Deep Learning for Natural Language Processing

The LSTM architecture

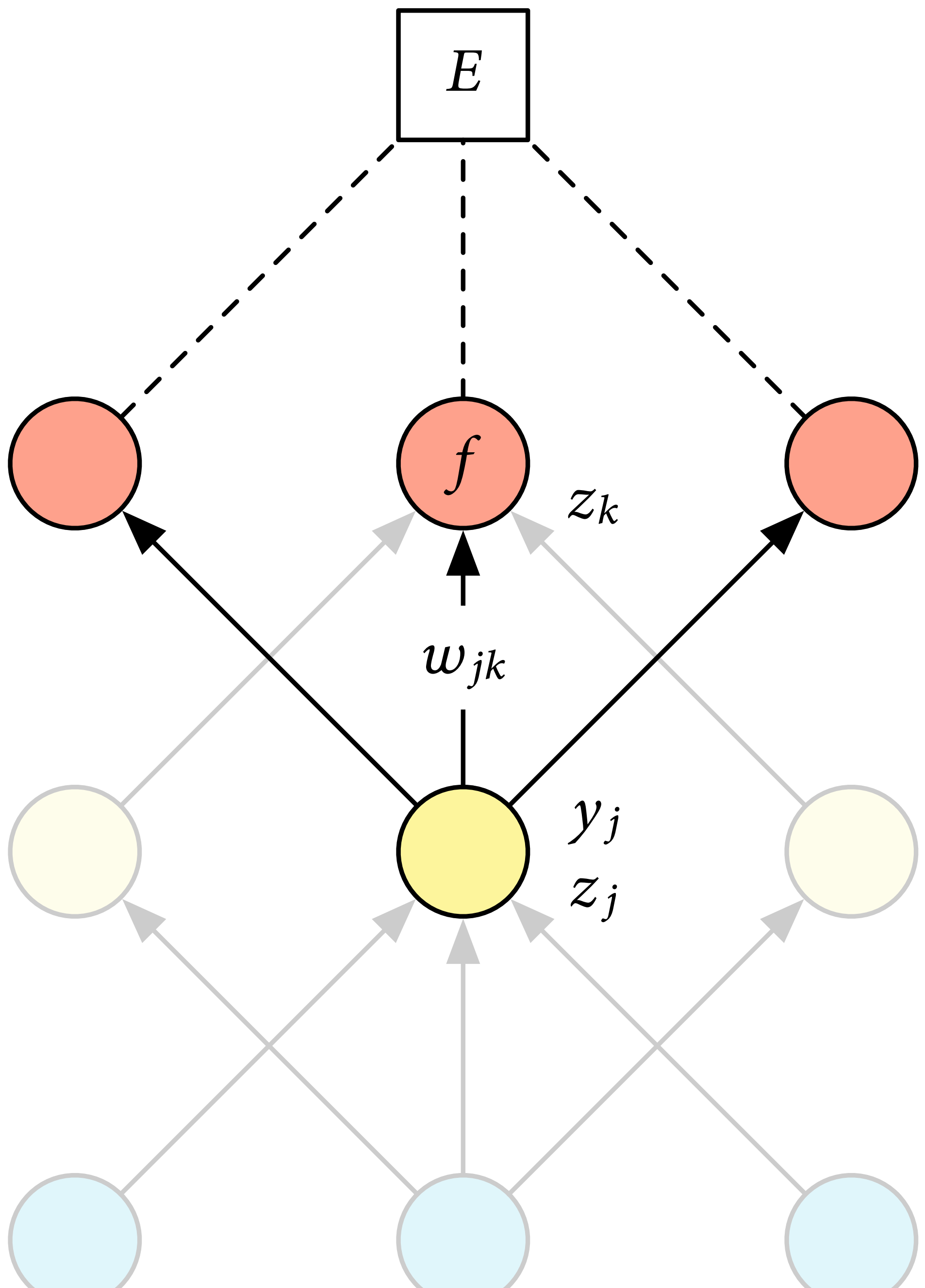
Marco Kuhlmann
Department of Computer and Information Science
Linköping University

WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

Challenges with recurrent neural networks

- In principle, recurrent neural networks are capable of learning long-distance dependencies in input sequences.
- In practice, training recurrent neural networks is challenging due to the large depth of the unrolled networks.

Vanishing and exploding gradients



$$\delta_k = \frac{\partial E}{\partial z_k} = \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial z_k} = \frac{\partial E}{\partial y_k} f'(z_k)$$

$$\delta_j = \frac{\partial E}{\partial z_j} = \frac{\partial y_j}{\partial z_j}\sum_k \frac{\partial E}{\partial z_k}\frac{\partial z_k}{\partial y_j} = f'(z_j)\sum_k \delta_k w_{jk}$$

Vanishing and exploding gradients

- In backpropagation there is a risk of gradients either vanishing or exploding, depending on the magnitude of the weights.
- This problem is exacerbated in recurrent networks, whose unrolled computation graphs can be very deep.
- Research on recurrent networks has proposed various methods to mitigate this problem.

  weight scaling and clipping, specialised architectures

Long Short-Term Memory (LSTM)

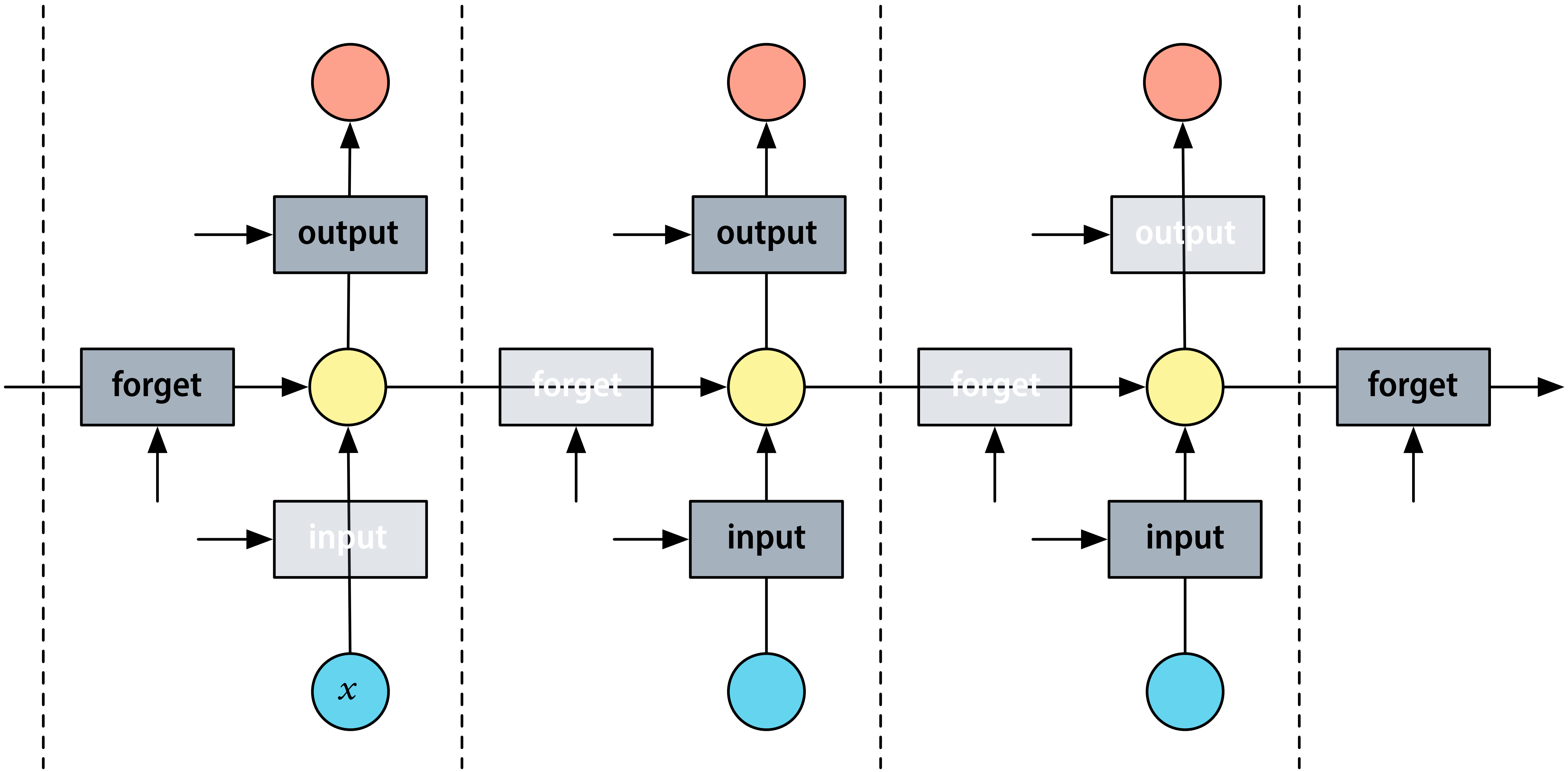
- The **Long Short-Term Memory (LSTM)** architecture was specifically designed to adress the vanishing gradients problem.
- Metaphor: The hidden state of the neural network can be considered as a short-term memory.
- The LSTM architecture tries to make this short-term memory last as long as possible by preventing vanishing gradients.

Memory cell and gating mechanism

The crucial innovation in an LSTM is the design of its memory cell.

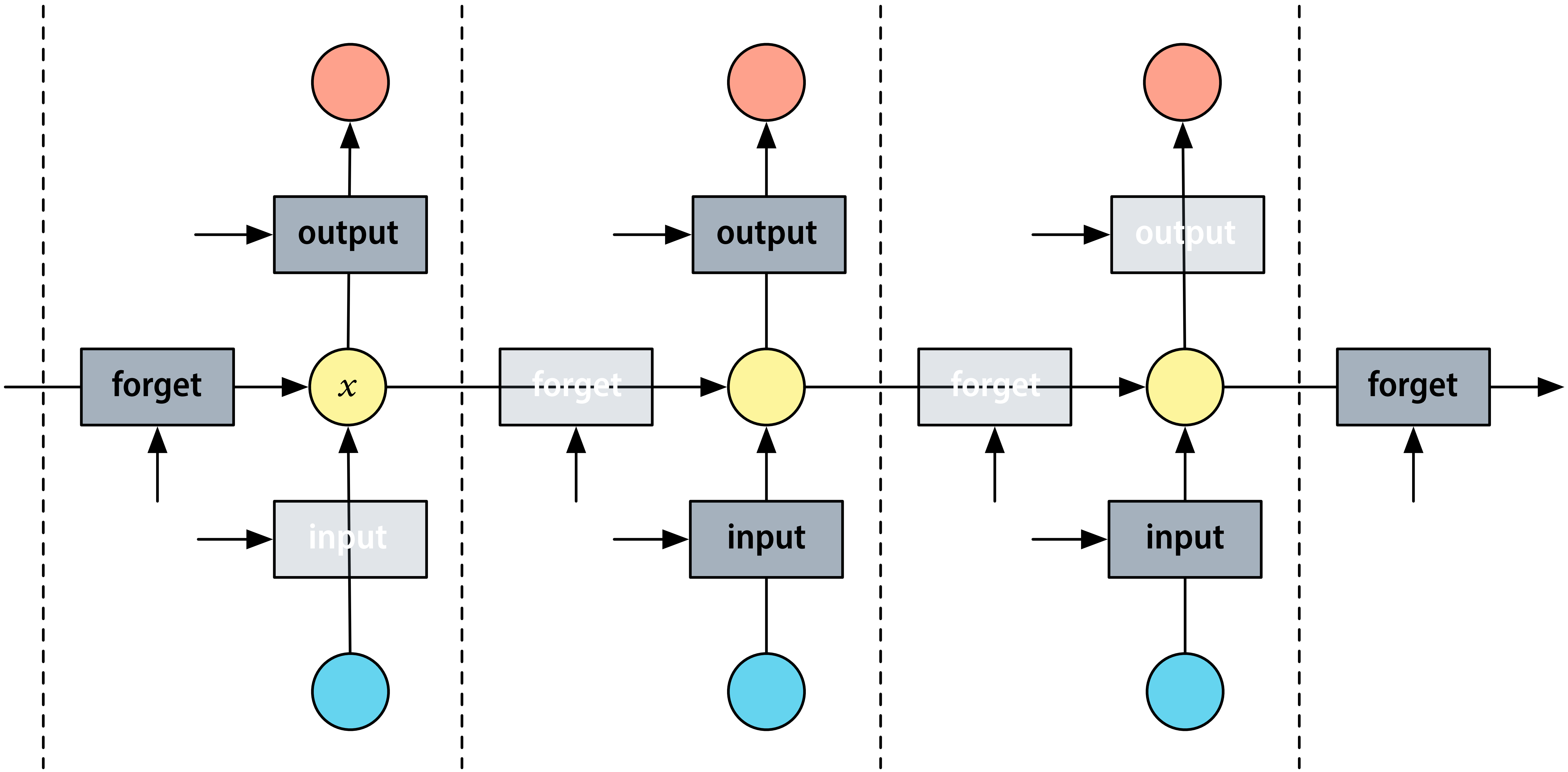
- Information is written into the cell if its INPUT gate is open.
- Information stays in the cell as long as its FORGET gate is closed.
- Information is read from the cell if its READ gate is open.

Information flow in an LSTM



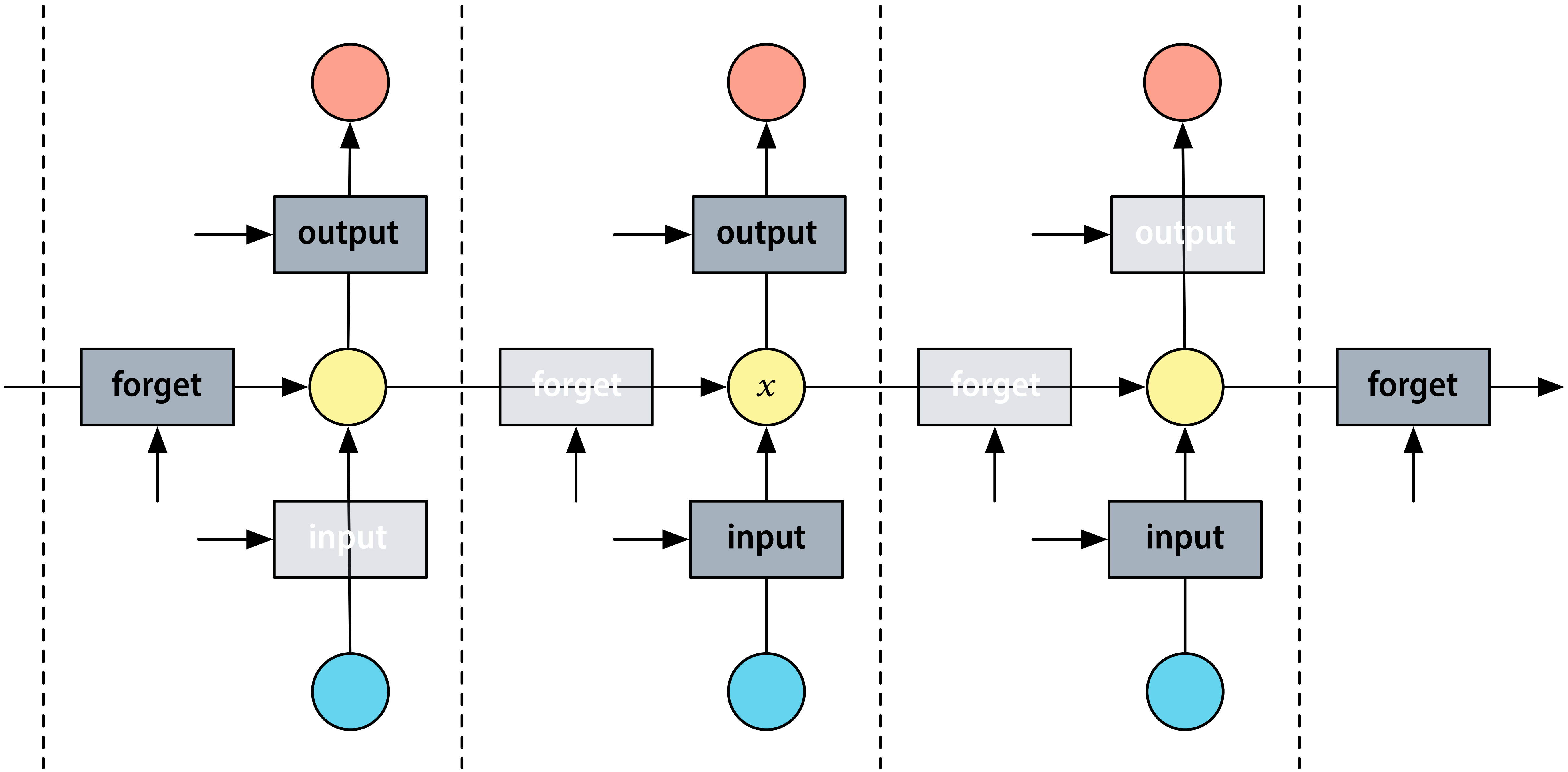
Attribution: Geoffrey Hinton

Information flow in an LSTM



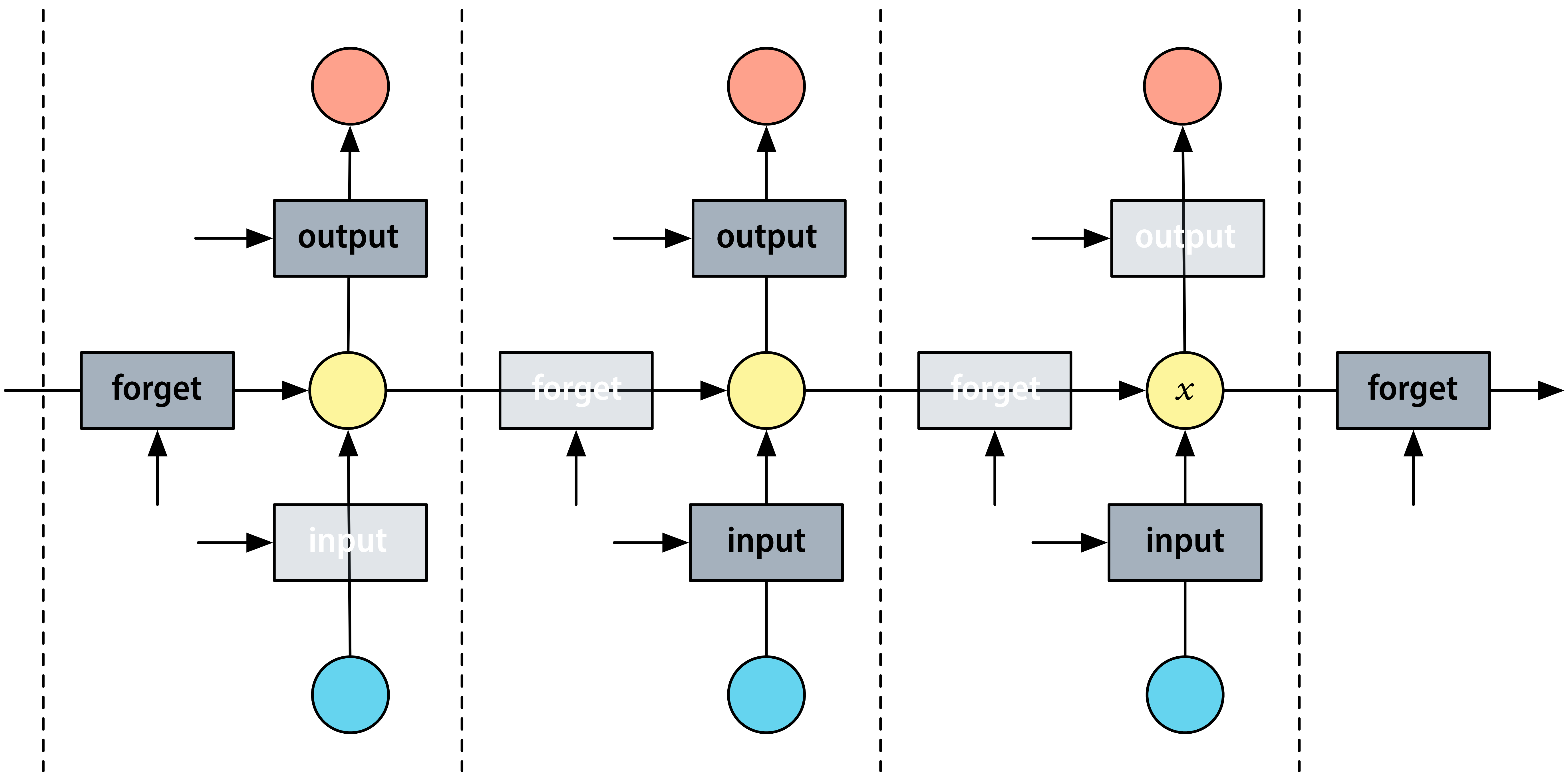
Attribution: Geoffrey Hinton

Information flow in an LSTM



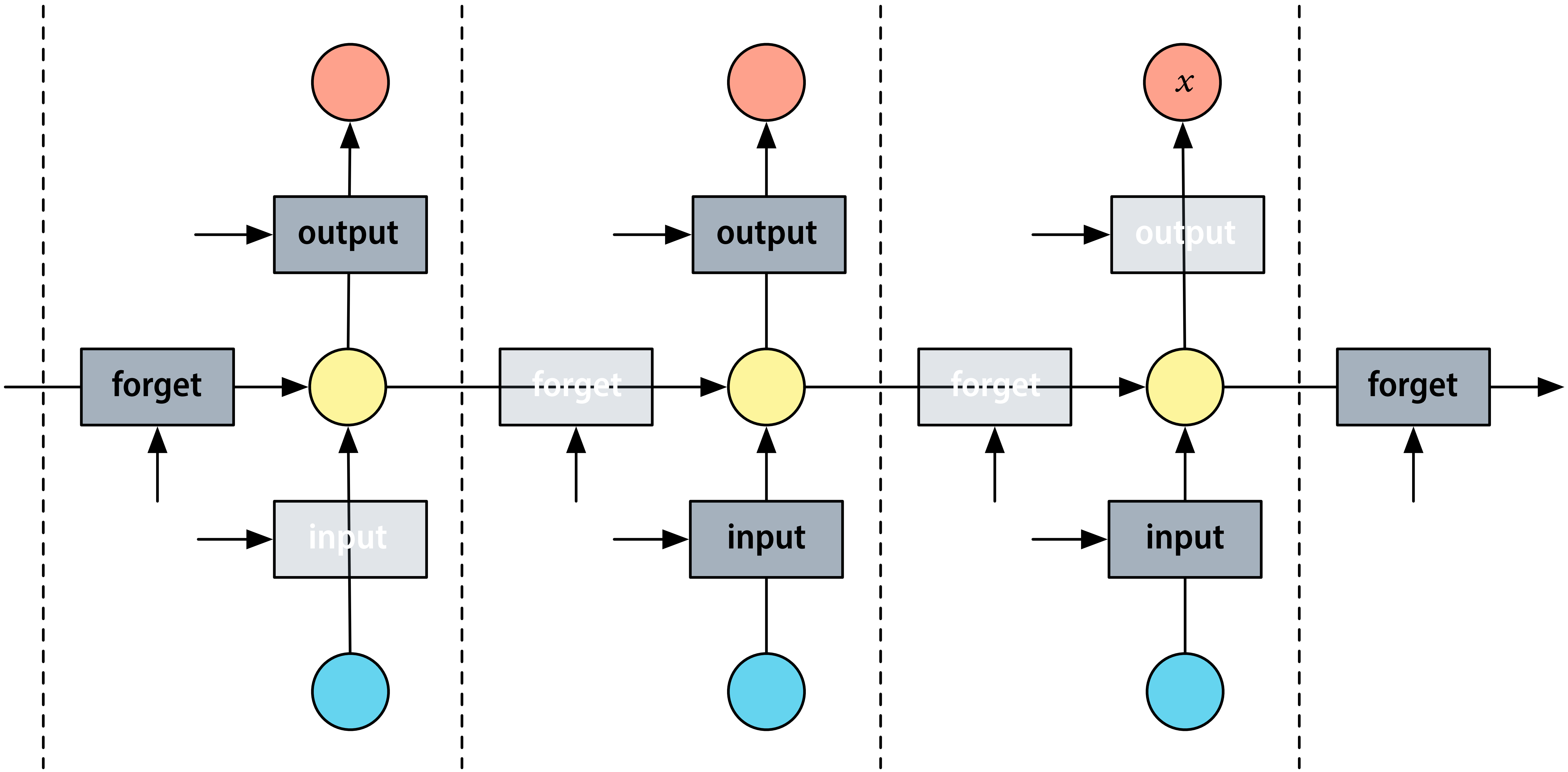
Attribution: Geoffrey Hinton

Information flow in an LSTM



Attribution: Geoffrey Hinton

Information flow in an LSTM



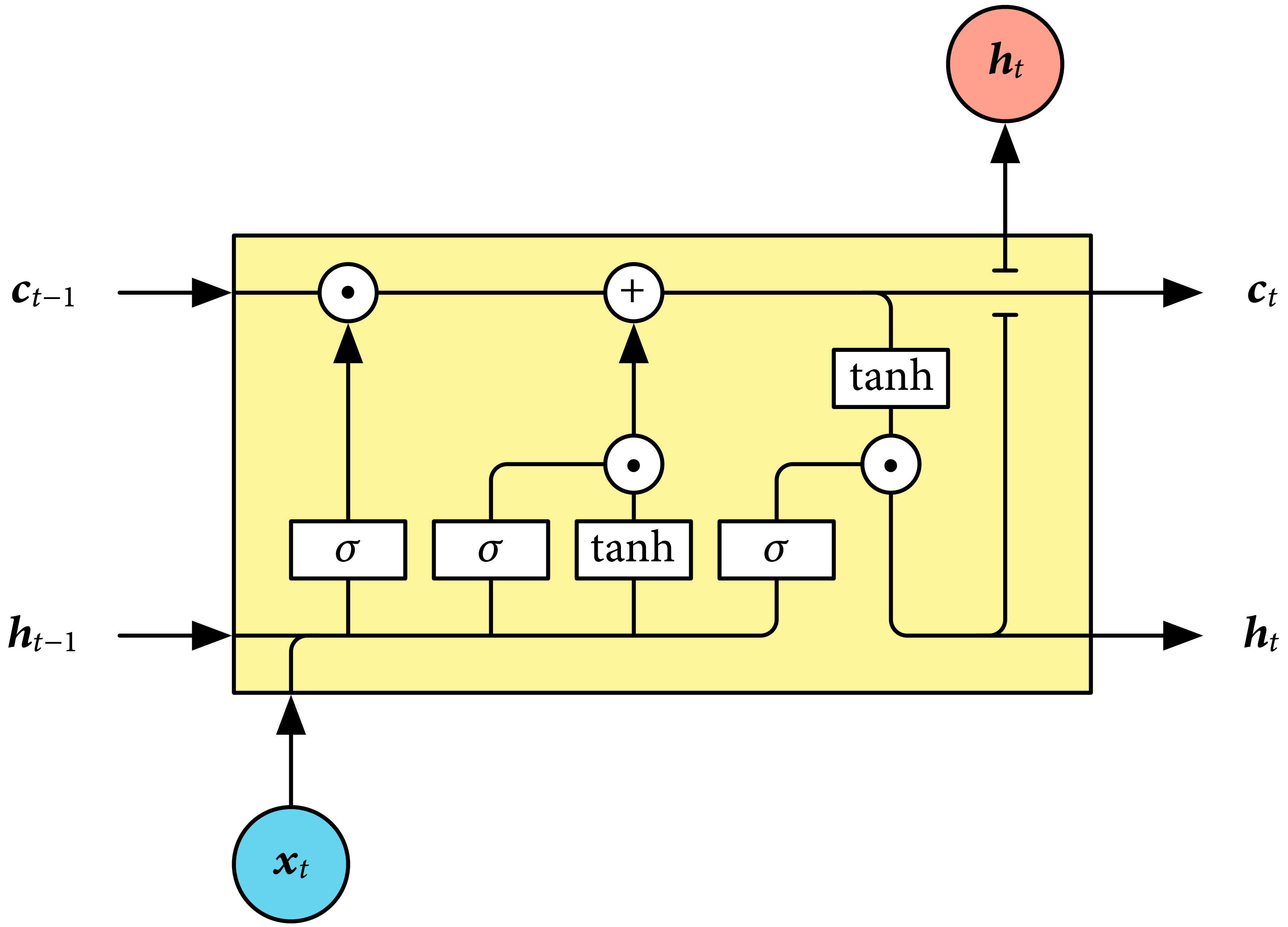
Attribution: Geoffrey Hinton

Gating mechanism

$$\underbrace{\begin{bmatrix}1\\2\\3\\4\end{bmatrix}}_{\boldsymbol{h}_{t-1}} \odot \underbrace{\begin{bmatrix}0\\1\\1\\0\end{bmatrix}}_{\boldsymbol{g}} + \underbrace{\begin{bmatrix}5\\6\\7\\8\end{bmatrix}}_{\boldsymbol{x}_t} \odot \underbrace{\begin{bmatrix}1\\0\\0\\1\end{bmatrix}}_{1-\boldsymbol{g}} = \underbrace{\begin{bmatrix}5\\2\\3\\8\end{bmatrix}}_{\boldsymbol{h}_t}$$

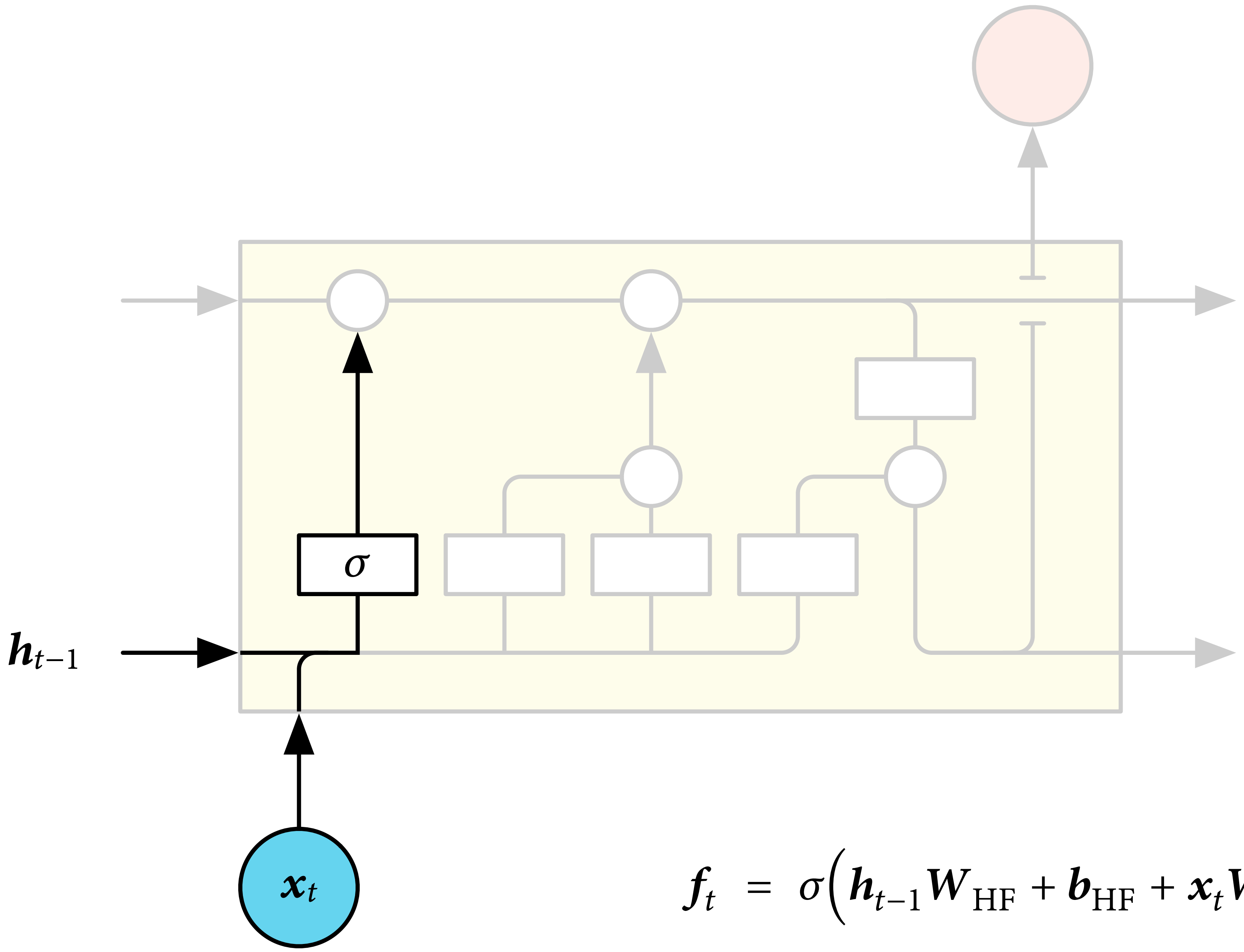
The gating masks $\boldsymbol{g}$ are learned values between 0 and 1.

A look inside an LSTM cell



Attribution: Chris Olah

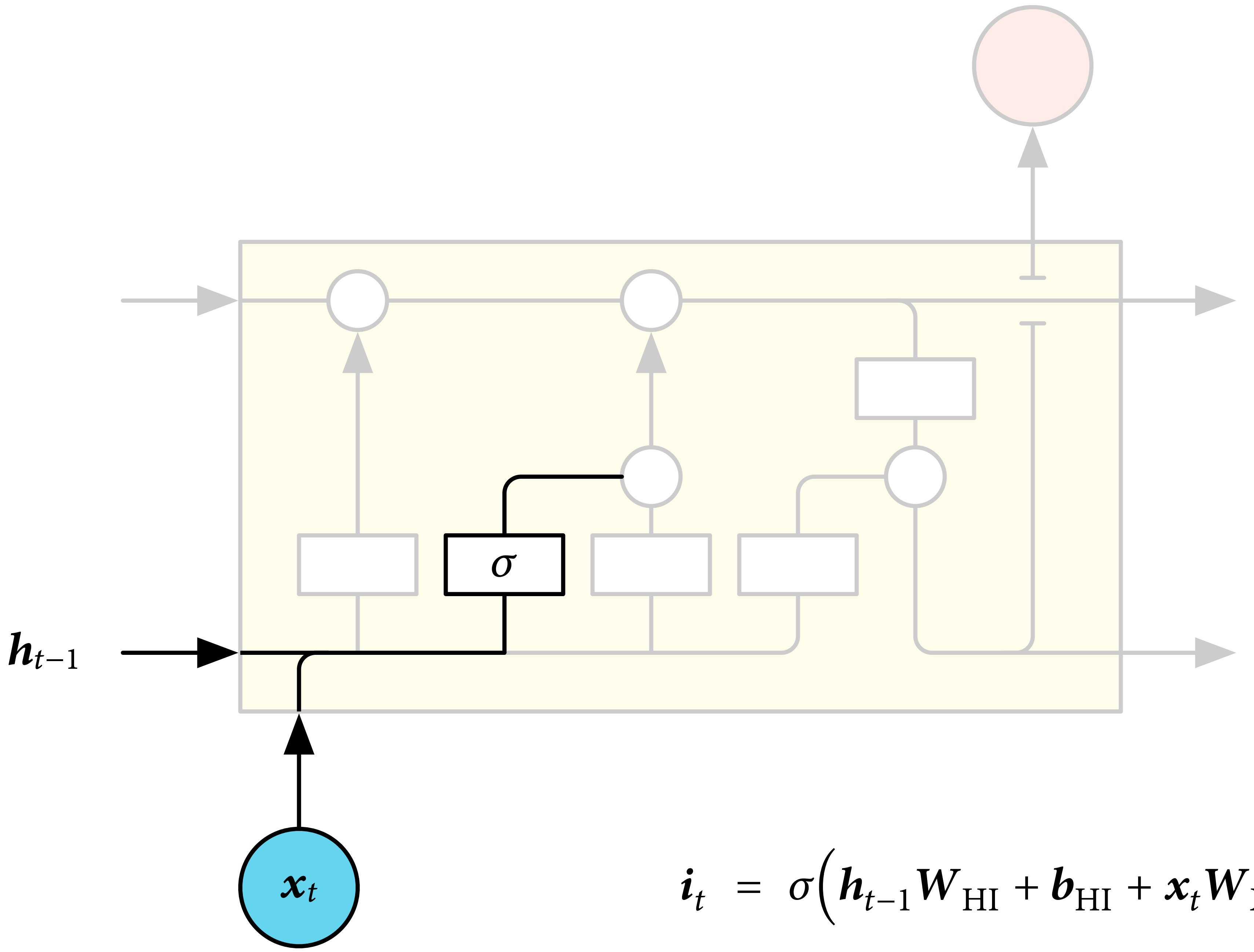
Forget gate



$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{h}_{t-1}\boldsymbol{W}_{\mathrm{HF}} + \boldsymbol{b}_{\mathrm{HF}} + \boldsymbol{x}_t\boldsymbol{W}_{\mathrm{XF}} + \boldsymbol{b}_{\mathrm{XF}}\right)$$

Attribution: Chris Olah

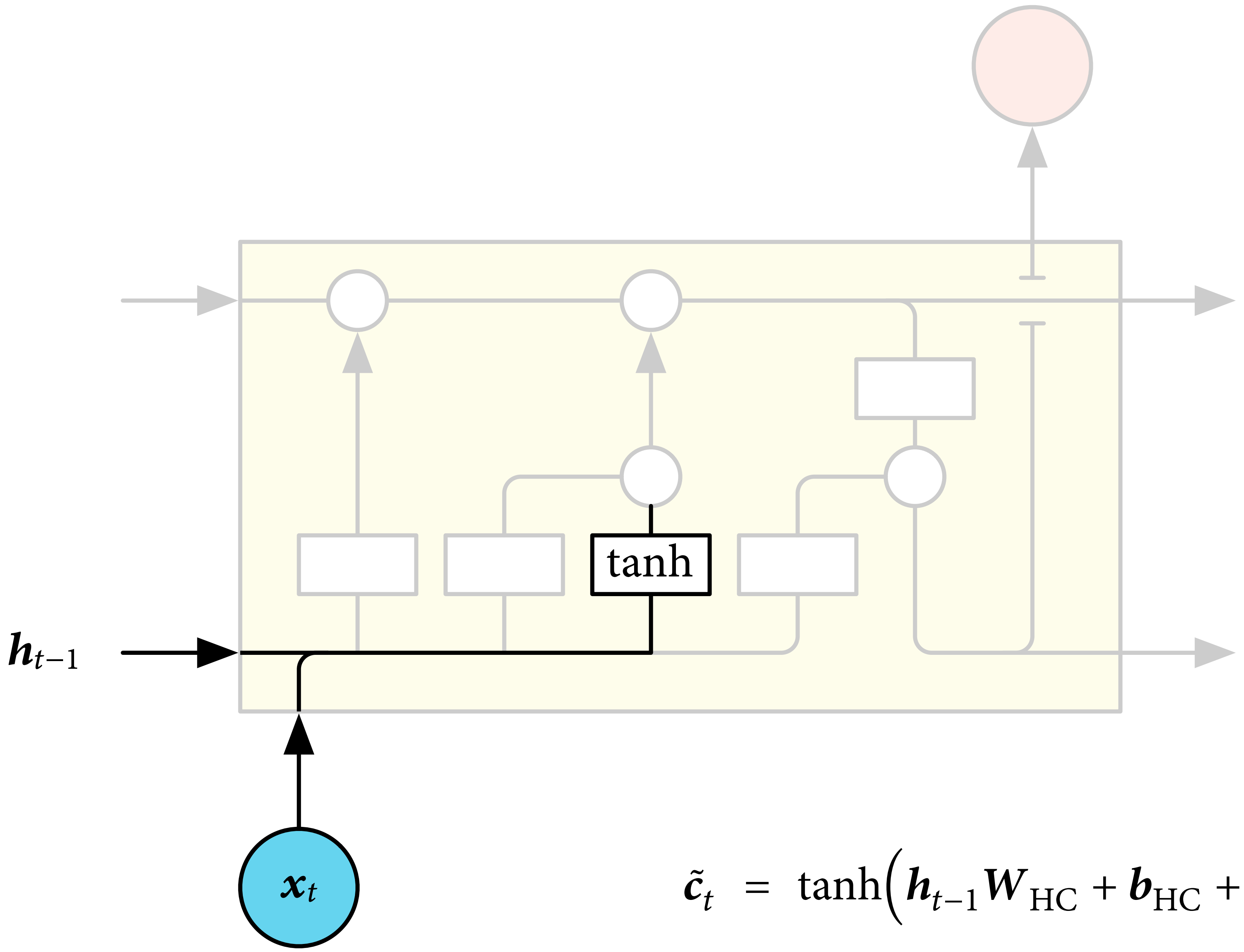
Input gate



$$i_t = \sigma\left(h_{t-1}W_{\mathrm{HI}} + b_{\mathrm{HI}} + x_t W_{\mathrm{XI}} + b_{\mathrm{XI}}\right)$$

Attribution: Chris Olah

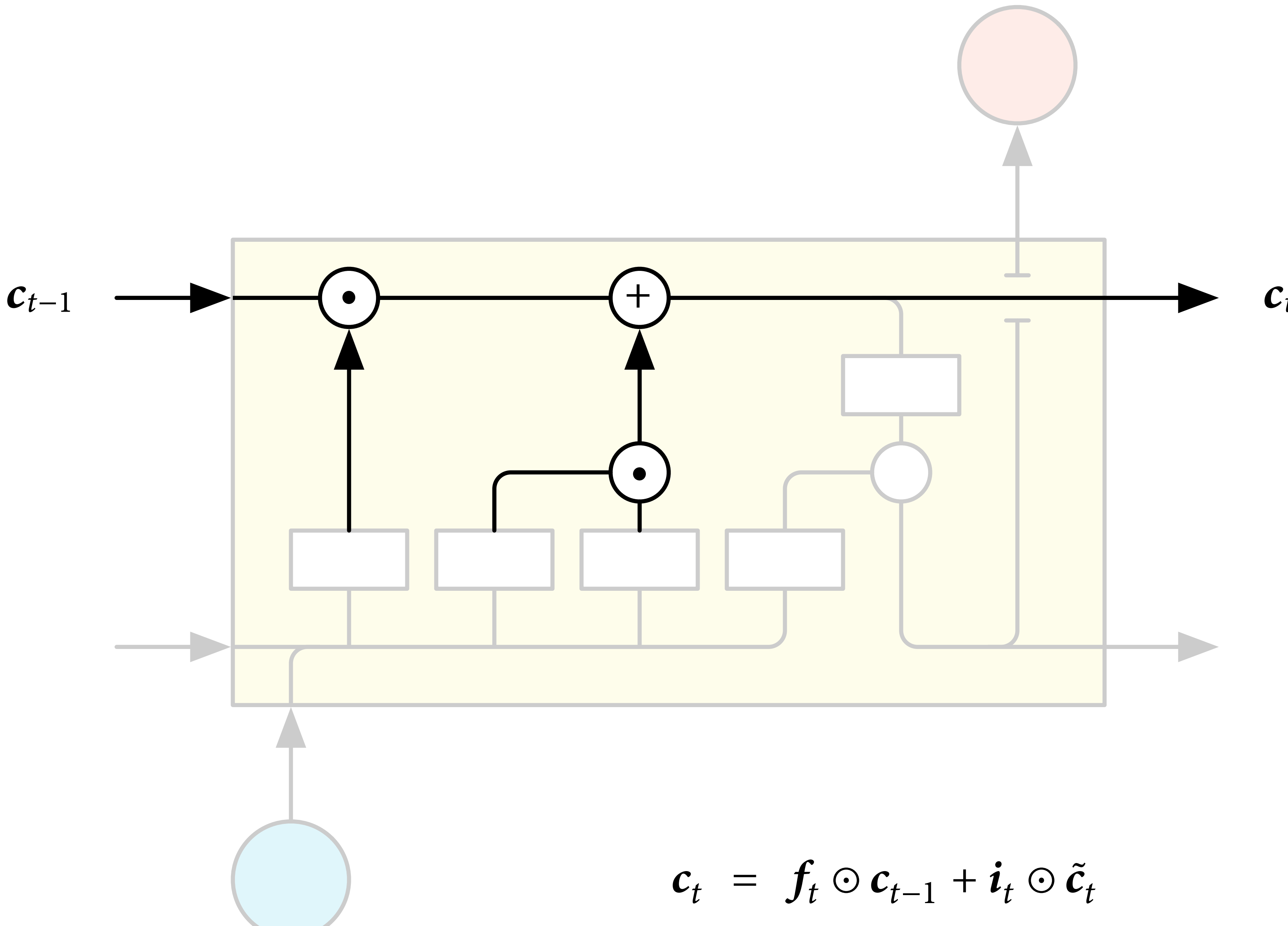
Update candidate



$$\tilde{c}_t = \tanh\left(\boldsymbol{h}_{t-1}\boldsymbol{W}_{\mathrm{HC}} + \boldsymbol{b}_{\mathrm{HC}} + \boldsymbol{x}_t\boldsymbol{W}_{\mathrm{XC}} + \boldsymbol{b}_{\mathrm{XC}}\right)$$

Attribution: Chris Olah

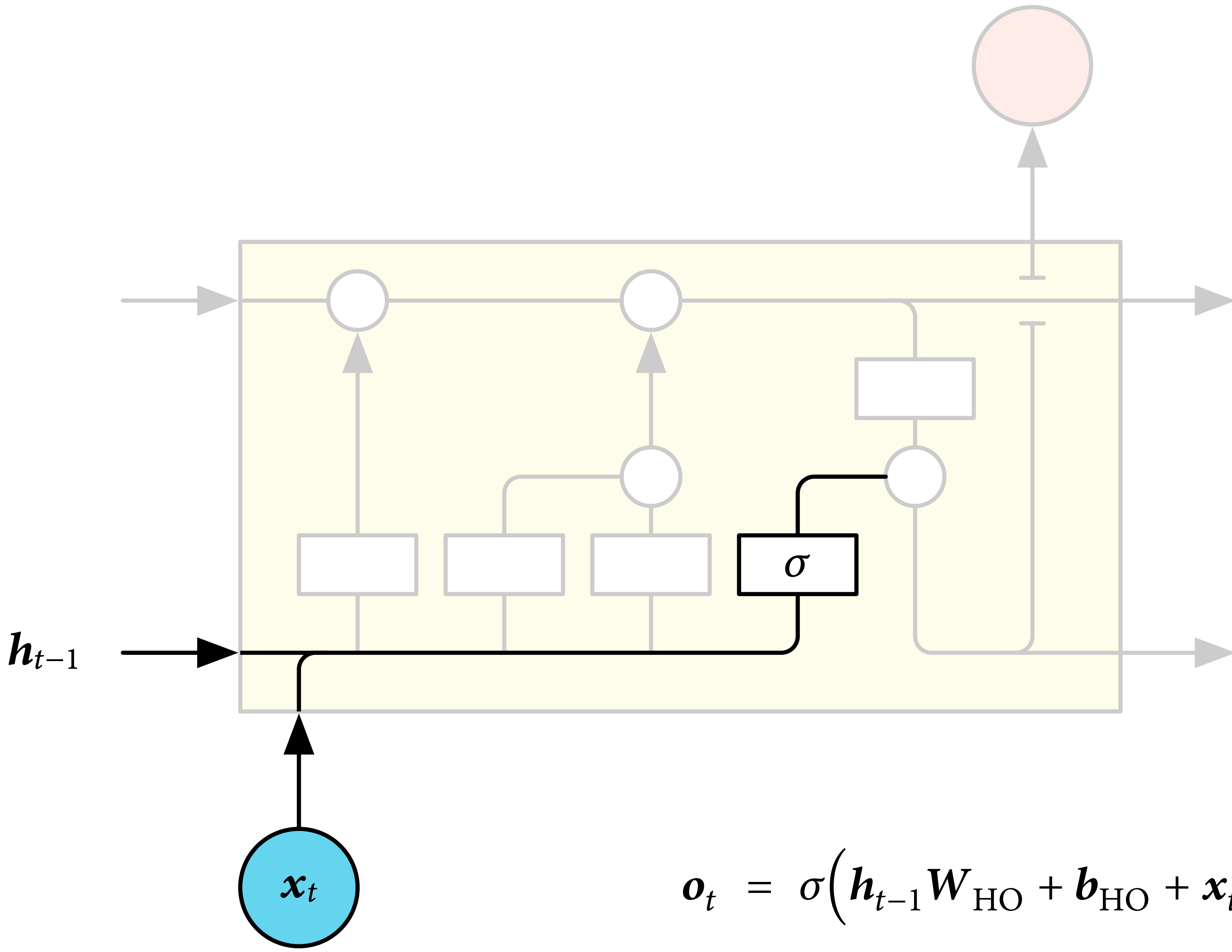
Memory cell update



$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

Attribution: Chris Olah

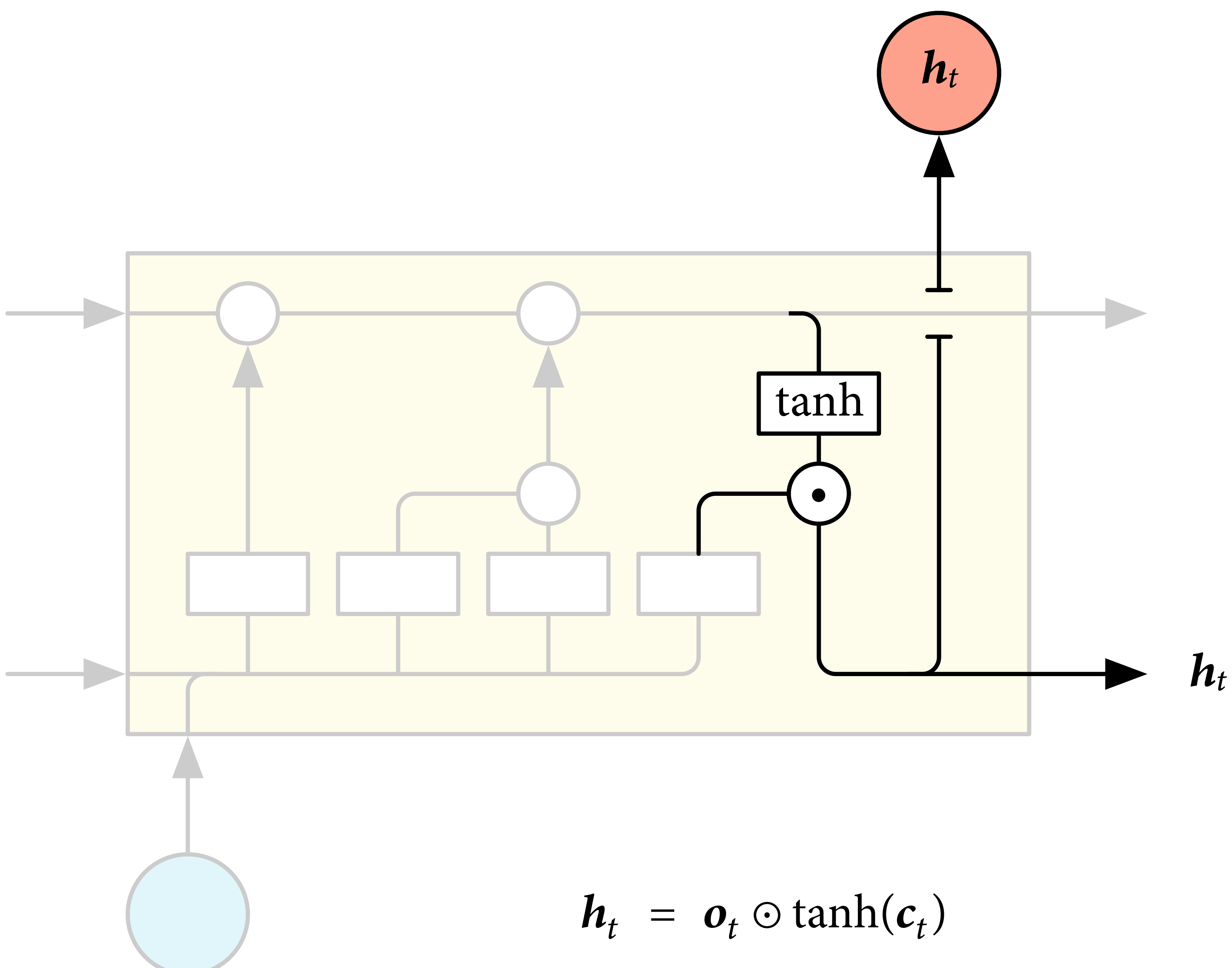
Output gate



$$\boldsymbol{o}_t = \sigma\left(\boldsymbol{h}_{t-1}\boldsymbol{W}_{\mathrm{HO}} + \boldsymbol{b}_{\mathrm{HO}} + \boldsymbol{x}_t\boldsymbol{W}_{\mathrm{XO}} + \boldsymbol{b}_{\mathrm{XO}}\right)$$

Attribution: Chris Olah

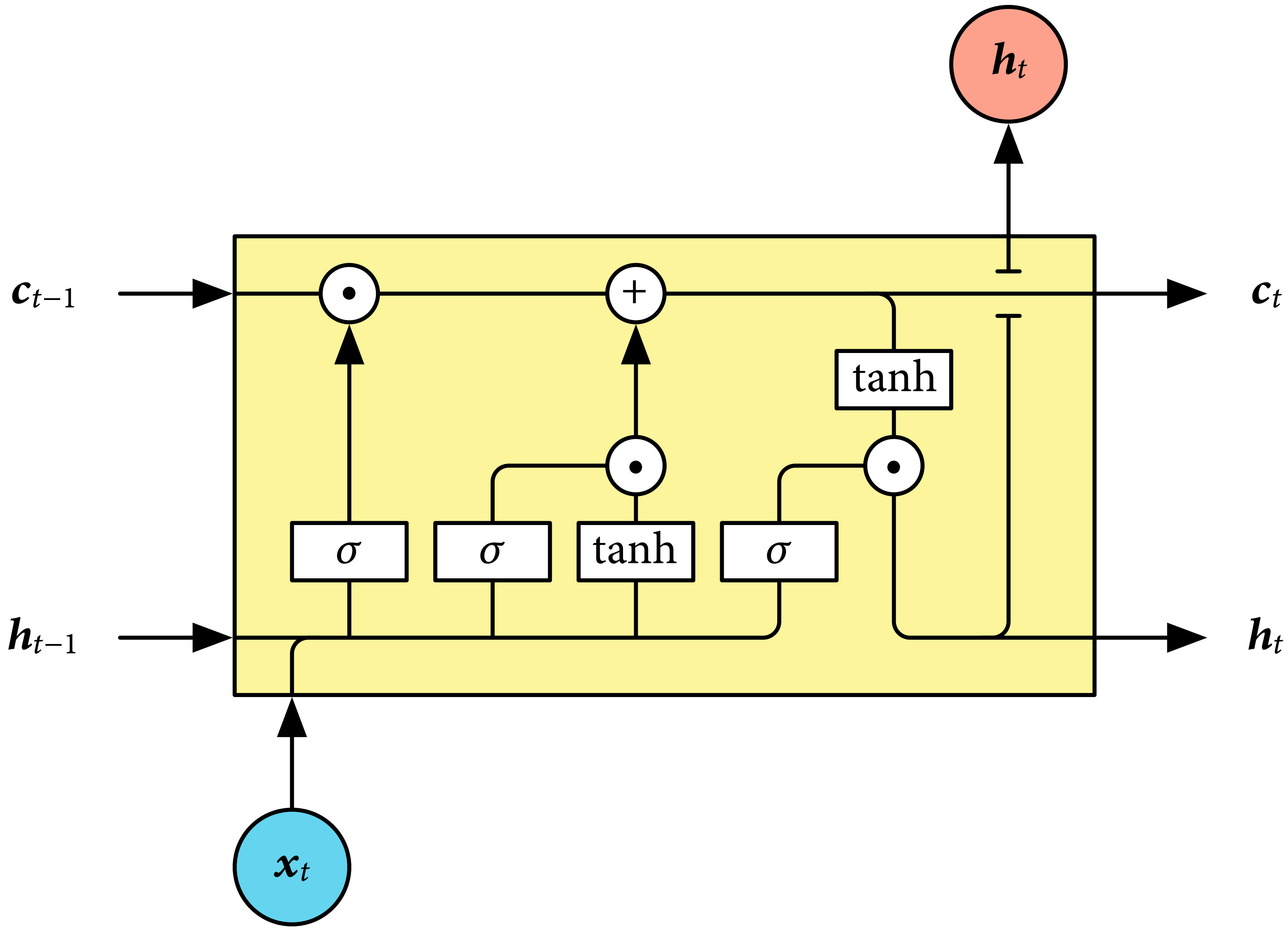
Output



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

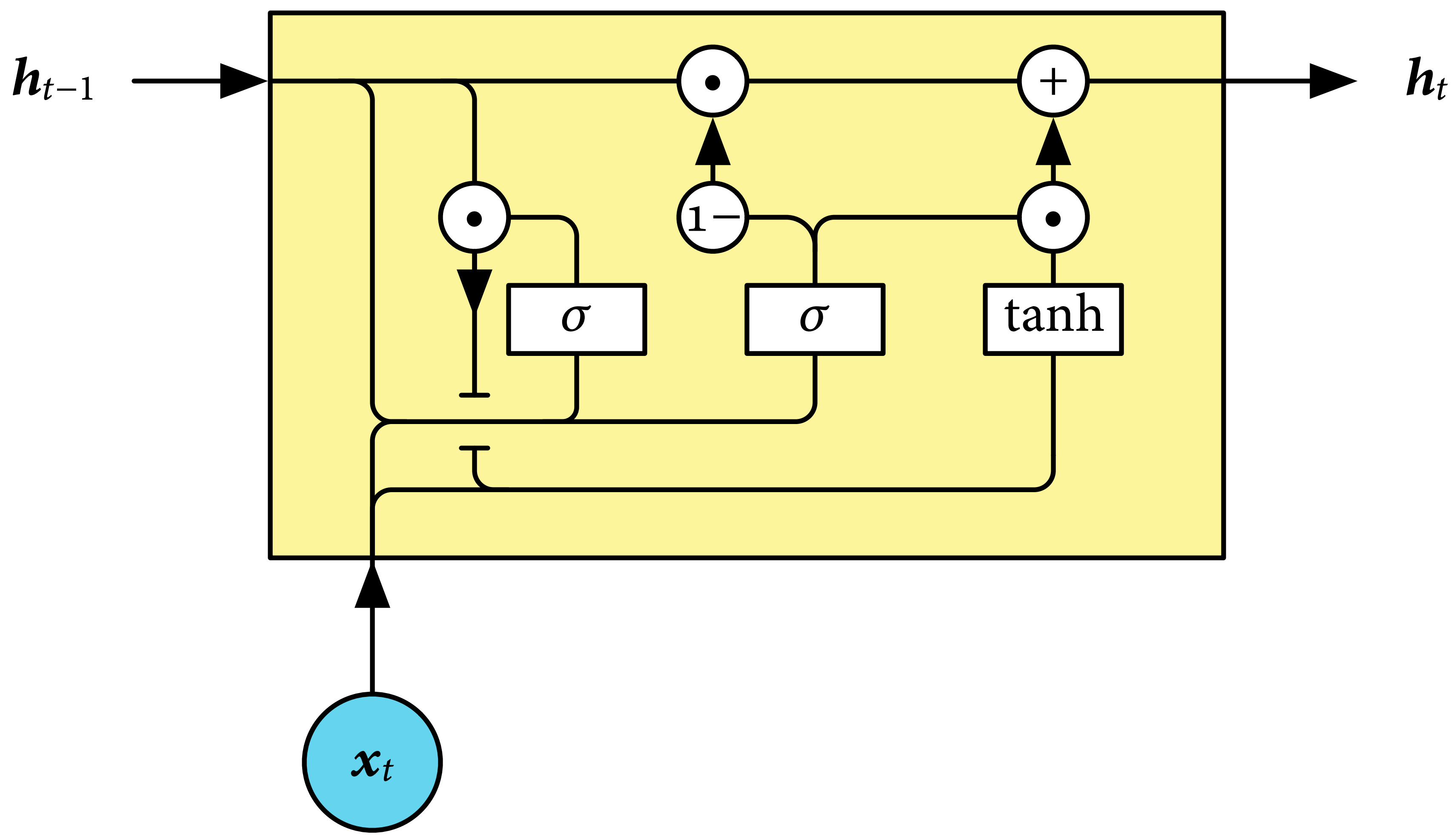
Attribution: Chris Olah

A look inside an LSTM cell



Attribution: Chris Olah

Gated Recurrent Unit (GRU)



Attribution: Chris Olah