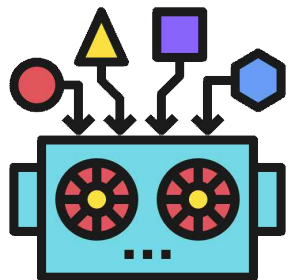


1

Convolutional Neural Network

CNN



이미지 데이터를 처리하는 능력이 좋은 신경망 모델



이미지의 공간 정보를 유지하면서

인접 이미지와의 특징을 효과적으로 인식

1

Convolutional Neural Network

CNN의 특징



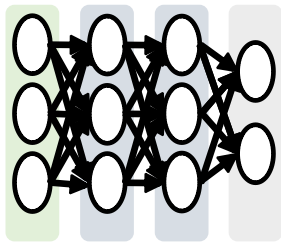
- 와 사진 및 영상의 공간적 특성을 최대한 활용
- 와 가중치 및 편향을 공유하여 학습시간 단축
- 와 사진 및 영상 구별 능력 UP

1

Convolutional Neural Network

CNN의 구성 요소

Fully Connected Layer



모든 유닛이 이전 층의 모든 유닛과
연결되어 있는 층



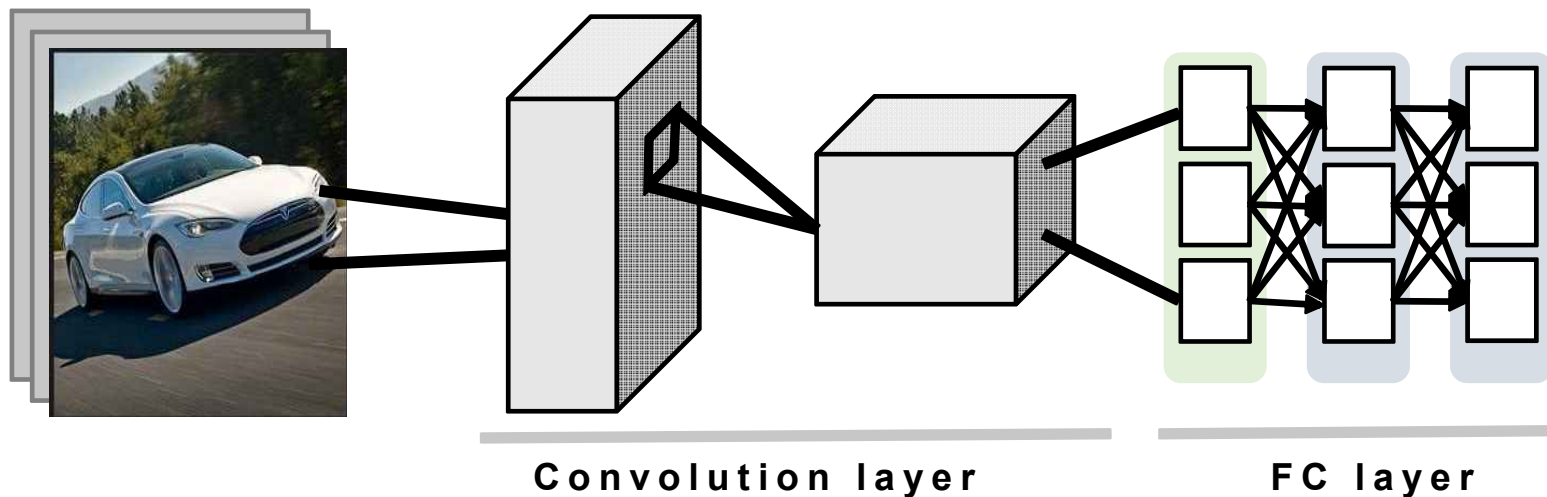
1차원 데이터를 사용하기에 데이터를 **평탄화**
따라서, **3차원 데이터의 공간적 정보 소실**

1

Convolutional Neural Network

CNN의 구성 요소

Convolutional Layer

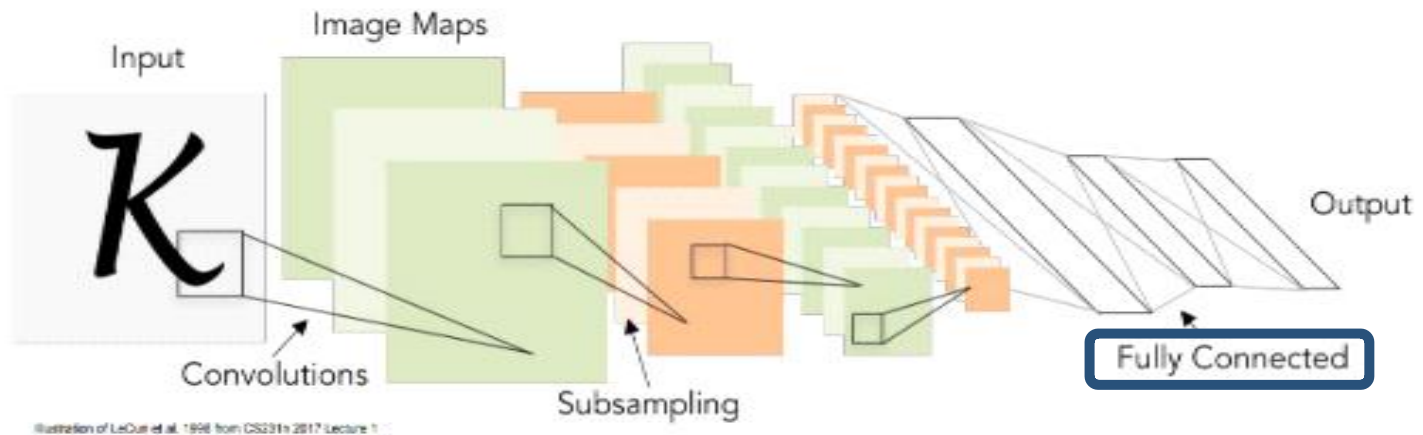


입력 받은 이미지에 대한 특징을 **Filter**를 통해 추출
추출 후 평탄화 하여 기존 신경망과 같이 FC Layer를 이용해 분류

1

Convolutional Neural Network

CNN



1차원 데이터를 입력 받는 FC layer에
최대한 유의미한 feature를 건네 주기 위해
Convolution layer를 통해 특징을 추출

2

Recurrent Neural Network

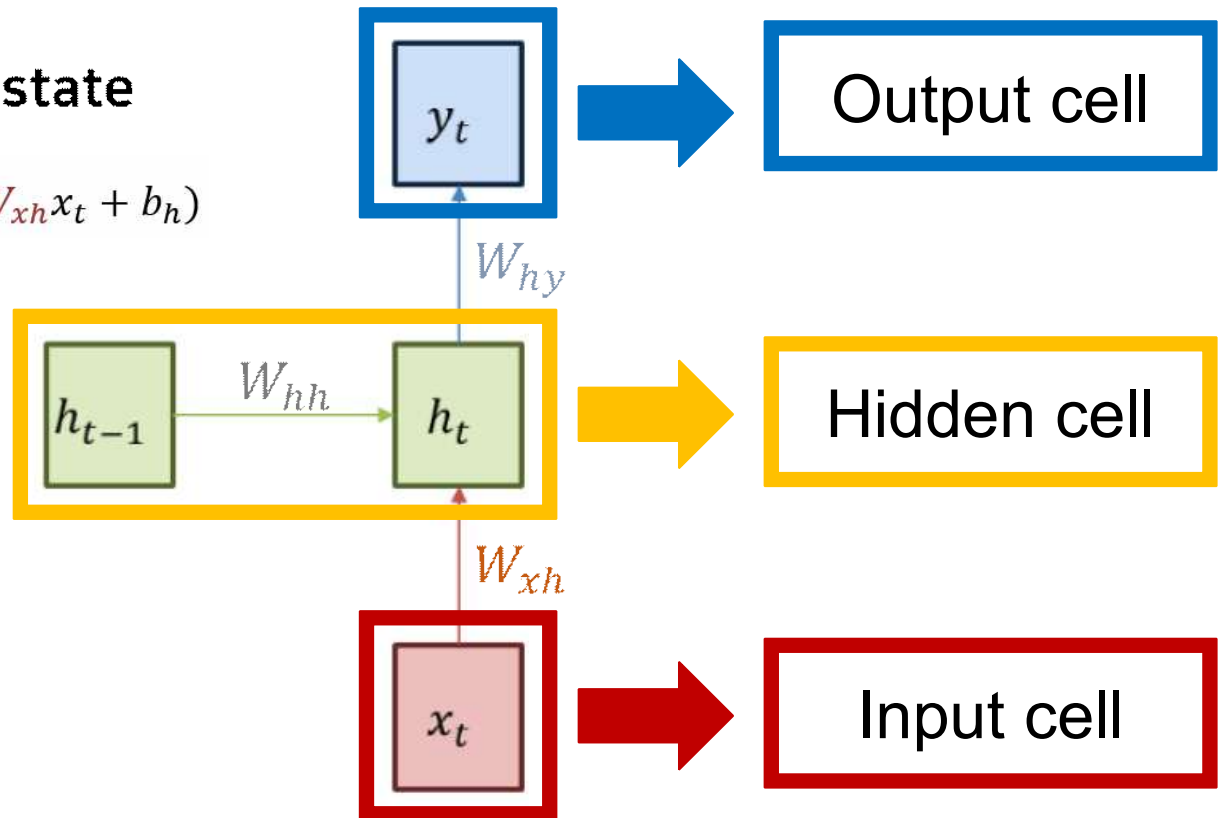
RNN의 구조



h_{t-1}, h_t : Hidden state



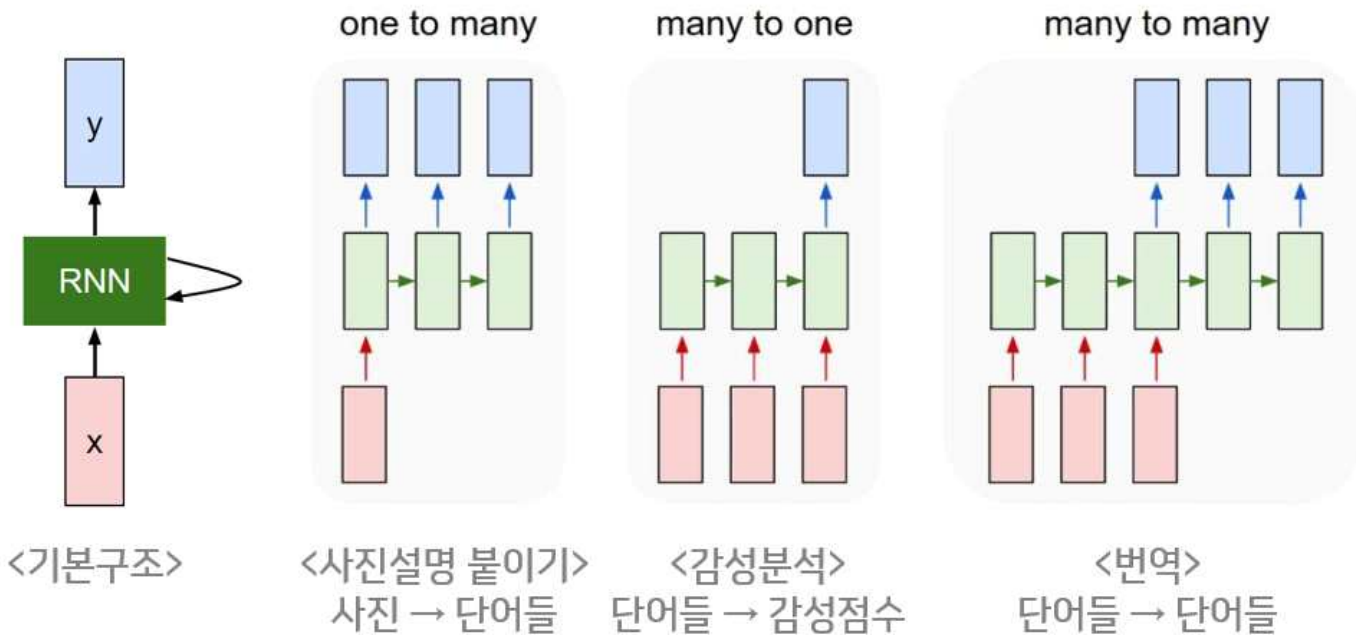
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h)$$



2

Recurrent Neural Network

RNN의 쓰임새



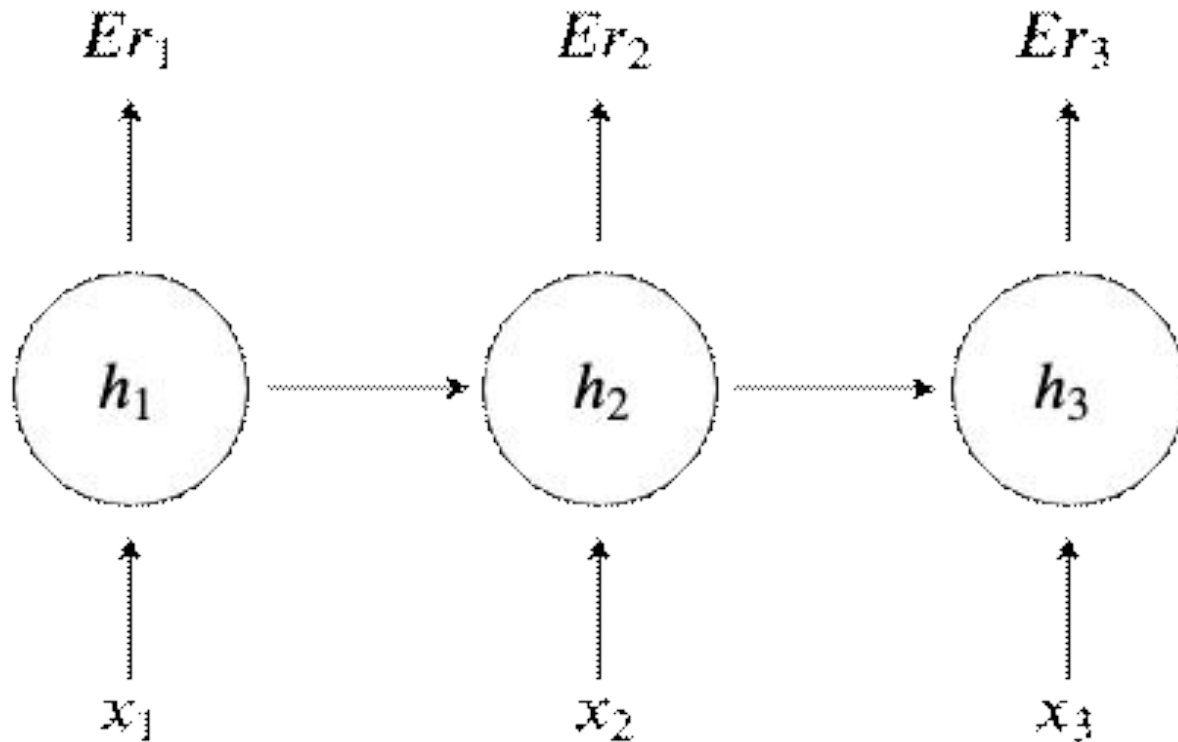
RNN은 입출력이 매우 자유로움 ➡ 필요에 따라 다양하고 유연한 구조 가능

2

Recurrent Neural Network

BPTT

Back-Propagation Through Time



Time Step 별로 역전파 진행

BPTT의 한계

BPTT는 모든 노드를 다 거치기 때문에 t 가 커질수록 계산량도 증가
장기의존성 (feat. vanishing gradient)

Truncated BPTT

BPTT의 계산량 한계를 보완한 알고리즘
전체 time step을 일정 구간(3 or 5 등)으로 나눠 역전파
각 구간 단위로 미래의 블록과는 독립적으로 수행

Activation 함수로 tanh를 사용하므로 time step이 길어질수록
gradient가 끝까지 전달되지 못함

➡ 결국 **Vanishing gradient** 문제에서 자유롭지 못함

실제 적용상 긴 문맥을 학습할 수 없다

ex) 나는 프랑스에서 자랐습니다. (대략 1억5천8백6십3만개 문장)
나는 000어를 잘합니다.

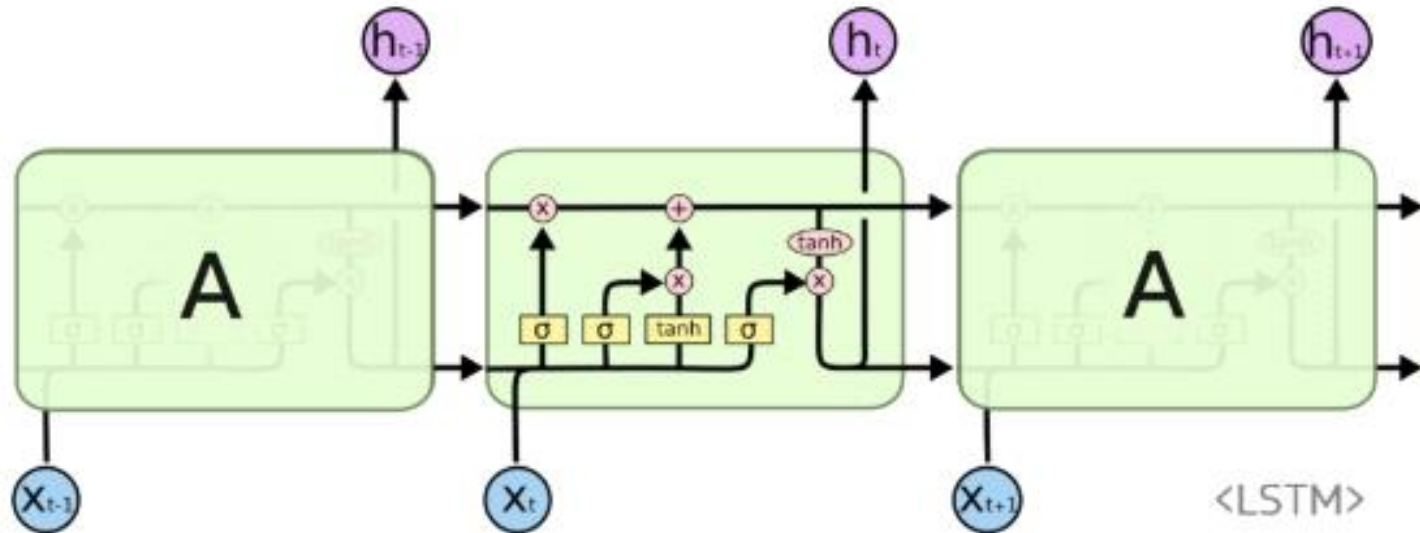
3

Long Short-Term Memory

LSTM이란?

Long Short-Term Memory

Cell state + Hidden state
장기기억 + 단기 기억



3

Long Short-Term Memory

LSTM의 학습과정



1

2

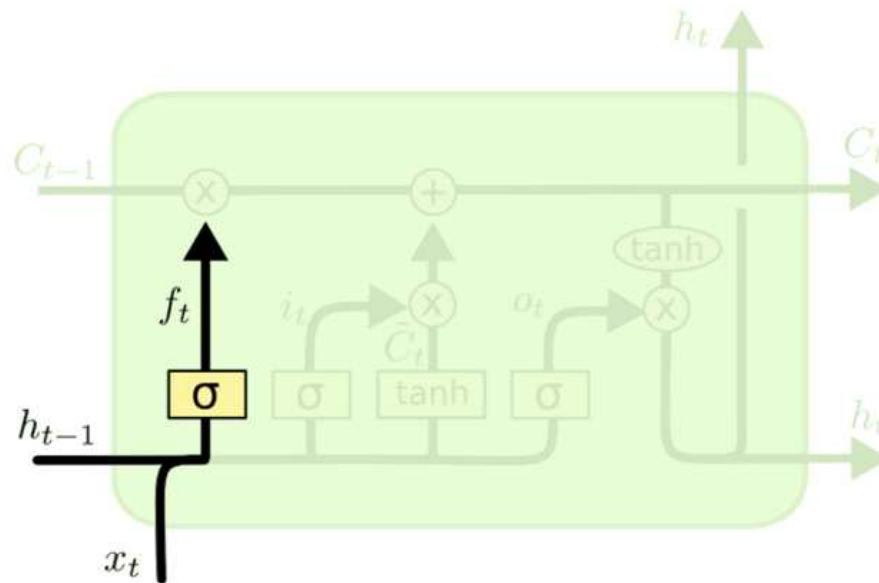
3

4



Forget gate

‘과거 정보를 잊기 위한 게이트’ : cell state에서 어떤 정보를 버릴지 결정



3

Long Short-Term Memory

LSTM의 학습과정

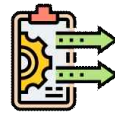


1

2

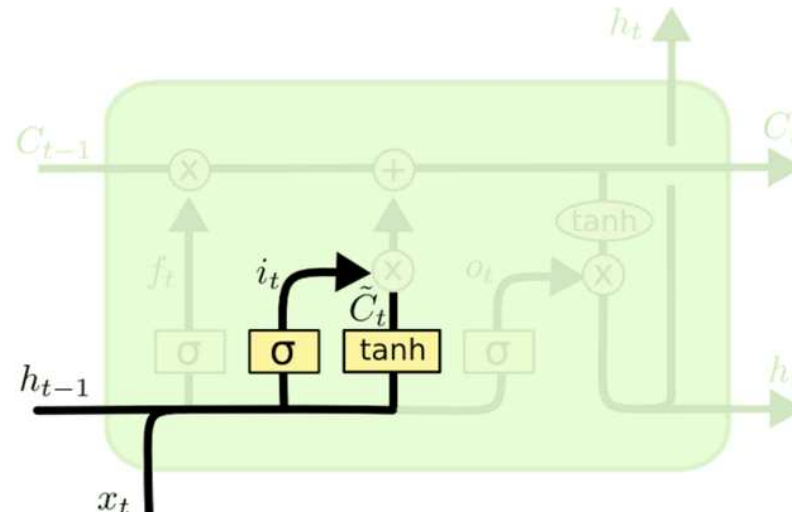
3

4



Input gate

‘현재 정보를 기억하기 위한 게이트’ :
새로운 정보를 얼마나 cell state에 저장할지 결정



3

Long Short-Term Memory

LSTM의 학습과정



1

2

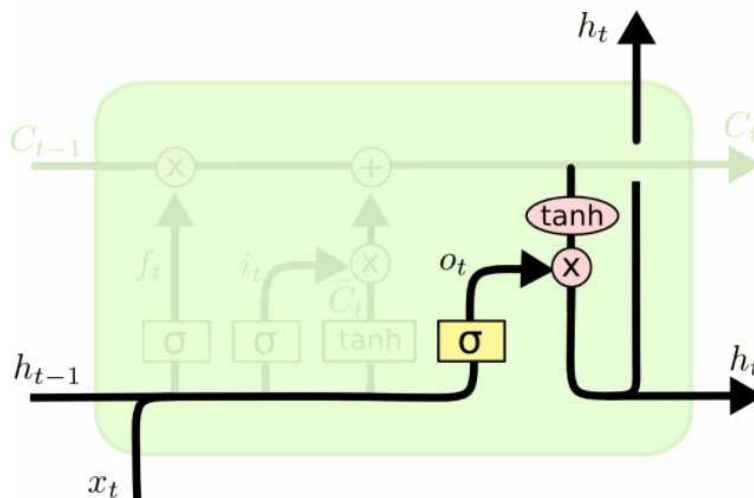
3

4



Output gate

‘무엇을 내보낼지 결정하는 게이트’ :
cell state의 어느 부분을 읽어서 출력해야 하는지 결정



3

Long Short-Term Memory

LSTM의 학습과정



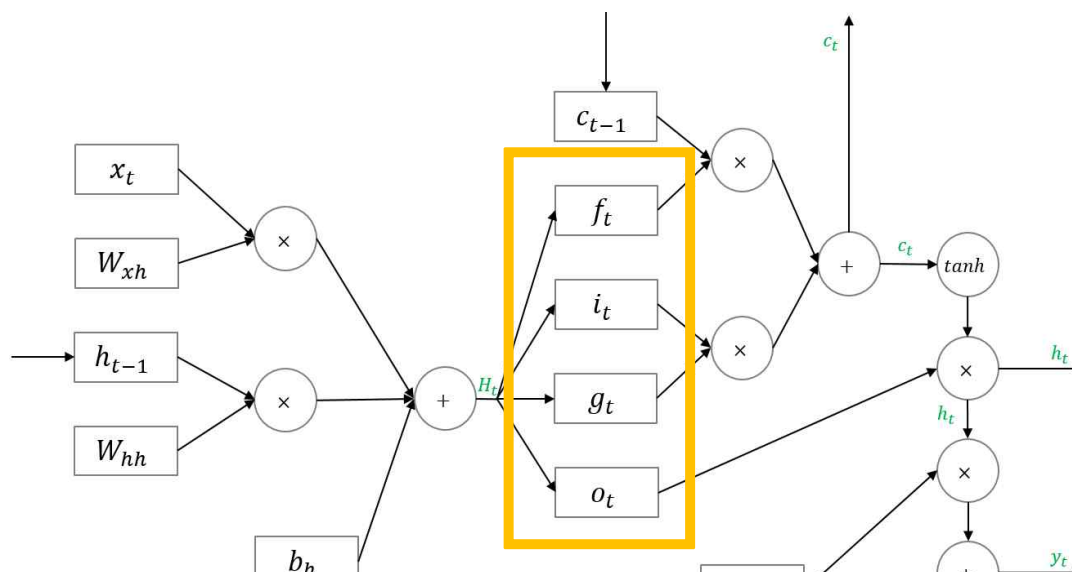
1

2

3

4

Forward propagation 방식으로 진행 !



Hidden state 행렬을 4등분해 위 세가지에 해당하는
활성화 함수를 적용해 계산 (매 time step마다)



1

2

3

4



Backpropagation

“RNN과 방식은 같음”

hidden state의 gradient는 위 세 단계의 계산값의 gradient를 합쳐 만듦

Cell state와 hidden state가 재귀적으로 구해짐

-> 이 둘의 gradient는 이전 시점의 gradient에 영향 받기 때문에

이를 역전파에 반영해야 함