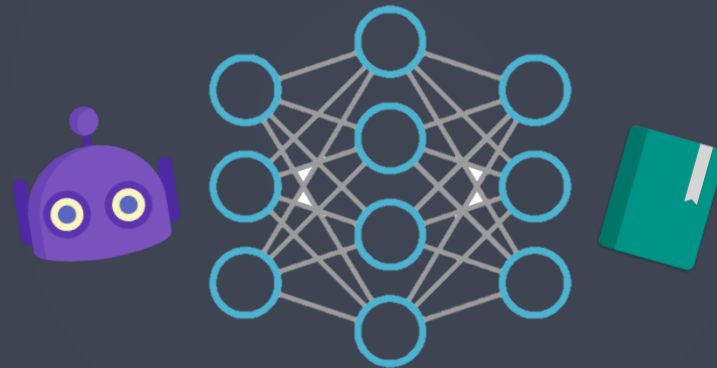


# Deep Learning

## Chapter 5 순환 신경망 (Recurrent Neural Network)



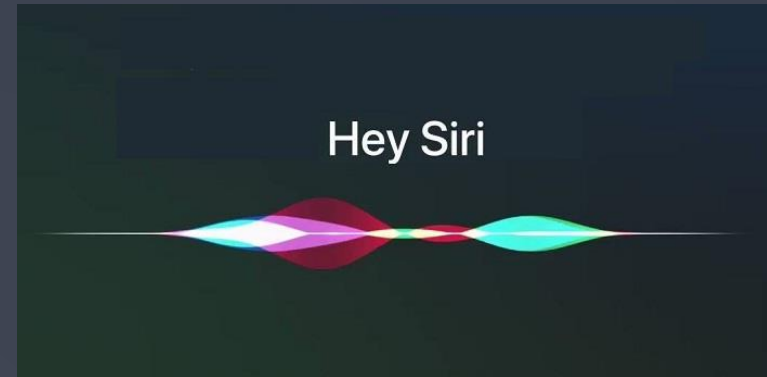
START

- 순환 신경망에 대해 알 수 있다.
- RNN의 활용분야를 알 수 있다.
- Keras를 활용해 순환 신경망을 구성 할 수 있다.

## 다음 문장을 완성하시오.

- 나는 오늘 짬뽕과 밥을 \_\_\_\_\_.
- 공부하려고 책을 \_\_\_\_\_.
- 친구가 나를 못생겼다고 \_\_\_\_\_.
- 그래서 나는 \_\_\_\_\_.

다음 단어를 쓰기위해서는 이전 단어를 기억하고 있어야한다.



- 문장을 듣고 무엇을 의미하는지 알아야 서비스 제공이 가능하다.
- 문장을 듣고 이해한다는 것은 많은 문장을 이미 학습해 놓았다는 것이다.
- 문장의 의미를 전달하려면 각 단어가 정해진 순서대로 입력되어야 한다.
- 과거에 입력된 데이터와 나중에 입력된 데이터 사이의 관계를 고려해야 하는 문제가 생긴다.
- 이를 해결하기 위해 **순환신경망(RNN)**이 고안되었다.

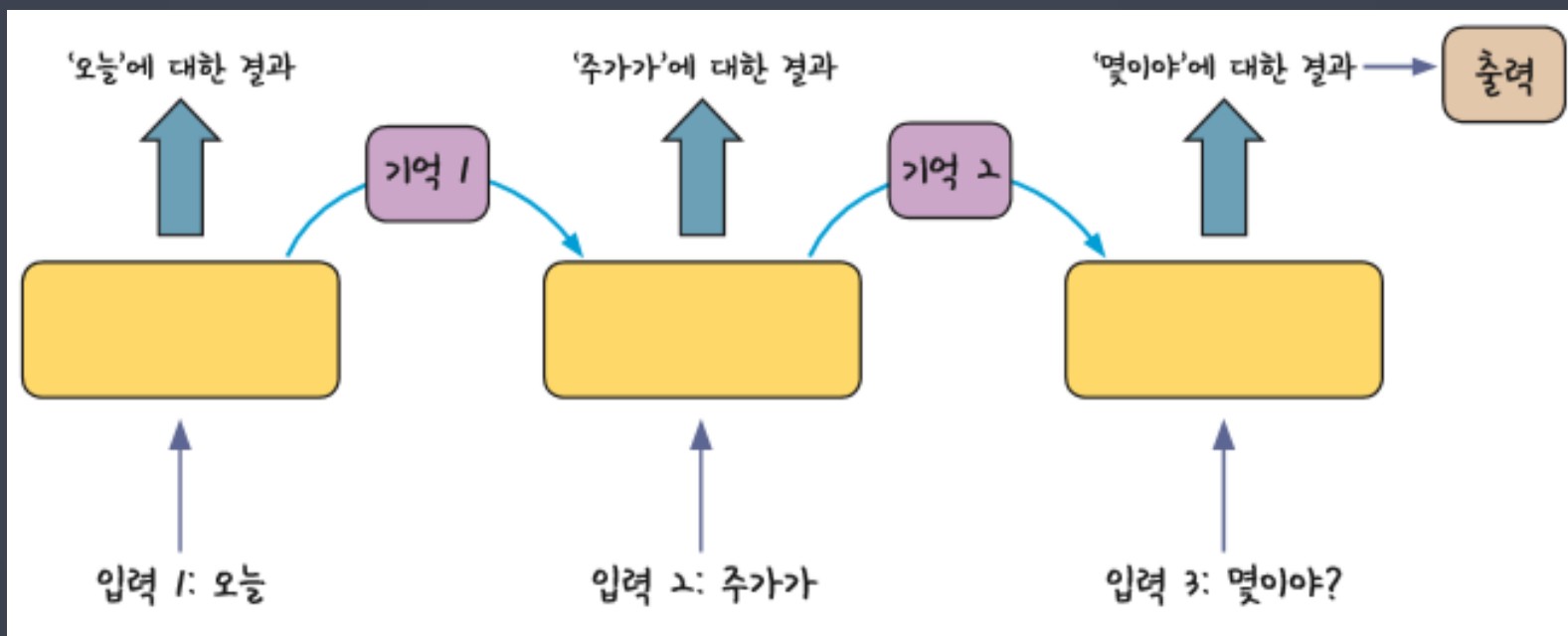


**R**ecurrent : 순환하는

**N**eural **N**etworks : 신경망

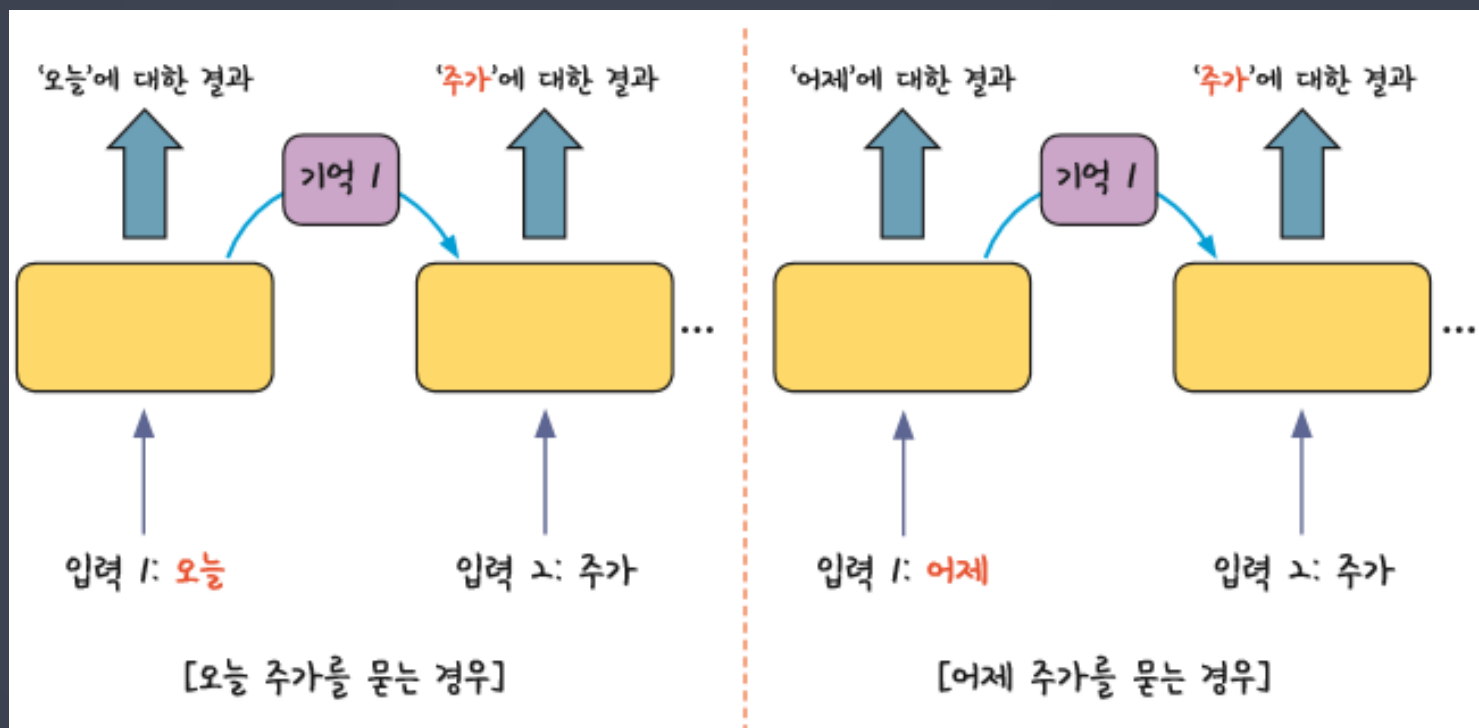
# 일반신경망과 순환신경망의 차이

- RNN은 여러 개의 데이터가 순서대로 입력되었을 때 앞서 입력 받은 데이터의 연산 결과를 잠시 기억해 놓는 방법이다.
- 기억된 데이터를 가지고 다음 데이터로 넘어가면서 함께 연산한다.

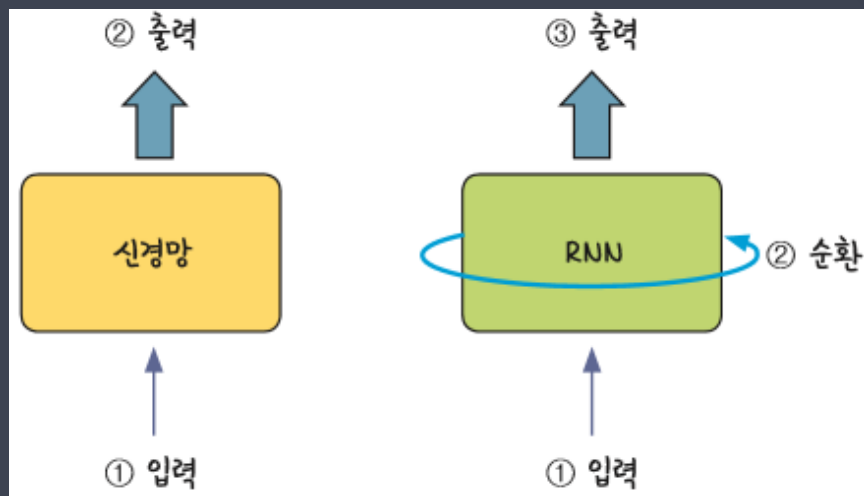


# 일반신경망과 순환신경망의 차이

- 앞에서 나온 입력에 대한 결과가 뒤에서 나오는 입력 값에 영향을 주는 것을 알 수 있다.
- 예를 들어 비슷한 두 문장이 입력되어도 앞에서 나온 입력 값을 구별하여 출력 값에 반영할 수 있다.







- 모든 입력 값에 이 작업을 순서대로 실행하므로 다음 층(layer)으로 넘어가기 전에 같은 층을 맴도는 것처럼 보인다.
- 같은 층 안에서 맴도는 성질때문에 순환 신경망이라고 부른다.

# Sequential Data (순차기반데이터)



Time series



Music



Sentence



Translation

분석에 사용되는 feature들이 시간적,순차적 특성을 지닌 데이터

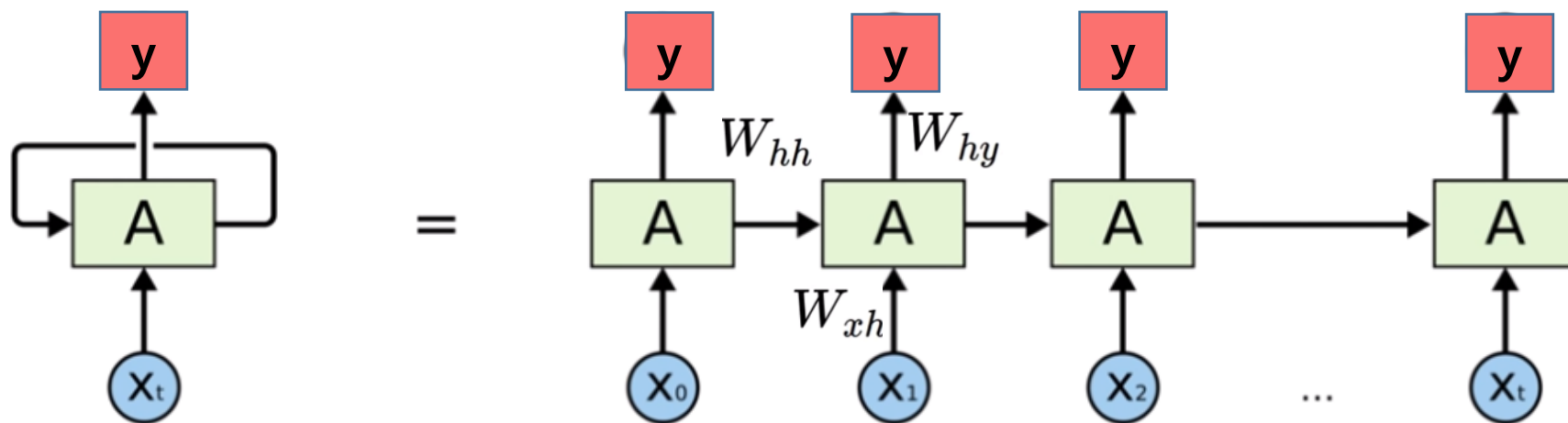
$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state

old state

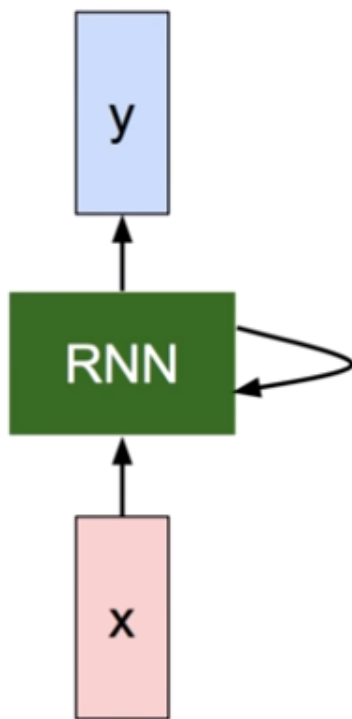
input vector at some time step

some function with parameters  $W$



# (Vanilla) Recurrent Neural Network

The state consists of a single “hidden” vector  $h$ :



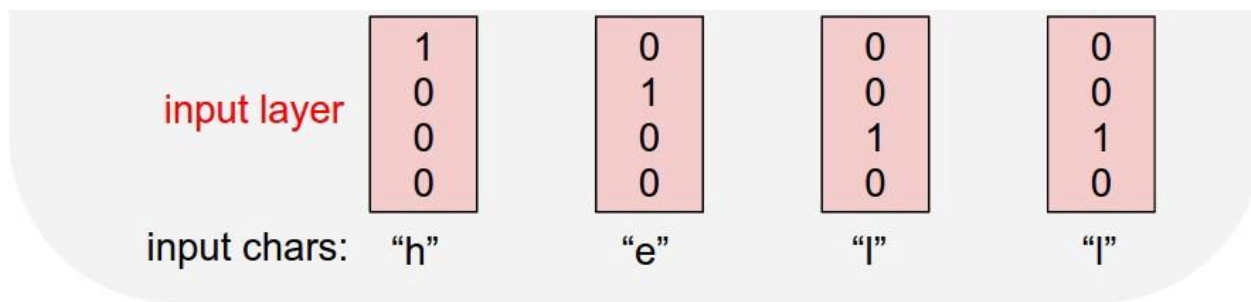
$$h_t = f_W(h_{t-1}, x_t)$$



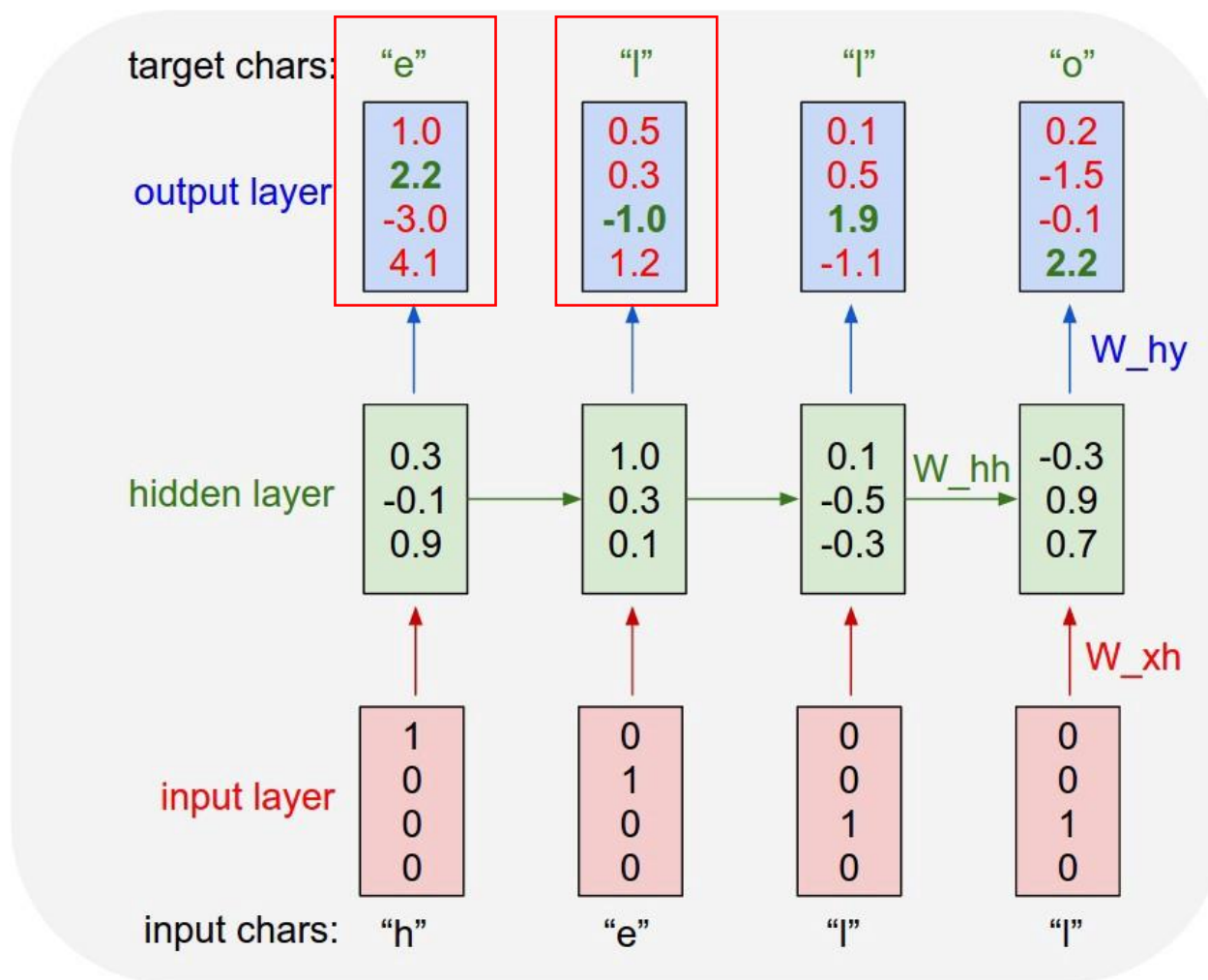
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

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



$$y_t = W_{hy}h_t$$



hello를 SimpleRNN 신경망으로 학습하기

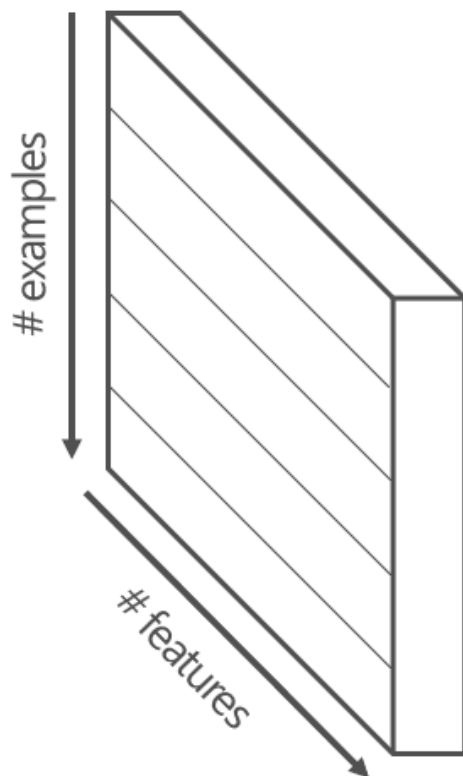
SimpleRNN (**units** = 3,  
**input\_shape** = (4, 4))

- units : 퍼셉트론(뉴런의 개수)
- input\_shape : (time steps, features) 형태의 튜플로 들어간다.

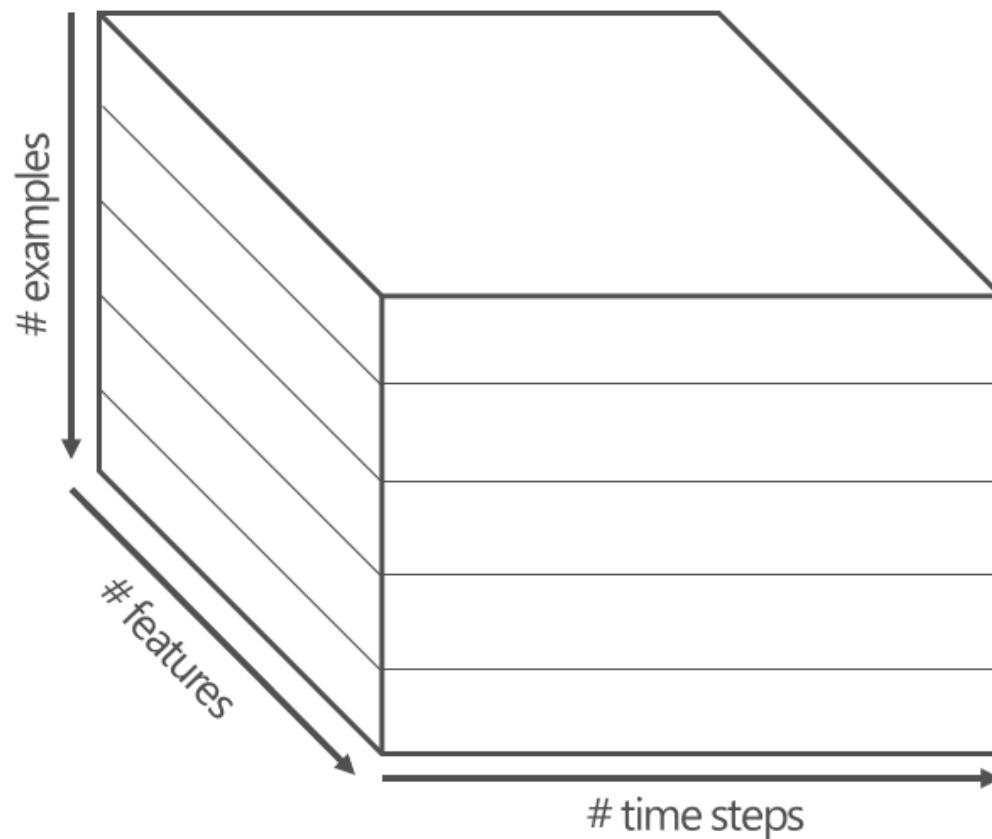


## RNN 데이터 구조

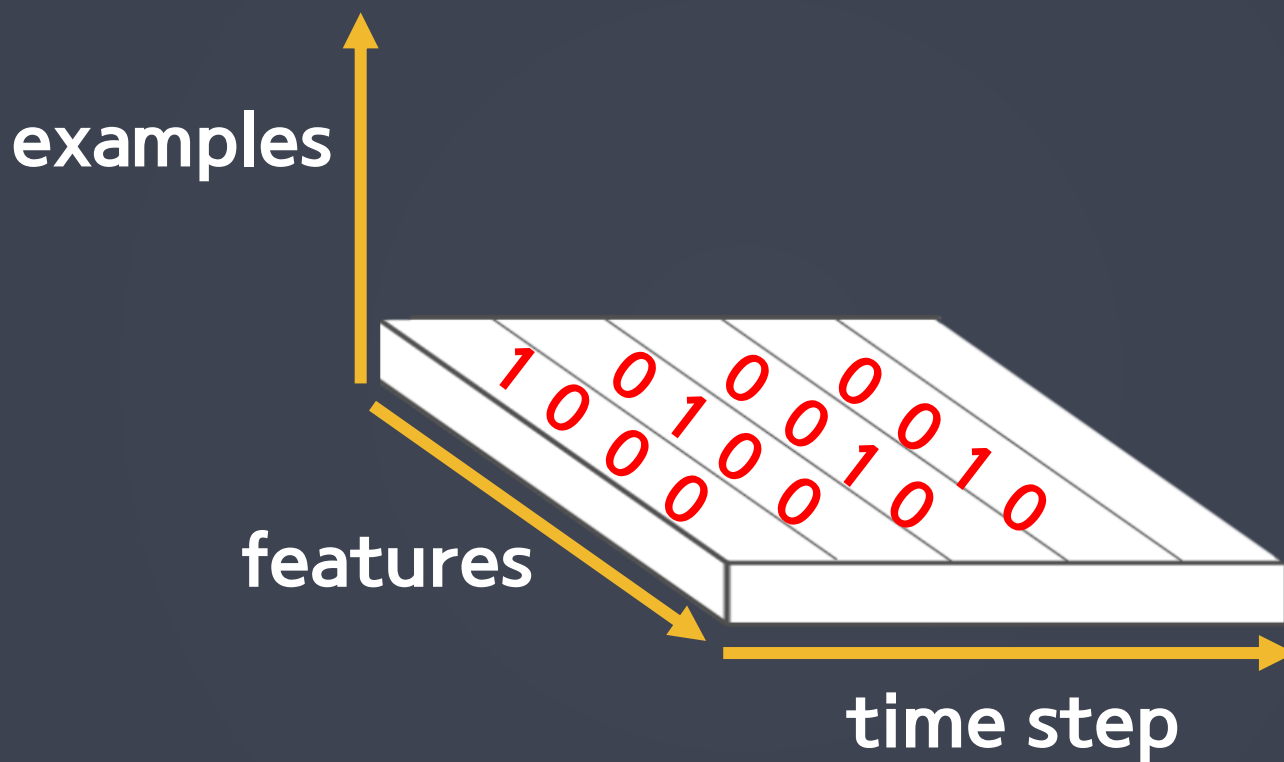
Feed Forward Network Data (2D)



Recurrent Network Data (3D)

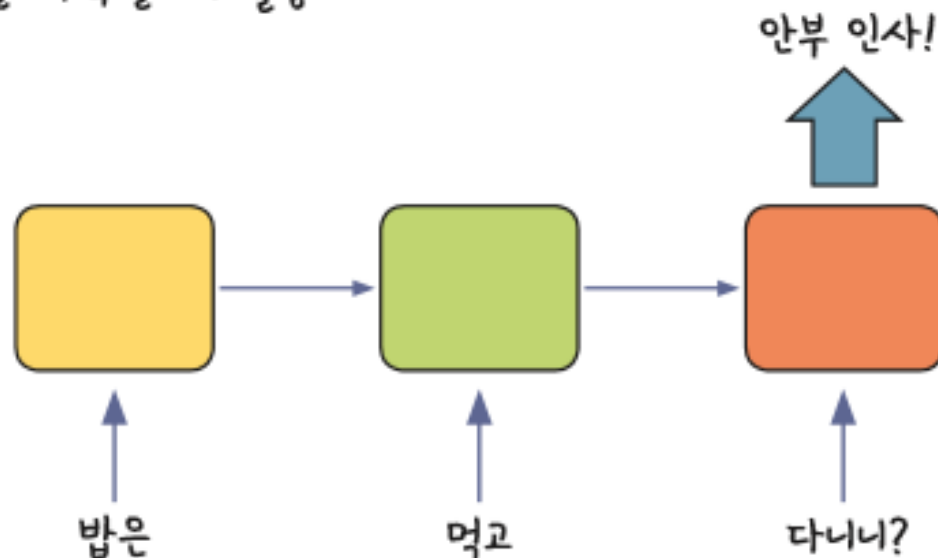


## hello를 one-hot-encoding하기



## ① 다수 입력 단일 출력

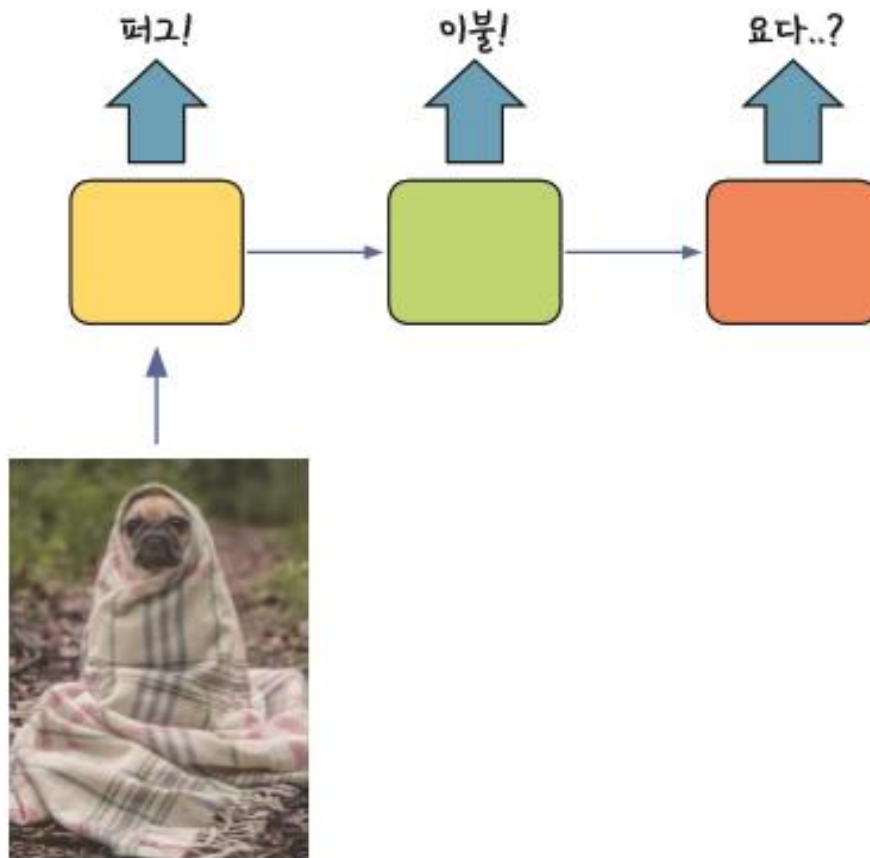
예: 문장을 읽고 뜻을 파악할 때 활용



```
model = Sequential()  
model.add(SimpleRNN(units = output_size,  
                    input_shape=(timesteps, features)))
```

## ② 단일 입력 다수 출력

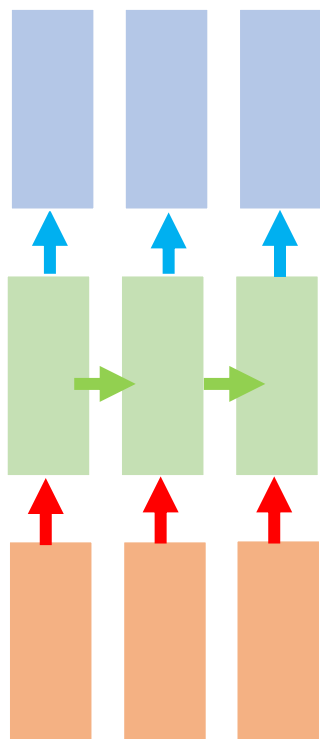
예: 사진의 캡션을 만들 때 활용



```
model = Sequential()
```

```
model.add(RepeatVector(number_of_times,  
                        input_shape=input_shape))
```

```
model.add(SimpleRNN(units = output_size,  
                    return_sequences=True))
```



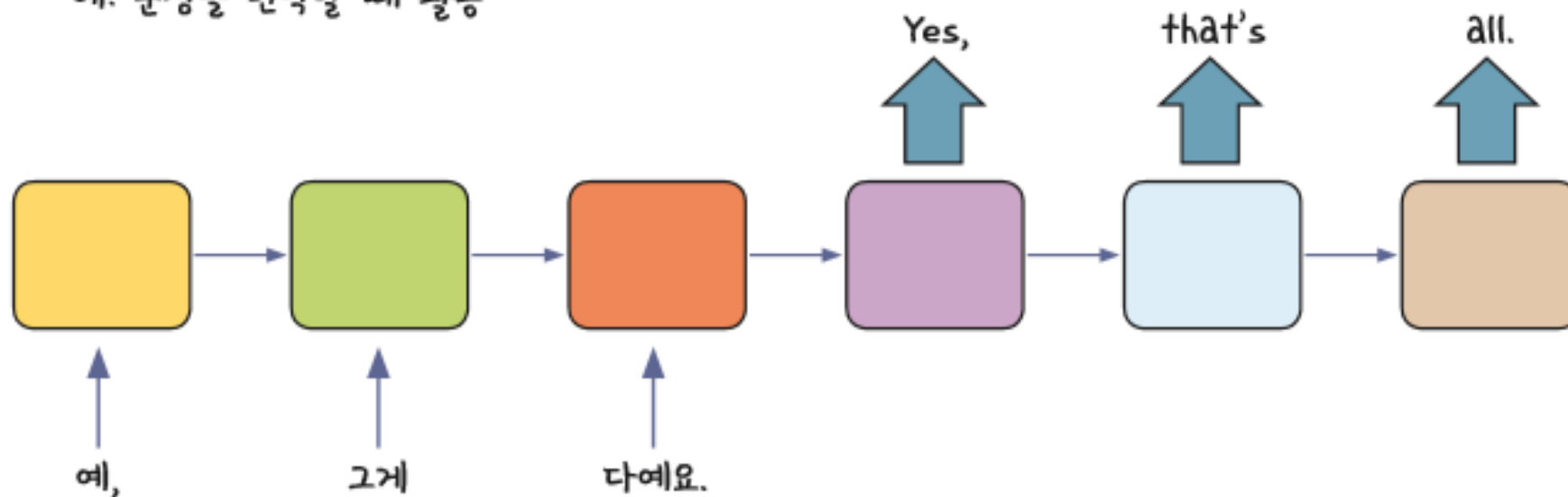
**Video에서 Frame단위 Classfication**

```
model = Sequential()  
model.add(SimpleRNN(units = output_size,  
                    input_shape=(timesteps, features),  
                    return_sequences=True))
```

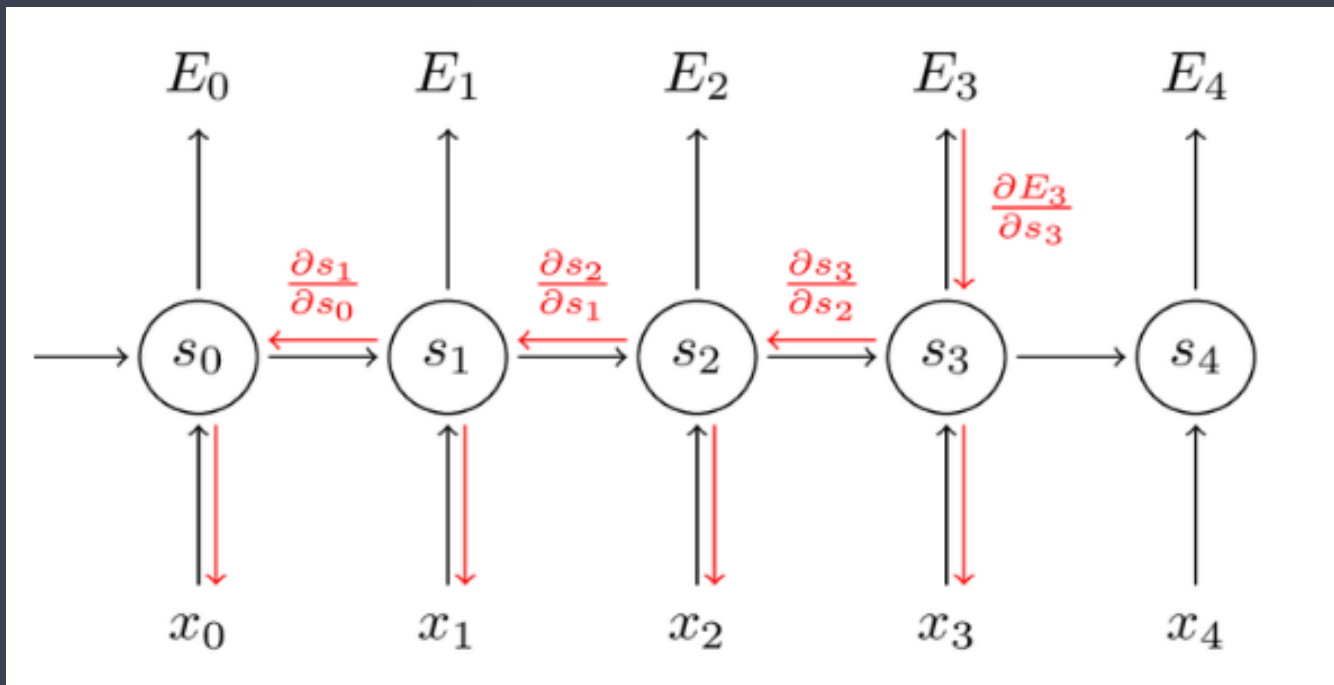


## ③ 다수 입력 다수 출력

예: 문장을 번역할 때 활용



## BPTT(Back Propagation Through Time)



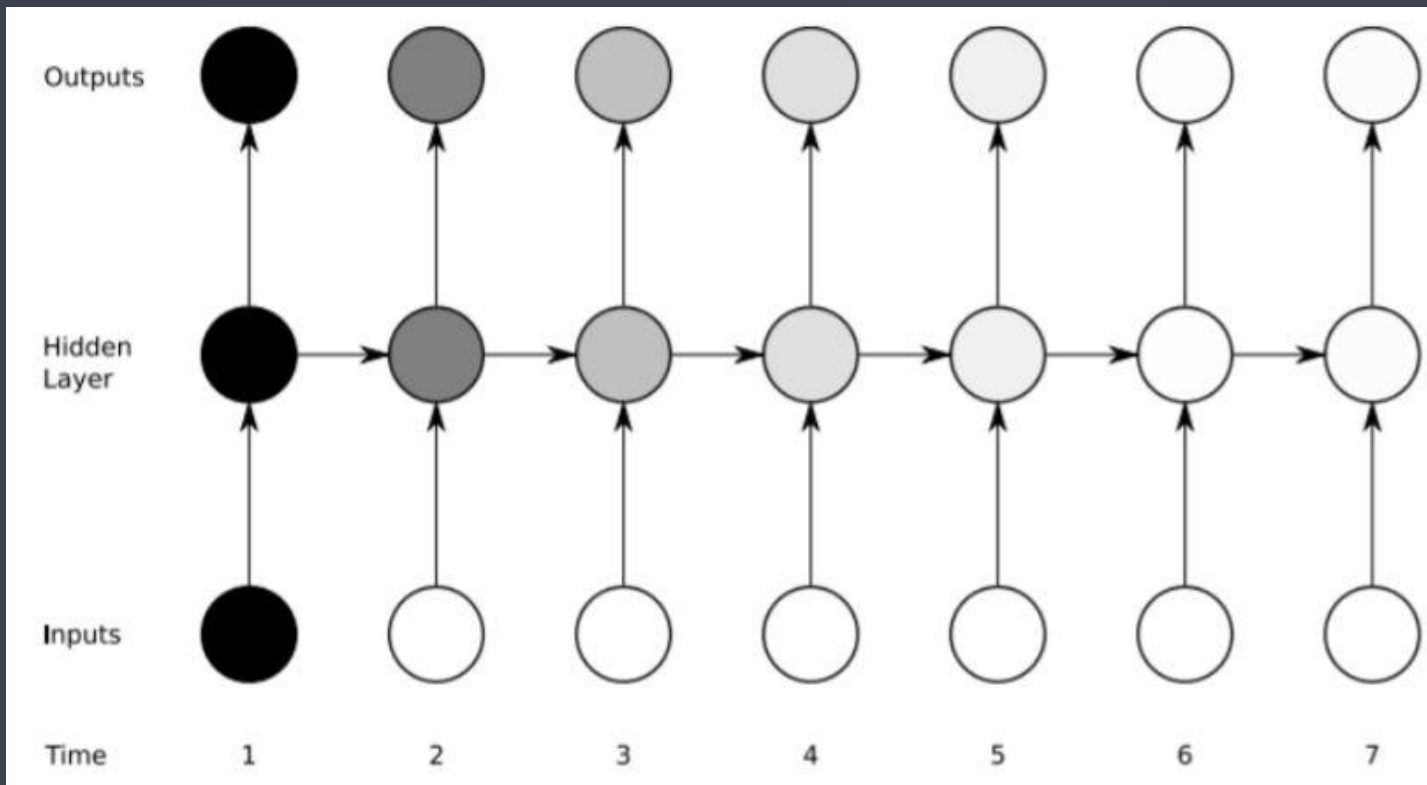
RNN 에서 사용되는  
BPTT 알고리즘은

- 계산 그래프에서 화살표 역방향으로
- 출력의 오차 정보를 가진 미분값 (gradient)을 이전 상태에 전달하여
- 순차적으로 신경망의 중간에 있는 매개변수를 갱신하는 절차입니다.

RNN에서 사용되는 오차역전파(BPTT)

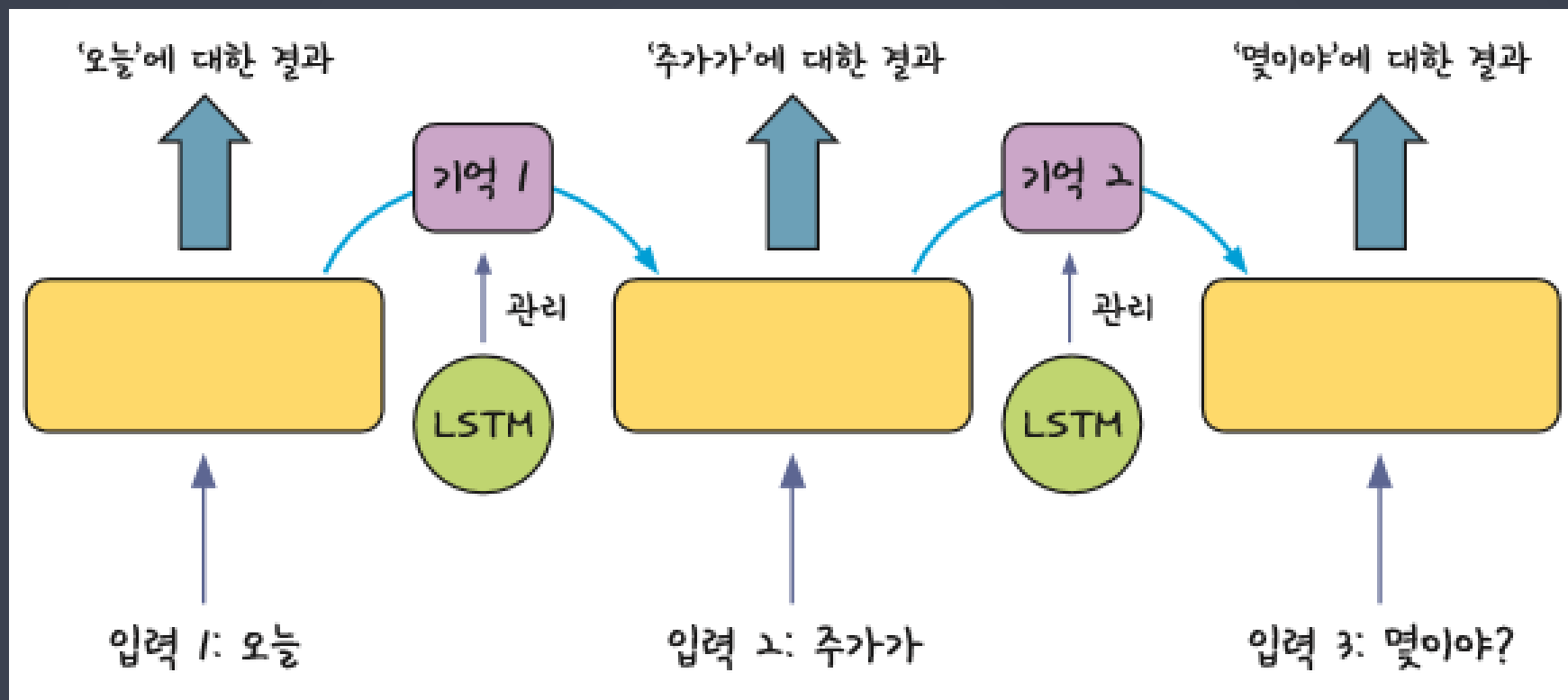
한쪽 ~~어느~~ 보석 ~~같은~~ ~~어려운~~ 보석 ~~같은~~ 아름다운 별들이 있었다.

# 소실 문제 (Vanishing Gradient)



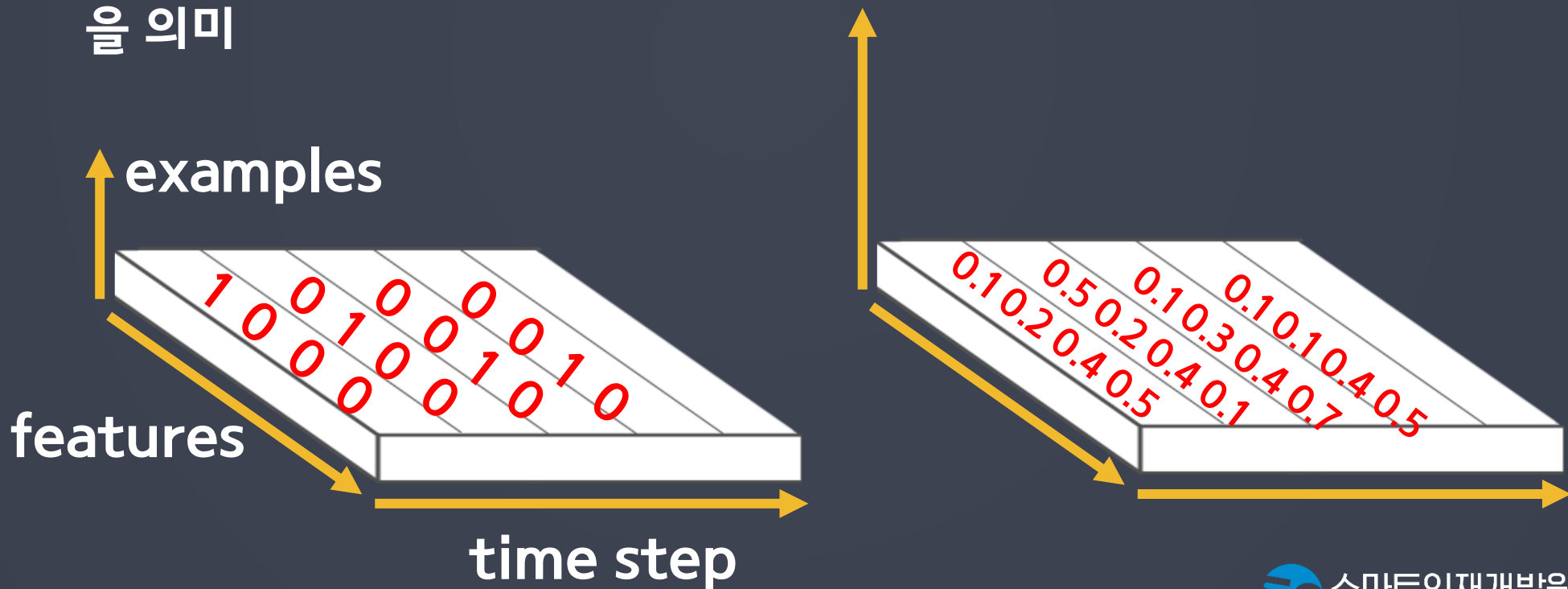
시간이 지나면  
이전의 입력값을  
잊어버리게 된다!

# LSTM(Long Short Term Memory)



## LSTM 로이터 뉴스 분류 + word Embedding

- 자연어를 컴퓨터가 이해하고, 효율적으로 처리하기 위해서는 컴퓨터가 이해할 수 있도록 변환 할 필요가 있다.
- 단어의 의미를 벡터화 하는 것을 워드 임베딩 이라 한다.
- 주로 희소 표현(one-hot-encoding)에서 밀집 표현으로 변환하는 것을 의미



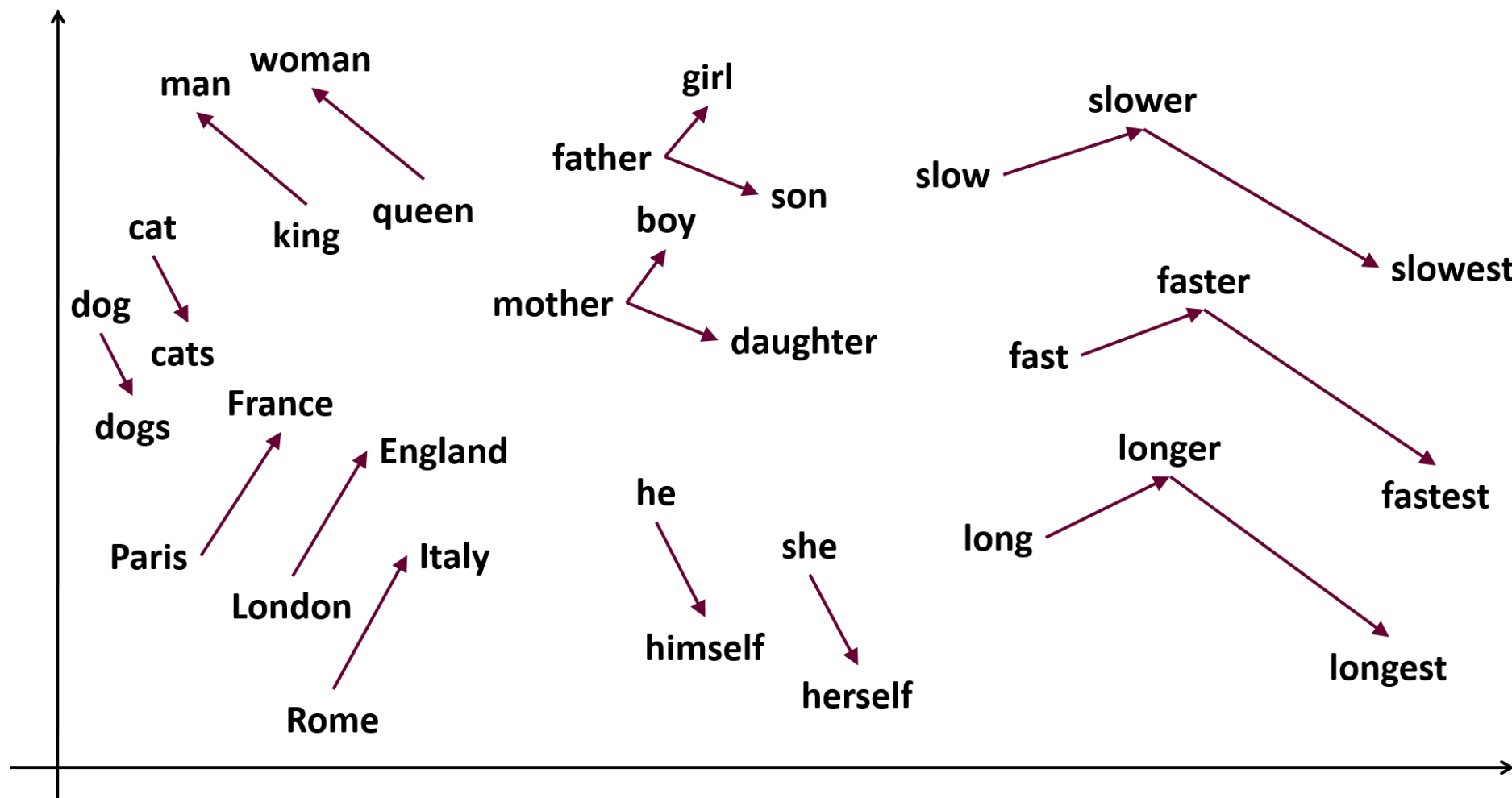
- 단어를 밀집 벡터의 형태로 표현하는 방법을 워드 임베딩(word embedding)이라고 한다.
- 임베딩 과정을 통해 나온 결과를 임베딩 벡터(embedding vector)라고 한다.
- 워드 임베딩 방법론으로는 Word2Vec, FastText, Glove 등이 있다.
- 케라스에도 제공하는 도구인 Embedding()은 위에서 사용하는 방법과 다르게 랜덤한 값을 가지는 밀집 벡터로 변환한 뒤에, 인공 신경망의 가중치를 학습한다.

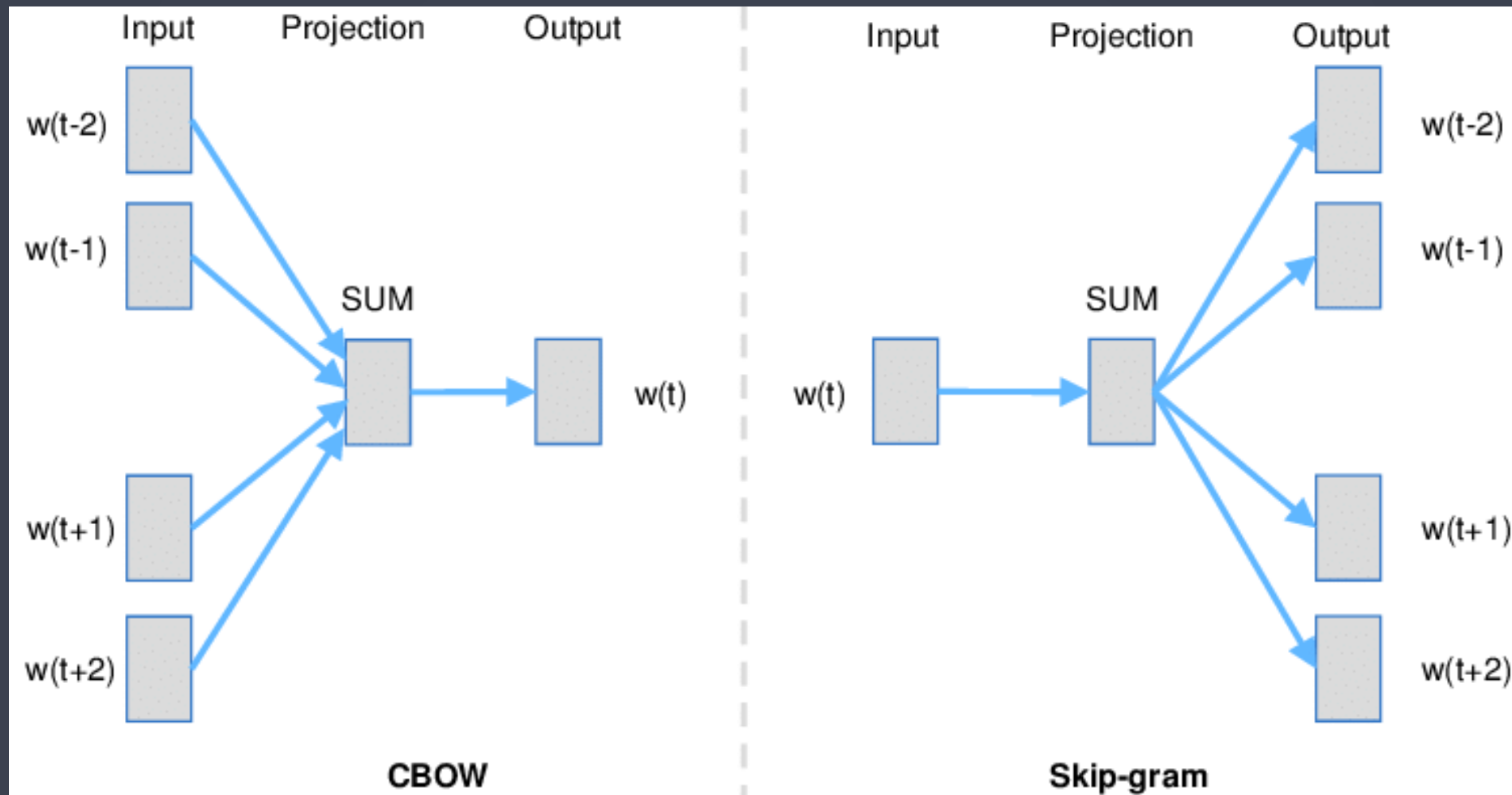


-	원-핫 벡터	임베딩 벡터
차원	고차원(단어 집합의 크기)	저차원
다른 표현	희소 벡터의 일종	밀집 벡터의 일종
표현 방법	수동	훈련 데이터로부터 학습함
값의 타입	1과 0	실수

## CNN + LSTM IMDB 분류

'비슷한 위치에서 등장하는 단어들은 비슷한 의미를 가진다'





## 에너지 사용량 예측 모델링