

혼자 공부하는 머신러닝 + 딥러닝

Chapter 09.

텍스트를 위한 인공 신경망

순차 데이터(sequential data)

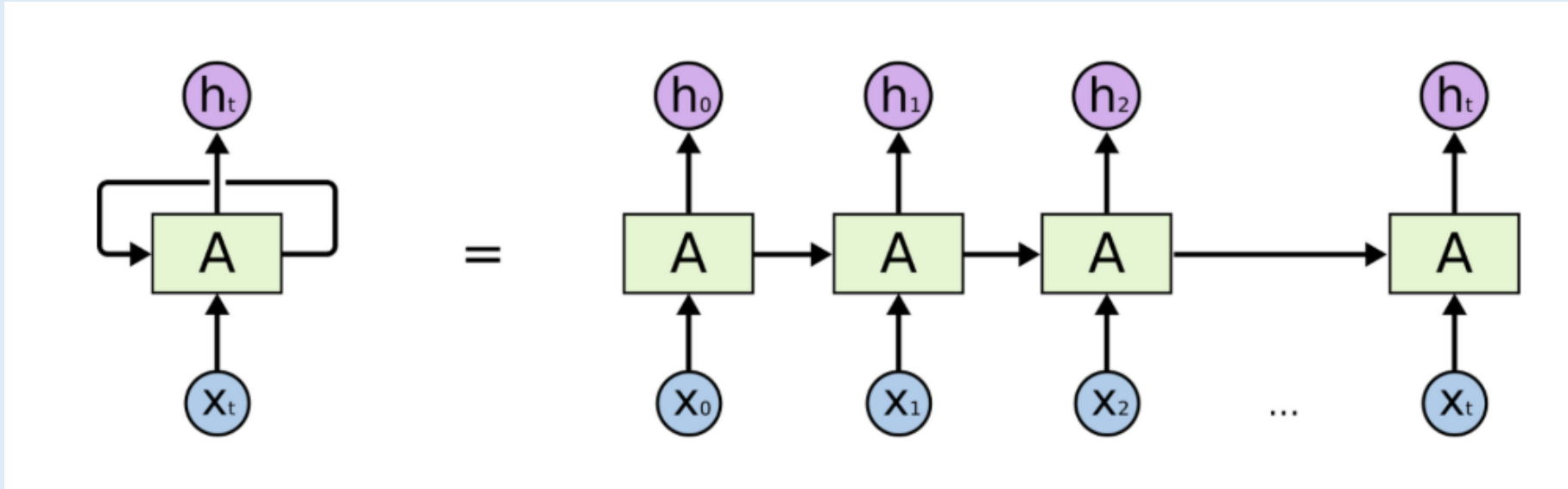
- 순서에 의미가 있는 데이터
- 순서를 유지하며 신경망에 주입해야 함.
- 이전에 입력한 데이터를 기억하는 기능 필요
- ex) 텍스트, 시계열 데이터

순환 신경망
(RNN)

Recurrent Neural Network, RNN

- 스스로를 반복하면서 이전 단계에서 얻은 정보가 지속되도록 함.

순환 신경망 (RNN)



cell (셀)

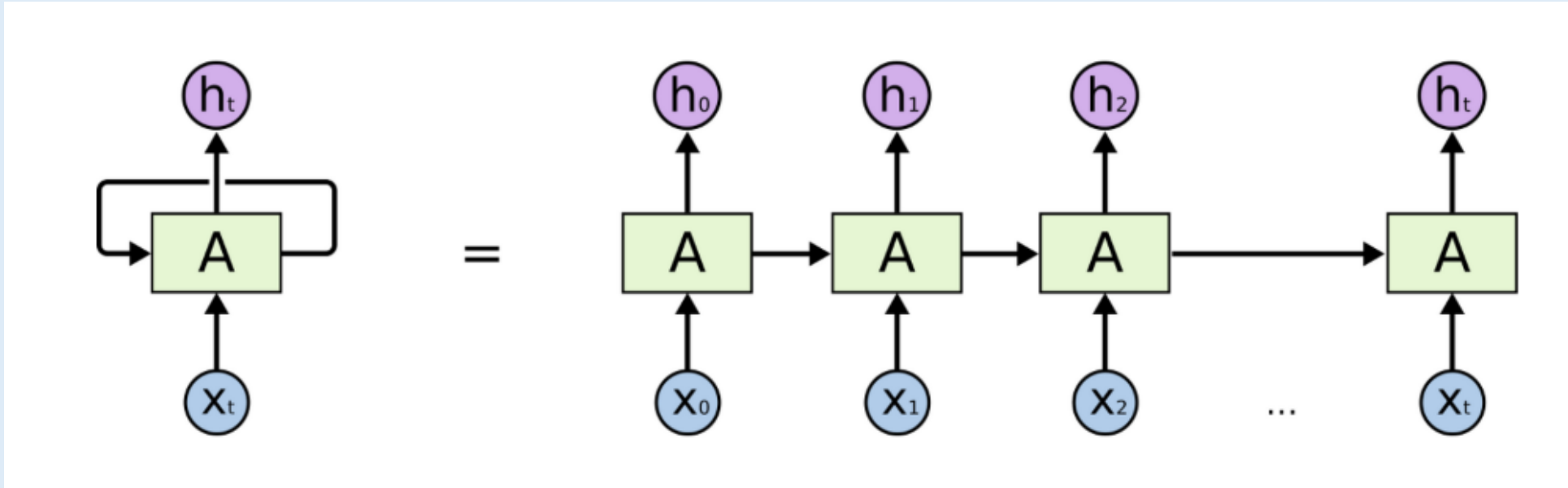
RNN에서 hidden layer에서 activation function을 통해 출력을 내보내는 역할을 하는 node

= memory cell, RNN cell

hidden state (은닉 상태)

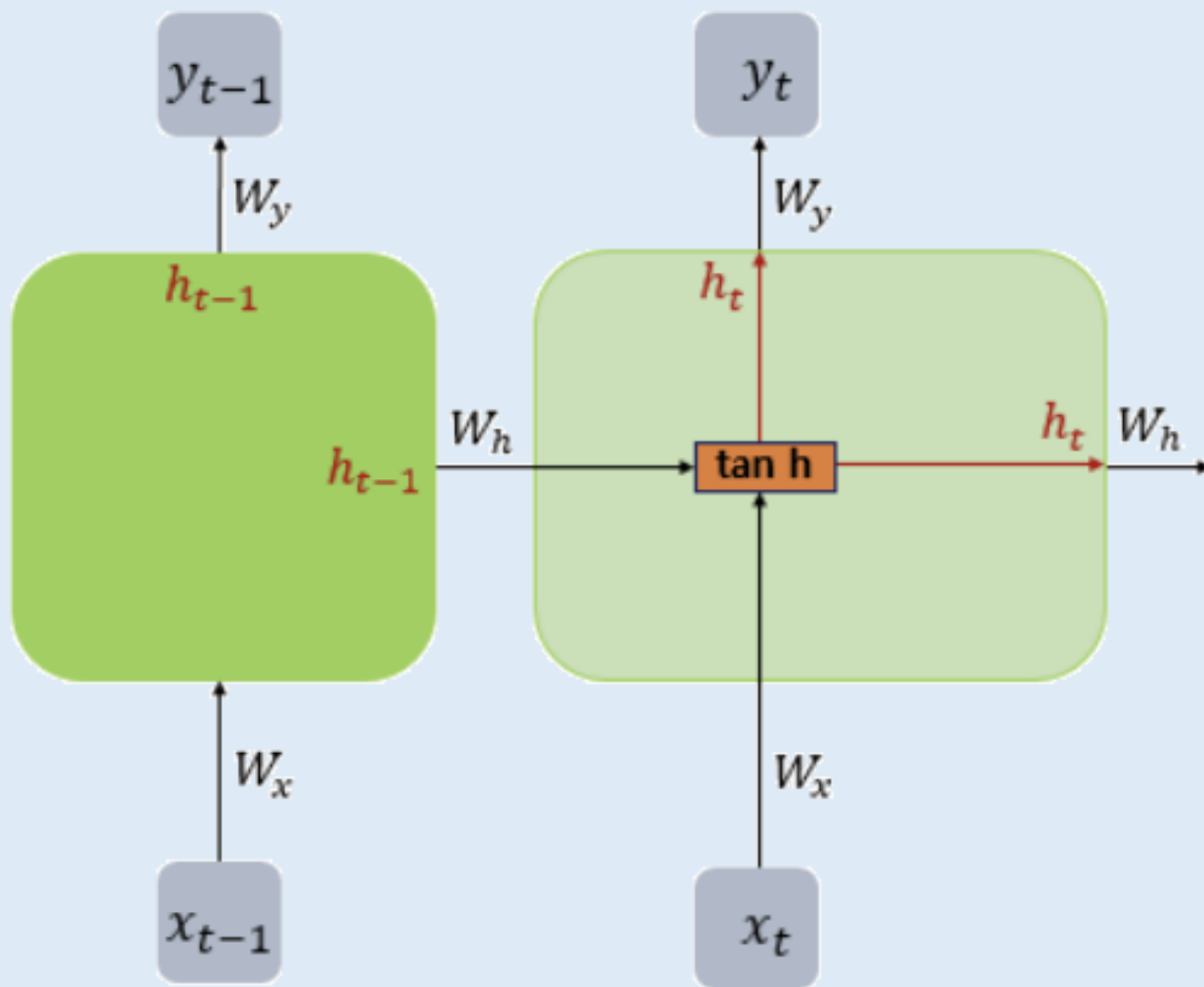
- memory cell이 출력층 방향으로 또는 다음 시점인 $t+1$ 의 자신에게 보내는 값
- 네트워크가 시간적인 의존성을 학습하고 과거 정보를 기억하며 데이터를 처리하는 데 사용됨

순환 신경망 (RNN)



- A : 입력으로 x_t 를 받아 h_t 를 출력
- A를 둘러싼 반복(화살표) : 다음 단계에서의 network가 이전 단계의 정보를 받음.
- t 시점의 memory cell은 t-1 시점의 memory cell이 보낸 hidden state 값을 t 시점의 hidden state 계산을 위한 입력값으로 사용함.

순환 신경망 (RNN)



h_t

현재 시점 t 에서의 은닉상태값

W_x

입력층을 위한 가중치

W_h

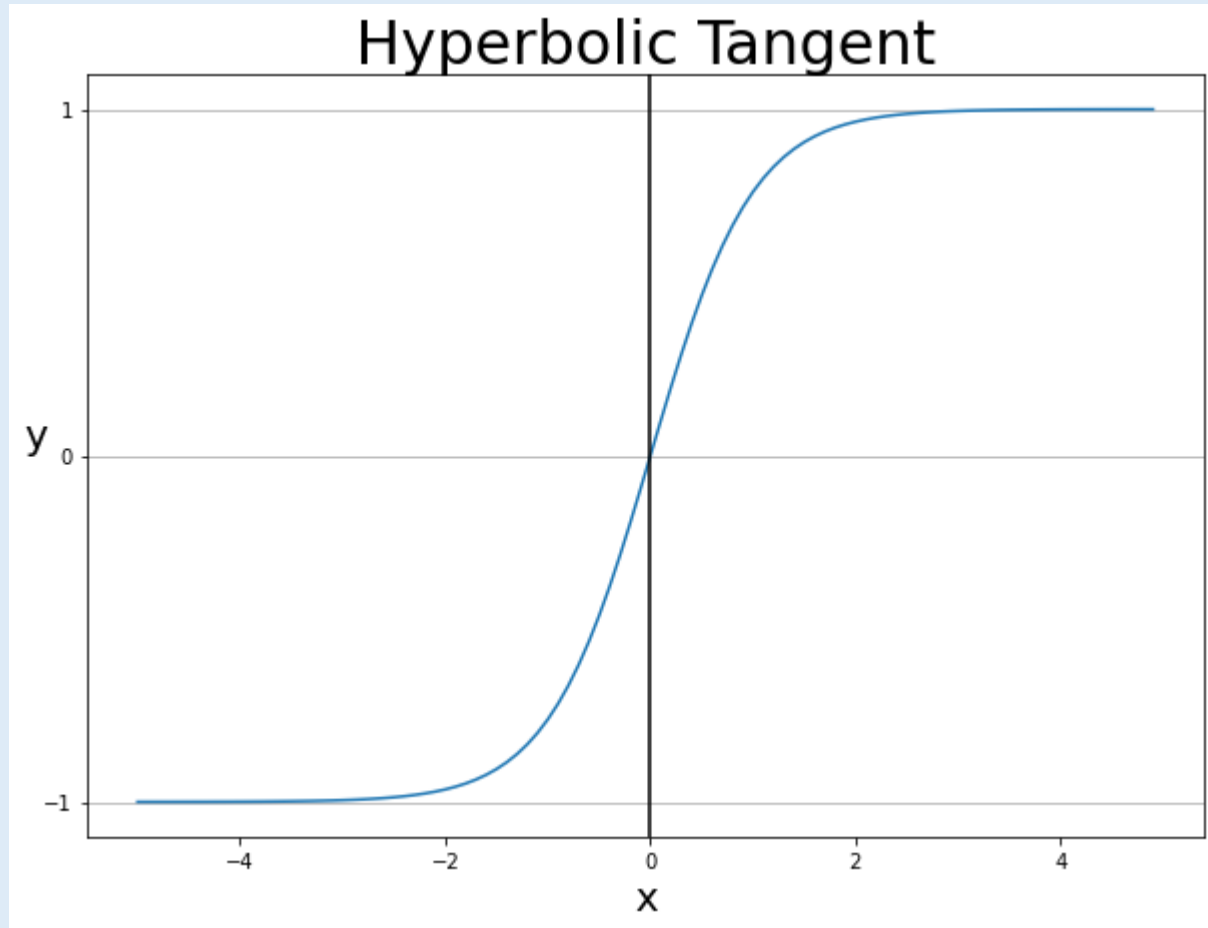
이전 시점 $t-1$ 의 은닉 상태값인 h_{t-1} 을 위한 가중치

식

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

출력층 : $y_t = f(W_y h_t + b)$

하이퍼볼릭 탄젠트 함수 (hyperbolic tangent, tanh)



- RNN에서 은닉층의 활성화 함수로 많이 사용함.
- 시그모이드 함수를 일부 보완함.

(1) 기울기가 양수, 음수 모두 가능 ->

시그모이드 함수보다 학습 효율성이 뛰어남.

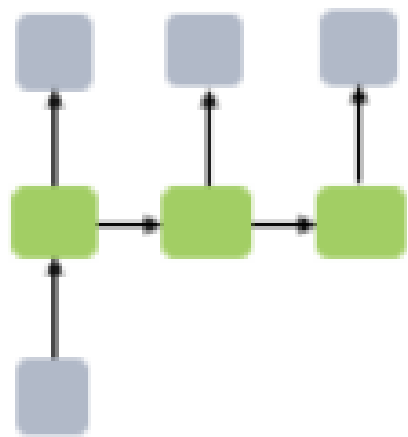
(2) -1~ 1 사이의 범위를 가짐 -> 범위가 더 넓기 때문에 출력값의 변화폭이 더 크고, 그로 인해 기울기 소실 증상이 더 적음.

기울기 소실 (Gradient Vanishing) :

미분 함수에 대하여, 값이 일정 이상 커지는 경우 미분값이 소실되는 현상

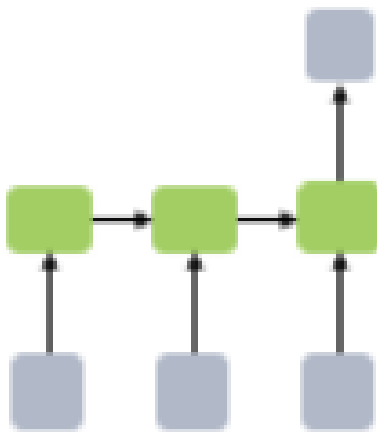
순환 신경망 (RNN)의 형태

- 입력과 출력의 길이(x_t 와 h_t 의 개수)를 다르게 설계할 수 있음.
- 다양한 용도로 사용 가능



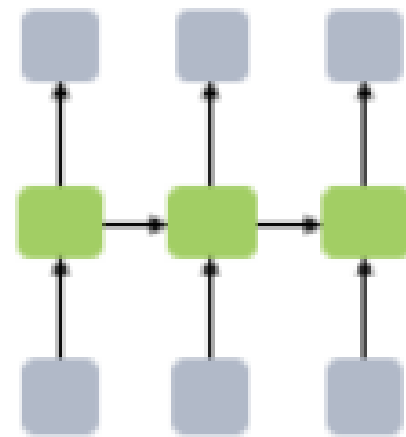
일 대 다
(one-to-many)

ex) 이미지 캡셔닝



다 대 일
(many-to-one)

ex) 감성 분류, 스팸 메일 분류

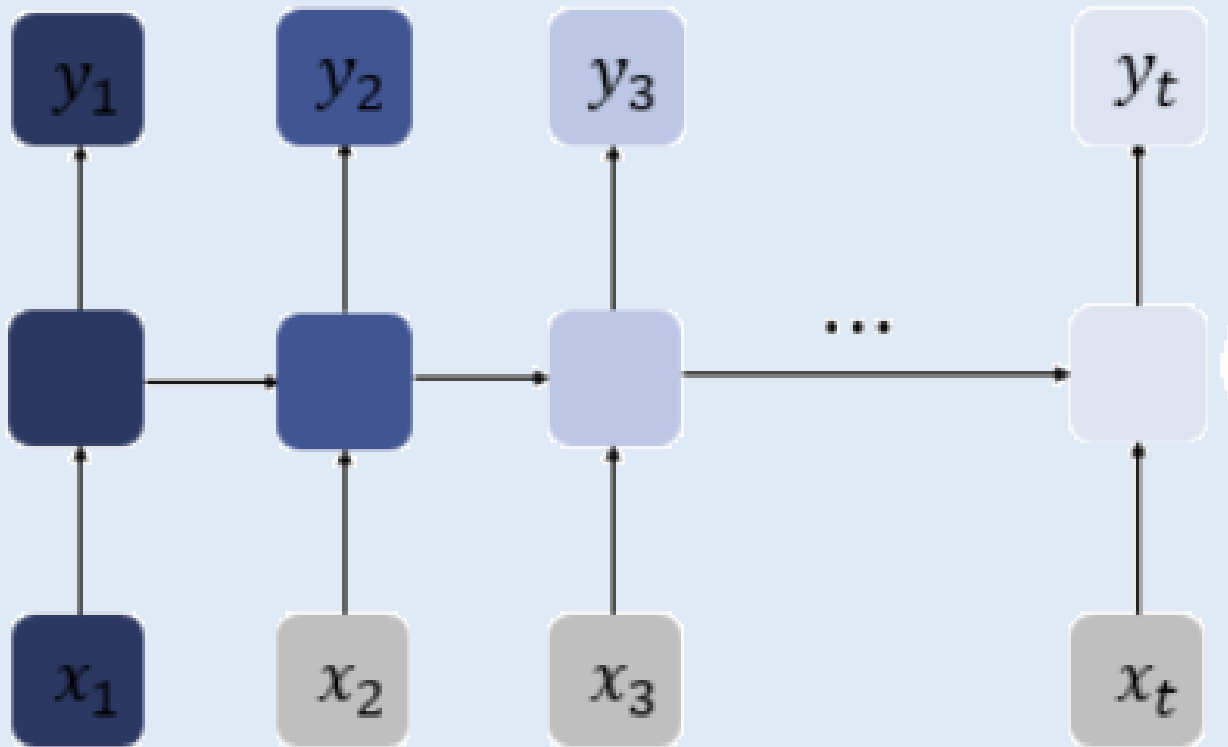


다 대 다
(many-to-many)

ex) 챗봇, 번역기

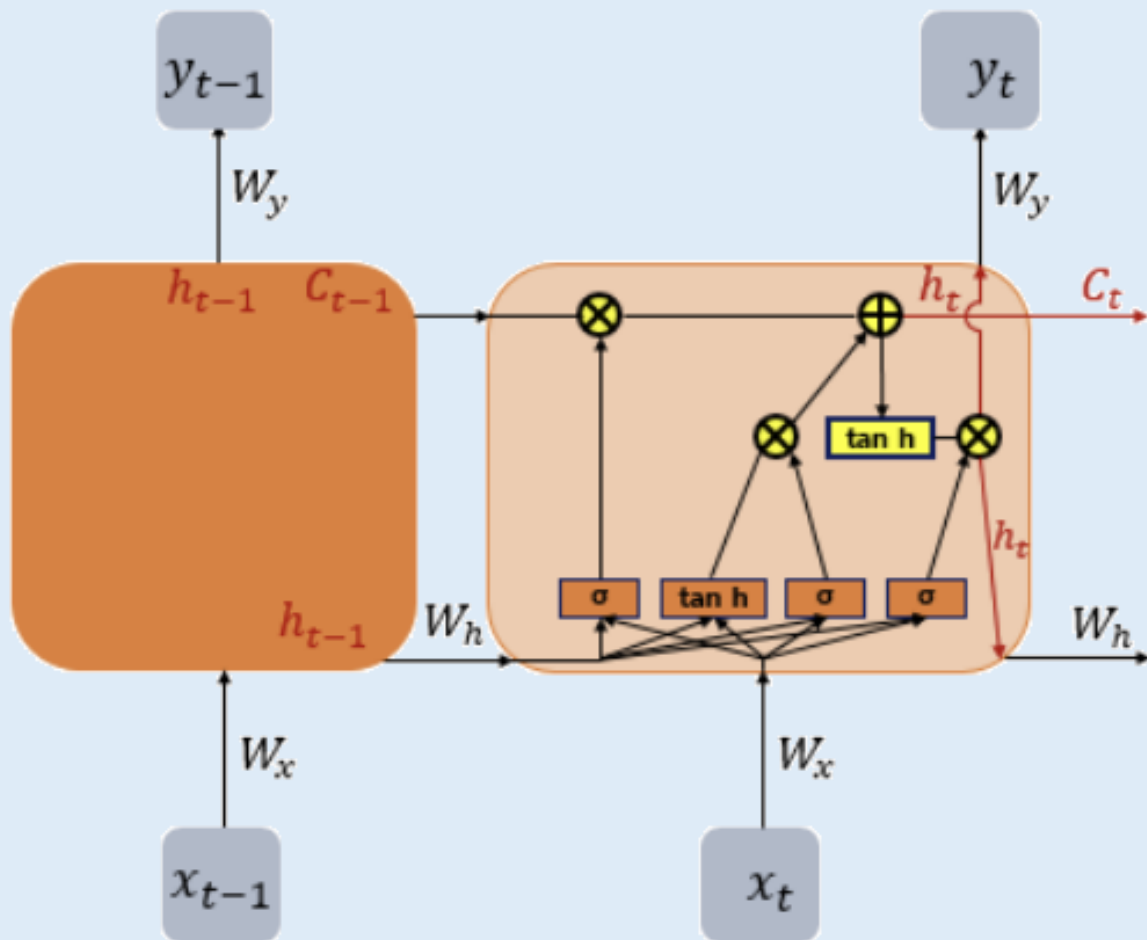
RNN의 한계점

- RNN은 출력 결과가 이전의 계산 결과에 의존함.
- 비교적 짧은 시퀀스에 대해서만 효과를 보임.
- RNN의 time step이 길어질수록 앞의 정보가 뒤로 충분히 전달되지 못하는 현상 발생함.



장기 의존성 문제

장단기 메모리 (Long Short-Term Memory, LSTM)



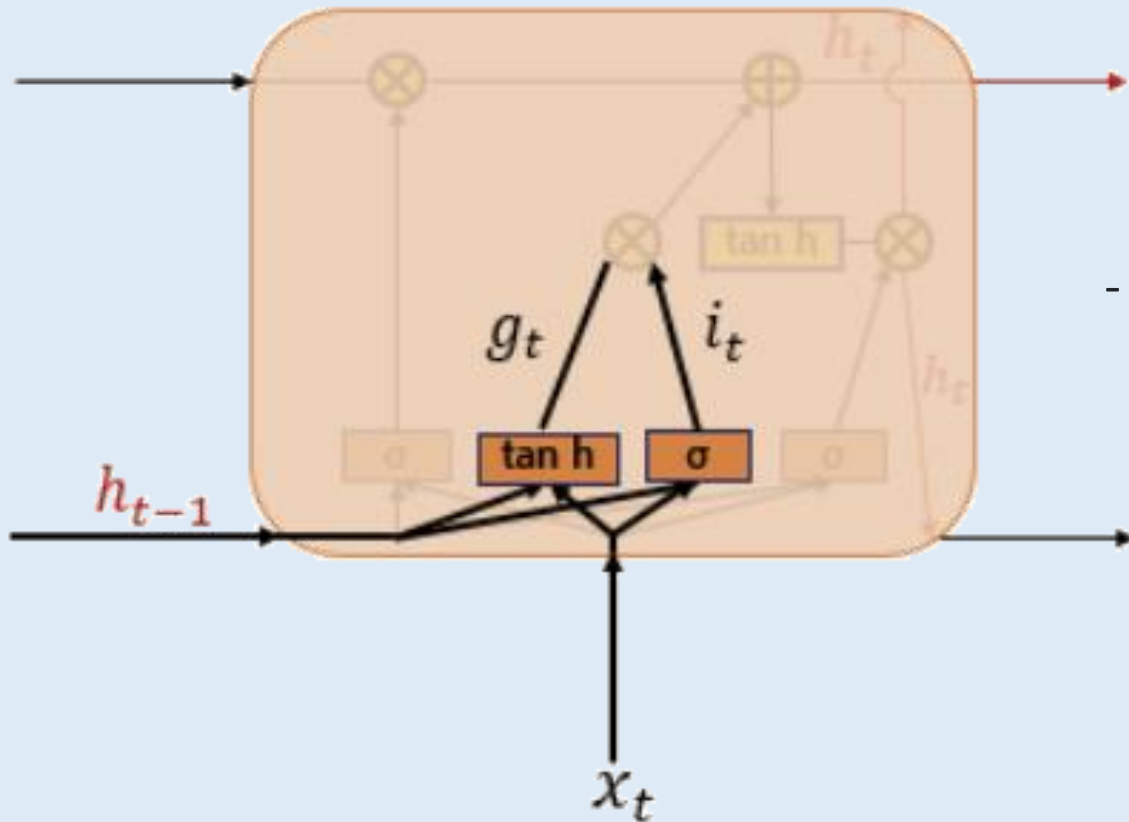
- 은닉층의 메모리 셀에 입력 게이트, 삭제 게이트, 출력 게이트를 추가하여 불필요한 기억을 지우고, 기억해야 할 것들을 정함.
- 긴 시퀀스의 입력을 처리하는데 탁월한 성능을 보임.

+ 셀 상태 (cell state) : C_t

- 왼쪽에서 오른쪽으로 가는 굵은 선
- 은닉 상태처럼 이전 시점의 셀 상태가 다음 시점의 셀 상태를 구하기 위한 입력으로 사용됨.

LSTM - 입력 게이트

- 현재 정보를 기억하기 위한 게이트
- 현재 시점의 입력을 얼마나 반영할지 결정
- 두 값으로 기억할 정보의 양 정함.



- i_t : 현재 시점 t 의 x 값과 입력 게이트로 이어지는 가중치를 곱한 값과 이전 시점 $t-1$ 의 은닉 상태가 입력 게이트로 이어지는 가중치를 곱한 값을 더하여 시그모이드 함수를 지난 값

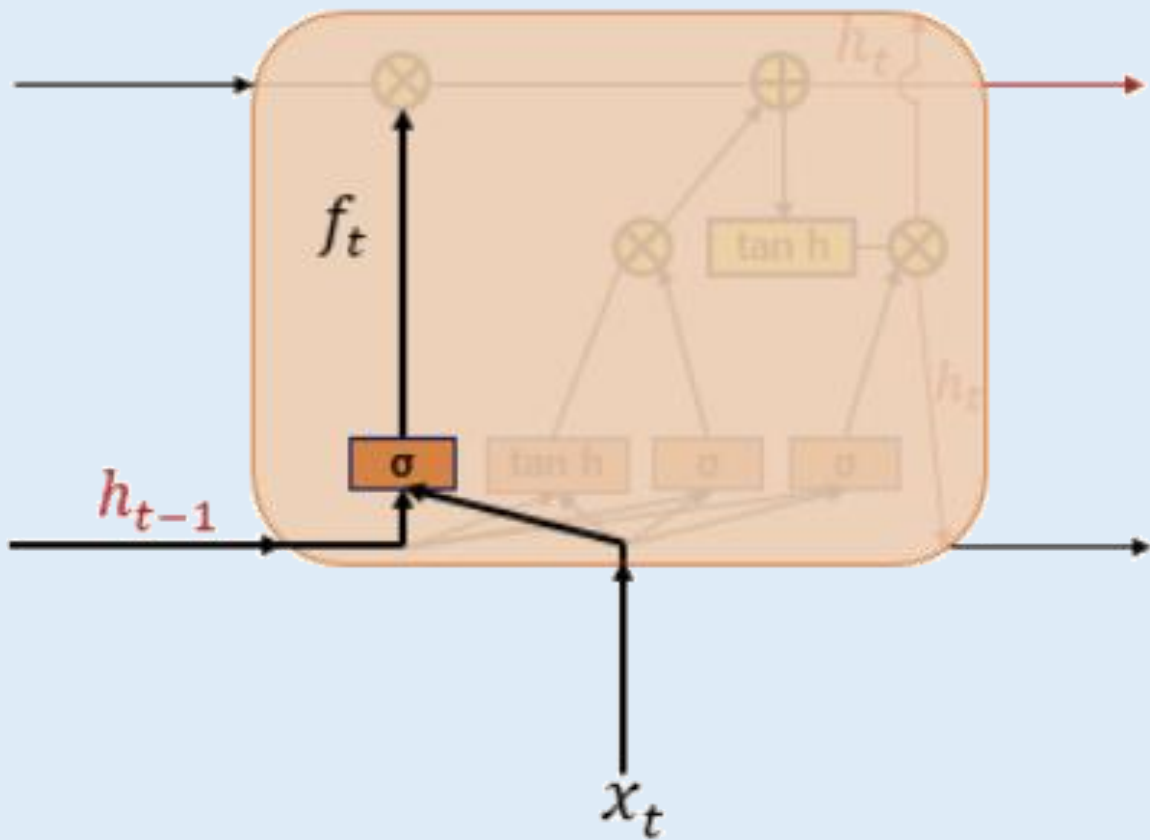
$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$$

- g_t : 현재 시점 t 의 x 값과 입력 게이트로 이어지는 가중치를 곱한 값과 이전 시점 $t-1$ 의 은닉 상태가 입력 게이트로 이어지는 가중치를 곱한 값을 더하여 하이퍼보릭탄젠트 함수를 지난 값

$$g_t = \tanh(W_{xg}x_t + W_{hg}h_{t-1} + b_g)$$

LSTM - 삭제 게이트

- 기억을 삭제하기 위한 게이트
- 이전 시점의 입력을 얼마나 반영할지 의미함.

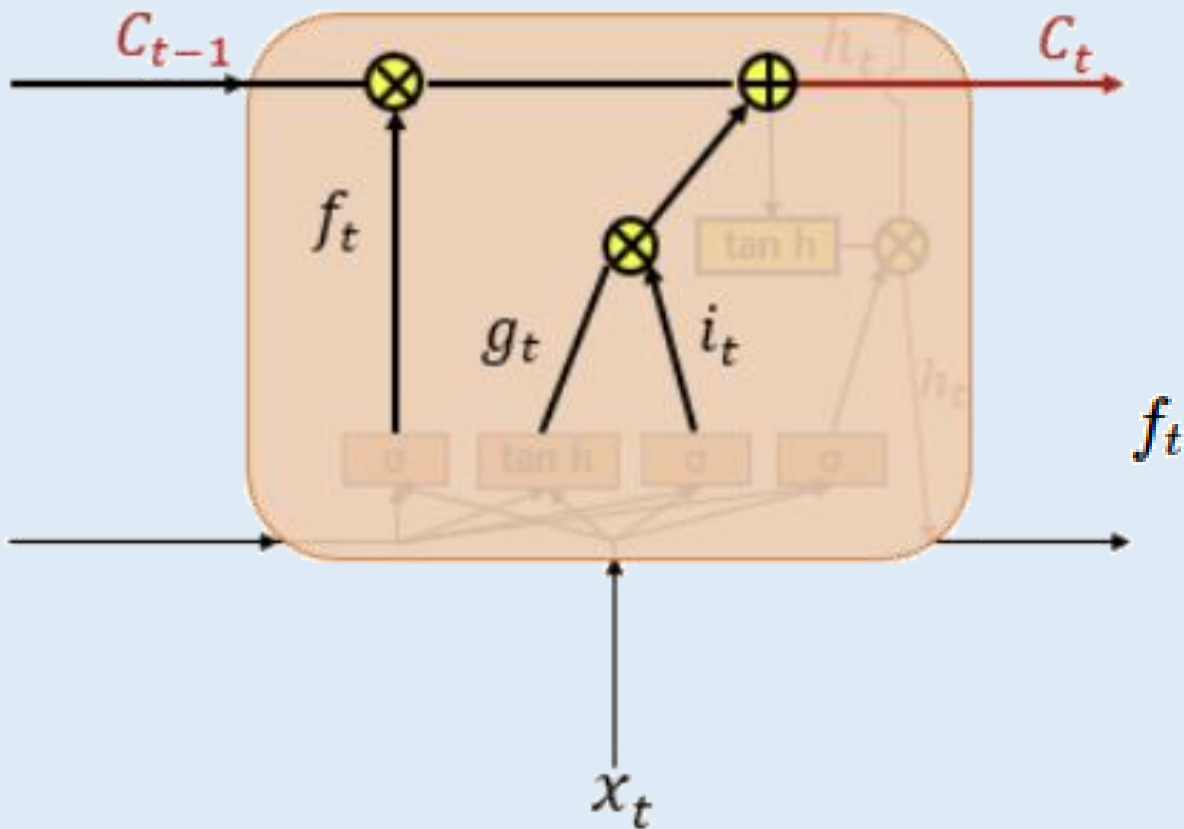


f_t : 현재 시점 t 의 x 값과 이전 시점 $t-1$ 의 은닉 상태가 시그모이드 함수를 지난 값

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$

- f_t - 0과 1 사이의 값
- 삭제 과정을 거친 정보의 양
 - 0에 가까울수록 -> 정보가 많이 삭제됨.
 - 1에 가까울수록 -> 정보를 기억함.

LSTM – 셀 상태 구하기



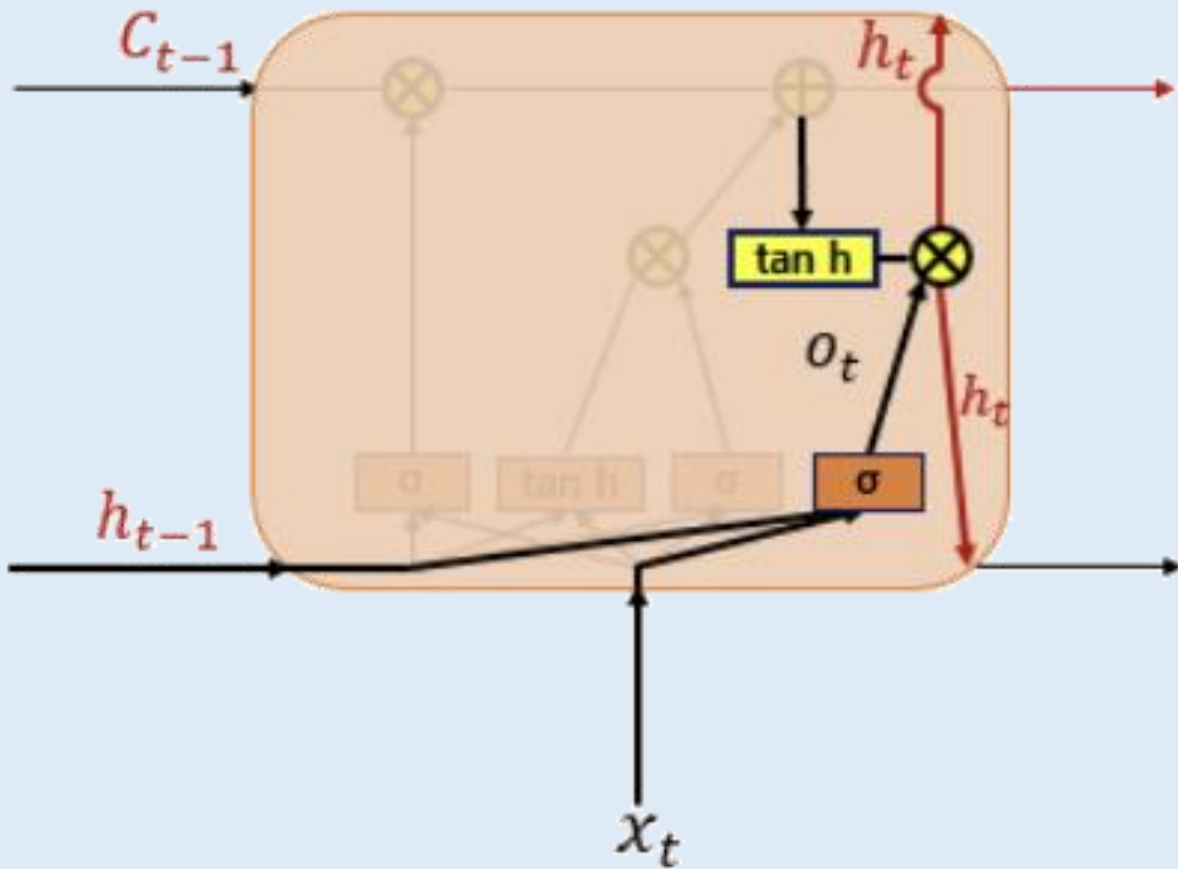
$$i_t \odot g_t$$

- 입력 게이트에서 구한 두 값에 대한 원소별 곱
- 기억할 값

$$C_t = f_t \odot C_{t-1} + i_t \odot g_t$$

- 입력 게이트에서 선택된 기억을 삭제 게이트의 결과값에 더함.
- 현재 시점 t 의 셀 상태
- 다음 $t+1$ 시점의 LSTM 셀로 넘겨짐.

LSTM - 출력 게이트 / 은닉 상태 구하기



- o_t :

현재 시점 t 의 x 값과 이전 시점 $t-1$ 의 은닉 상태가 시그모이드 함수를 지난 값

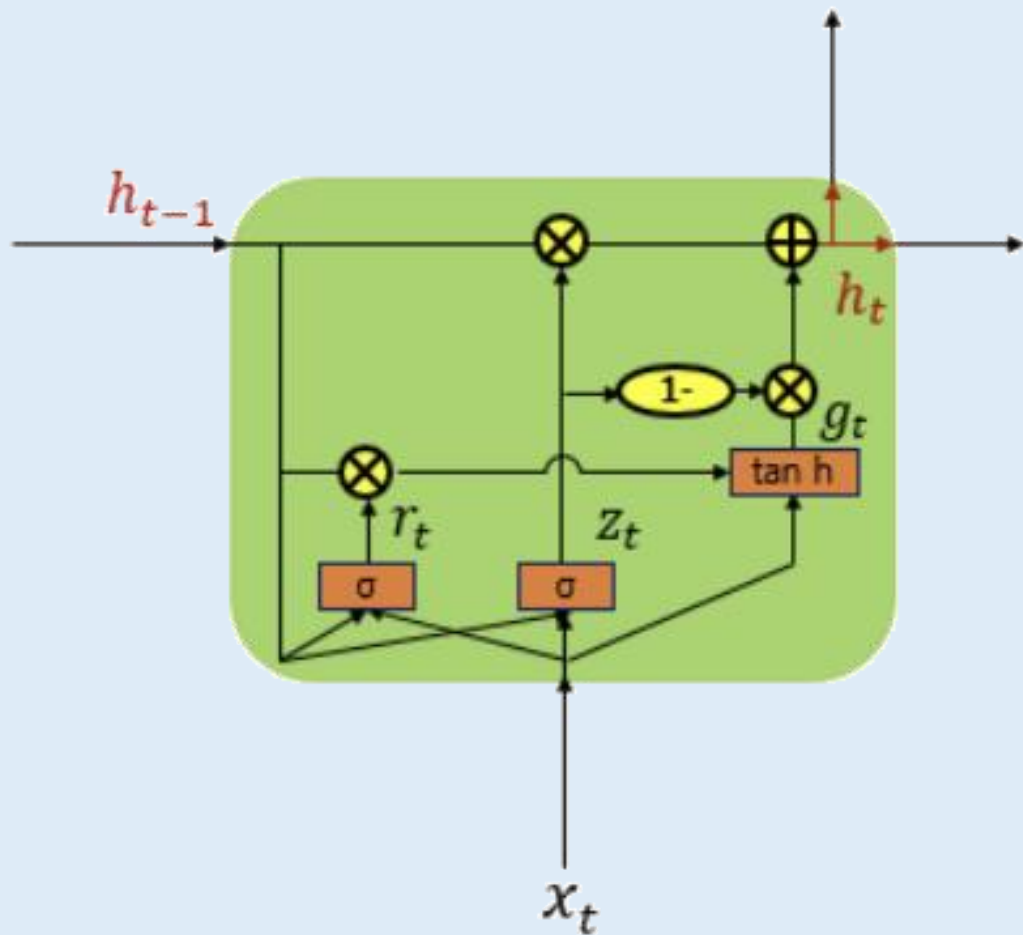
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$

현재 시점 t 의 은닉 상태를 결정하는 데 쓰임.

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

- h_t : 셀 상태의 값이 \tanh 함수를 지나 -1과 1 사이의 값이 되고, 해당 값은 출력 게이트의 값과 연산되면서, 값이 걸러지는 효과가 발생하여 은닉상태가 됨.

게이트 순환 유닛 (Gated Recurrent Unit, GRU)



GRU란?

- LSTM의 장기 의존성 문제에 대한 해결책을 유지하면서, 은닉 상태를 업데이트하는 계산을 줄인 것

LSTM과의 차이점

- 업데이트 게이트, 리셋 게이트 두 가지 게이트만 존재함.
- 가중치가 적어 계산량이 적음. (but 성능은 비슷)

$$r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r)$$

$$z_t = \sigma(W_{xz}x_t + W_{hz}h_{t-1} + b_z)$$

$$g_t = \tanh(W_{hg}(r_t \circ h_{t-1}) + W_{xg}x_t + b_g)$$

$$h_t = (1 - z_t) \circ g_t + z_t \circ h_{t-1}$$

NLP에서의 문자 -> 숫자 변형 방법

원-핫 인코딩 (One-Hot Encoding)

1st) 단어 집합 만들기

단어 집합 : 텍스트의 모든 단어를 중복 허용 없이 모아 놓은 것

2nd) 단어 집합에 고유한 정수 부여 (정수 인코딩)

각 단어에 고유한 정수 인덱스를 부여함.

3rd) 숫자로 바뀐 단어들을 벡터로 다루기

- 표현하고 싶은 단어의 고유한 정수를 인덱스로 간주하고 해당 위치에 1을 부여하고, 다른 단어의 인덱스의 위치에는 0을 부여함.
- 벡터 또는 행렬의 값이 대부분이 0으로 표현되는 방법
- 원 - 핫 벡터는 희소 벡터임.
- 문제점 : 단어의 개수가 늘어나면 벡터의 차원이 한없이 커짐, 공간적 낭비, 단어의 의미 표현 불가

희소 표현
(Sparse Representation)

NLP에서의 문자 -> 숫자 변형 방법

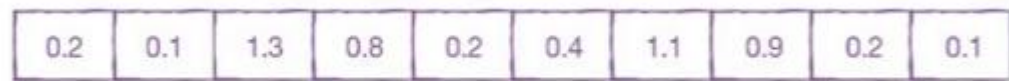
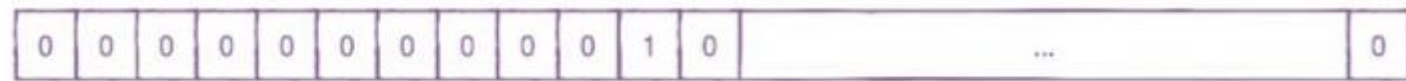
밀집 표현 (Dense Representation)

- 0과 1 x -> 실수 값 가짐.
- 벡터의 차원 : 단어 집합의 크기 x -> 사용자 설정값
- 벡터의 차원이 조밀해짐 -> 밀집 벡터

워드 임베딩 (Word Embedding)

단어를 밀집 벡터(dense vector)의 형태로 표현하는 방법

	원-핫 벡터	임베딩 벡터
차원	고차원(단어 집합의 크기)	저차원
다른 표현	희소 벡터의 일종	밀집 벡터의 일종
표현 방법	수동	훈련 데이터로부터 학습함
값의 타입	1과 0	실수



워드 임베딩 방법론 : LSA, Word2Vec, FastText, Glove 등

