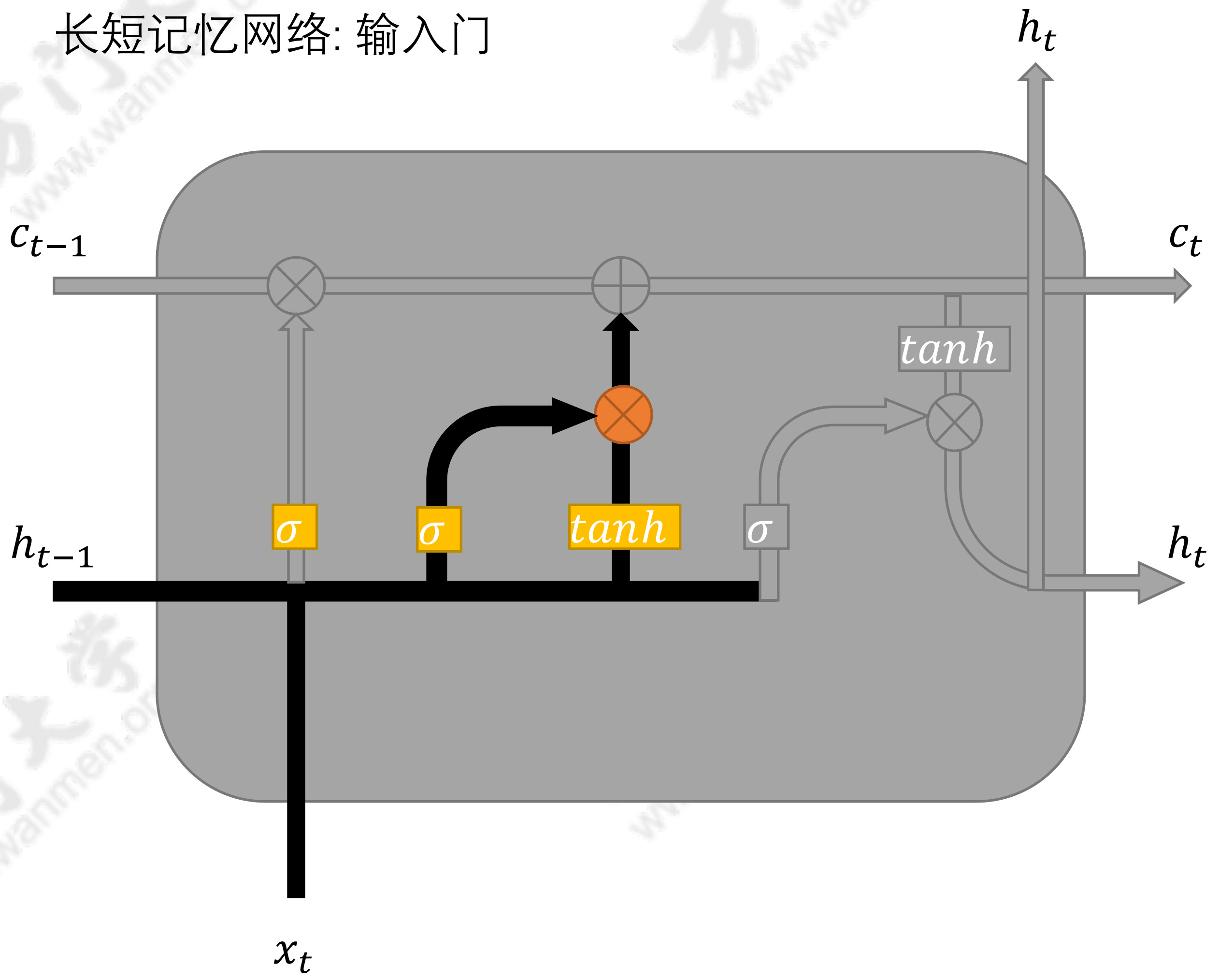
长短记忆网络: 遗忘门



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

Long-short Term Memory Network: Input Gate

长短记忆网络: 输入门

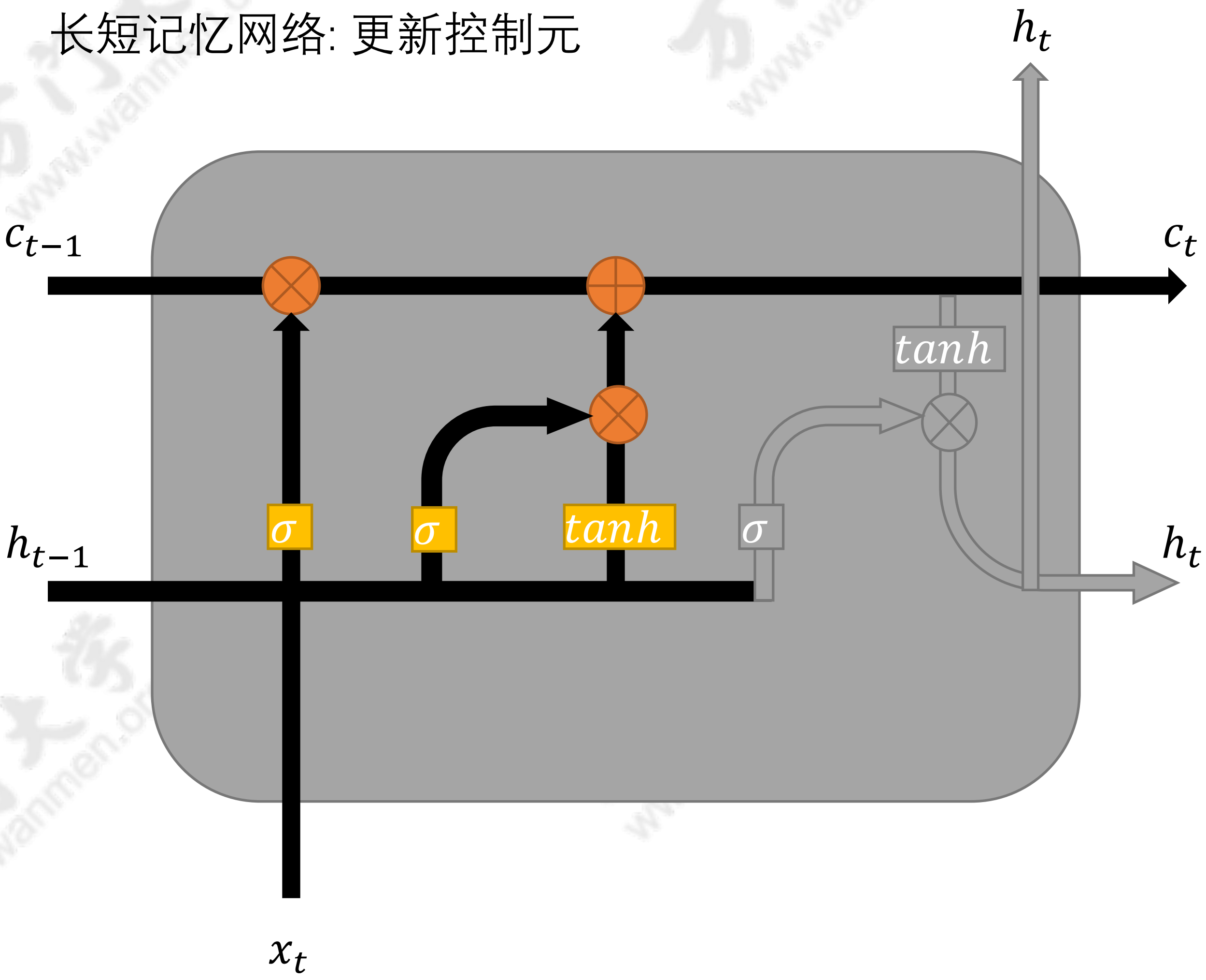


$$i_t = \sigma\left(U^i x_t + W^i h_{t-1}\right)$$

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

Long-short Term Memory Network: Update Cell State

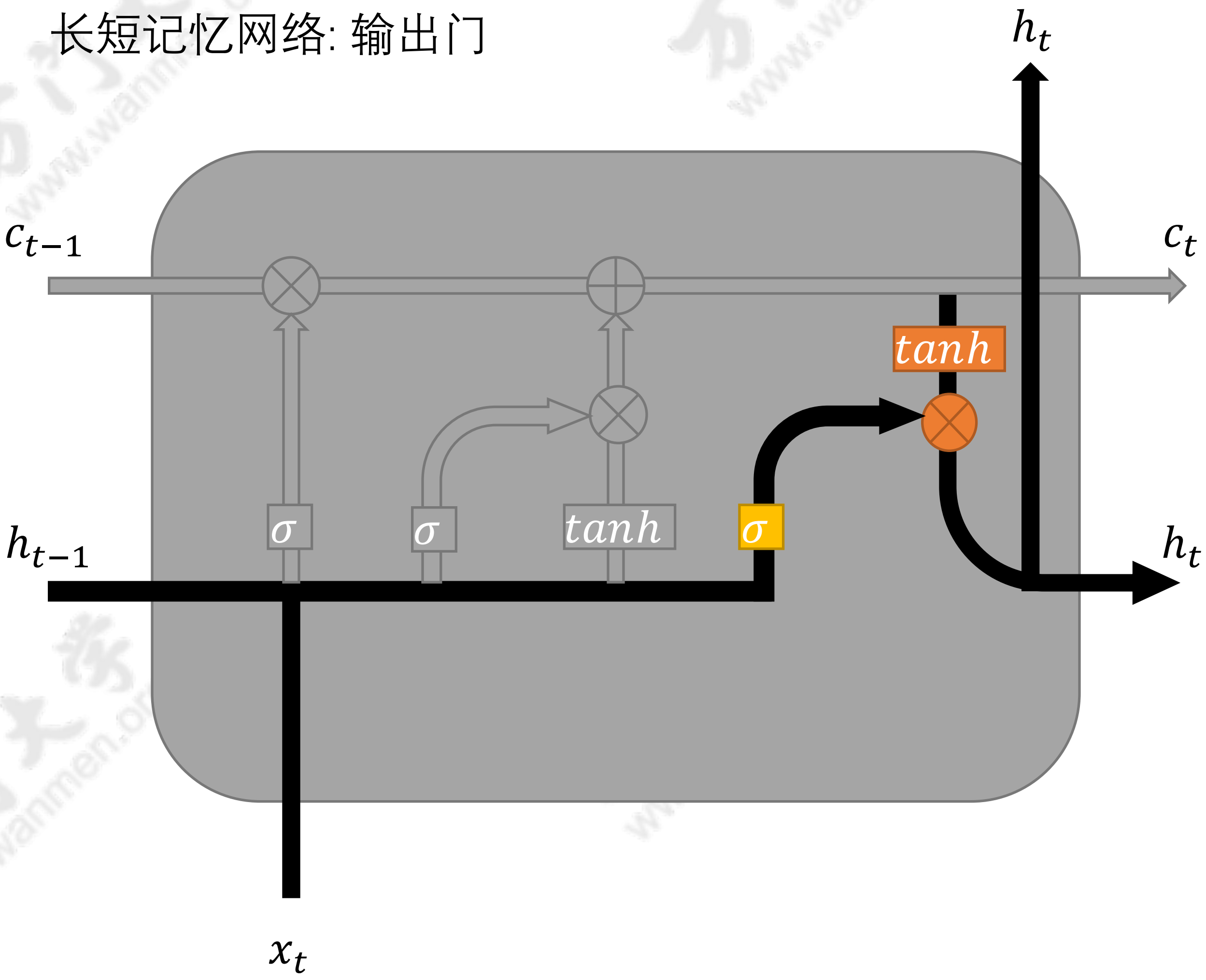
长短记忆网络: 更新控制元



$$c_t = c_{t-1} \circ f_t + g_t \circ i_t$$

Long-short Term Memory Network: Output Gate

长短记忆网络: 输出门

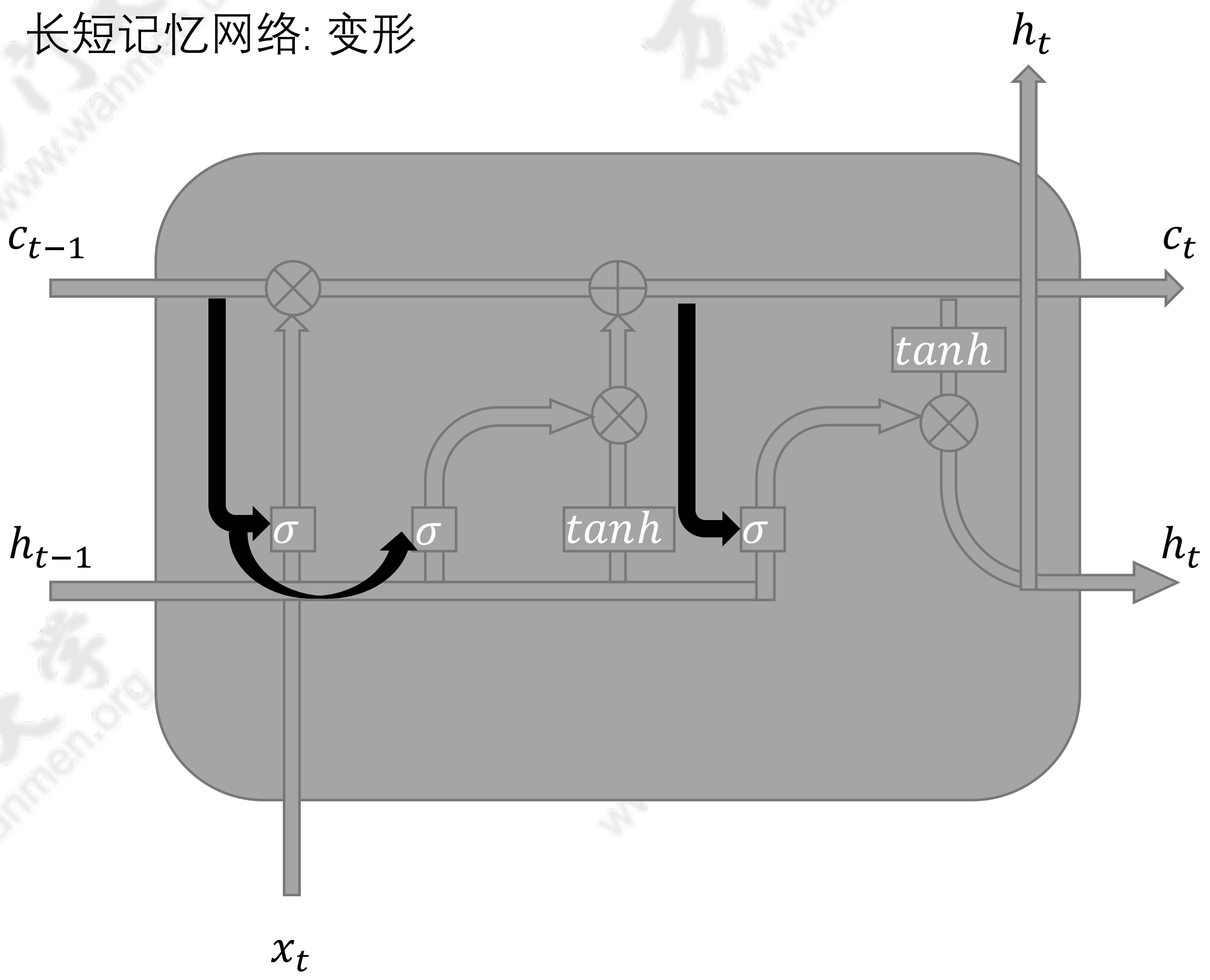


$$o_t = \sigma(U^o x_{t-1} + W^o h_{t-1})$$

$$h_t = o_t * \tanh(c_t)$$

Long-short Term Memory Network: Variants

长短记忆网络: 变形



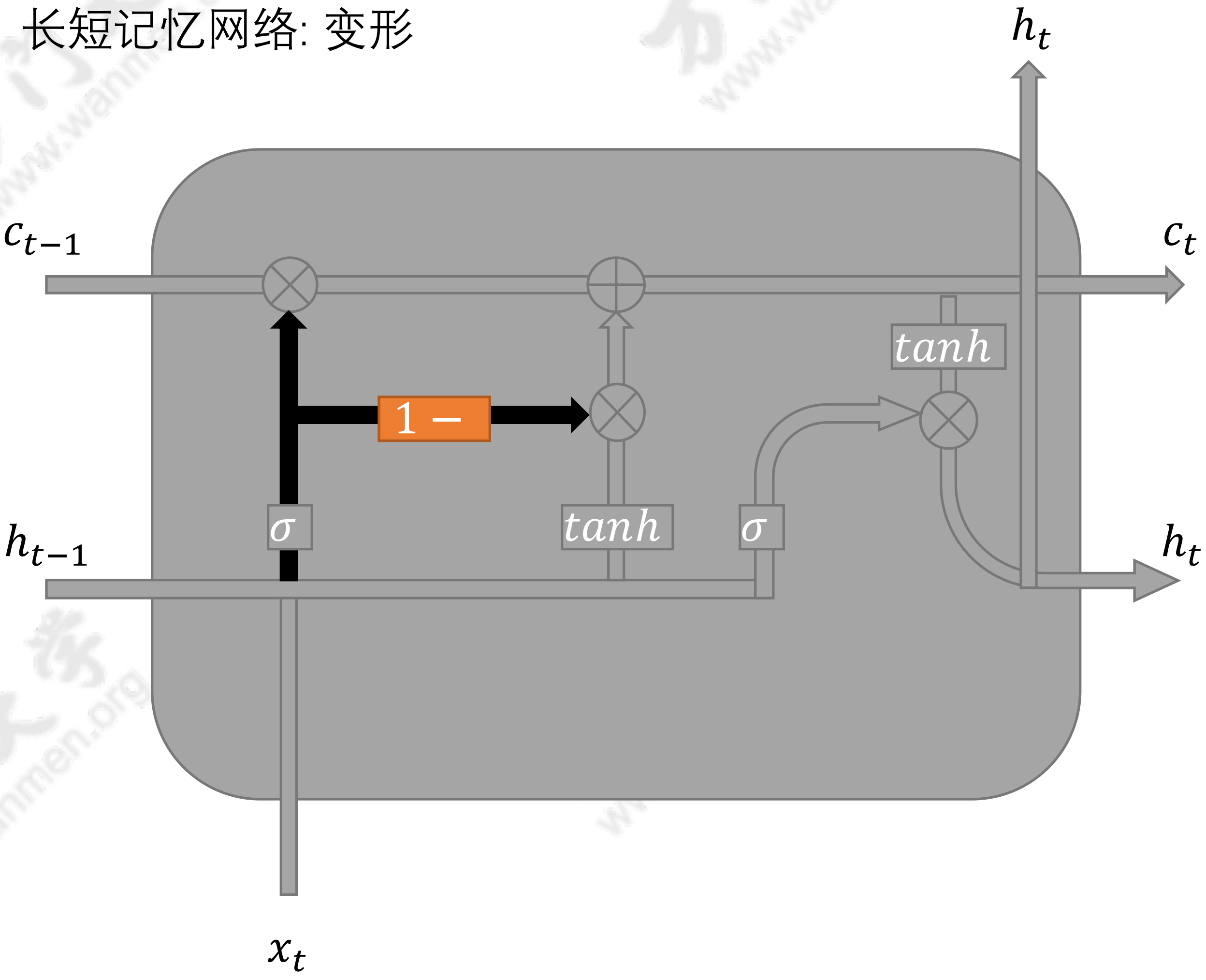
$$f_t = \sigma\left(U^f x_t + W^f h_{t-1} + V^f c_{t-1}\right)$$

$$i_t = \sigma\left(U^i x_t + W^i h_{t-1} + V^i c_{t-1}\right)$$

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

Long-short Term Memory Network: Variants

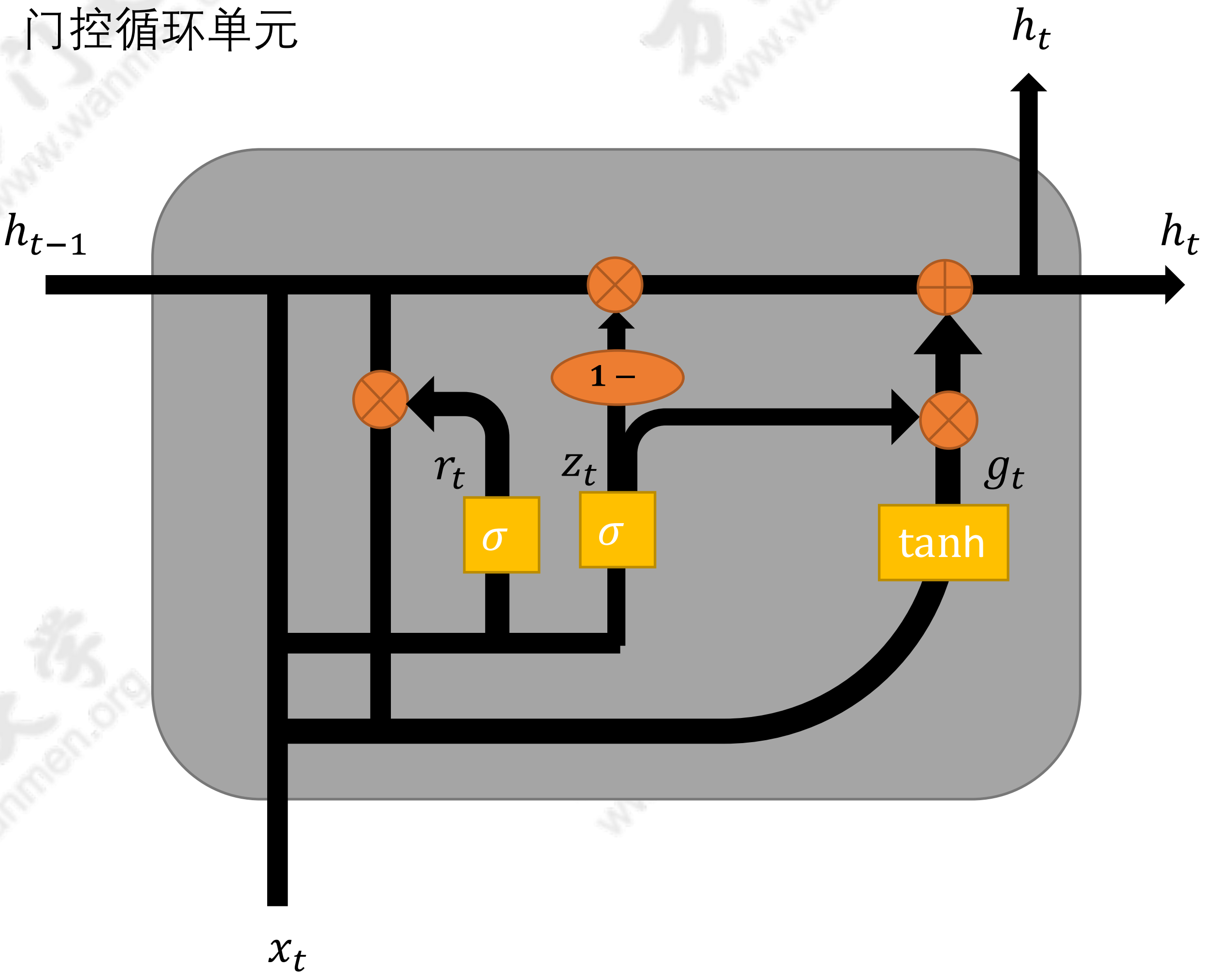
长短记忆网络: 变形



$$c_{t-1} = f_t * c_{t-1} + (1 - f_t) * g_t$$

Gated Recurrent Unit

门控循环单元



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

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

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

$$h_t = (1 - z_t) * h_{t-1} + z_t * g_t$$