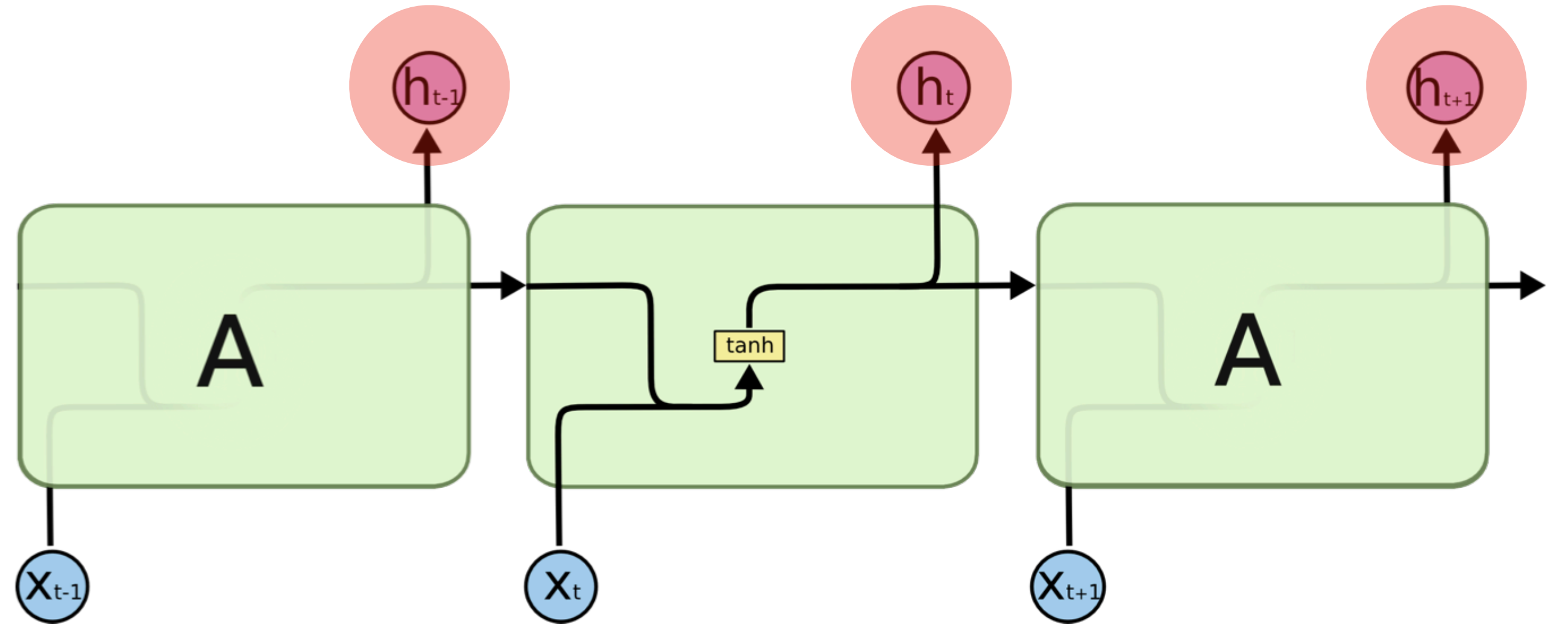


# 6. 게이트가 추가된 RNN

# 현재까지의 RNNLM

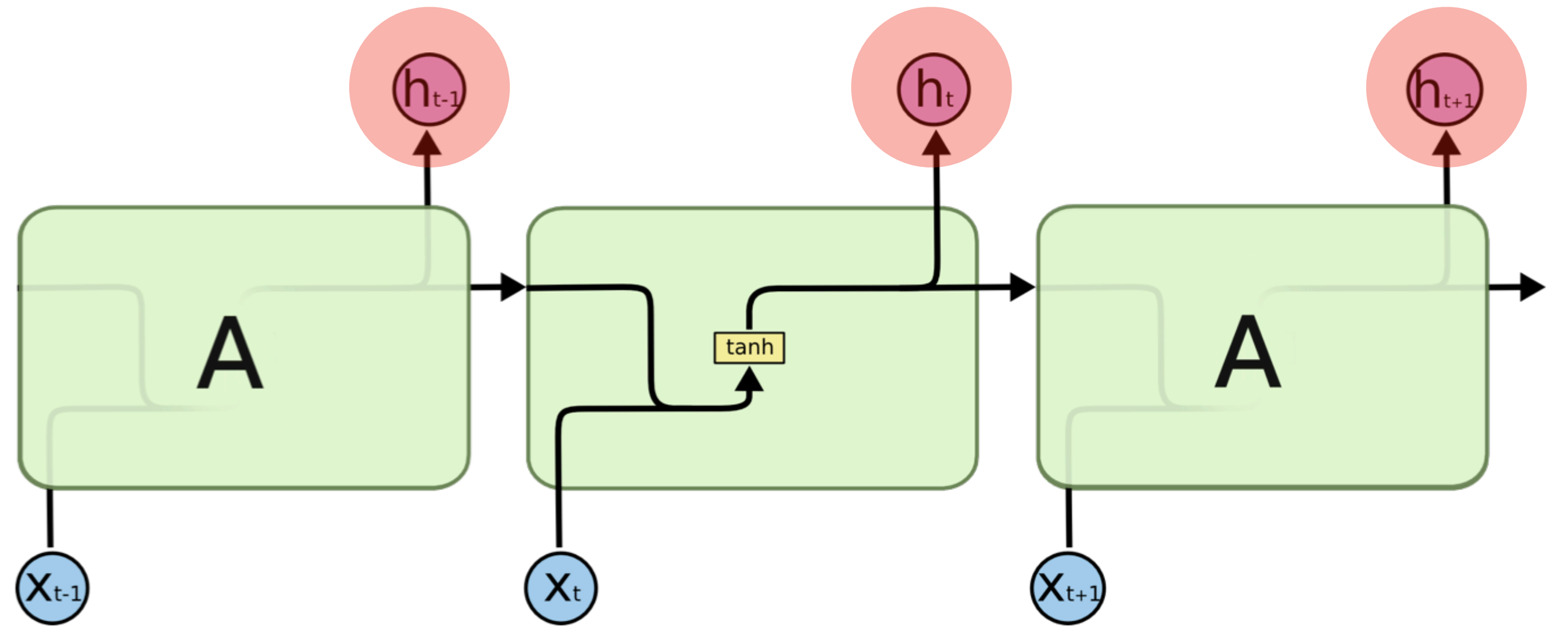
$$h_t = \tanh(h_{t-1}W_h + x_tW_x + b)$$



- 과거 정보를 저장하는 은닉 상태(노드)  $h_t$
- **장기 의존 관계**를 학습: 과거 방향으로 ‘의미 있는 기울기’를 전달함으로써 시간 방향의 의존 관계를 학습

# RNNLM 문제점

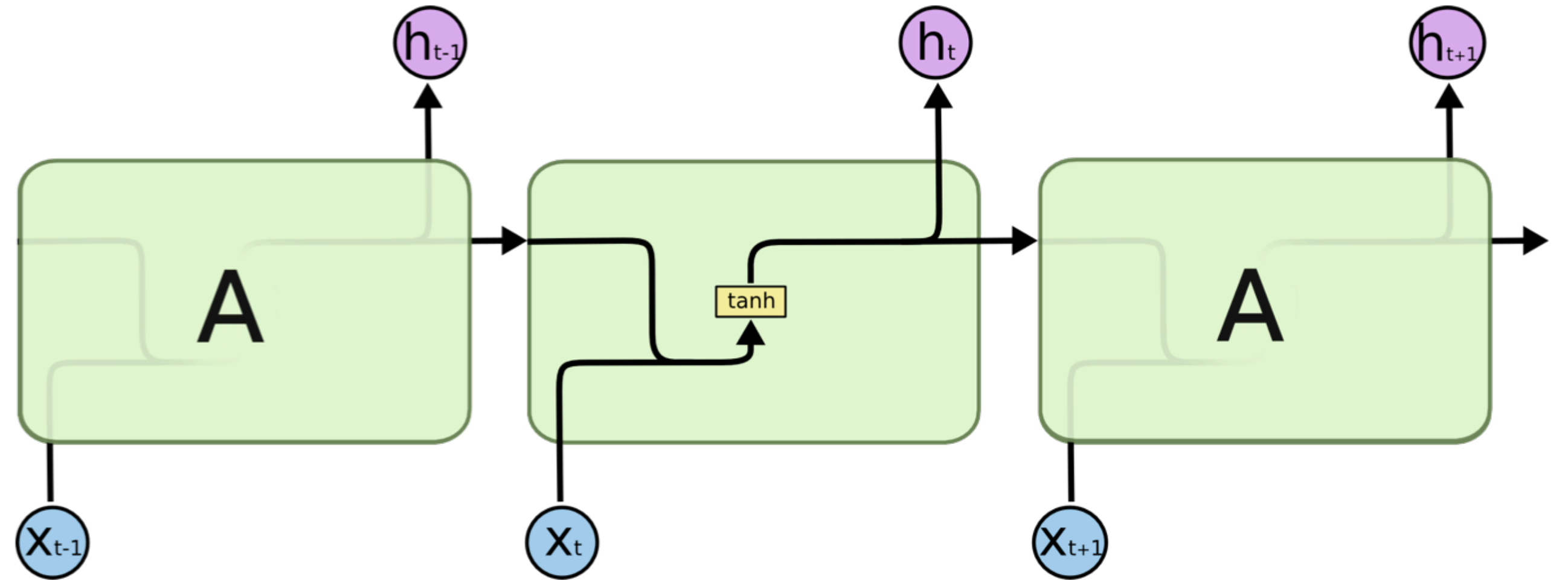
$$h_t = \tanh(h_{t-1}W_h + x_tW_x + b)$$



- 과거 정보를 저장하는 은닉 상태(노드)  $h_t$
- **장기 의존 관계**를 학습: 과거 방향으로 ‘의미 있는 기울기’를 전달함으로써 시간 방향의 의존 관계를 학습
  - Cons: 현재의 단순한 RNN 계층은 시간을 거슬러 올라갈수록 반영 X
    - $dhW_h^T \dots W_h^T$  만큼 가중치 반영하기에 기울기 소실/폭발

# RNNLM 해결책

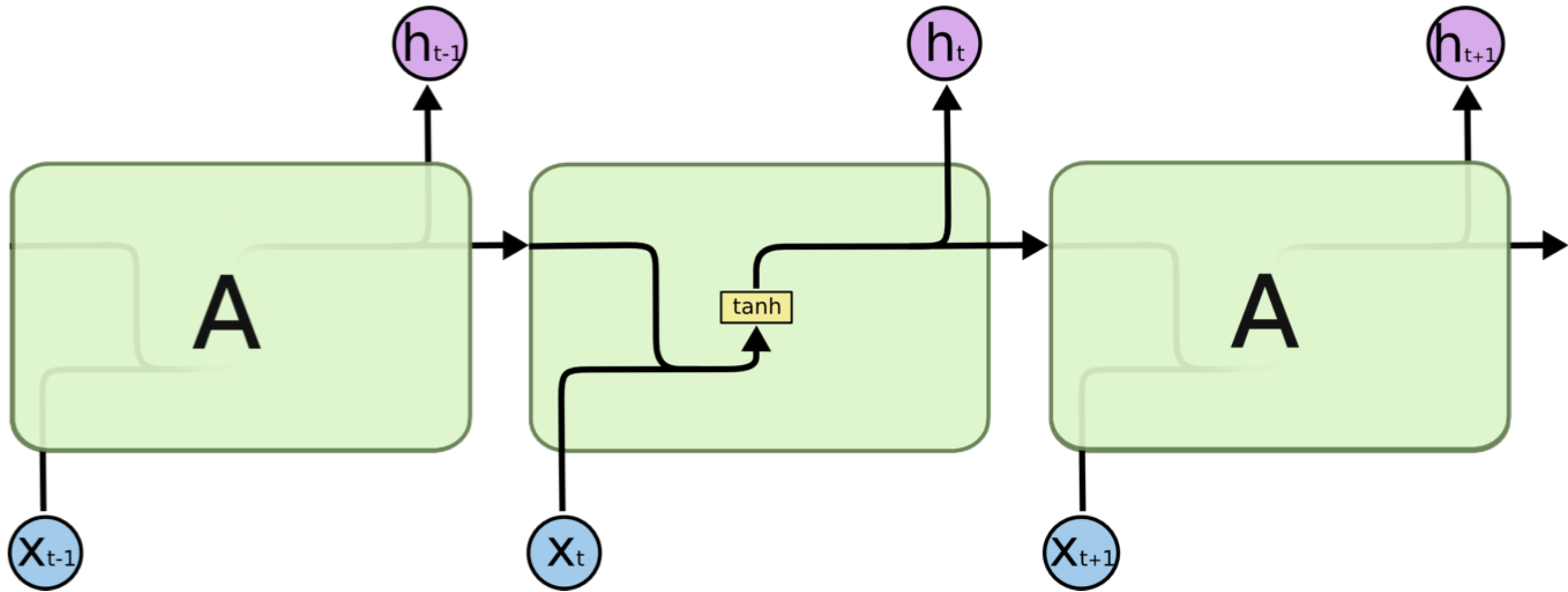
$$h_t = \tanh(h_{t-1}W_h + x_tW_x + b)$$



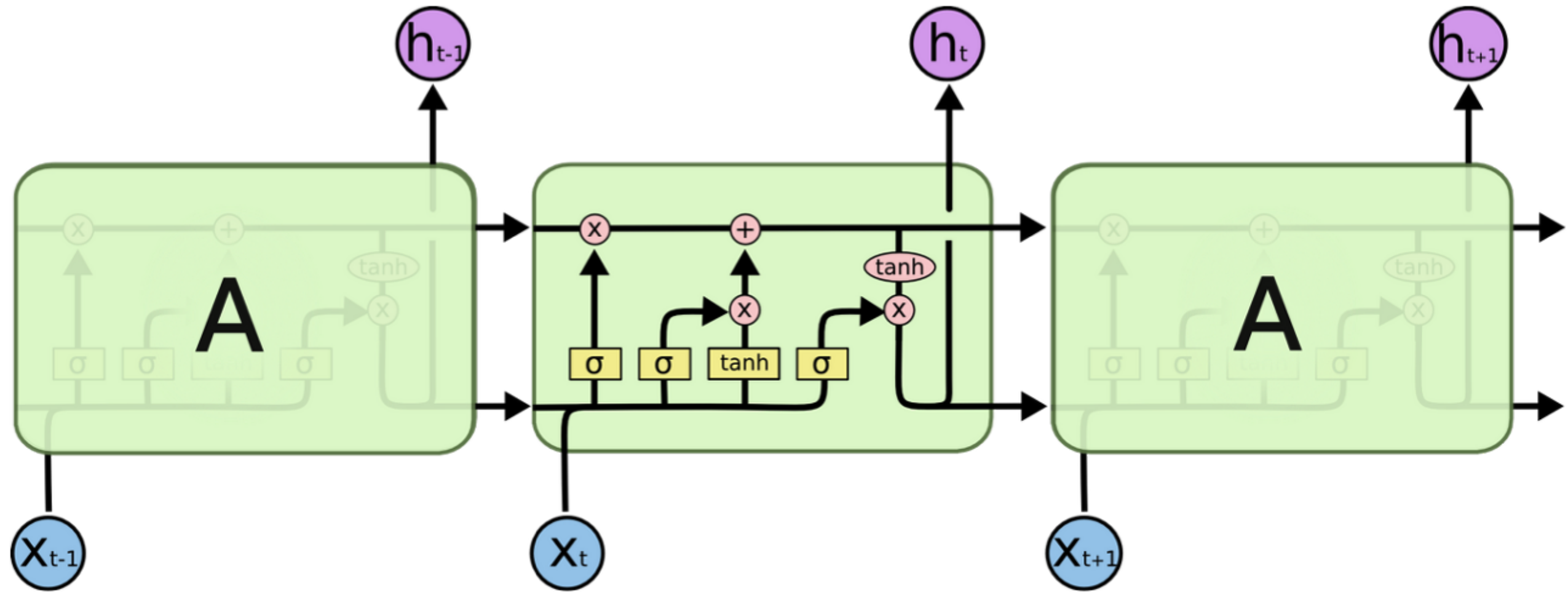
- 기울기가 tanh 노드를 지날 때마다 기울기 소실/폭발
  - 폭발 대책. Gradient clipping
    - 기울기의 L2 이 threshold보다 크면 기울기를 수정하는 방법
  - 소실 대책. Gated RNN(LSTM, GRU)

# RNN vs. LSTM

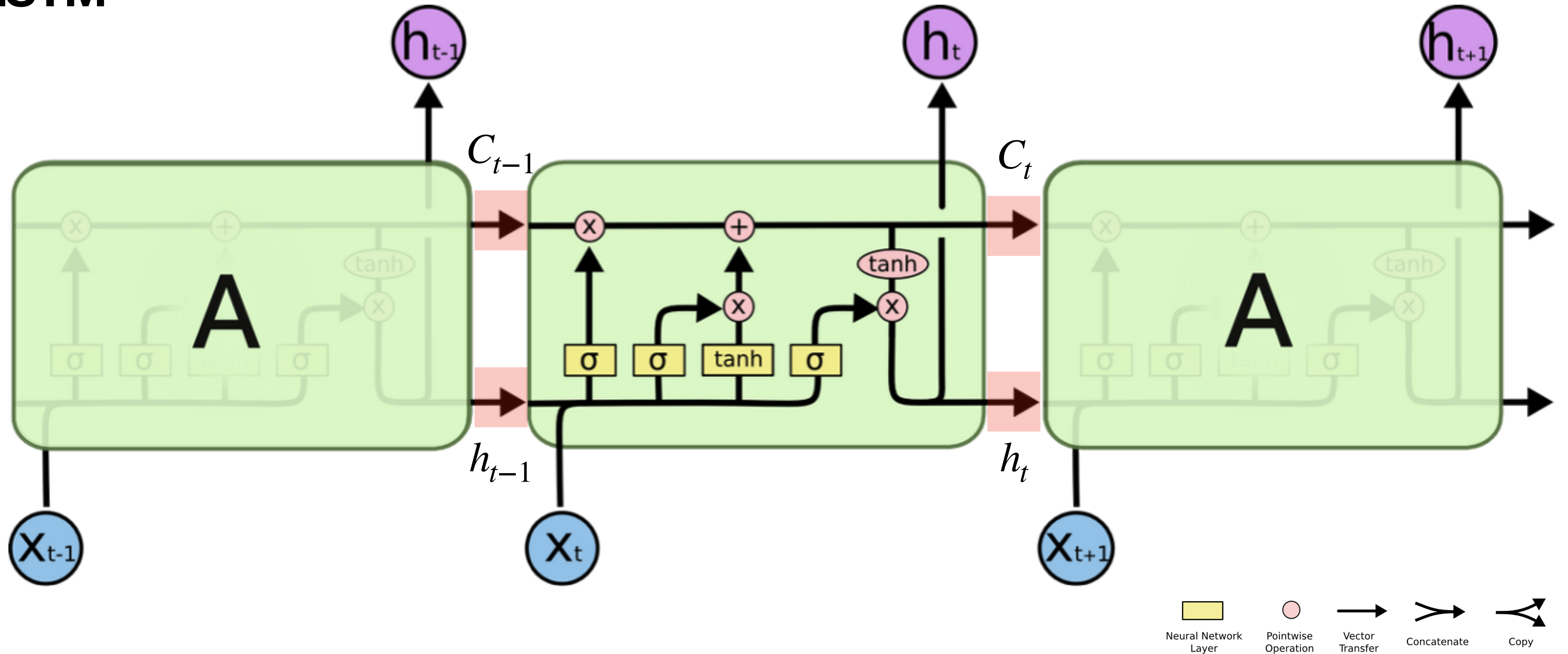
**RNN layer**  
(single layer)



**LSTM layer**  
(Four interacting layer)



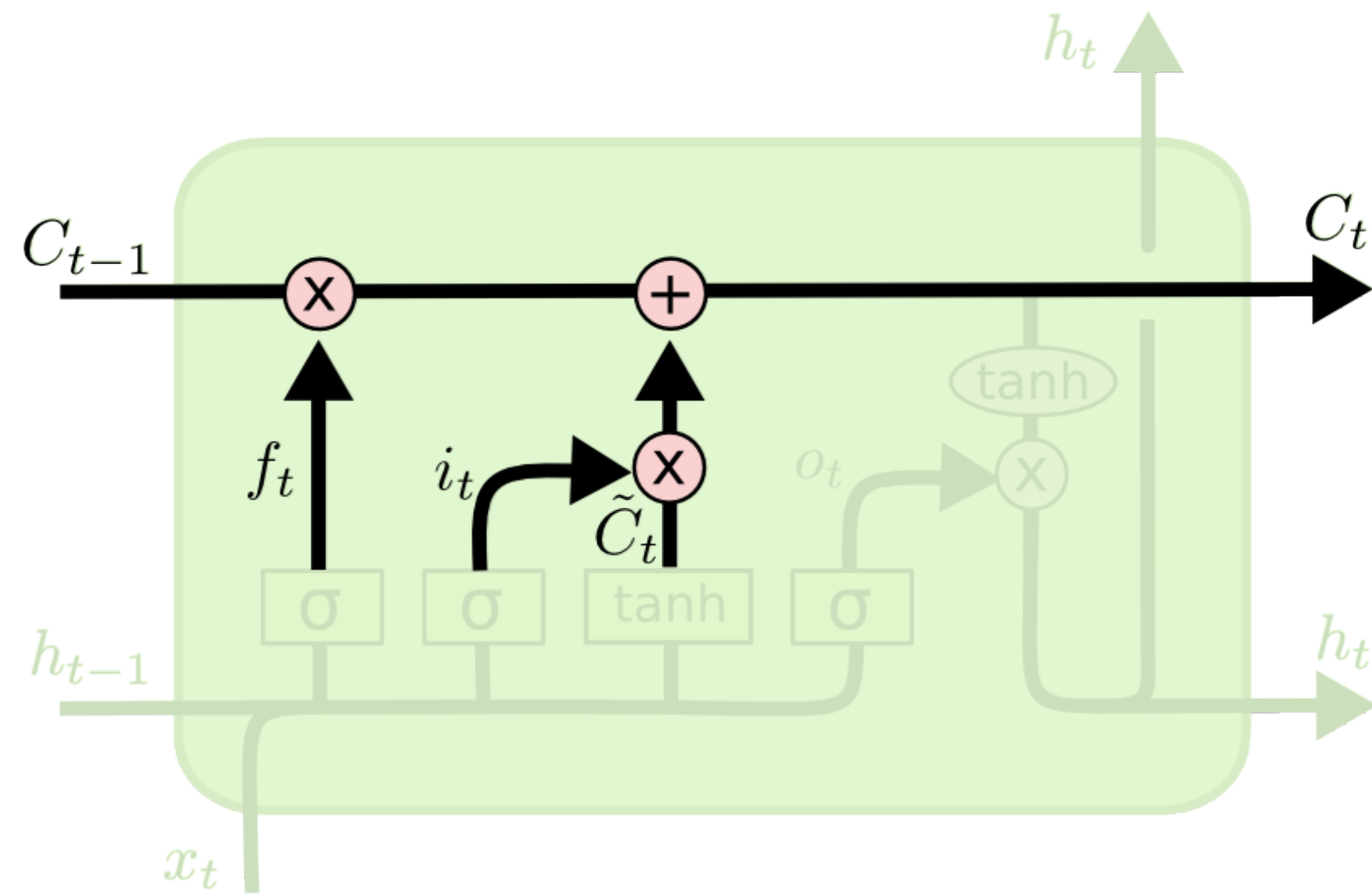
# LSTM



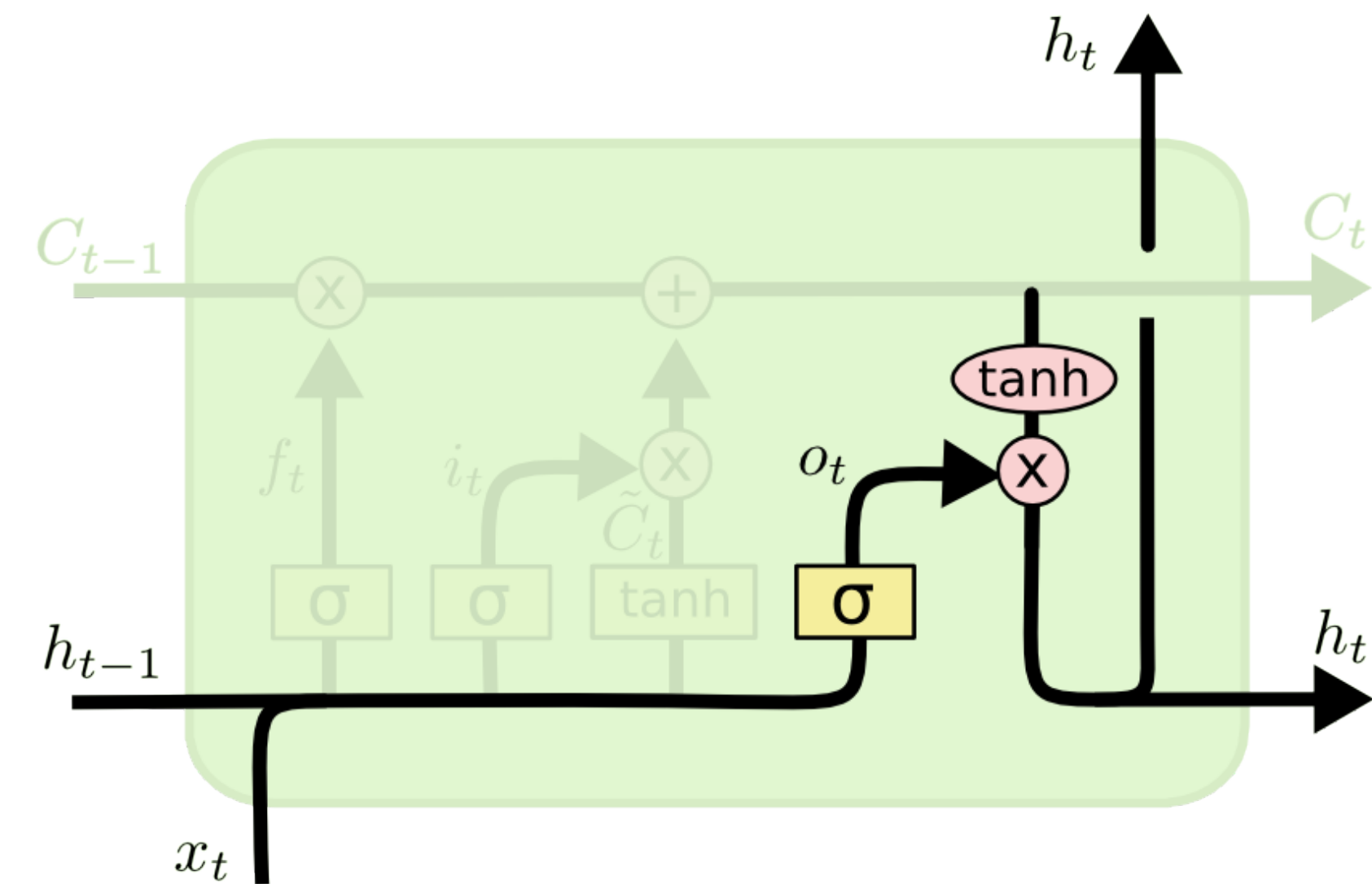
1. 과거 정보를 저장하는 은닉 상태(노드)  $h_t$  와 기억 셀  $c_t$
2. Gate: 게이트의 열림 상태를 조종
  - Input gate, Output gate, Forget gate

# LSTM

1. 과거 정보를 저장하는 은닉 상태  $h_t$  와 기억 셀  $c_t$ 
  - 기억 셀  $C_t$ : 시각  $t$ 에서의 LSTM 기억 = 시각  $0 \sim t$  까지의 필요한 모든 정보 저장
  - 기억 셀  $C_t$ 를 바탕으로 외부 계층에 은닉 상태  $h_t$ 를 제공



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

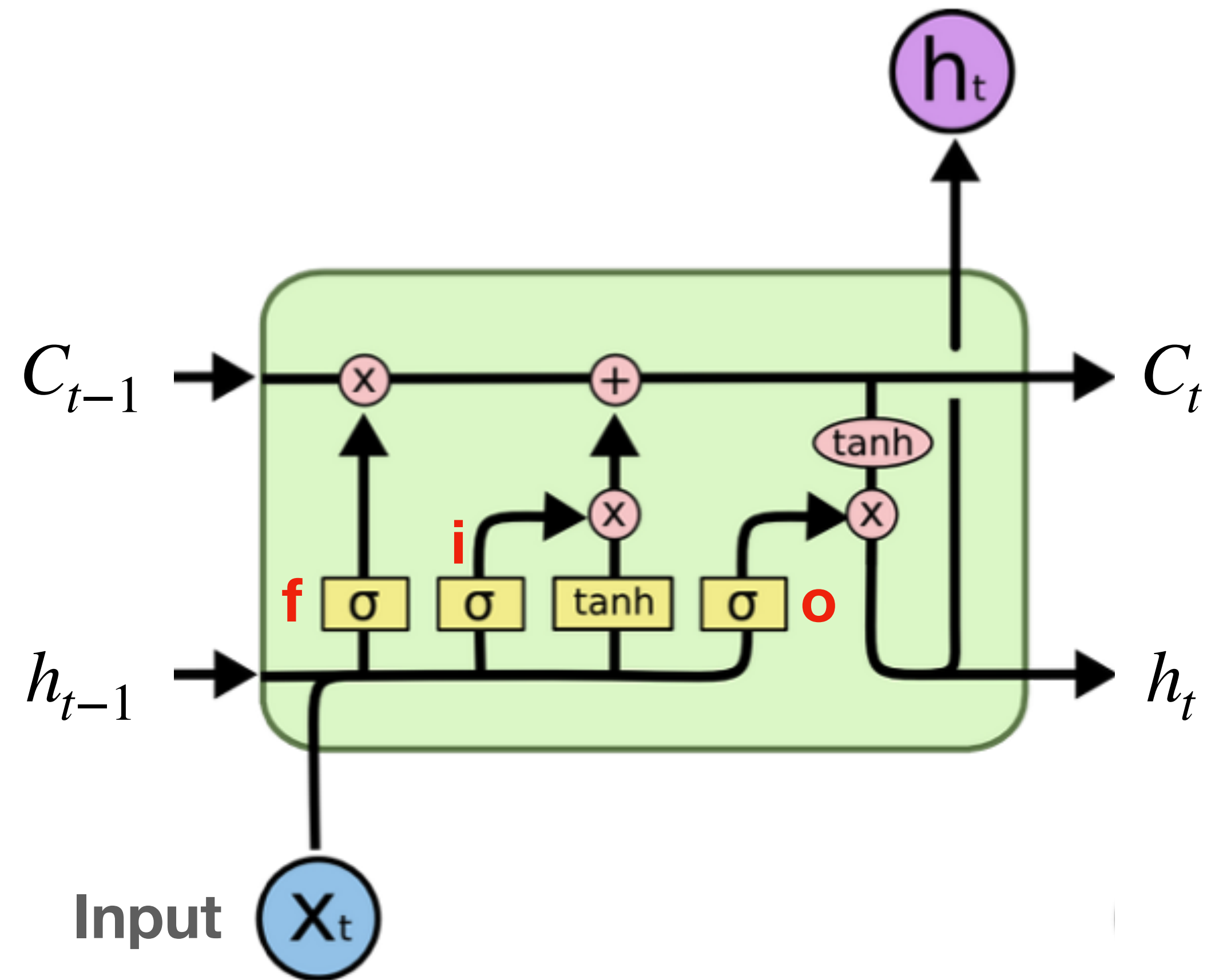


$$o_t = \sigma (W_o [h_{t-1}, x_t] + b_o)$$

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

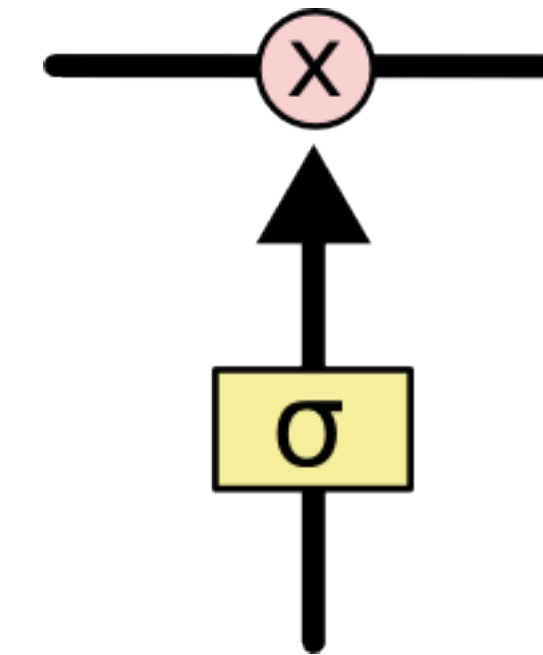


# LSTM



2. Gate: 게이트의 열림 상태를 조종

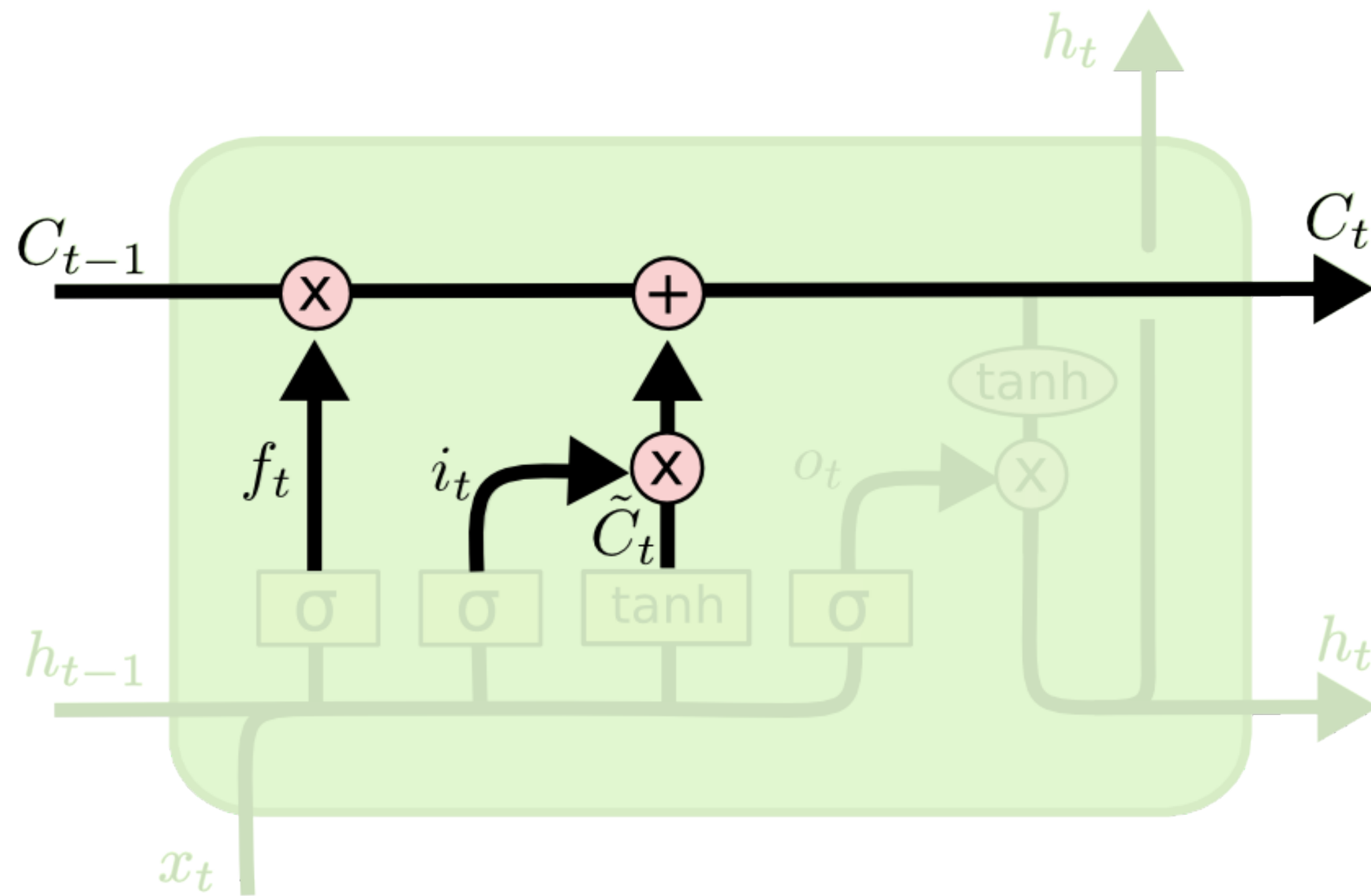
- Format: sigmoid



- Forget gate  $f$
- Output gate  $o$
- Input gate  $i$

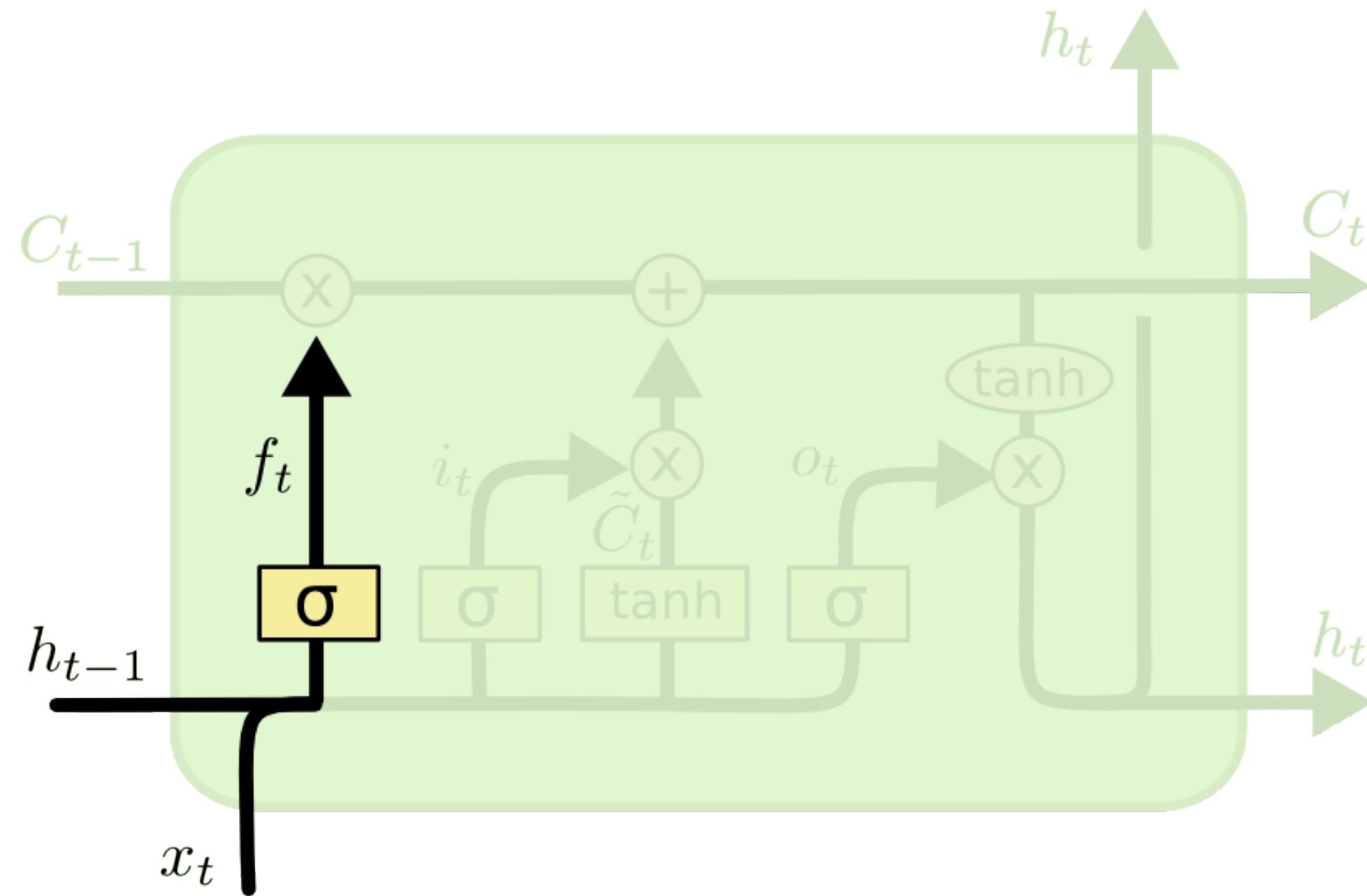


# LSTM



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

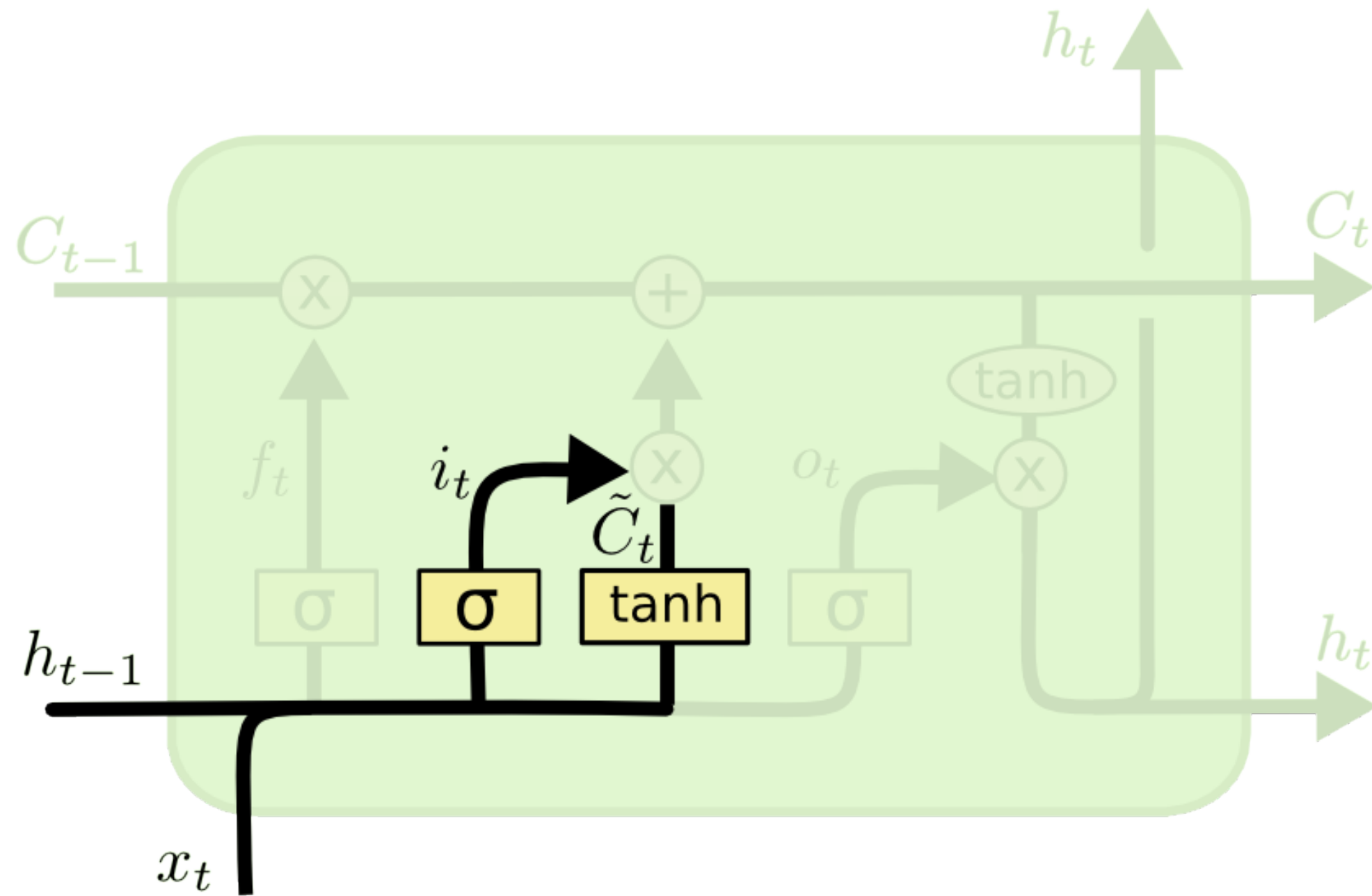
# LSTM: forget gate $f_t$



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

- Gate  $f_t$ : 이전 은닉 상태  $h_{t-1}$ 와 입력값  $x_t$ 를 바탕으로 기억 셀  $C_{t-1}$  중에 필요한 기억만 남기도록 0.0~1.0의 가중치 설정
- 이전 기억셀에 반영하여 현재 기억셀  $C_t$  생성 예정

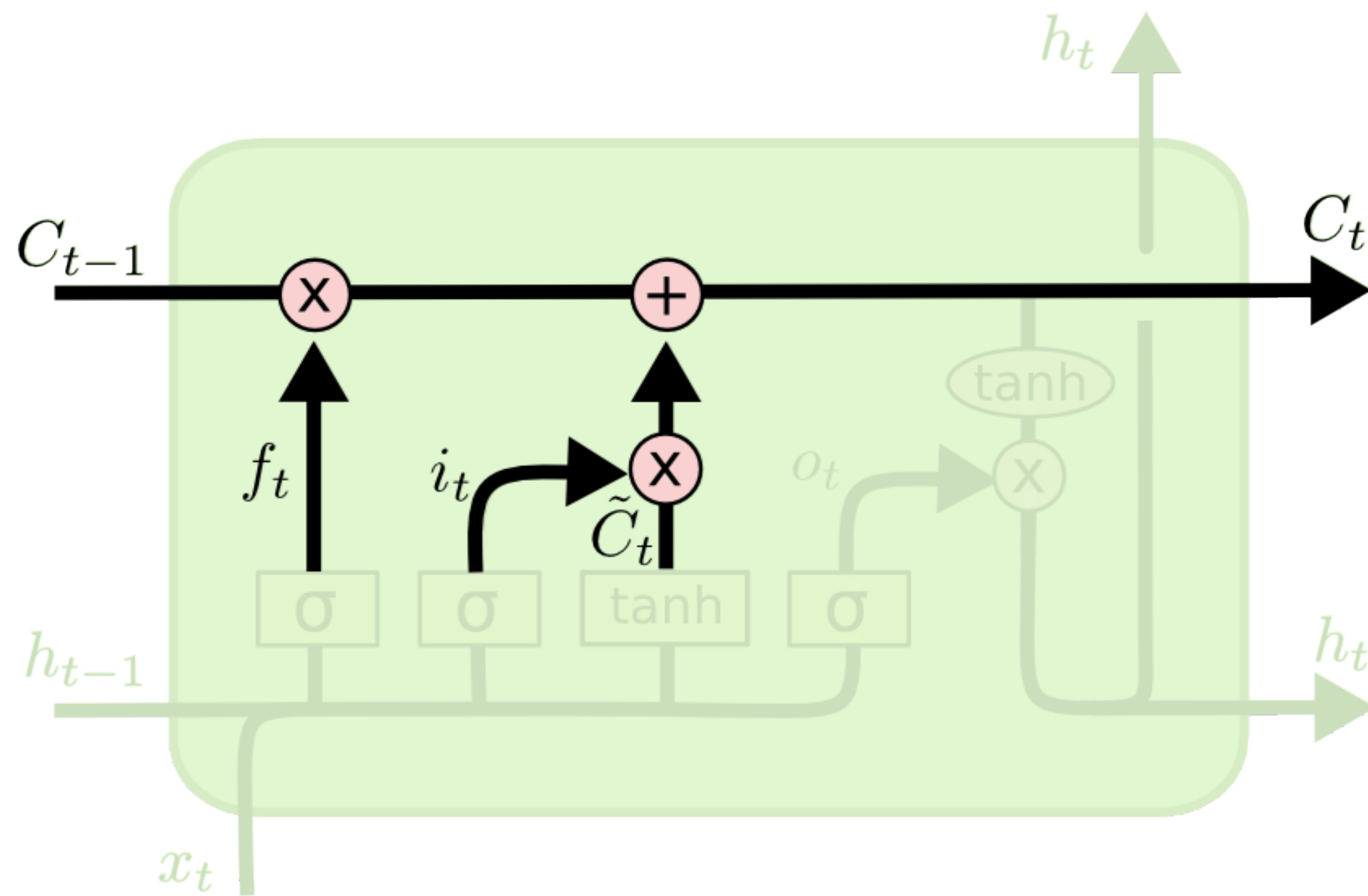
# LSTM: input gate $i_t$



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

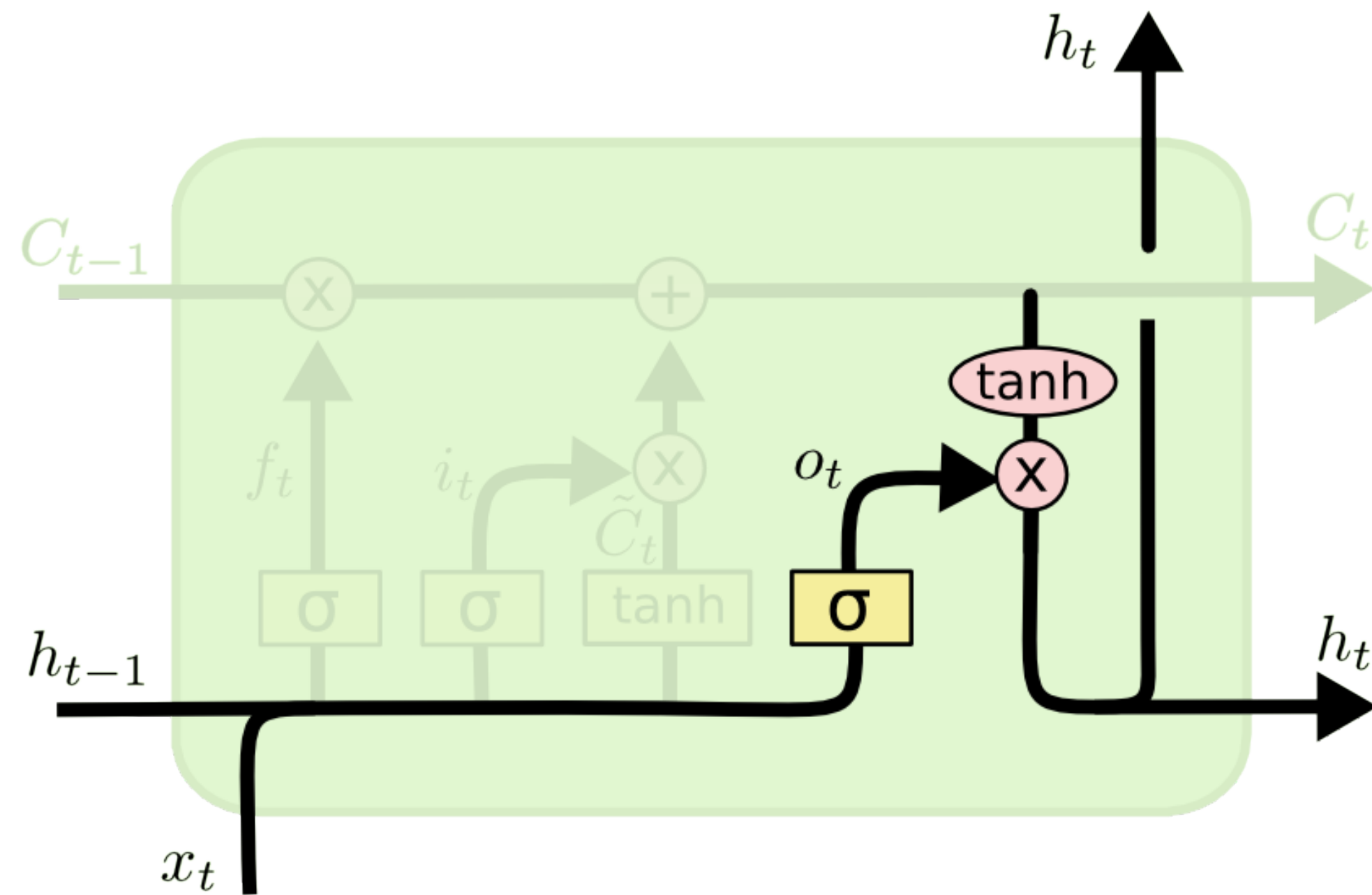
- Gate  $i_t$ : 새로 추가되는 정보로써의 가치를 0.0~1.0의 가중치로 판단
- $\tilde{C}_t$

## LSTM: input gate $i_t$



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

# LSTM: output gate $o_t$

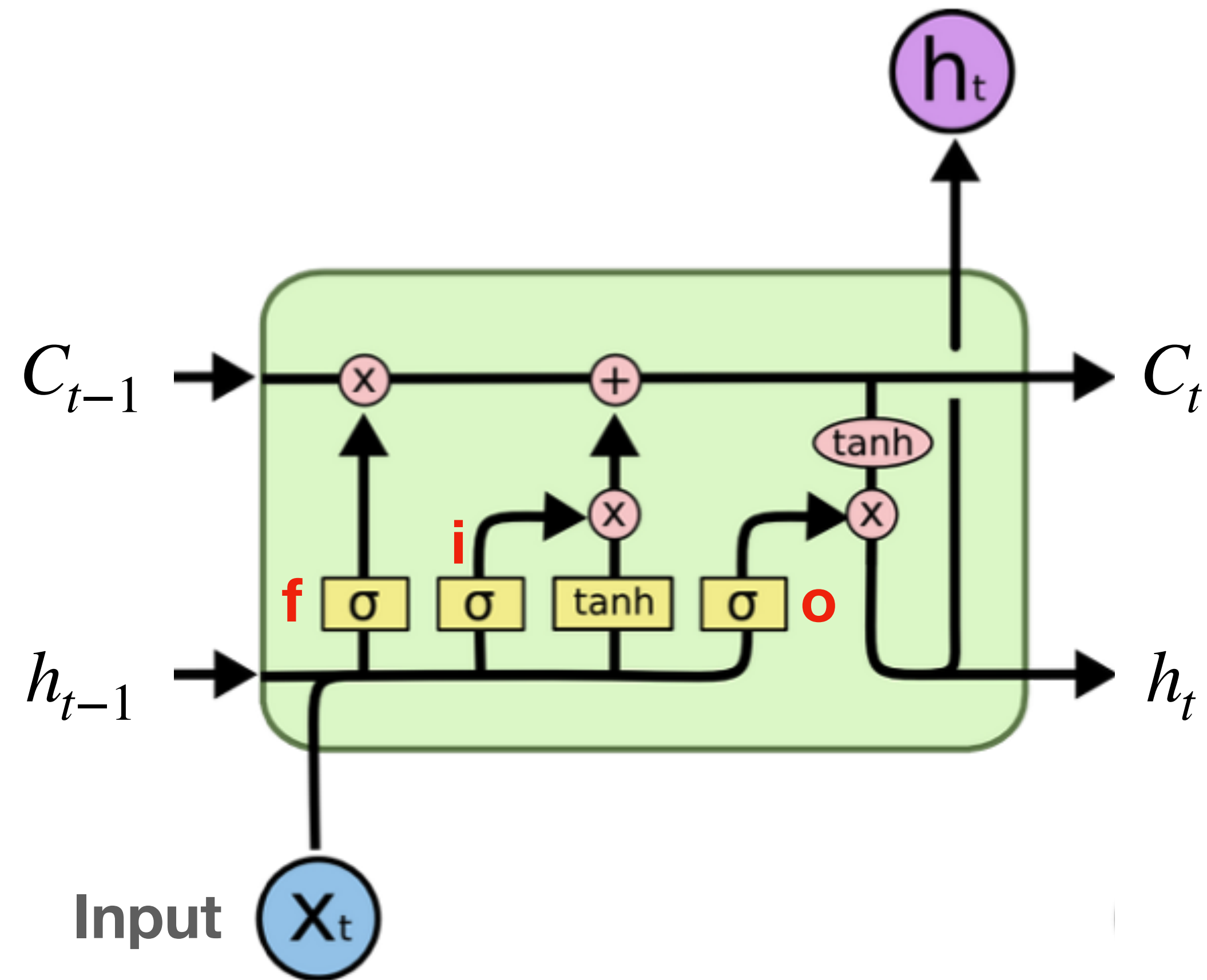


$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

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

- Gate  $o_t$ : 이전 은닉 상태  $h_{t-1}$ 와 입력  $x_t$ 로부터 다음 은닉 상태에 얼마나 중요한가 0.0~1.0의 가중치 설정
- $h_t = o_t \odot \tanh(C_t)$ 
  - 기억 셀  $C_t$ 를 바탕으로 외부 계층에 은닉 상태  $h_t$ 를 제공

# LSTM



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

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

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

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

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

# Additional Improvements

- Dropout
- Variational Dropout
- Weight tying 가중치 공유