Dong-A Univ. (ISPL)

동아대학교
DONG-A UNIVERSITY

LLM 기초 – RNN, LSTM

컴퓨터AI공학부

2025년 1학기 머신러닝

Large Language Model (LLM)

- **Language Model**
  - 인간의 언어를 이해하고 생성하기 위해 설계된 인공지능 모델
- **Large Language Model**
  - 방대한 양의 데이터에 기반하여 Training된 수많은 Parameter를 가지는 인공지능 모델

Large Language Model (LLM)

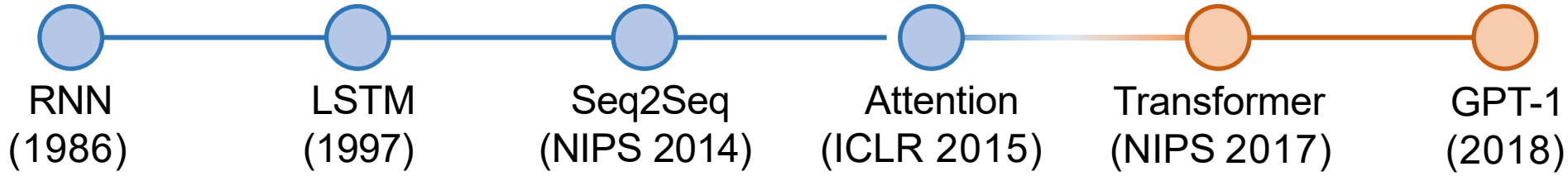
- **Language Model**
  - 인간의 언어를 이해하고 생성하기 위해 설계된 인공지능 모델
- **Large Language Model**
  - 방대한 양의 데이터에 기반하여 Training된 수많은 Parameter를 가지는 인공지능 모델
  - 대표적인 LLM: **Chat GPT**

ChatGPT

Generative **P**re-trained **T**ransformer

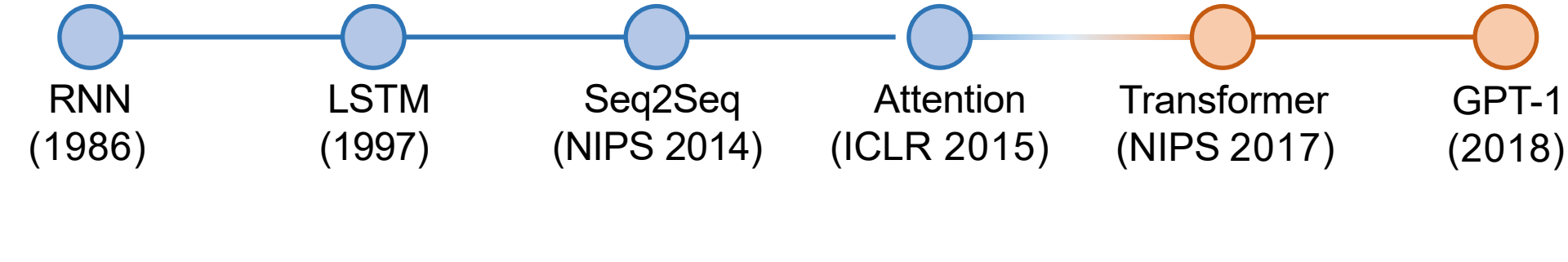
Large Language Model (LLM)

- **History of Language Model**
  - 기존 Language model은 RNN 기반으로 설계 되었음
  - RNN 기반 모델들의 문제점을 Transformer가 해결하면서 높은 성능을 도출함



Large Language Model (LLM)

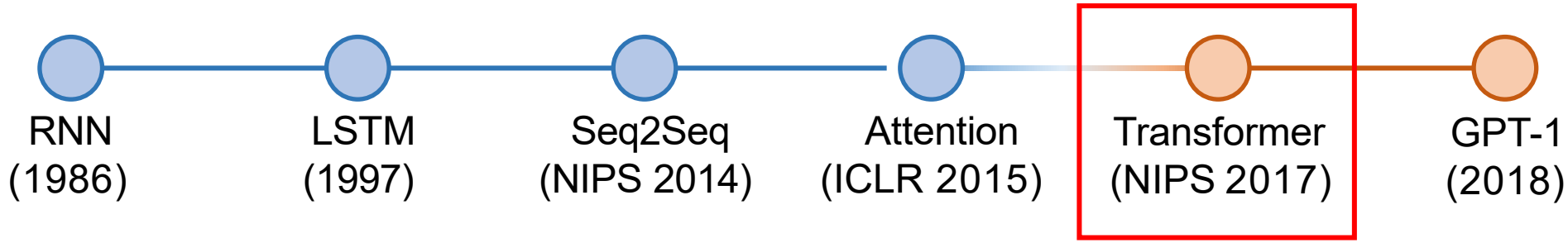
- **History of Language Model**
  - 기존 Language model은 RNN 기반으로 설계 되었음
  - RNN 기반 모델들의 문제점을 Transformer가 해결하면서 높은 성능을 도출함



RNN을 기반으로 설계된 model

Large Language Model (LLM)

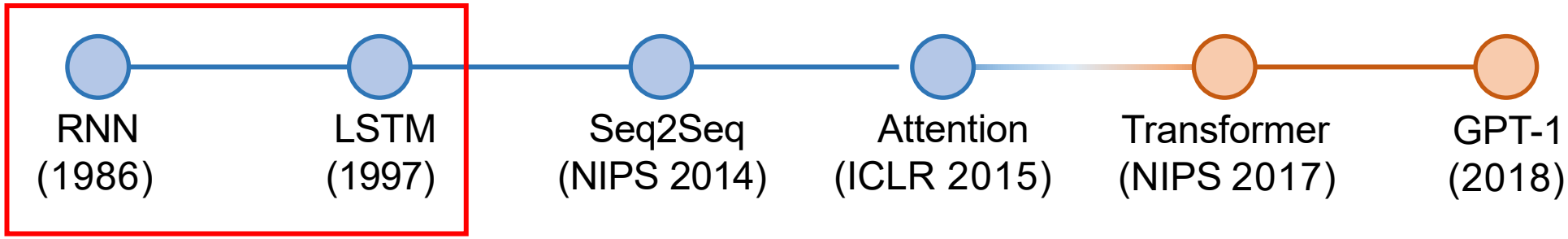
- **History of Language Model**
  - 기존 Language model은 RNN 기반으로 설계 되었음
  - RNN 기반 모델들의 문제점을 Transformer가 해결하면서 높은 성능을 도출함



RNN 기반 Model의 한계점 해결

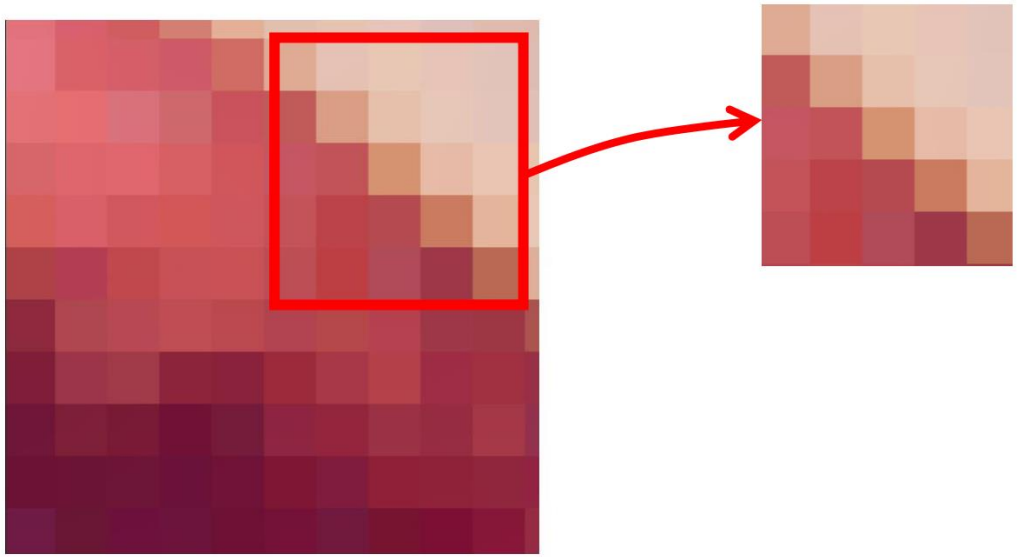
Large Language Model (LLM)

- **History of Language Model**
  - 기존 Language model은 RNN 기반으로 설계 되었음
  - RNN 기반 모델들의 문제점을 Transformer가 해결하면서 높은 성능을 도출함



RNN

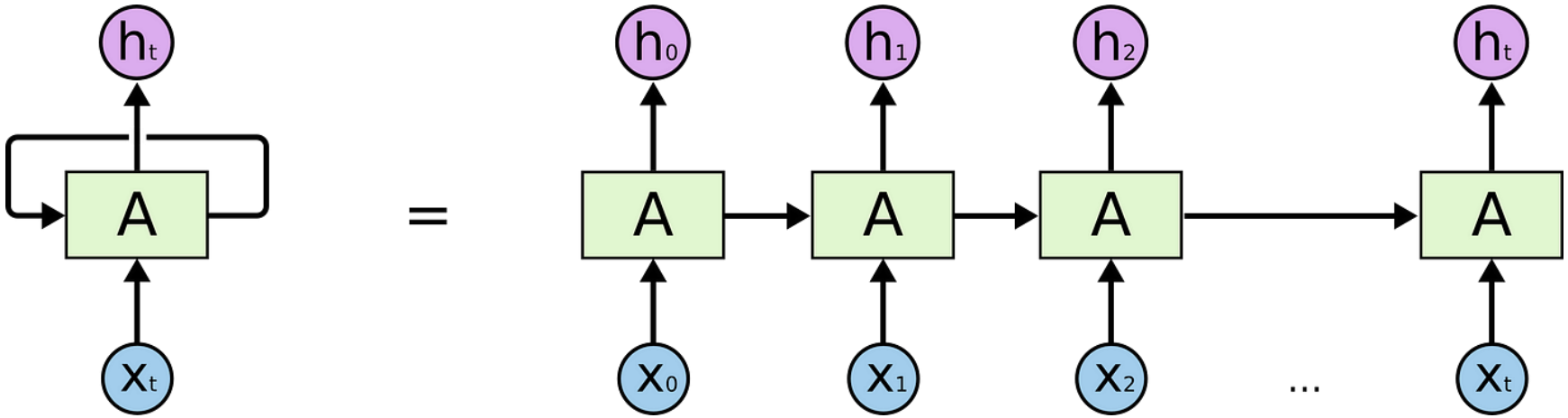
- **Overview**
  - 기존의 CNN은 데이터의 공간적인 정보만을 학습함



CNN은 이미지 데이터의 공간적 특징을 추출하여 학습

RNN

- **Overview**
  - 시간 순서가 있는 데이터 (Time Series Data)를 효율적으로 학습하기 위해 등장
  - 순환 구조를 통해서 이전 상태의 정보를 함께 사용하여 현재 상태의 정보 학습



순서가 있는 데이터를 순차적으로 입력하여 학습

- **시계열 데이터 (Time Series Data)**
  - 시간의 흐름에 따라 관측되어 시간의 영향을 받게 되는 데이터
  - 현재 상태가 과거의 상태에 영향을 받음

Index	Bedrooms	Bathrooms	Sqft_living	Price
0	3	1	1180	221900
1	3	2.25	2570	538000
2	2	1	770	180000
3	4	3	1960	604000
…	…	…	…	…
N	3	2	1680	510000

다변량 데이터 예시 (Kc house dataset)

- **시계열 데이터 (Time Series Data)**
  - 시간의 흐름에 따라 관측되어 시간의 영향을 받게 되는 데이터
  - 현재 상태가 과거의 상태에 영향을 받음

Index	Bedrooms	Bathrooms	Sqft_living	Price
0	3	1	1180	221900
1	3	2.25	2570	538000
2	2	1	770	180000
3	4	3	1960	604000
…	…	…	…	…
N	3	2	1680	510000

각각의 데이터가 서로 독립적

다변량 데이터 예시 (Kc house dataset)

- **시계열 데이터 (Time Series Data)**
  - 시간의 흐름에 따라 관측되어 시간의 영향을 받게 되는 데이터
  - 현재 상태가 과거의 상태에 영향을 받음

시간	센서1	센서2	센서3	상태
12:00	0	98.9	3.9	정상
13:00	0	98.9	3.9	정상
14:00	0.3	74.5	6.7	불량
15:00	6.8	98.9	7.9	불량
…	…	…	…	…
24:00	-0.9	78.3	6.6	정상

현재 상태의 데이터가 이전 시점 상태에 영향을 받음

시계열 데이터 예시

RNN

- **시계열 데이터 (Time Series Data)**
  - 시간의 흐름에 따라 관측되어 시간의 영향을 받게 되는 데이터
  - 현재 상태가 과거의 상태에 영향을 받음
  - **인간의 언어 또한 각 단어별 순서와 맥락이 시간적 의존성을 가짐**

시간	센서1	센서2	센서3	상태
12:00	0	98.9	3.9	정상
13:00	0	98.9	3.9	정상
14:00	0.3	74.5	6.7	불량
15:00	6.8	98.9	7.9	불량
…	…	…	…	…
24:00	-0.9	78.3	6.6	정상

시계열 데이터 예시

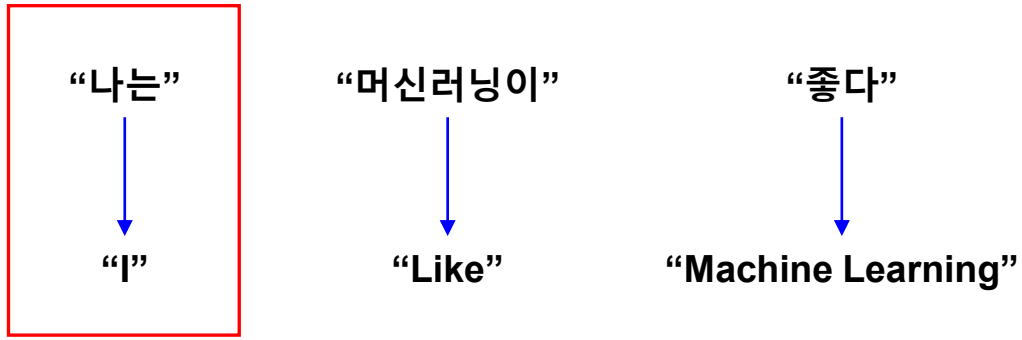
“안녕하세요 머신러닝 강의 너무 좋아요”

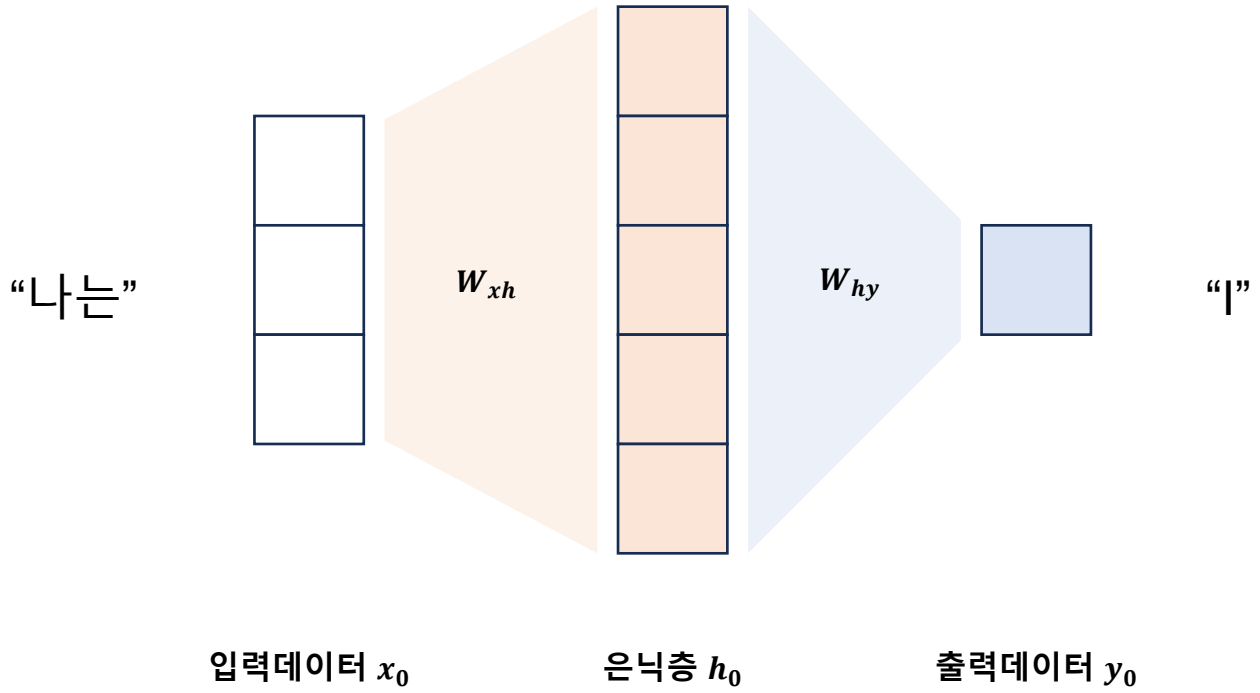
시간 순서에 영향을 받음

RNN

- **인공신경망 (Deep Neural Network, DNN)**
  - Ex) 한글 → 영어 번역 예시 (단어 단위 입력 및 출력)

- **인공신경망 (Deep Neural Network, DNN)**
  - Ex) 한글 → 영어 번역 예시 (단어 단위 입력 및 출력)



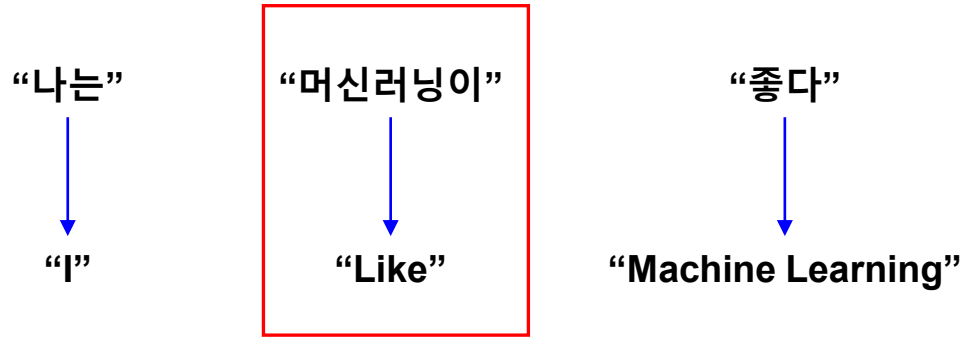


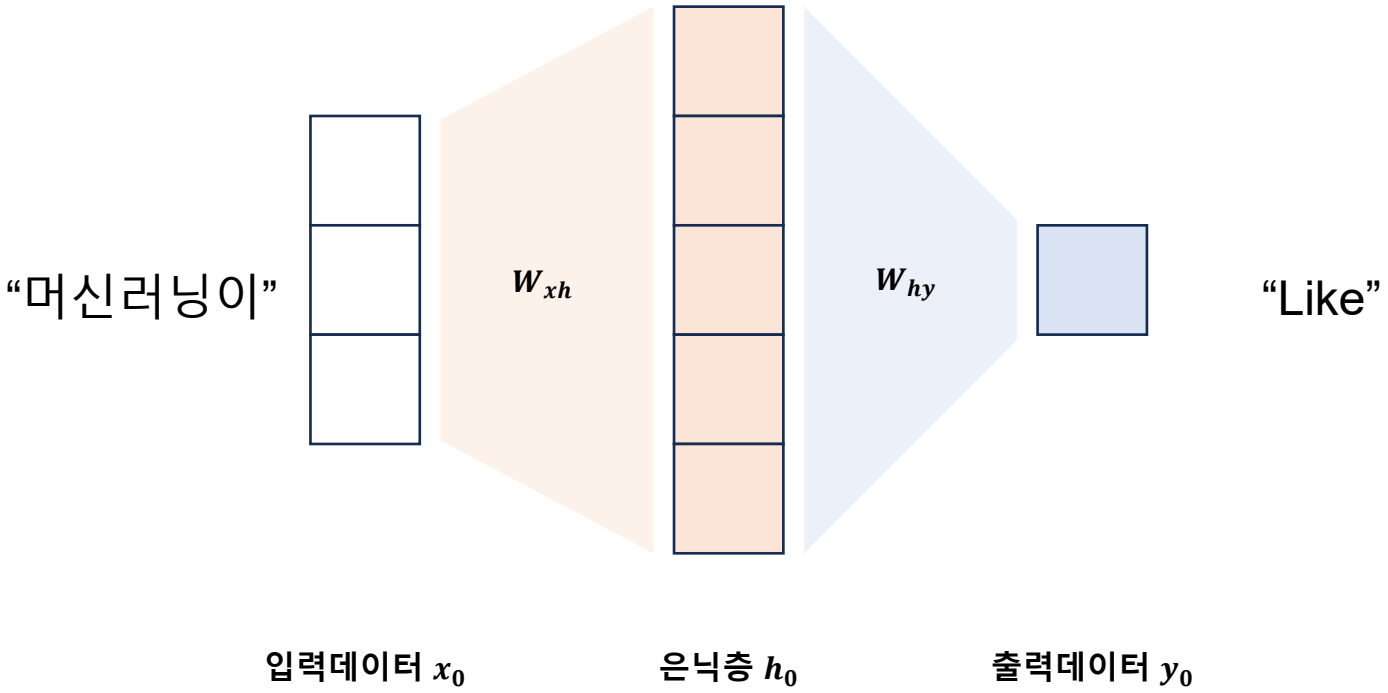
$$h_i = f(W_{xh}x_i + bias)$$

$$y_i = f(W_{hy}h_i + bias)$$

❖ f: activation function

- **인공신경망 (Deep Neural Network, DNN)**
  - Ex) 한글 → 영어 번역 예시 (단어 단위 입력 및 출력)





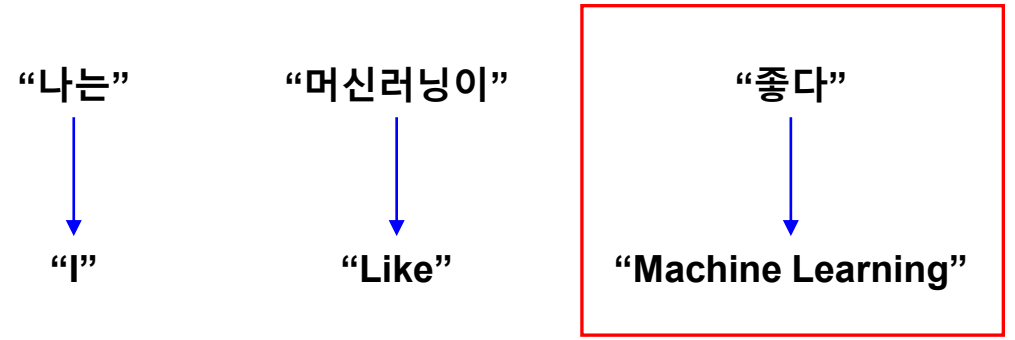
$$h_i = f(W_{xh}x_i + bias)$$

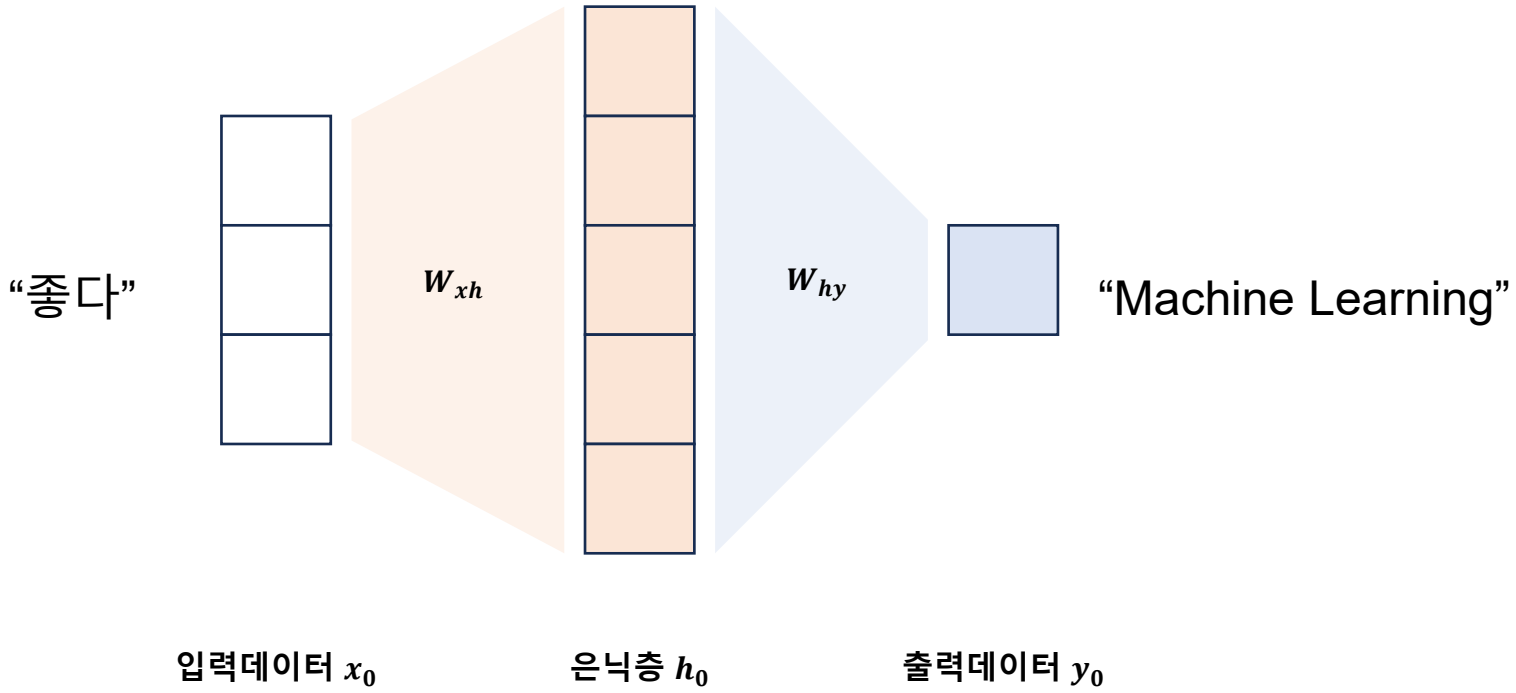
$$y_i = f(W_{hy}h_i + bias)$$

❖ f: activation function

RNN

- **인공신경망 (Deep Neural Network, DNN)**
  - Ex) 한글 → 영어 번역 예시 (단어 단위 입력 및 출력)
  - 이전 시점의 정보를 반영하지 않아 예측 어려움





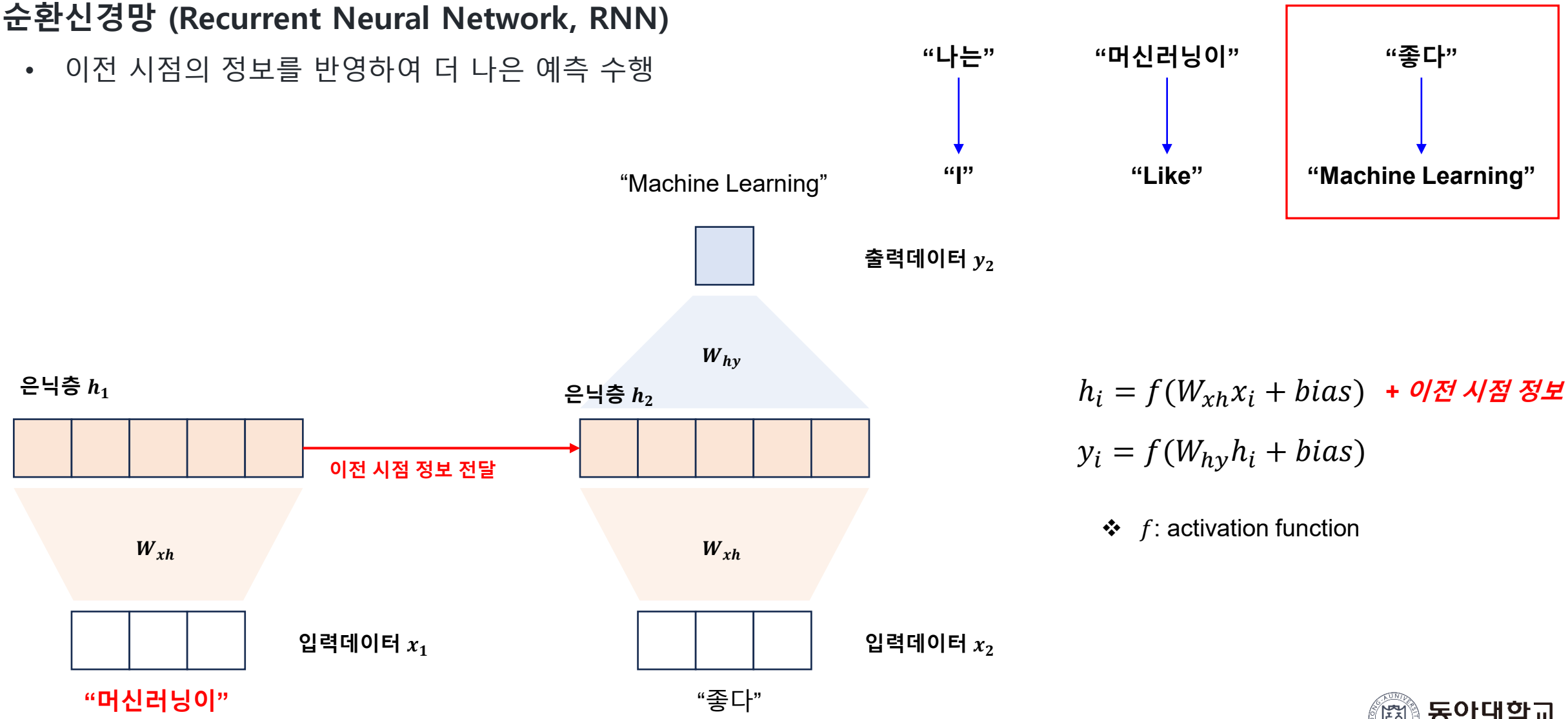
$$h_i = f(W_{xh}x_i + bias)$$

$$y_i = f(W_{hy}h_i + bias)$$

❖ f: activation function

RNN

- **순환신경망 (Recurrent Neural Network, RNN)**
  - 이전 시점의 정보를 반영하여 더 나은 예측 수행



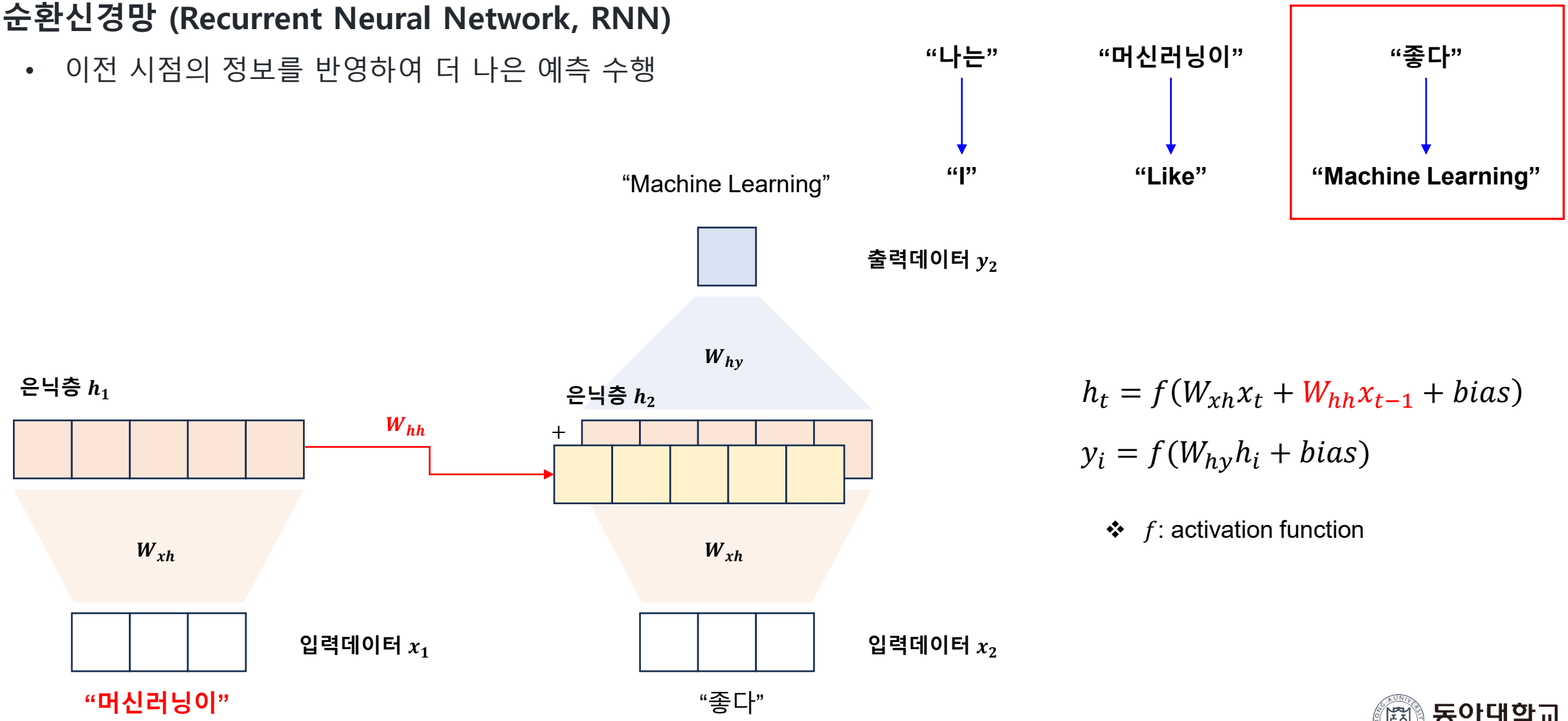
$$h_i = f(W_{xh}x_i + bias)$$ + 이전 시점 정보

$$y_i = f(W_{hy}h_i + bias)$$

❖ f: activation function

RNN

- **순환신경망 (Recurrent Neural Network, RNN)**
  - 이전 시점의 정보를 반영하여 더 나은 예측 수행



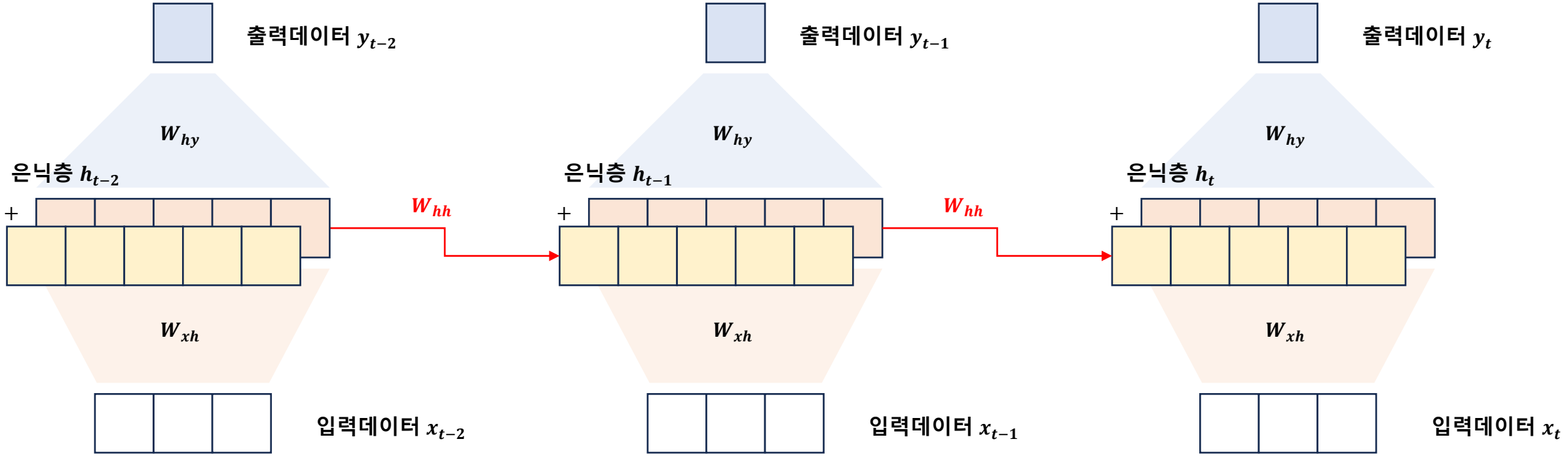
$$h_t = f(W_{xh}x_t + W_{hh}x_{t-1} + bias)$$

$$y_i = f(W_{hy}h_i + bias)$$

❖ f: activation function

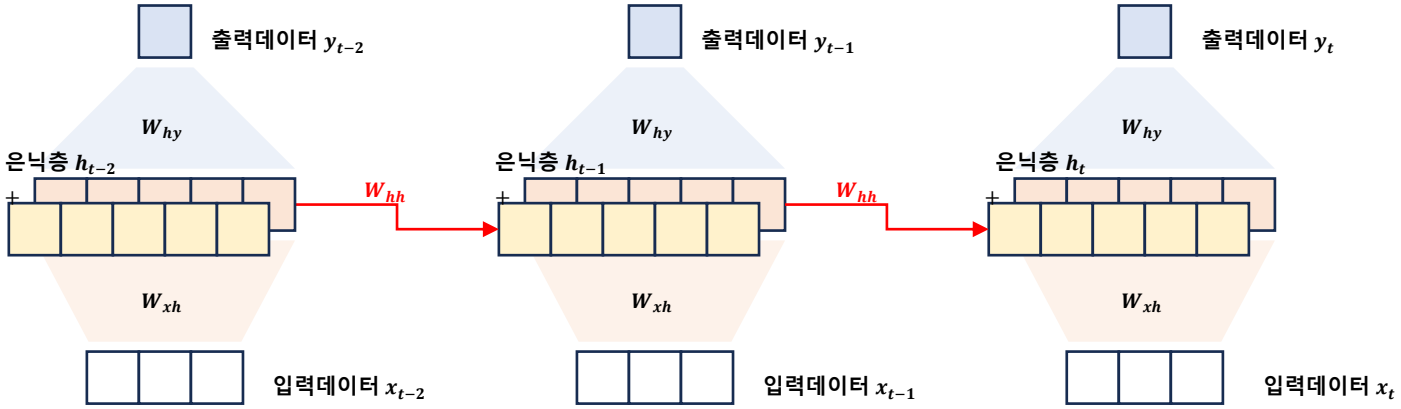
RNN

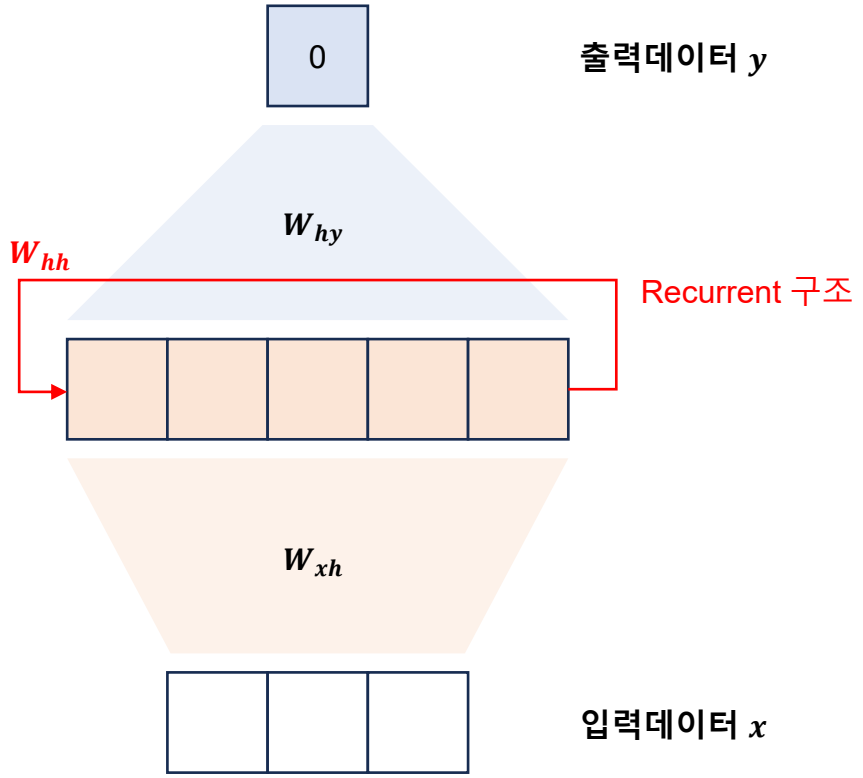
- **순환신경망 (Recurrent Neural Network, RNN)**
  - 이전 시점의 정보를 반영하여 더 나은 예측 수행
  - 이전 정보들이 순환 (반복)하여 입력



RNN

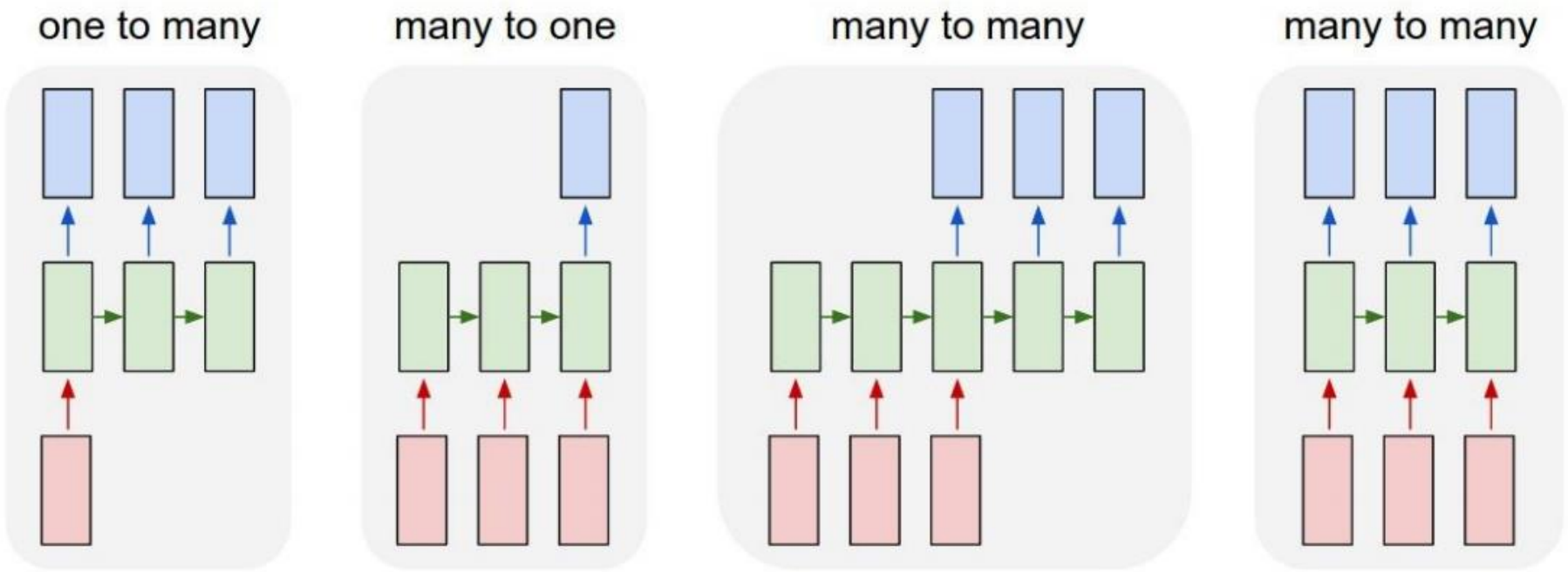
- **순환신경망 (Recurrent Neural Network, RNN)**
  - 이전 시점의 정보를 반영하여 더 나은 예측 수행
  - 이전 정보들이 순환 (반복)하여 입력





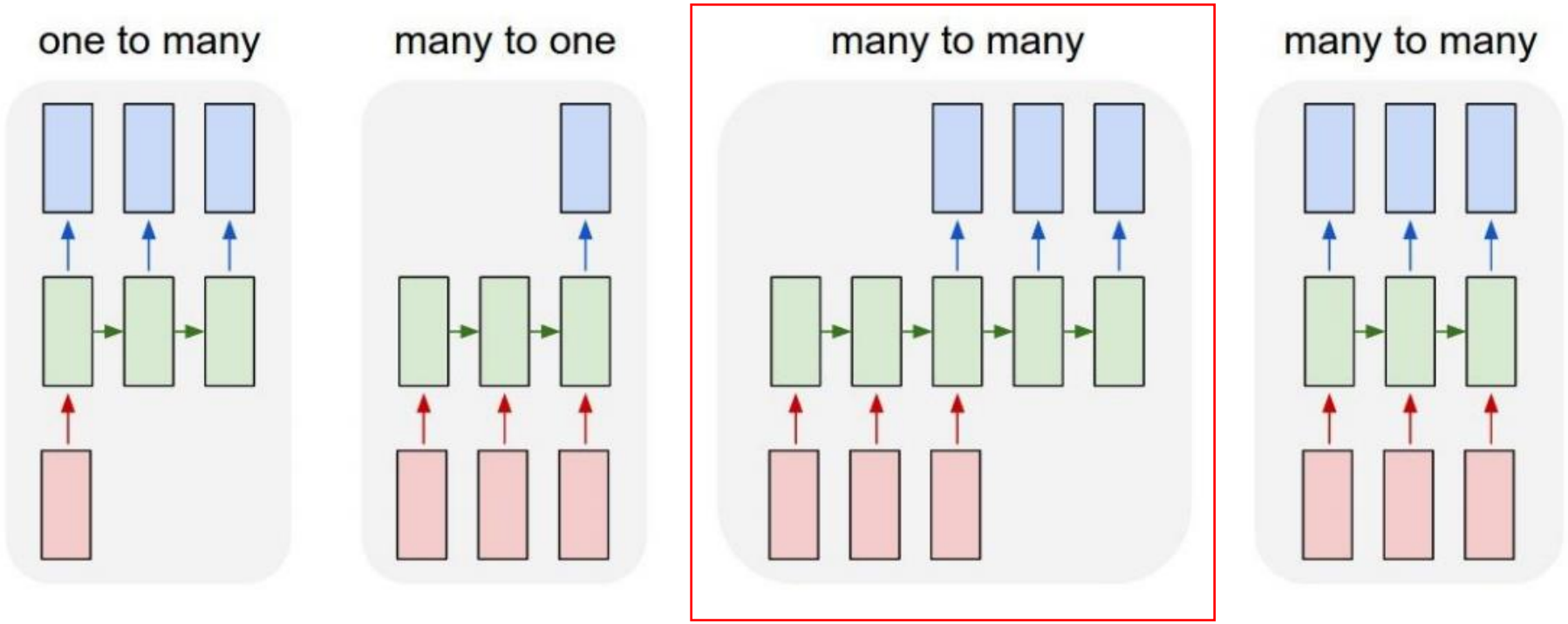
RNN

- **순환신경망 구조의 종류**
  - 순차적인 입력의 길이, 순차적인 예측의 길이에 따라 다음과 같이 구분 가능



RNN

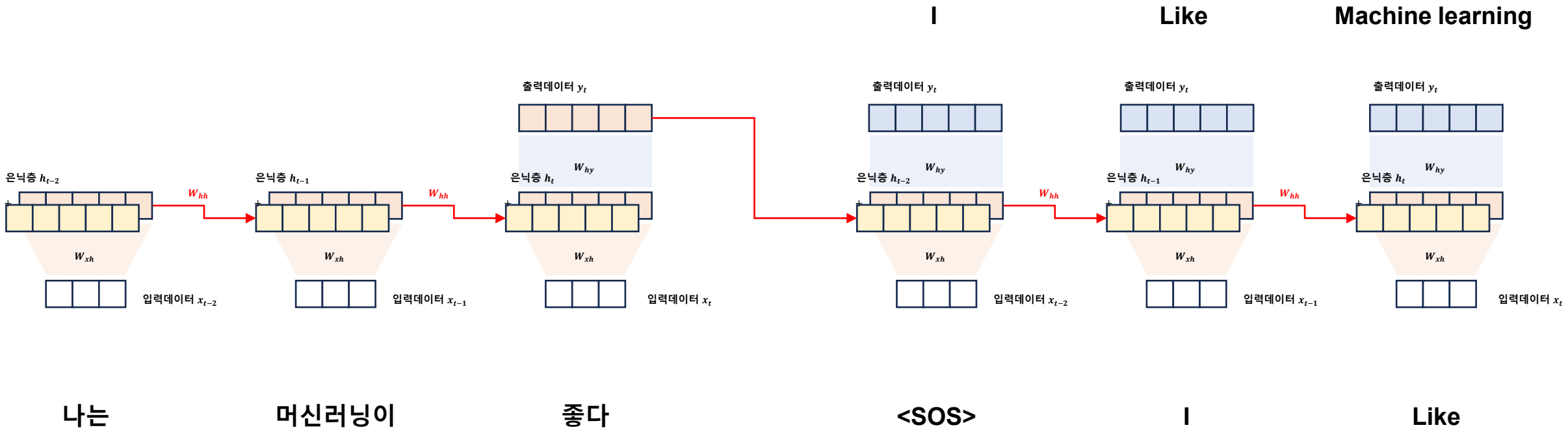
- **순환신경망 구조의 종류**
  - 순차적인 입력의 길이, 순차적인 예측의 길이에 따라 다음과 같이 구분 가능



Sequence to Sequence

RNN

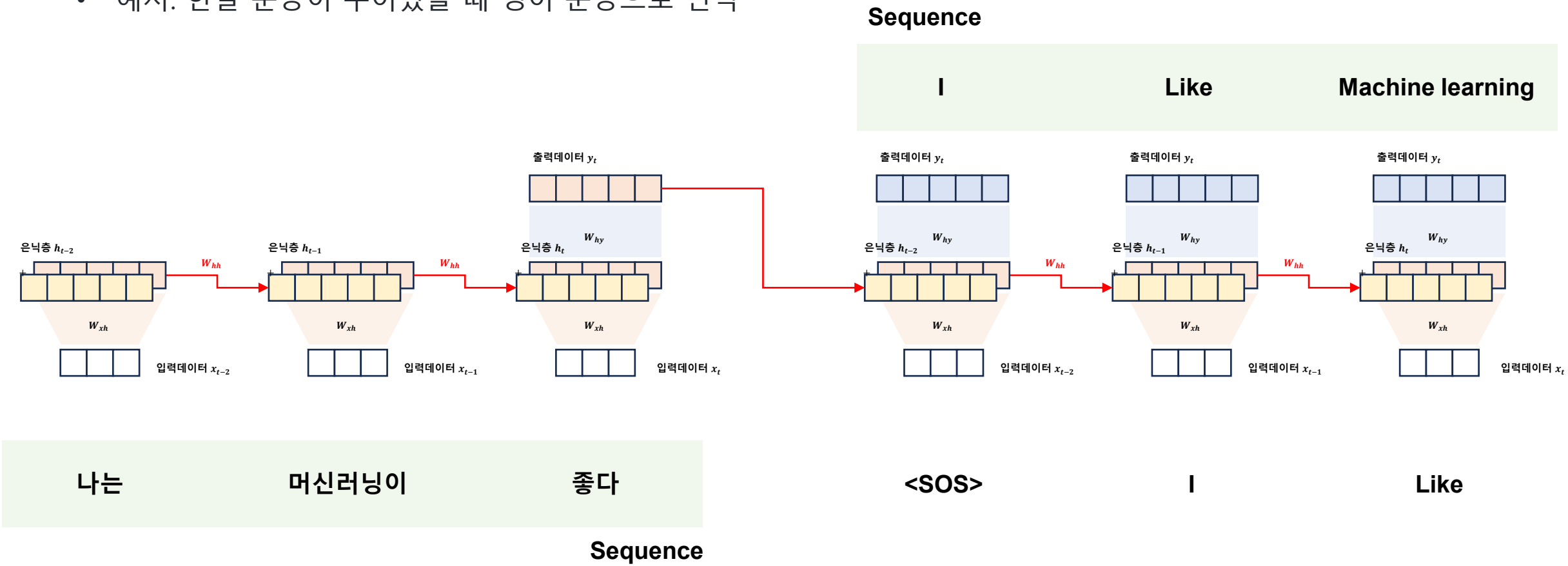
- **Sequence to Sequence**
  - 순차적인 X (Many)로 순차적인 Y (Many)를 예측하는 문제
  - 예시: 한글 문장이 주어졌을 때 영어 문장으로 번역



❖ <SOS> Start Of Sentence

RNN

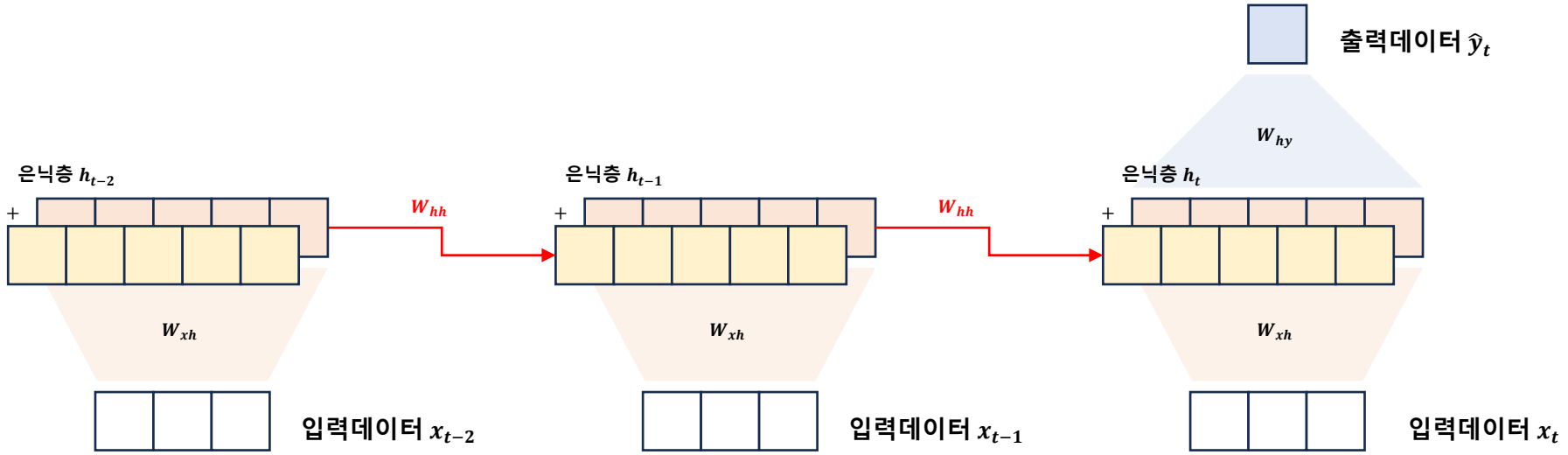
- **Sequence to Sequence**
  - 순차적인 X (Many)로 순차적인 Y (Many)를 예측하는 문제
  - 예시: 한글 문장이 주어졌을 때 영어 문장으로 번역



❖ <SOS> Start Of Sentence

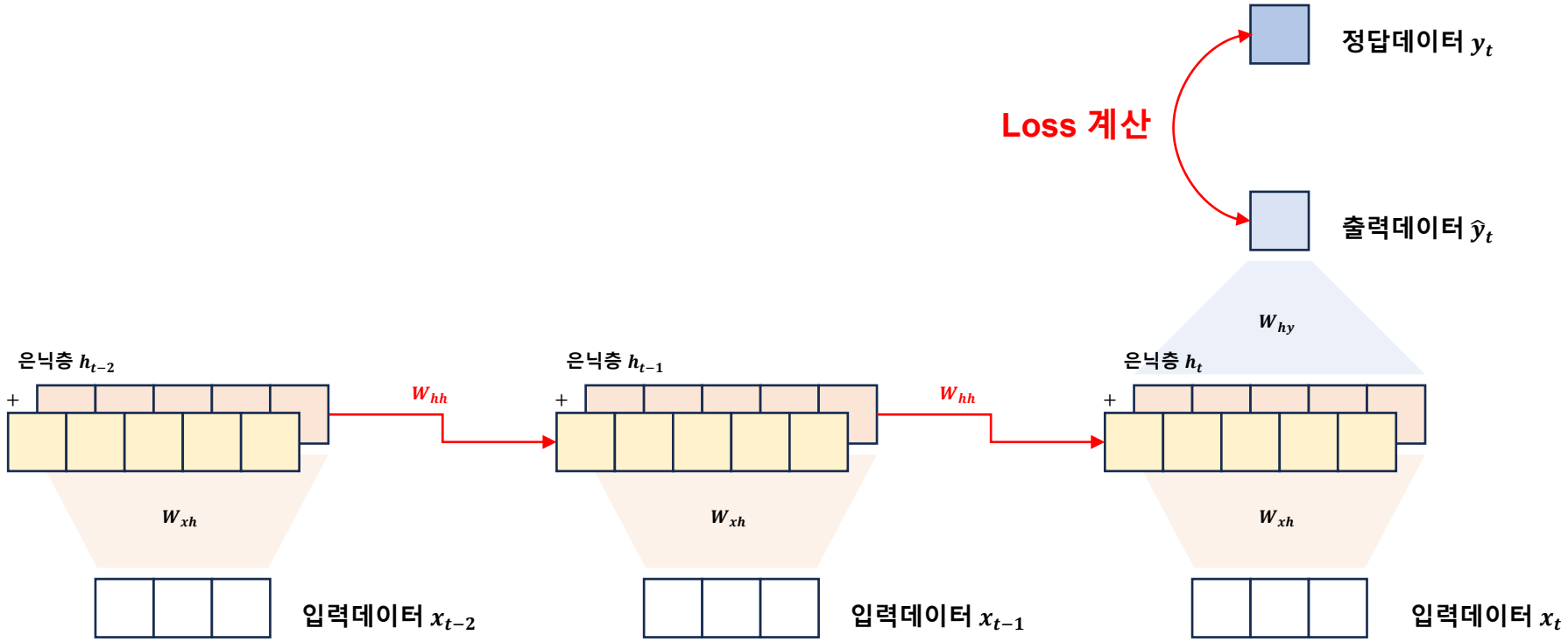
RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



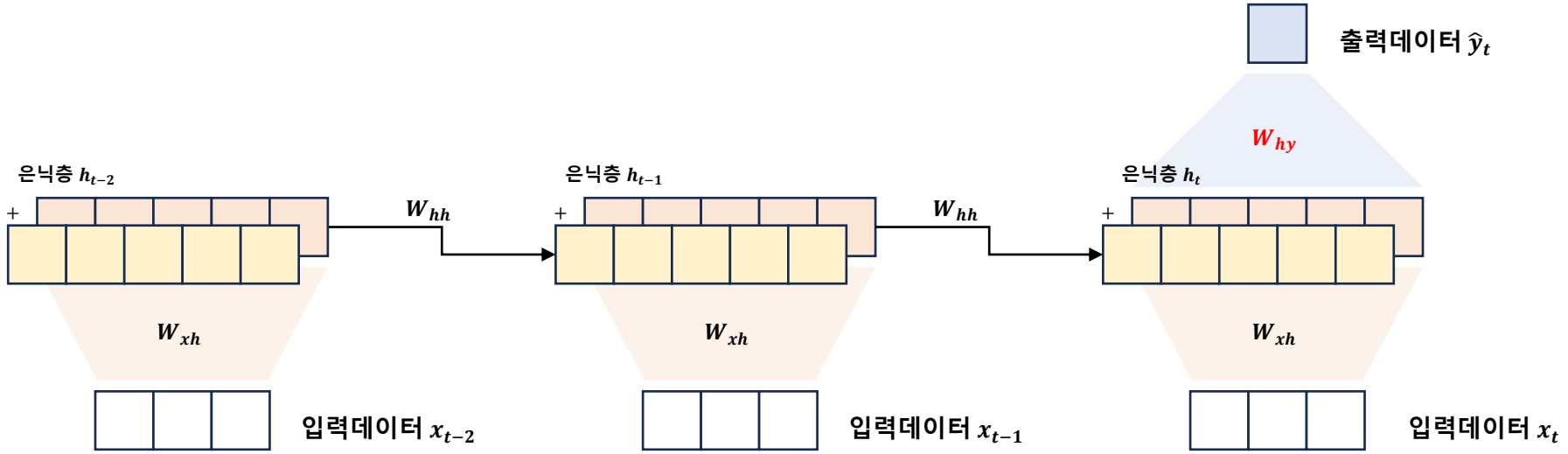
RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



RNN 학습

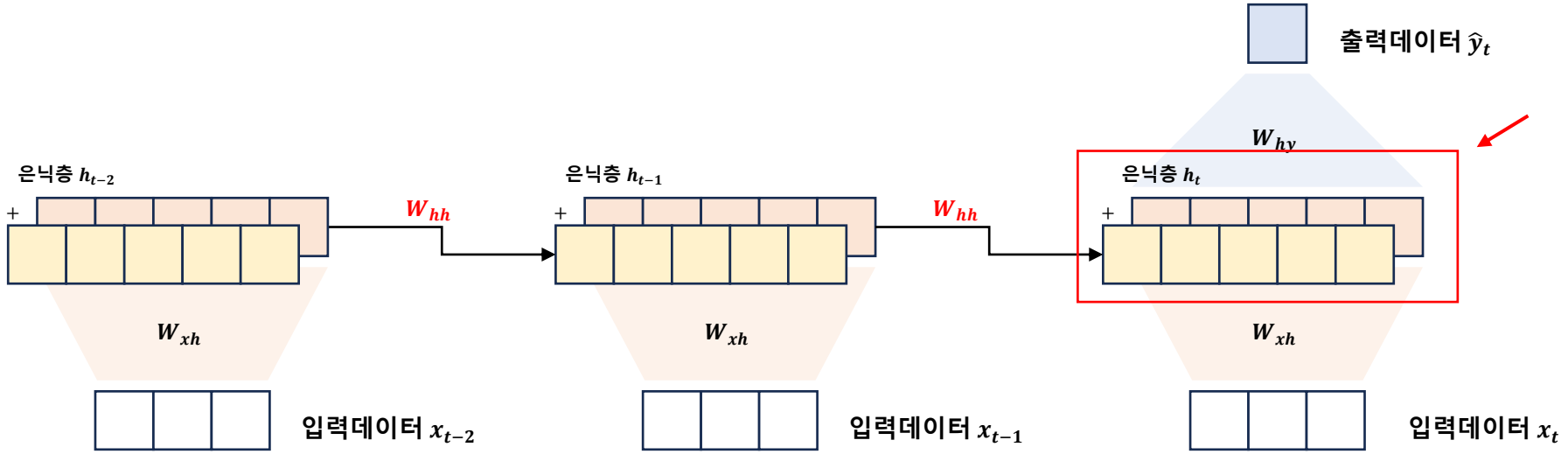
- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



$$\frac{\partial Loss}{\partial W_{hy}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial W_{hy} h_t} \times \frac{\partial W_{hy} h_t}{\partial W_{hy}}$$

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

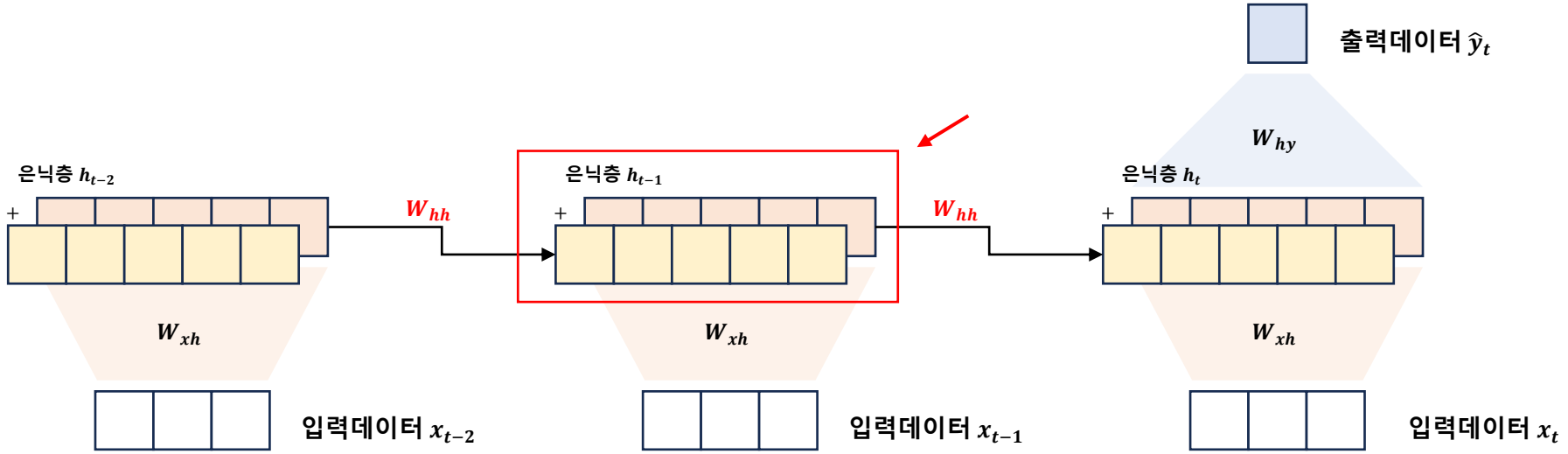


$$\frac{\partial Loss}{\partial W_{hh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{hh}}$$

t 시점에서의 영향

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

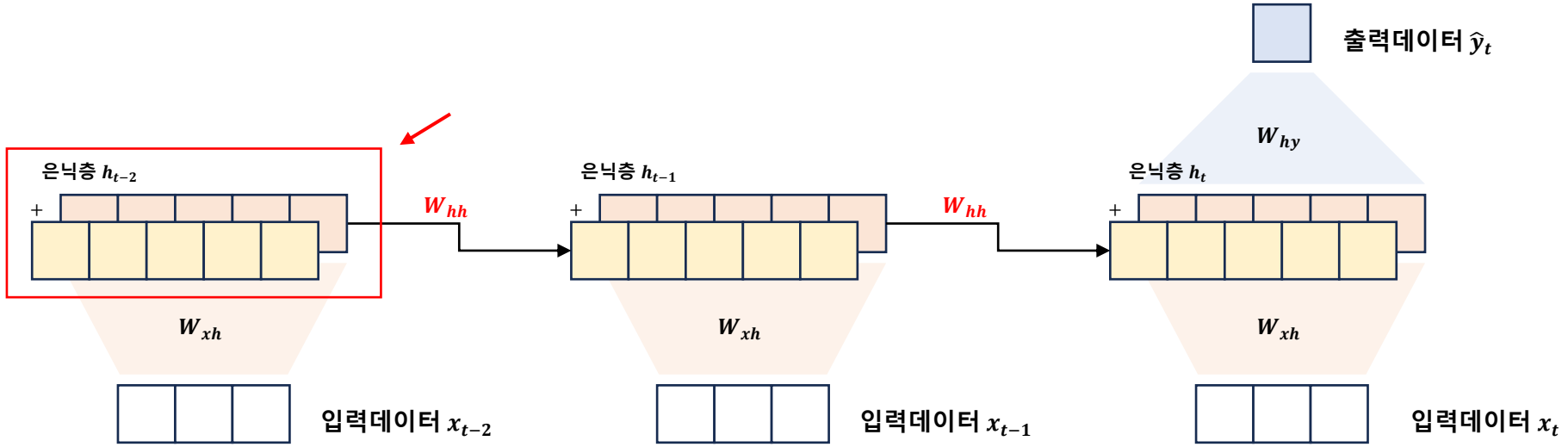


$$\frac{\partial Loss}{\partial W_{hh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{hh}}$$

t -1 시점에서의 영향

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

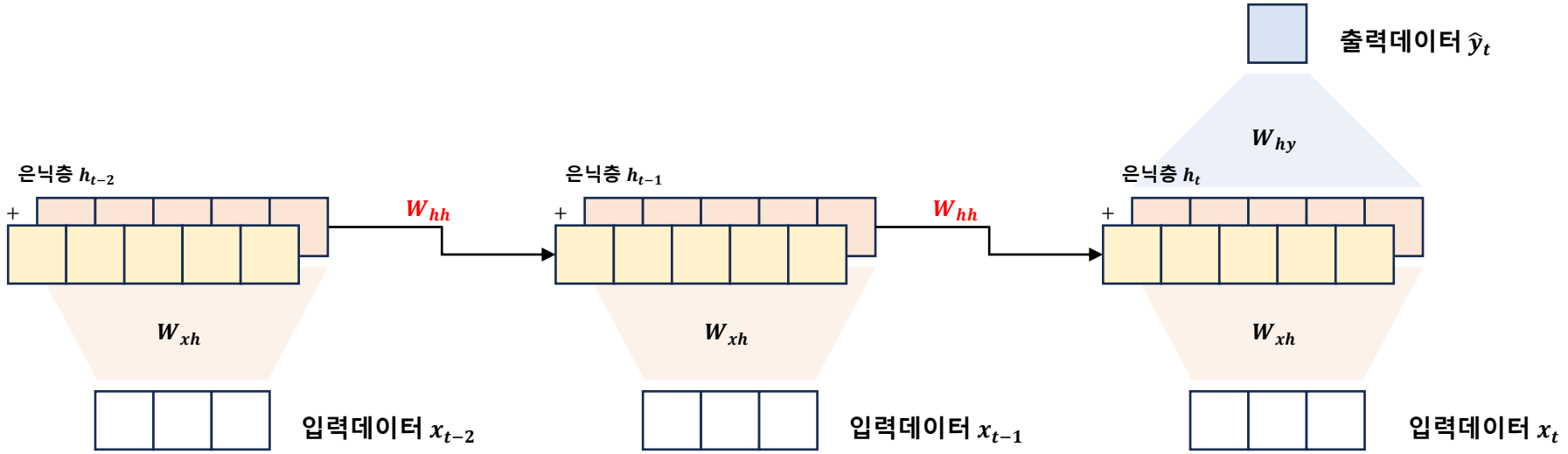


$$\frac{\partial Loss}{\partial W_{hh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{hh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{hh}}$$

t -2 시점에서의 영향

RNN 학습

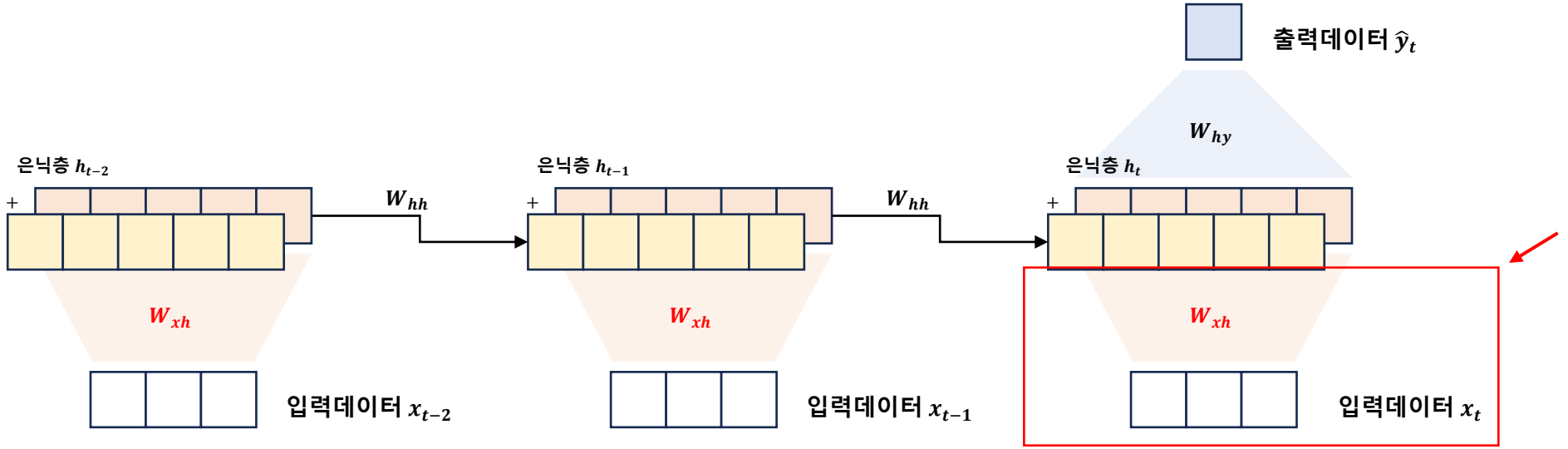
- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



$$\frac{\partial Loss}{\partial W_{hh}} = \sum_{k=0}^{t} \left(\frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_k} \times \frac{\partial h_k}{\partial W_{hh}} \right)$$

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

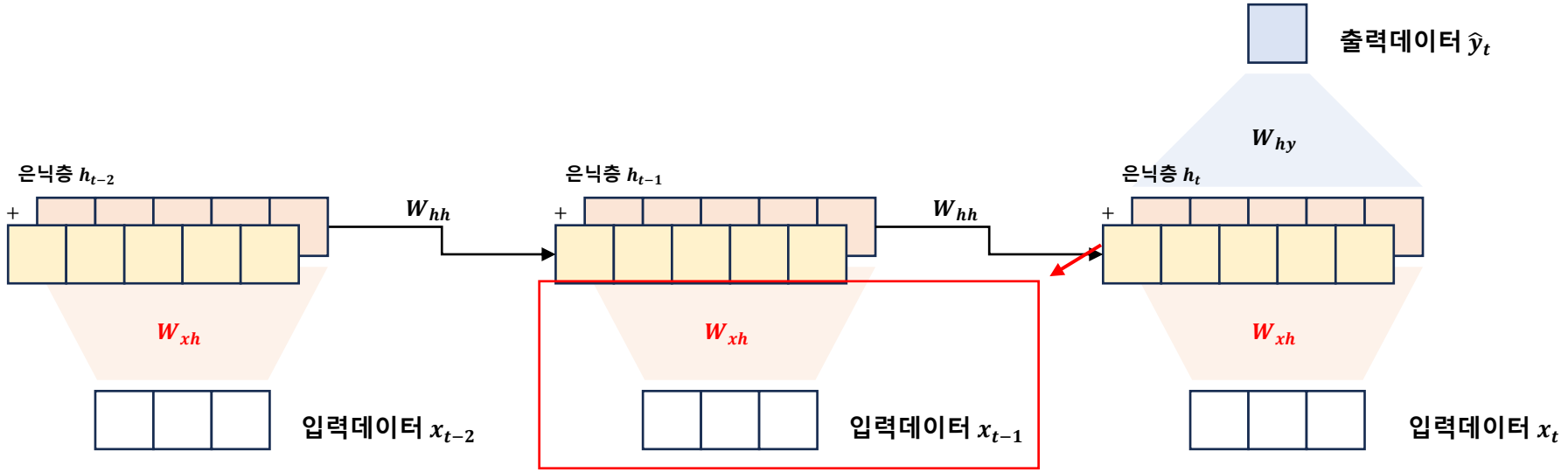


$$\frac{\partial Loss}{\partial W_{xh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{xh}}$$

t 시점에서의 영향

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

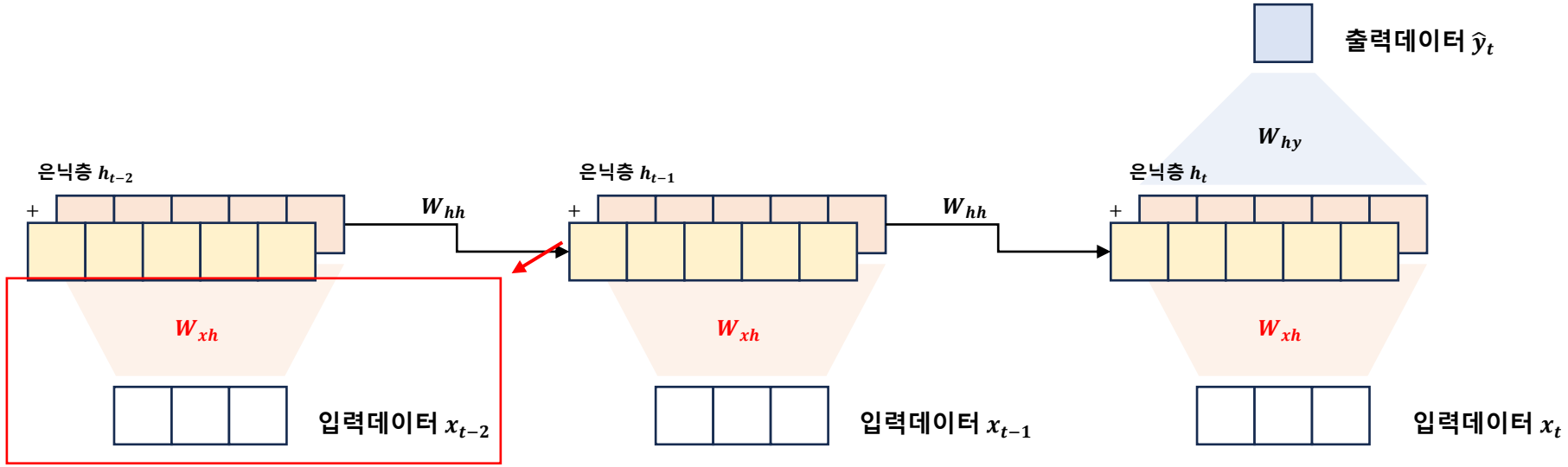


$$\frac{\partial Loss}{\partial W_{xh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{xh}}$$

t-1 시점에서의 영향

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)

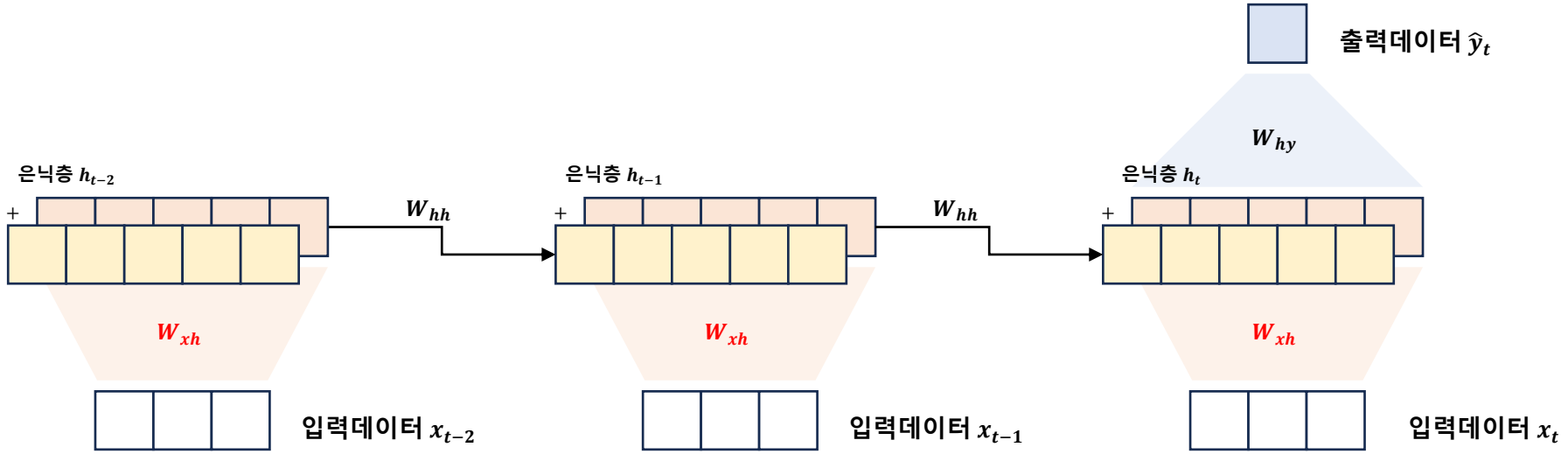


$$\frac{\partial Loss}{\partial W_{xh}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial W_{xh}} + \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_{t-1}} \times \frac{\partial h_{t-1}}{\partial h_{t-2}} \times \frac{\partial h_{t-2}}{\partial W_{xh}}$$

t-2 시점에서의 영향

RNN 학습

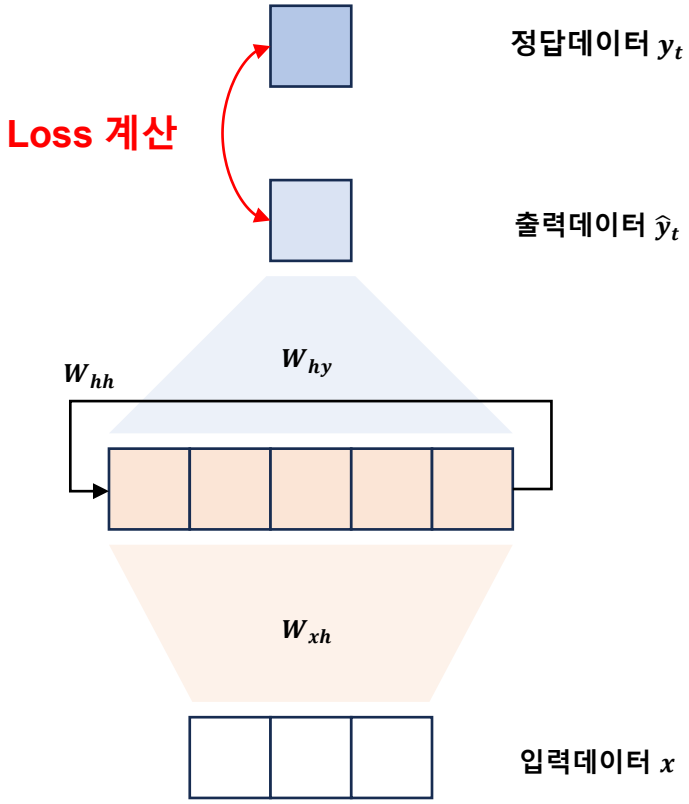
- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



$$\frac{\partial Loss}{\partial W_{xh}} = \sum_{k=0}^{t} \left(\frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_k} \times \frac{\partial h_k}{\partial W_{hh}} \right)$$

RNN 학습

- **학습 파라미터 (W_{hy}, W_{hh}, W_{xh})**
  - 각 파라미터는 매 시점마다 동일한 값을 사용함 (Shared parameter)



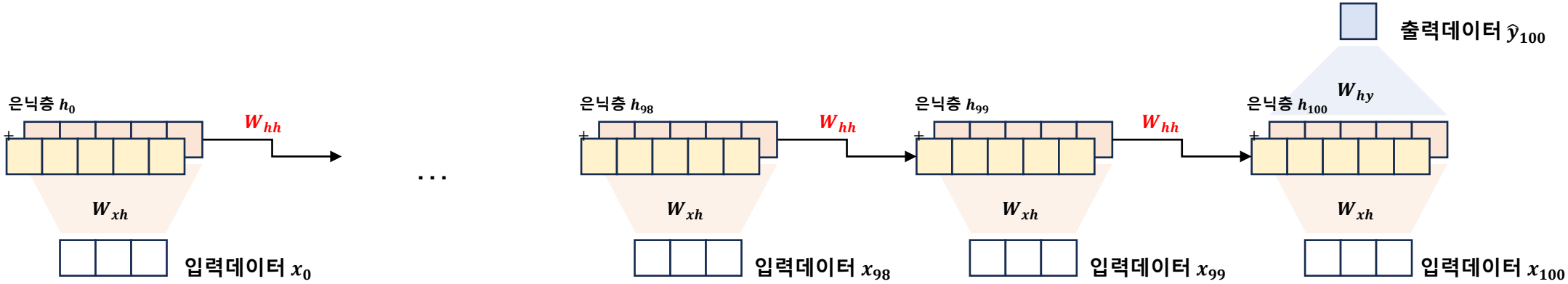
$$\frac{\partial Loss}{\partial W_{hy}} = \frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial W_{hy} h_t} \times \frac{\partial W_{hy} h_t}{\partial W_{hy}}$$

$$\frac{\partial Loss}{\partial W_{hh}} = \sum_{k=0}^{t} \left(\frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_k} \times \frac{\partial h_k}{\partial W_{hh}} \right)$$

$$\frac{\partial Loss}{\partial W_{xh}} = \sum_{k=0}^{t} \left(\frac{\partial Loss}{\partial \hat{y}_t} \times \frac{\partial \hat{y}_t}{\partial h_t} \times \frac{\partial h_t}{\partial h_k} \times \frac{\partial h_k}{\partial W_{hh}} \right)$$

RNN의 한계점

- **장기 의존성 문제 (Long-term dependency problem)**
  - Sequence의 길이가 길어질수록, 과거 정보 학습에 어려움이 발생함
  - 학습 과정 중 기울기 소실 (Vanishing Gradient) 발생

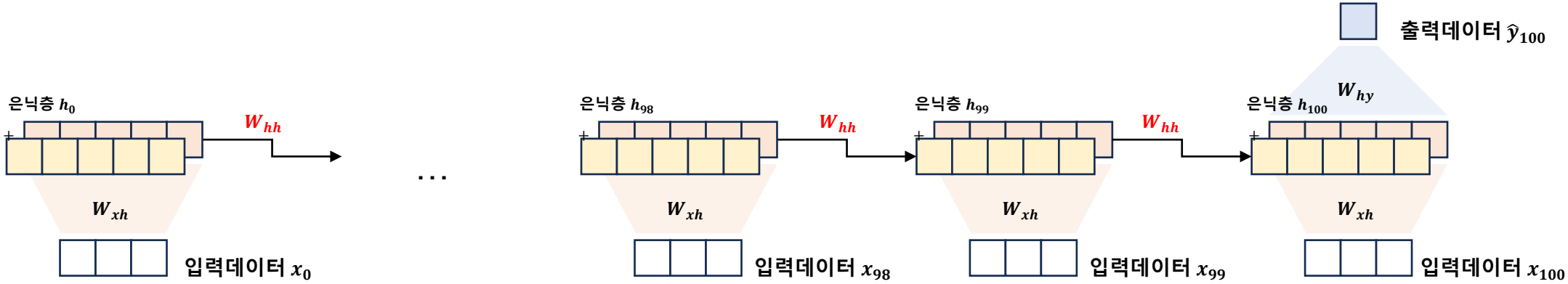


$$\frac{\partial Loss}{\partial W_{hh}} = \ \ldots + \frac{\partial Loss}{\partial \hat{y}_{100}} \times \frac{\partial \hat{y}_{100}}{\partial h_{100}} \times \frac{\partial h_{100}}{\partial h_{99}} \times \frac{\partial h_{99}}{\partial h_{98}} \times \frac{\partial h_{98}}{\partial h_{97}} \times \frac{\partial h_{97}}{\partial h_{96}} \times \ \ldots \times \frac{\partial h_3}{\partial h_2} \times \frac{\partial h_2}{\partial h_1} \times \frac{\partial h_1}{\partial h_0} \times \frac{\partial h_0}{\partial W_{hh}}$$

t = 0 시점에서의 영향

RNN의 한계점

- **장기 의존성 문제 (Long-term dependency problem)**
  - Sequence의 길이가 길어질수록, 과거 정보 학습에 어려움이 발생함
  - 학습 과정 중 기울기 소실 (Vanishing Gradient) 발생



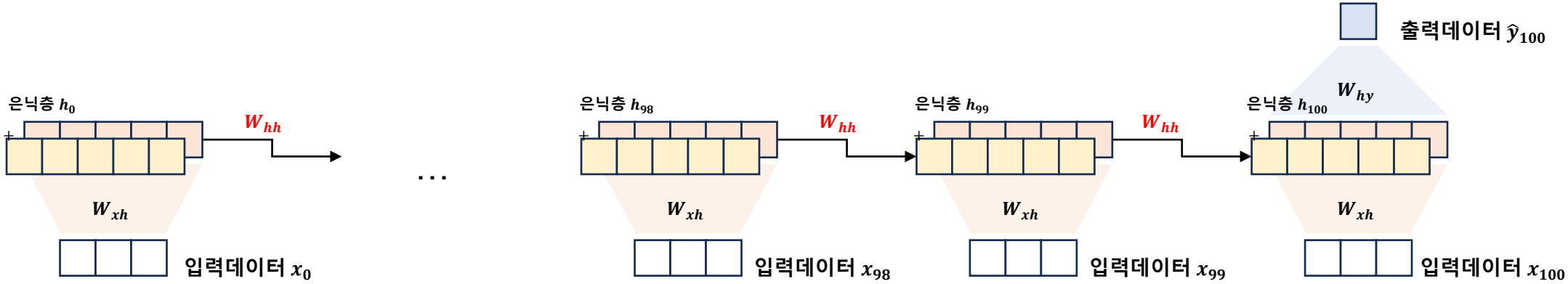
$$\frac{\partial Loss}{\partial W_{hh}} = \ldots + \frac{\partial Loss}{\partial \hat{y}_{100}} \times \frac{\partial \hat{y}_{100}}{\partial h_{100}} \times \frac{\partial h_{100}}{\partial h_{99}} \times \frac{\partial h_{99}}{\partial h_{98}} \times \frac{\partial h_{98}}{\partial h_{97}} \times \frac{\partial h_{97}}{\partial h_{96}} \times \ldots \times \frac{\partial h_3}{\partial h_2} \times \frac{\partial h_2}{\partial h_1} \times \frac{\partial h_1}{\partial h_0} \times \frac{\partial h_0}{\partial W_{hh}} \approx 0$$

0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1

t = 0 시점에서의 영향

RNN의 한계점

- **장기 의존성 문제 (Long-term dependency problem)**
  - Sequence의 길이가 길어질수록, 과거 정보 학습에 어려움이 발생함
  - 학습 과정 중 기울기 소실 (Vanishing Gradient) 발생



$$\frac{\partial Loss}{\partial W_{hh}} = \ldots + \frac{\partial Loss}{\partial \hat{y}_{100}} \times \frac{\partial \hat{y}_{100}}{\partial h_{100}} \times \frac{\partial h_{100}}{\partial h_{99}} \times \frac{\partial h_{99}}{\partial h_{98}} \times \frac{\partial h_{98}}{\partial h_{97}} \times \frac{\partial h_{97}}{\partial h_{96}} \times \ldots \times \frac{\partial h_3}{\partial h_2} \times \frac{\partial h_2}{\partial h_1} \times \frac{\partial h_1}{\partial h_0} \times \frac{\partial h_0}{\partial W_{hh}} \approx 0$$

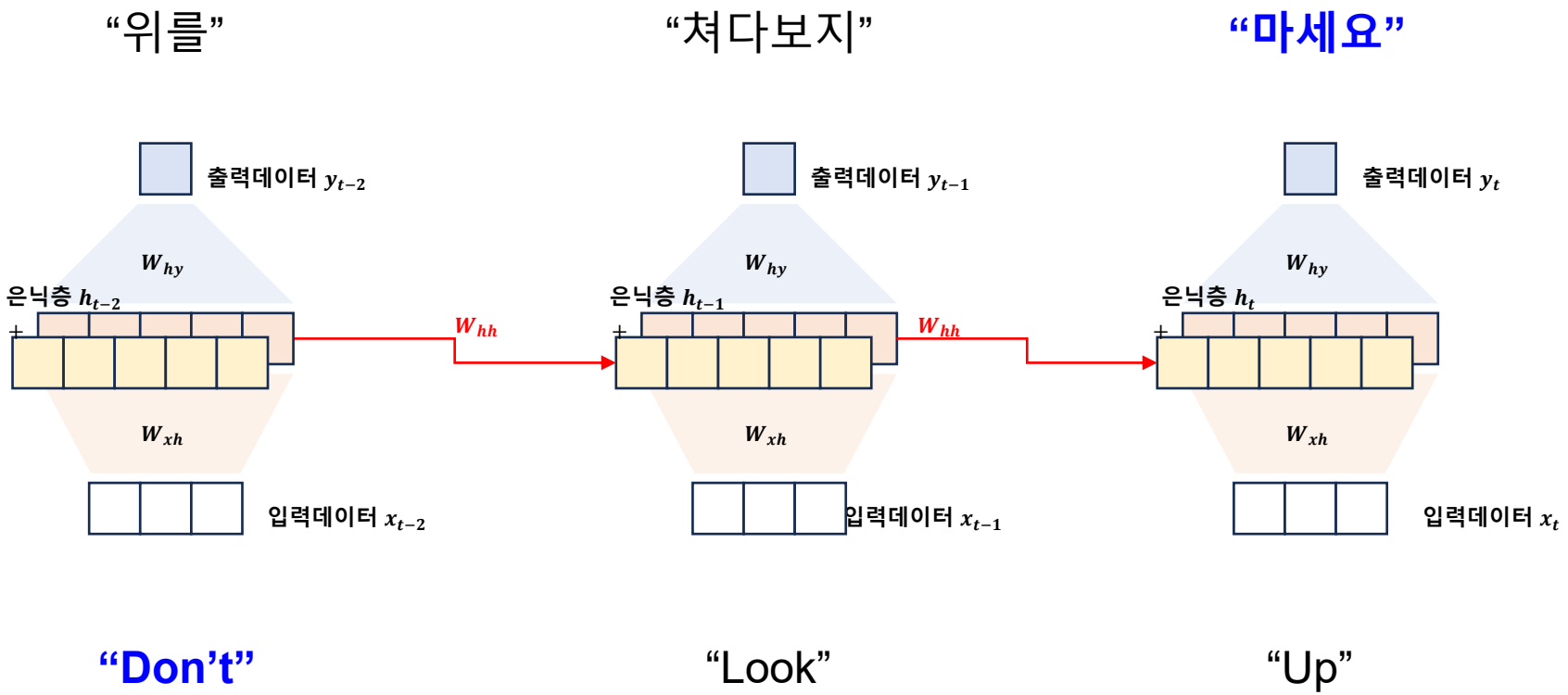
0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1 0 ~ 1

→ 기울기가 소실되어 parameter가 업데이트 되지 않음

t = 0 시점에서의 영향

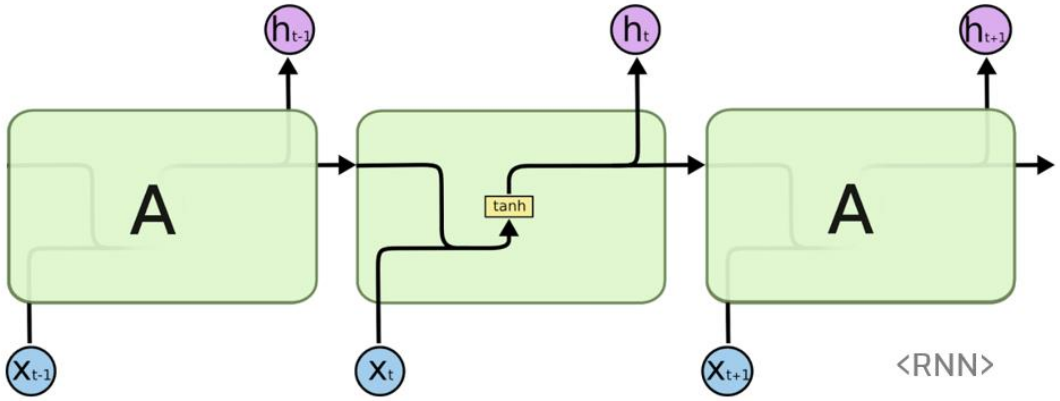
RNN의 한계점

- **장기 의존성 문제 (Long-term dependency problem)**
  - Ex) 영어 → 한글 번역 (Don't look up → 위를 쳐다보지 마세요.)
  - 입력의 "Don't" 과 출력의 "마세요"는 의미상으로 가까운 단어이지만 학습에 반영되지 않는다면 정확도 감소 가능성 존재



LSTM

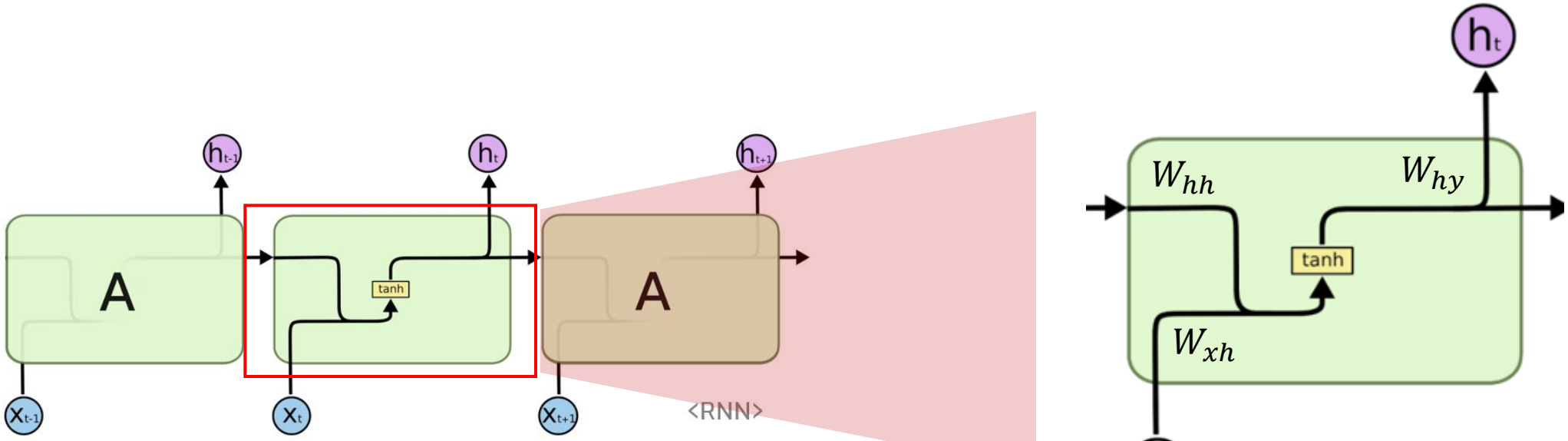
- **장단기 메모리 순환신경망 (Long Short-Term Memory)**
  - RNN의 장기 의존성 문제를 완화한 개선 모델
  - Cell state 구조를 제안하고 세가지 gate 추가함



RNN

LSTM

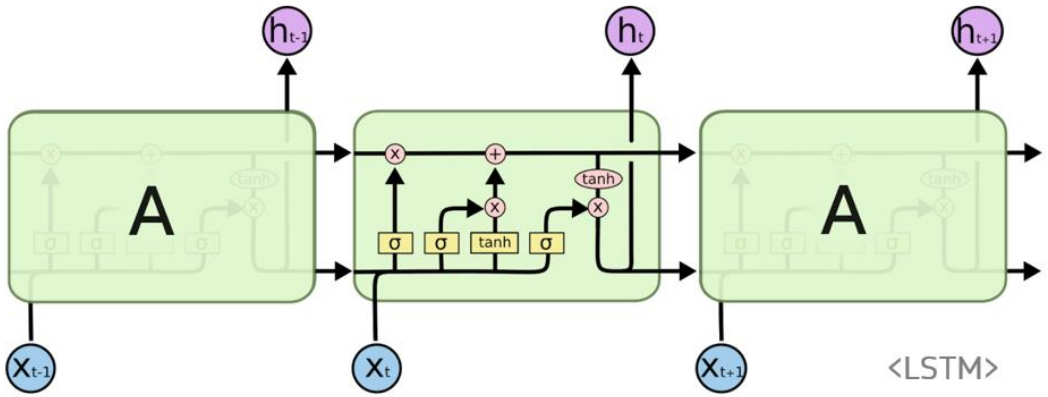
- **장단기 메모리 순환신경망 (Long Short-Term Memory)**
  - RNN의 장기 의존성 문제를 완화한 개선 모델
  - Cell state 구조를 제안하고 세가지 gate 추가함



RNN

LSTM

- **장단기 메모리 순환신경망 (Long Short-Term Memory)**
  - RNN의 장기 의존성 문제를 완화한 개선 모델
  - Cell state 구조를 제안하고 세가지 gate 추가함
    - Forget gate(f_t), Input gate(i_t), Output gate(o_t)



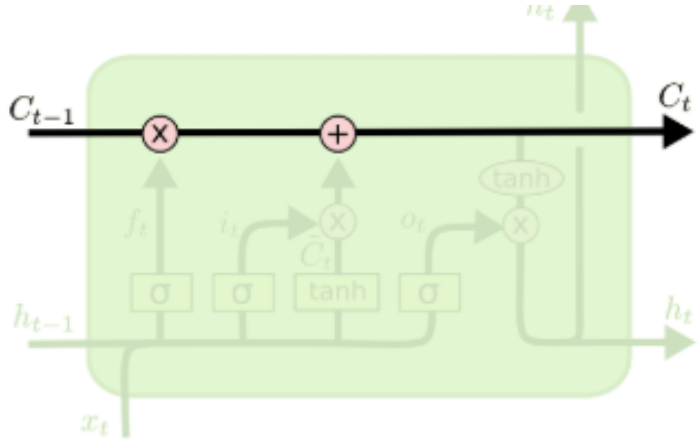
Cell state (C_t)

Hidden state (h_t)

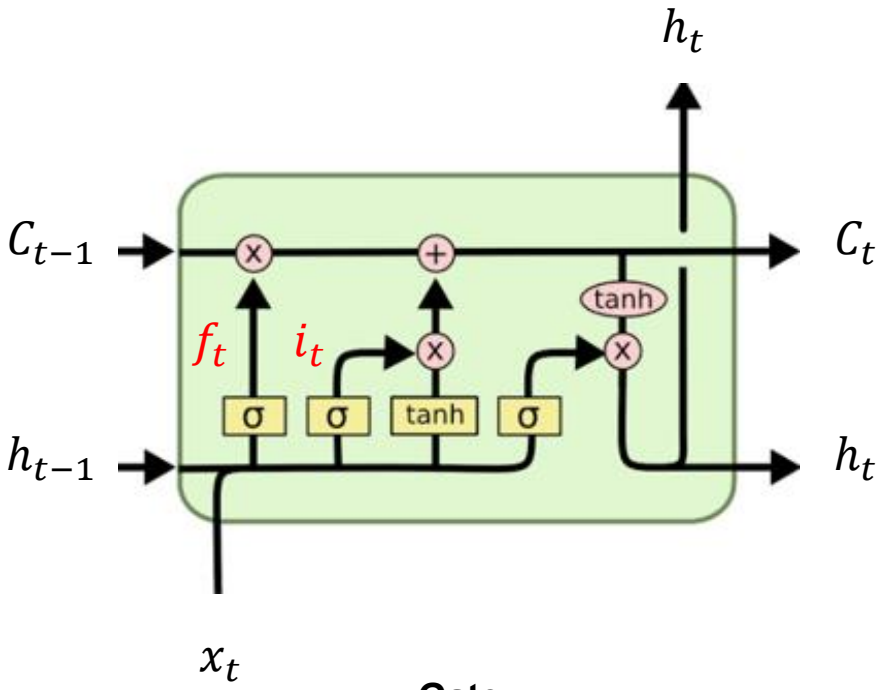
LSTM

LSTM

- **Cell state (C_t)**
  - LSTM의 핵심 구조로써, **장기적인 정보 (Long term)들을 유지**
  - 두가지 gate (Forget gate(f_t), Input gate(i_t))를 통해 cell state 업데이트



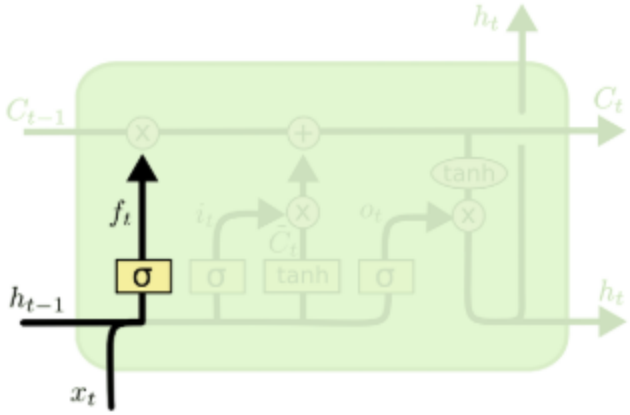
Cell state



Gate

LSTM

- **Cell state (C_t)**
  - Forget gate(f_t): 불필요한 과거 정보를 잊기 위한 gate



Forget gate

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

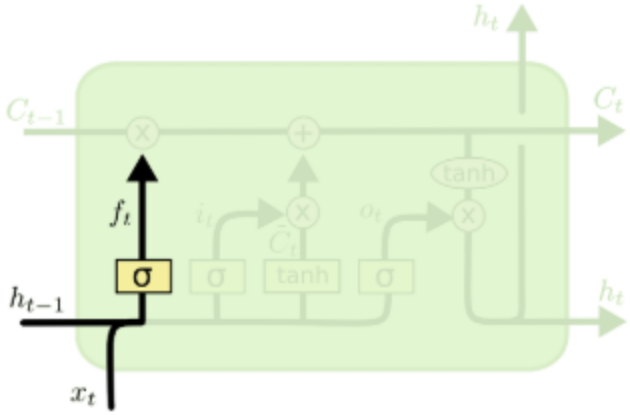
Sigmoid (0 ~ 1 사이 가중치)

다 잊는 경우

f_t	0	0	0	0	0	0	0
C_{t-1}	0.2	0.1	0.3	0.5	0.2	0.2	0.3
$f_t \otimes C_{t-1}$	0	0	0	0	0	0	0

LSTM

- **Cell state (C_t)**
  - Forget gate(f_t): 불필요한 과거 정보를 잊기 위한 gate



Forget gate

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

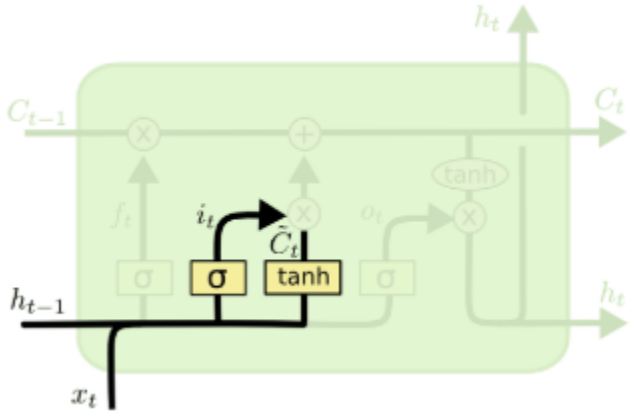
Sigmoid (0 ~ 1 사이 가중치)

모두 기억하는 경우

f_t	1	1	1	1	1	1	1
C_{t-1}	0.2	0.1	0.3	0.5	0.2	0.2	0.3
$f_t \otimes C_{t-1}$	0.2	0.1	0.3	0.5	0.2	0.2	0.3

LSTM

- **Cell state (C_t)**
  - Forget gate(f_t): 불필요한 과거 정보를 잊기 위한 gate
  - Input gate(i_t): 현재 정보를 기억하기 위한 gate



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

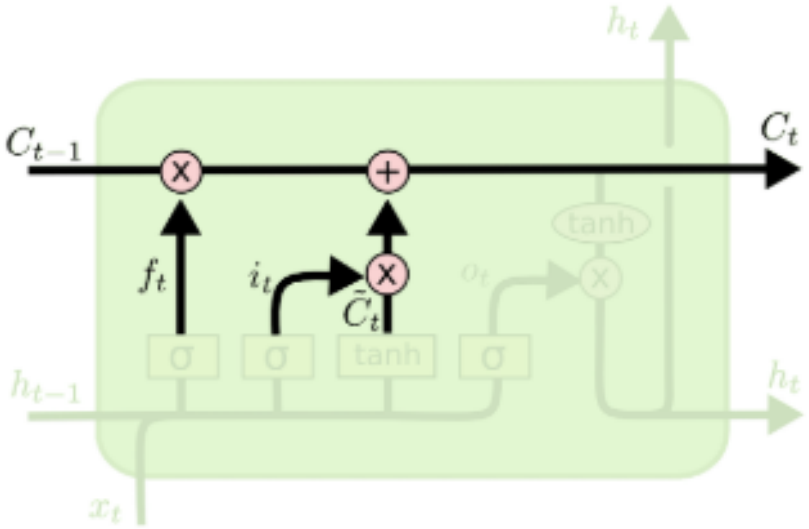
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

i_t	0.1	0	0.8	0.2	0.8	0.7	1
$\tilde{C}_t$	0.1	0.3	0.6	0.2	0.9	0.1	0.4
$i_t \otimes \tilde{C}_t$	0.01	0	0.48	0.04	0.72	0.07	0.4

Input gate

LSTM

- **Cell state (C_t)**
  - Forget gate(f_t): 불필요한 과거 정보를 잊기 위한 gate
  - Input gate(i_t): 현재 정보를 기억하기 위한 gate
  - Cell state (C_t) = 불필요한 정보를 제거한 이전 시점의 cell state (C_{t-1}) + 현재 시점의 cell state ($\tilde{C}_t$)

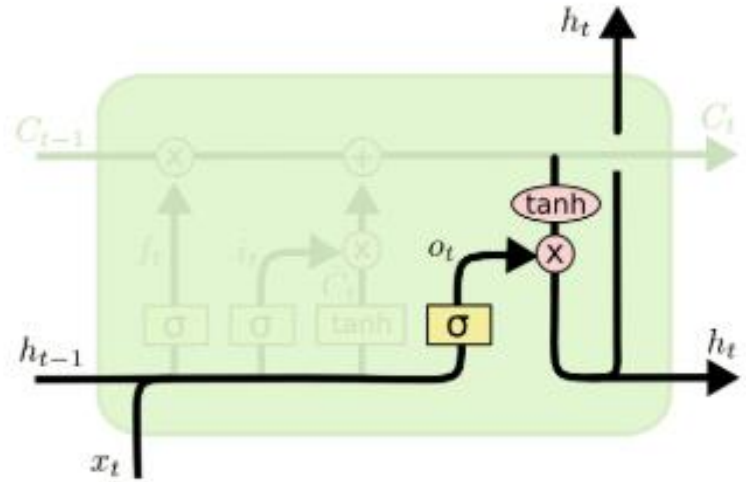


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Cell state 업데이트

LSTM

- **Hidden state (h_t)**
  - 단기적인 정보 (Short term)을 유지
  - Output gate(o_t): Hidden state에 cell state를 얼마나 반영할 것인지에 대한 가중치



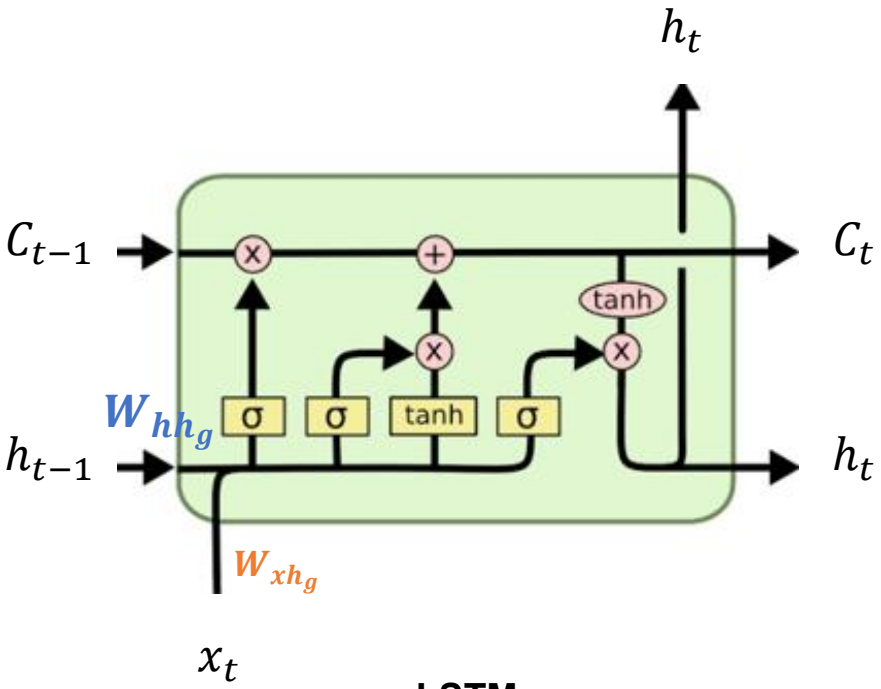
$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

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

Hidden state 업데이트

LSTM

- **Review**
  - Cell state (C_t): 현 시점에 대한 장기적인 정보 (Long term)들을 유지
  - Hidden state (h_t): 현 시점에 대한 단기적인 정보 (Short term)을 유지



LSTM

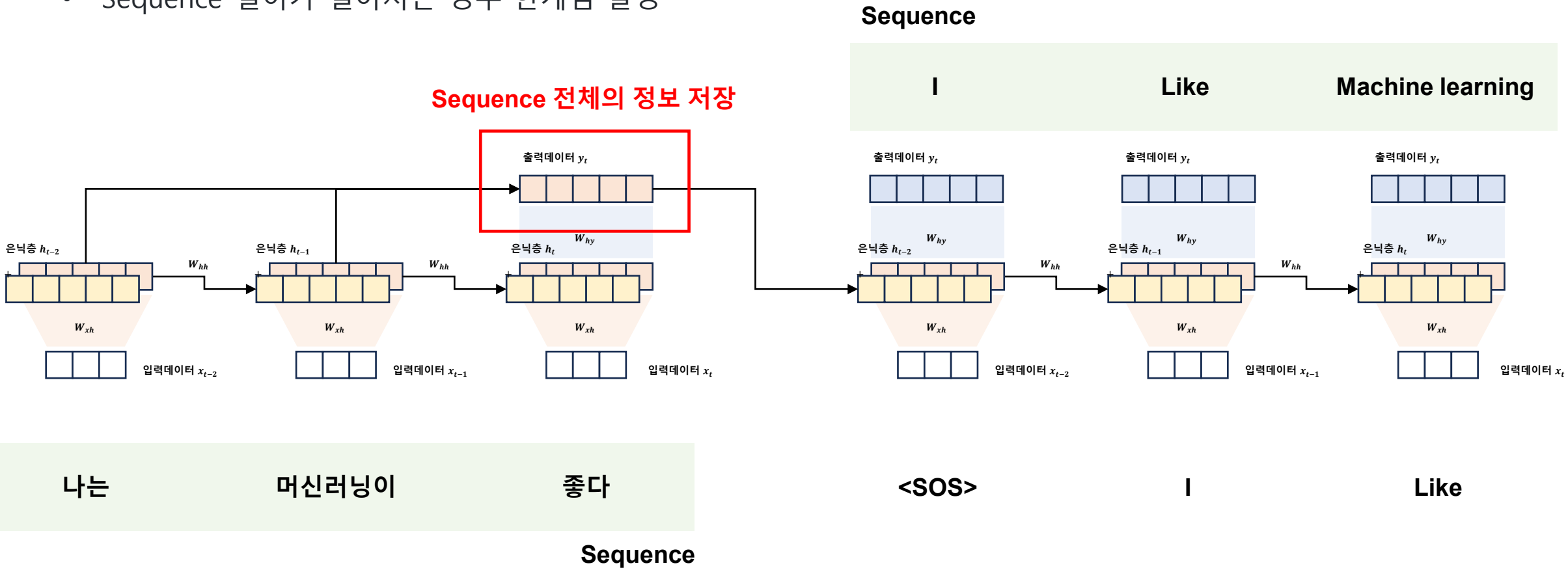
$$f_t = \sigma(W_{xhf}x_t + W_{hhf}h_{t-1} + bias)$$

$$i_t = \sigma(W_{xhi}x_t + W_{hhi}h_{t-1} + bias)$$

$$o_t = \sigma(W_{xho}x_t + W_{hho}h_{t-1} + bias)$$

LSTM의 한계점 및 Transformer

- **장기의존성 문제**
  - LSTM을 이용하여 문제를 완화하였으나 여전히 장기의존성 문제가 존재함 → Seq2Seq 방법으로 완화함
  - Sequence 길이가 길어지는 경우 한계점 발생



LSTM의 한계점 및 Transformer

- **장기의존성 문제**
  - LSTM을 이용하여 문제를 완화하였으나 여전히 장기의존성 문제가 존재함
  - Sequence 길이가 길어지는 경우 한계점 발생

- **병렬처리 문제**
  - RNN 및 LSTM은 순차적으로 입력, 출력하는 구조이기 때문에 병렬처리가 어려움

- **Transformer**
  - Self attention 기법을 활용한 transformer가 위의 문제점을 해결함
  - 이후 많은 분야에서 transformer를 이용한 연구 진행 중

Questions & Answers

Dongsan Jun (dsjun@dau.ac.kr)
Image Signal Processing Laboratory (www.donga-ispl.kr)
Division of Computer·AI Engineering
Dong-A University, Busan, Rep. of Korea