

# 딥러닝팀

## 1팀

이수경  
이승우  
이은서  
주혜인  
홍현경

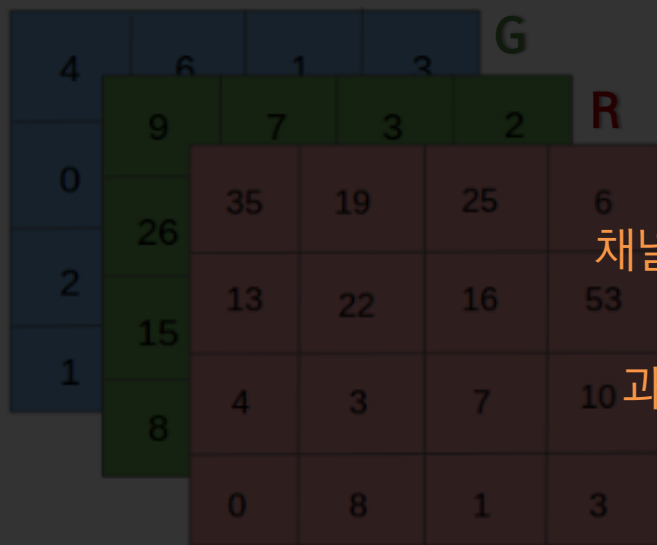
# 1 합성곱 신경망(CNN)

- 이미지 데이터의 특징

## DNN의 한계 2) 채널 별 특성 파악 불가



이미지 데이터를 DNN으로 다루는 경우



공간 정보 손실

채널 특성 파악 어려움

과도하게 큰 연산량

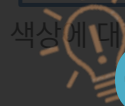
이미지는 RGB 각 채널마다 값의 특성이 다름

그러나 DNN은 각 채널의 값들을 평균 내어 하나의 채널 사용

채널 별 특성 파악 어려움

3 Colour Channels

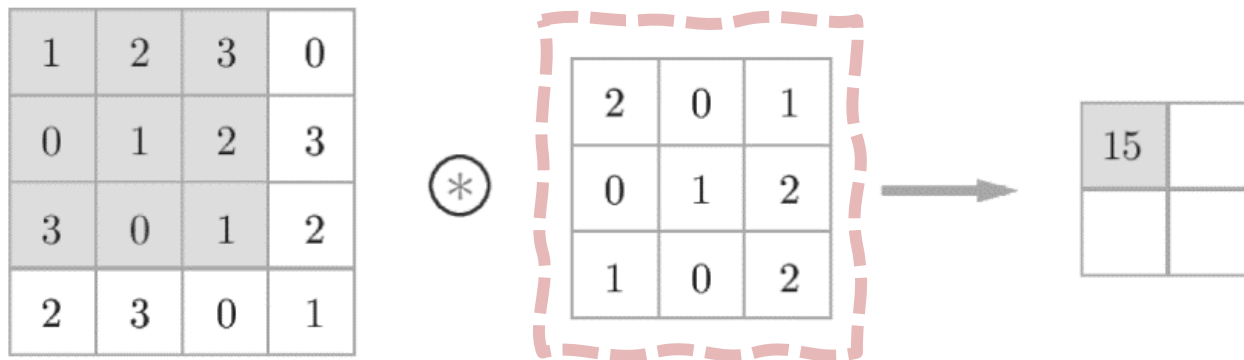
색상에 대한 정보를 담아 두는 곳



## CNN 알고리즘 이용

- Convolution Layer

## (1) 필터



필터 : DNN의 ‘가중치’에 해당

필터의 윈도우를 일정 간격으로 이동하며 입력 데이터에 필터 적용

## ● Convolution Layer

### (4) 패딩

0	0	0	0	0	0
0	1	2	3	0	0
0	0	1	2	3	0
0	3	0	1	2	0
0	2	3	0	1	0
0	0	0	0	0	0

raw size :  $4 \times 4$   
after padding :  $6 \times 6$

⊛

2	0	1
0	1	2
1	0	2



7	12	10	2
4	15	16	10
10	6	15	6
8	10	4	3

Output size :  $4 \times 4$

모서리 정보가 적게 반영되는 현상 완화

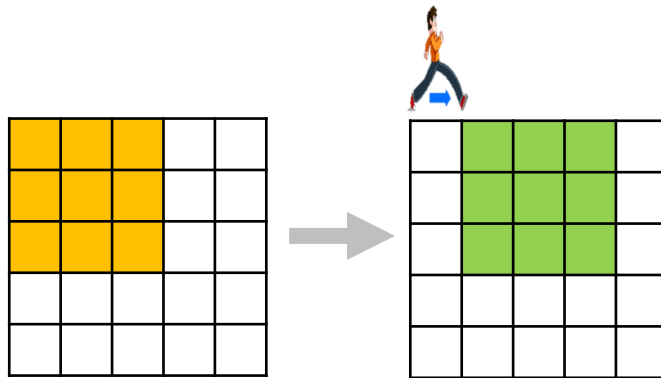
output featuremap의 크기 유지

# 1 합성곱 신경망(CNN)

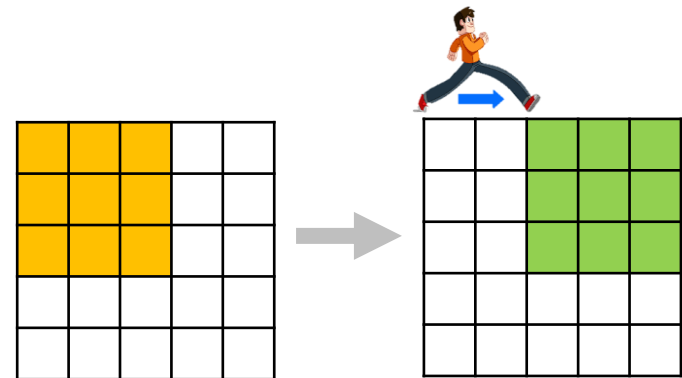
필터 합성곱 연산 편향 패딩 스트라이드

## ● Convolution Layer

### (5) 스트라이드



스트라이드 = 1



스트라이드 = 2

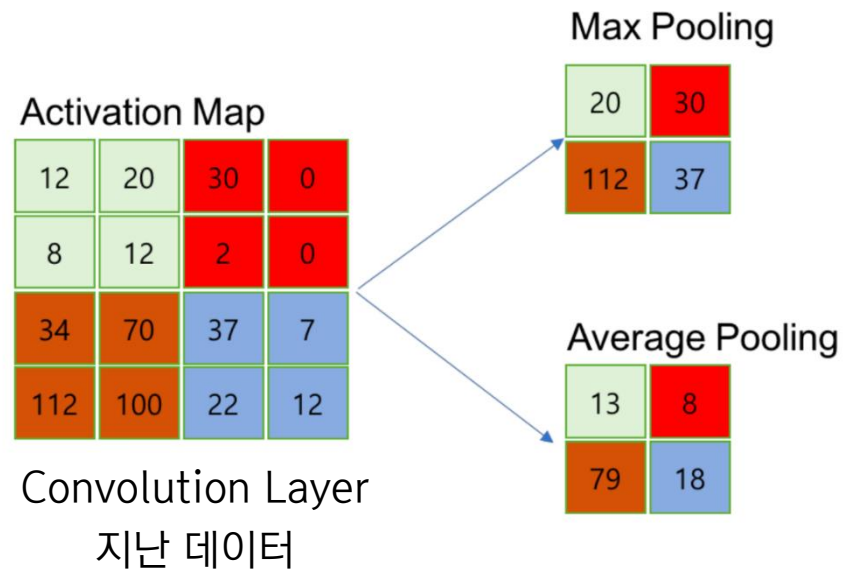
stride = 보폭

필터가 한번에 이동하는 **간격**

# 1 합성곱 신경망(CNN)

- Pooling Layer

## 풀링 Pooling



피쳐맵의 **크기**를 줄이는 연산

이미지의 크기를 줄여 **특징**을 잘 표현하는 값을 뽑아냄

- RNN이란?

# 순환신경망 Recurrent Neural Network

$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state      some function with parameters W      old state      input vector at some time step

RNN의 출력은 **이전의 모든 입력**에 영향을 받음  
➔ 순차적 데이터인 자연어의 시간적 특징 반영 가능

- RNN 모델 구조

# RNN의 한계점

## 장기 의존성 문제

어제 어제 어제 어제 어제 어제 어제  
어제 주문했던 로제 떡볶이의 맵기는?



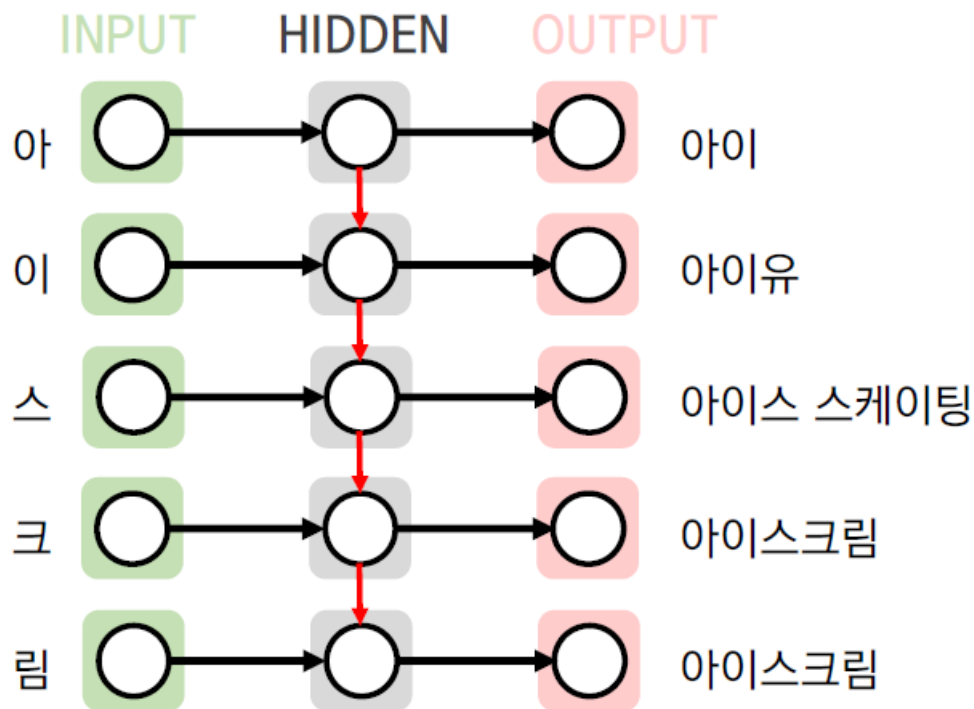
→ 은닉층 값 연쇄적으로 연결  
앞쪽의 타입 스텝 영향력 ↓ (잊혀짐)



## 2 RNN

### ● RNN 모델 구조

## RNN의 구조



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

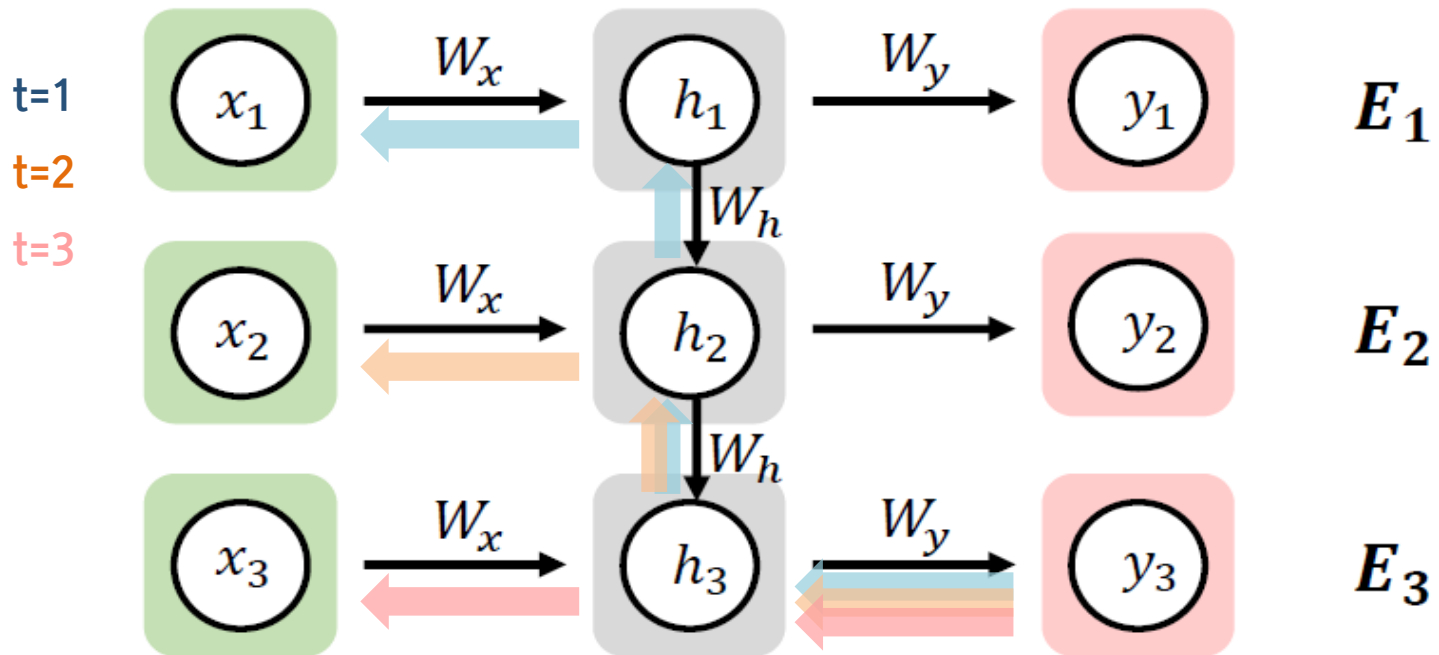
$$y_t = f(W_v h_t + b)$$

↳ 각 시점 별 출력값  
시점마다 다르게 출력

## 2 RNN

- RNN의 역전파

### BPTT Back Propagation Through Time



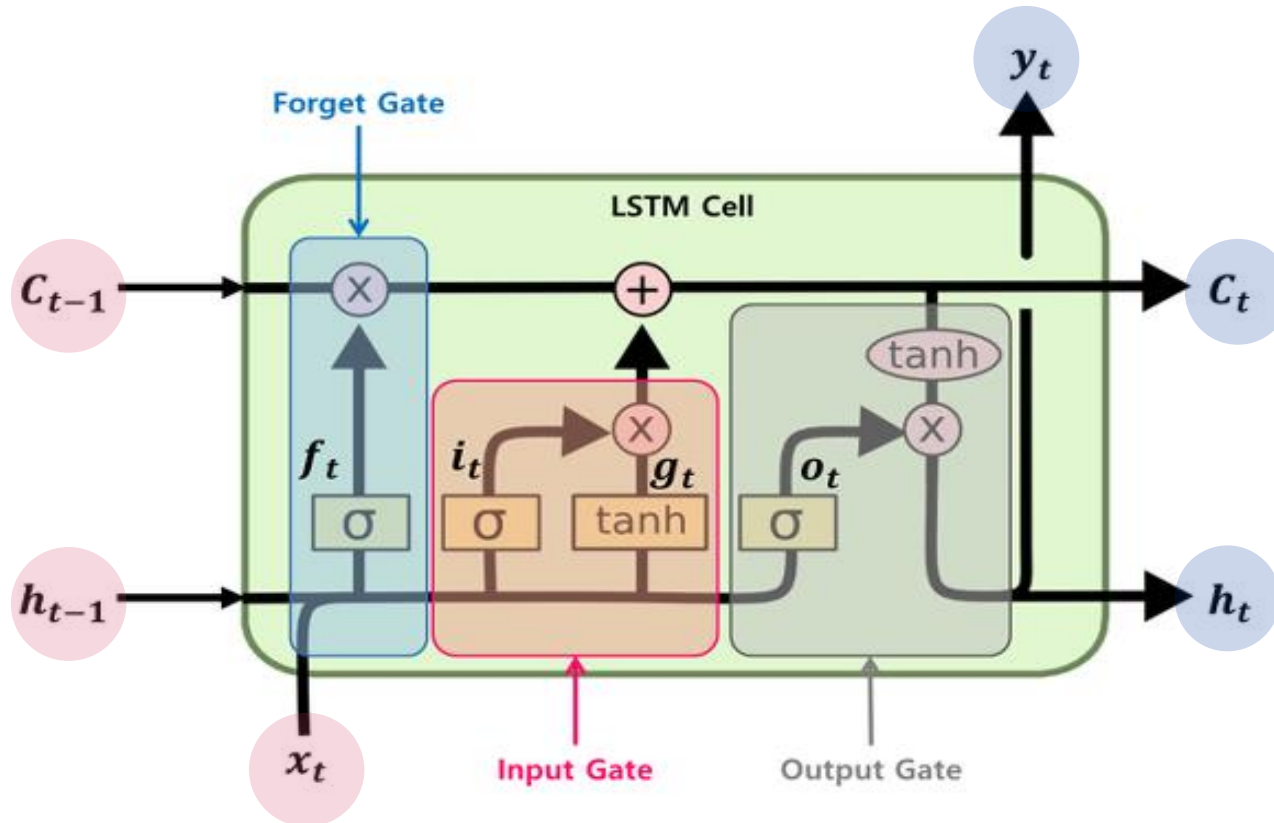
매 시점마다  $W_x$ 에 대해 역전파가 이루어짐

모든 시점의 역전파 값을 더하여  $W_x$ 에 대한 역전파 값으로 사용

# 3 LSTM

Forget input output

- LSTM의 구조



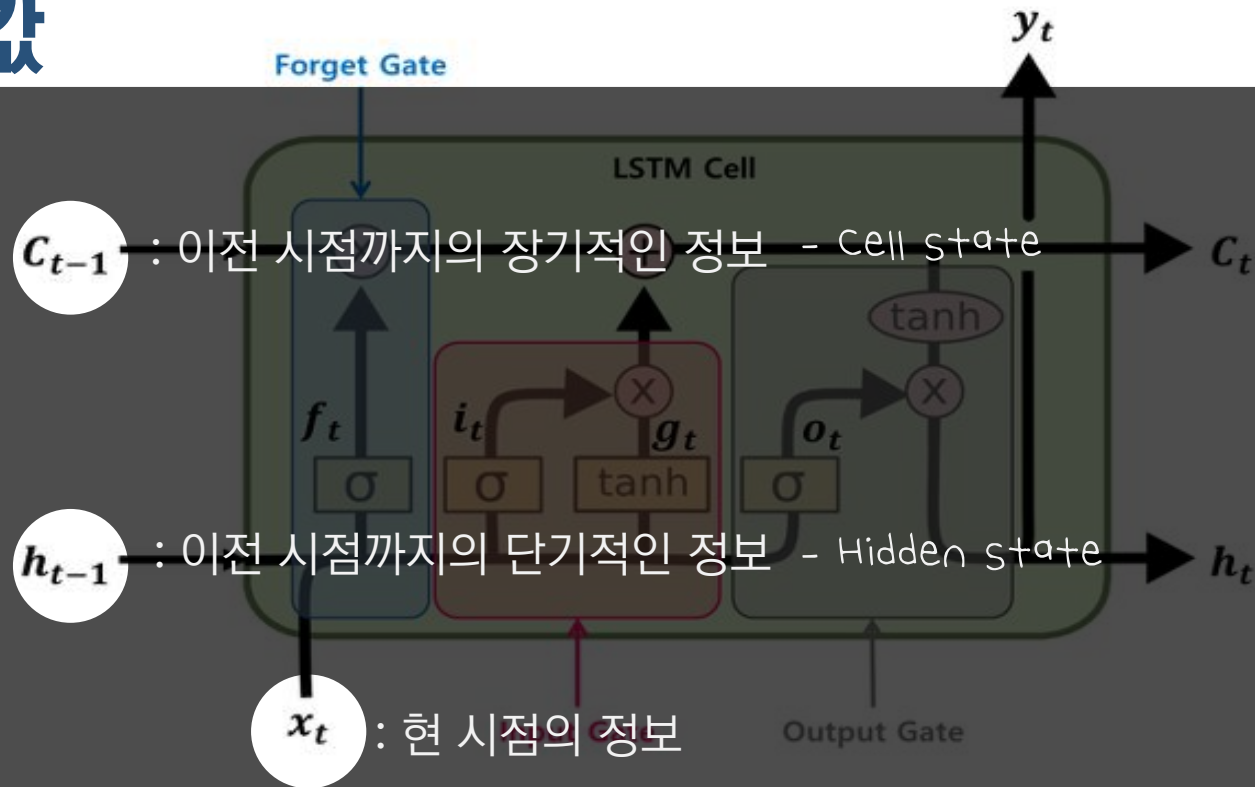
3 Inputs

3 Gates

3 Outputs

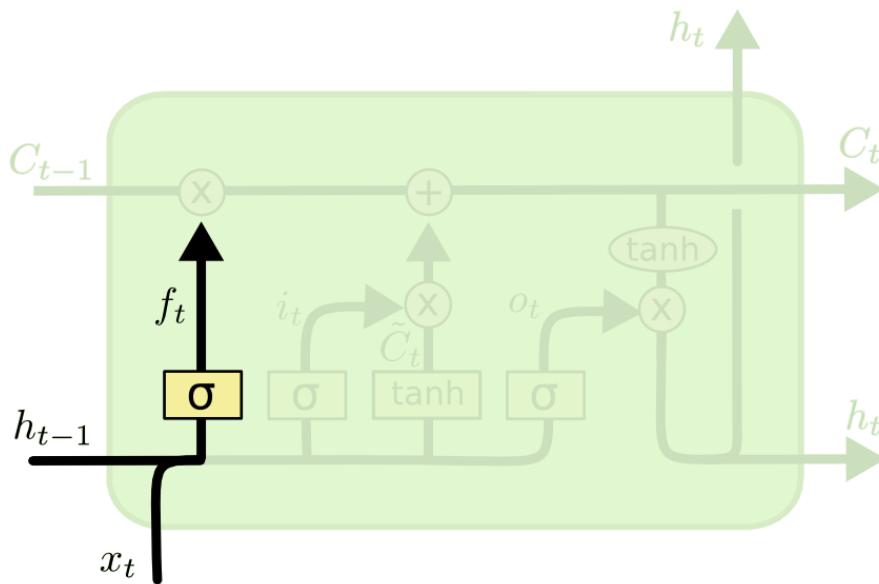
## ● LSTM의 구조

### 입력값



- LSTM의 구조

## Gate – Forget Gate



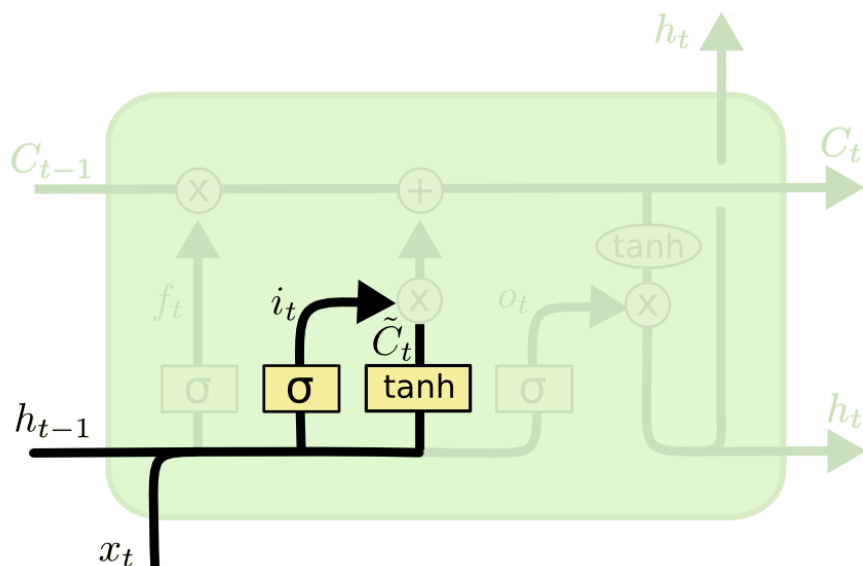
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

$$\hookrightarrow 0 \leq f_t \leq 1$$

$h_{t-1}$ 와  $x_t$ 를 고려해 이전 시점까지의 **장기적인 정보**  $C_{t-1}$ 에서  
어떤 정보가 쓸모가 없고 이를 **얼마나 잊을 지** 결정

- LSTM의 구조

## Gate – Input Gate



정보를 얼마나 업데이트 할 것인지

$$0 \leq i_t \leq 1$$

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

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

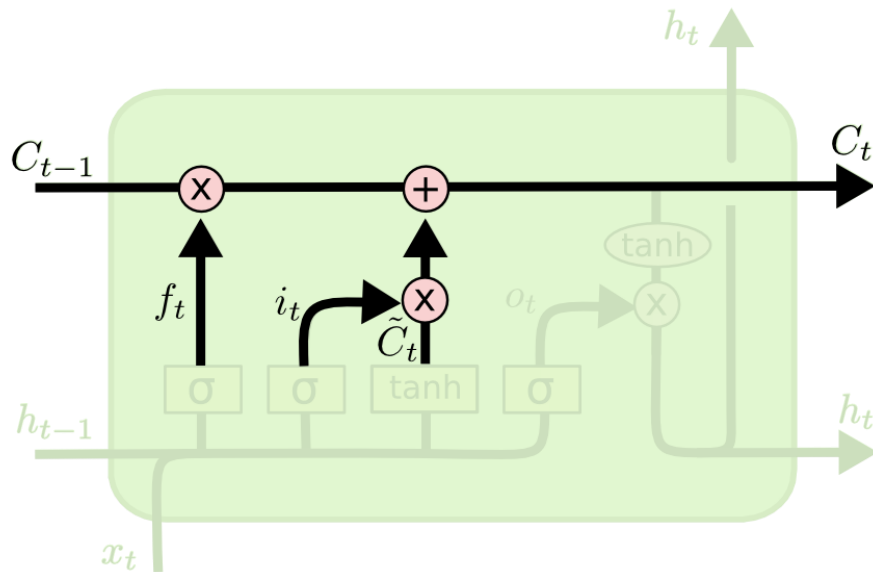
$$-1 \leq \tilde{C}_t \leq 1$$

어떤 정보를 더할 것인지

장기적인 정보( $C_t$ )에 새로운 정보( $\tilde{C}_t$ )를 얼마나( $i_t$ ) 저장할지 결정하는 역할

- LSTM의 구조

## Cell State



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

forget gate만큼 정보를 잃고 input gate만큼 새로운 정보를 갖도록 업데이트 후  
다음 time step으로 전달