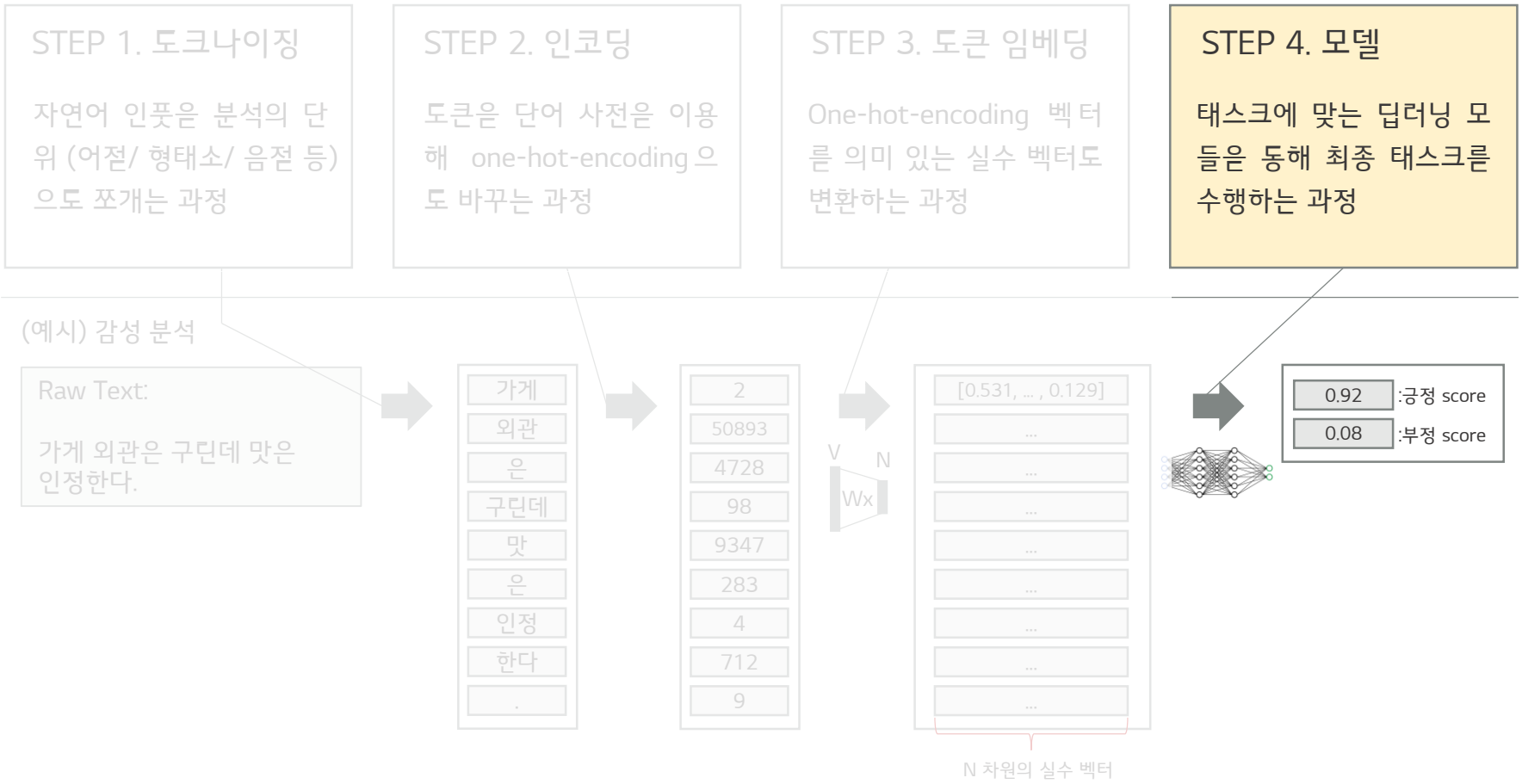
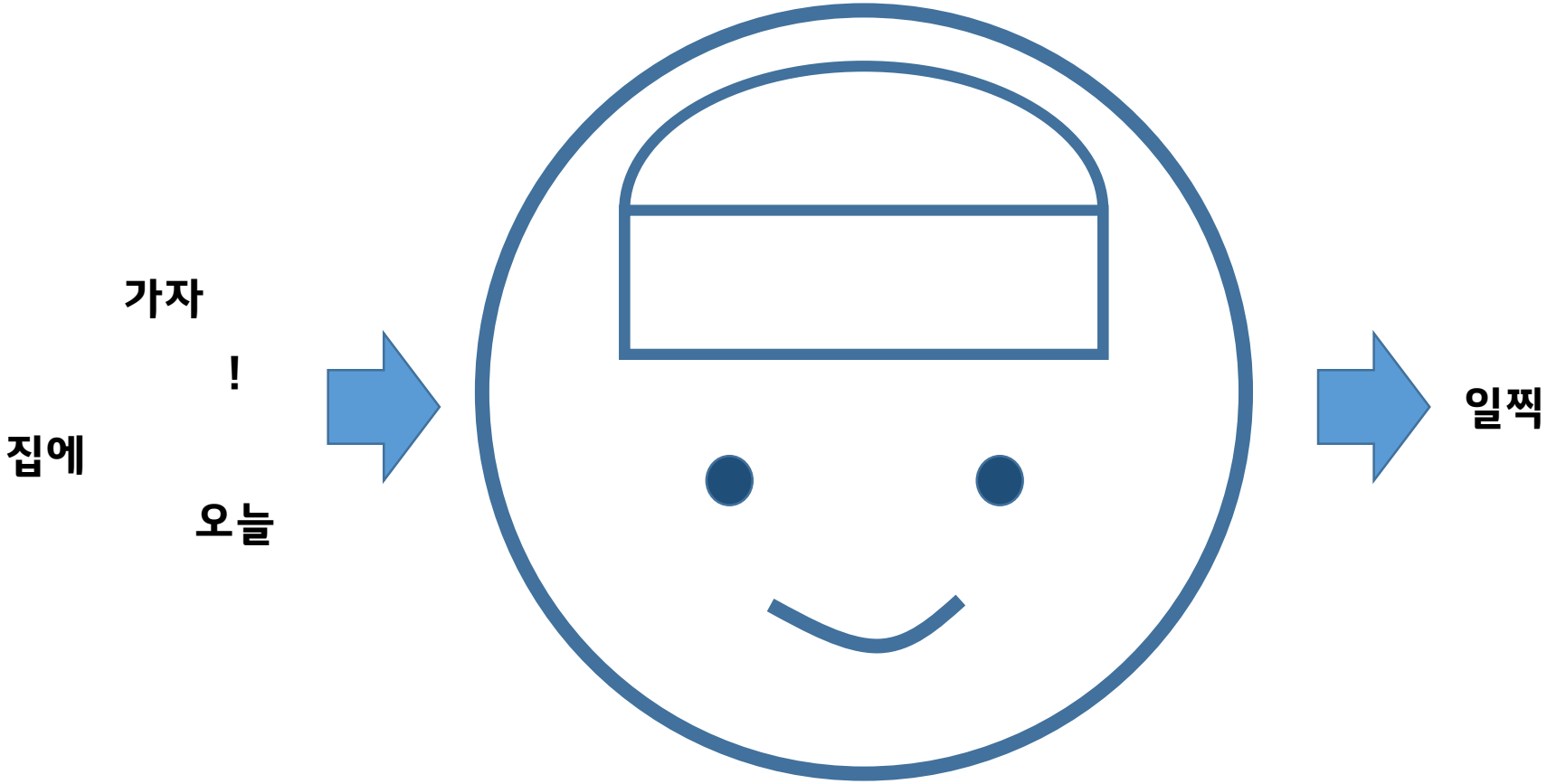
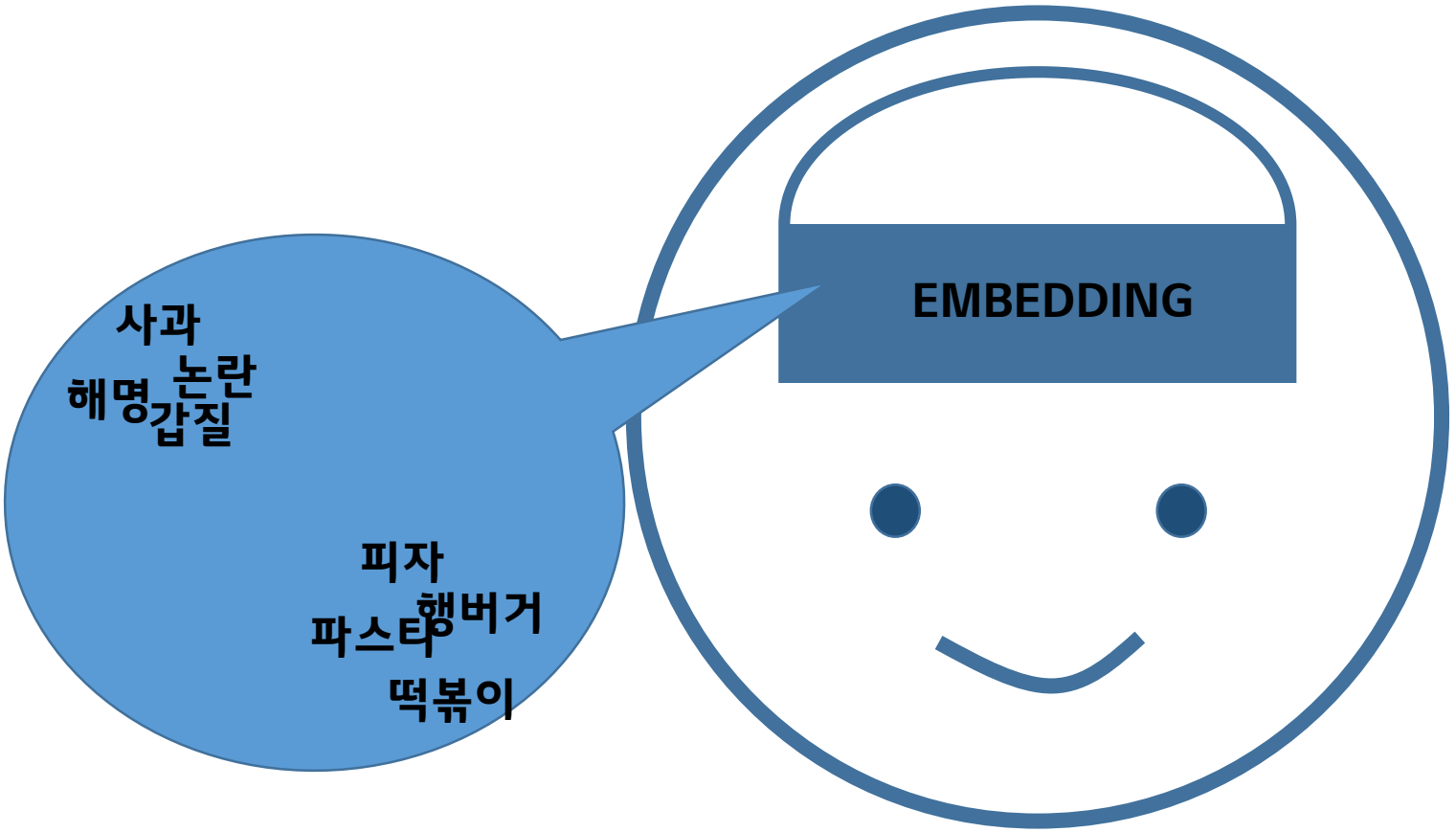
M4. RNN-basic


STEP 1. 도크나이징
자연어 인풋은 분석의 단위 (어절/ 형태소/ 음절 등)으로 쪼개는 과정
STEP 2. 인코딩
도큰은 단어 사전을 이용해 one-hot-encoding으로 바꾸는 과정
STEP 3. 도큰 임베딩
One-hot-encoding 벡터를 의미 있는 실수 벡터로 변환하는 과정
STEP 4. 모델
태스크에 맞는 딥러닝 모들은 동해 최종 태스크를 수행하는 과정
(예시) 감성 분석
Raw Text:
가게 외관은 구딘데 맛은 인정한다.
가게
외관
은
구딘데
맛
은
인정
한다
.
2
50893
4728
98
9347
283
4
712
9
V
N
Wx
[0.531, ... , 0.129]
...
...
...
...
...
...
...
...
N 차원의 실수 벡터
0.92
:긍정 score
0.08
:부정 score

토큰 임베딩 : CBOW/SKIPGRAM

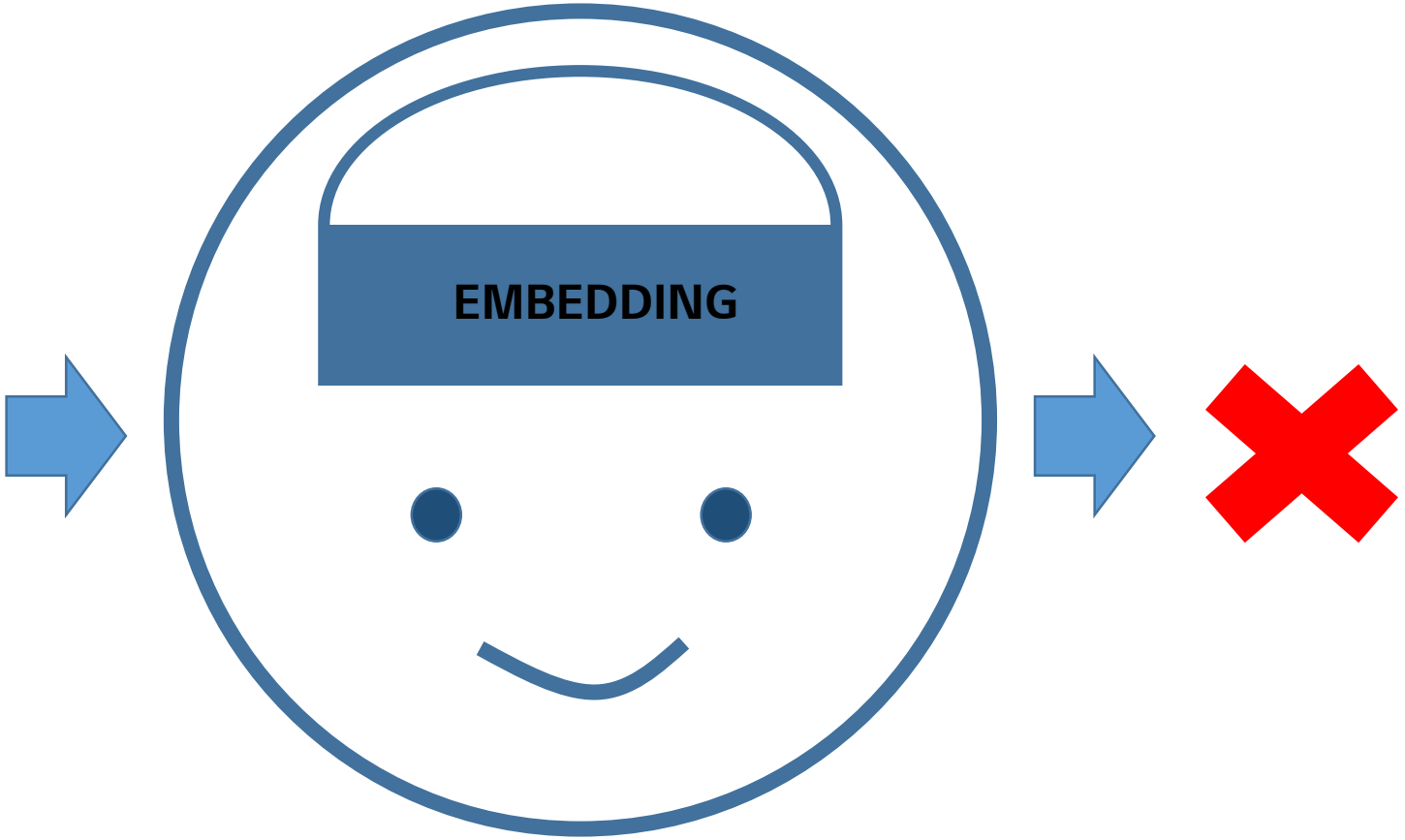


토큰 임베딩 : CBOW/SKIPGRAM

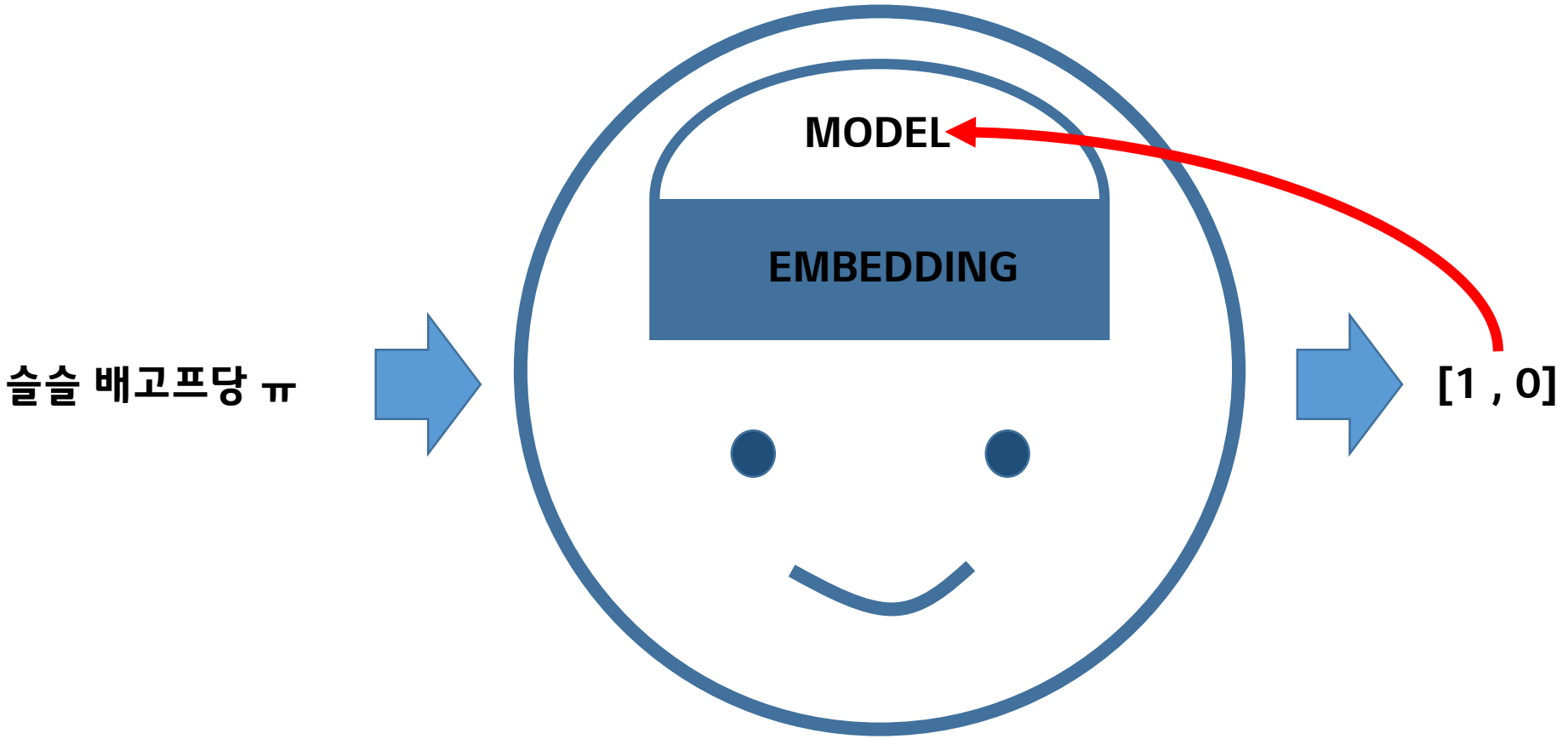


MODELING

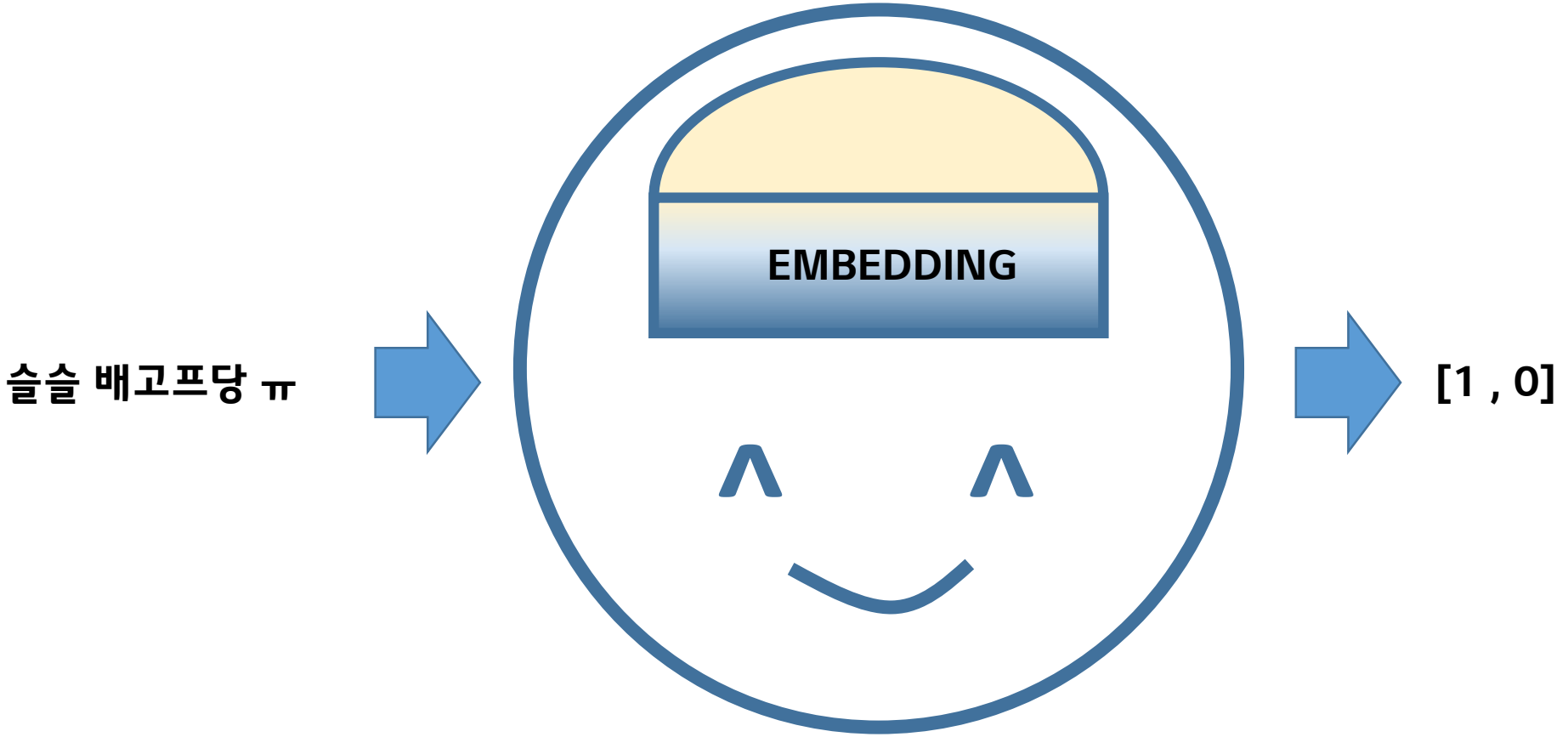
슬슬 배고프당 ㅠ



MODELING



MODELING



MODELING

RNN

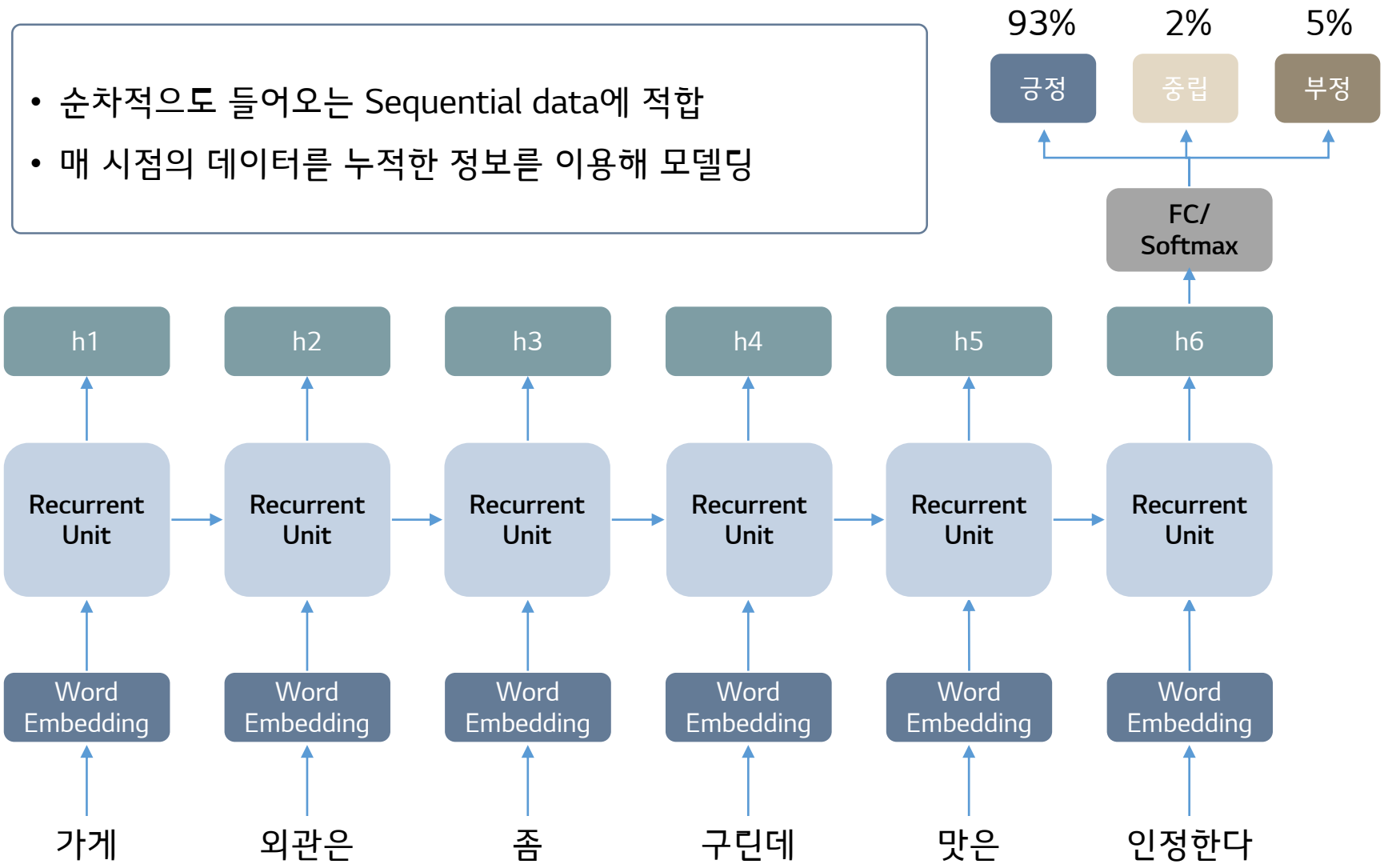
LSTM

GRU

토큰의 시퀀스로 이루어져 있는 자연어 문장에서
문맥적인 의미를 담은 representation(hidden state)을 만드는 알고리즘

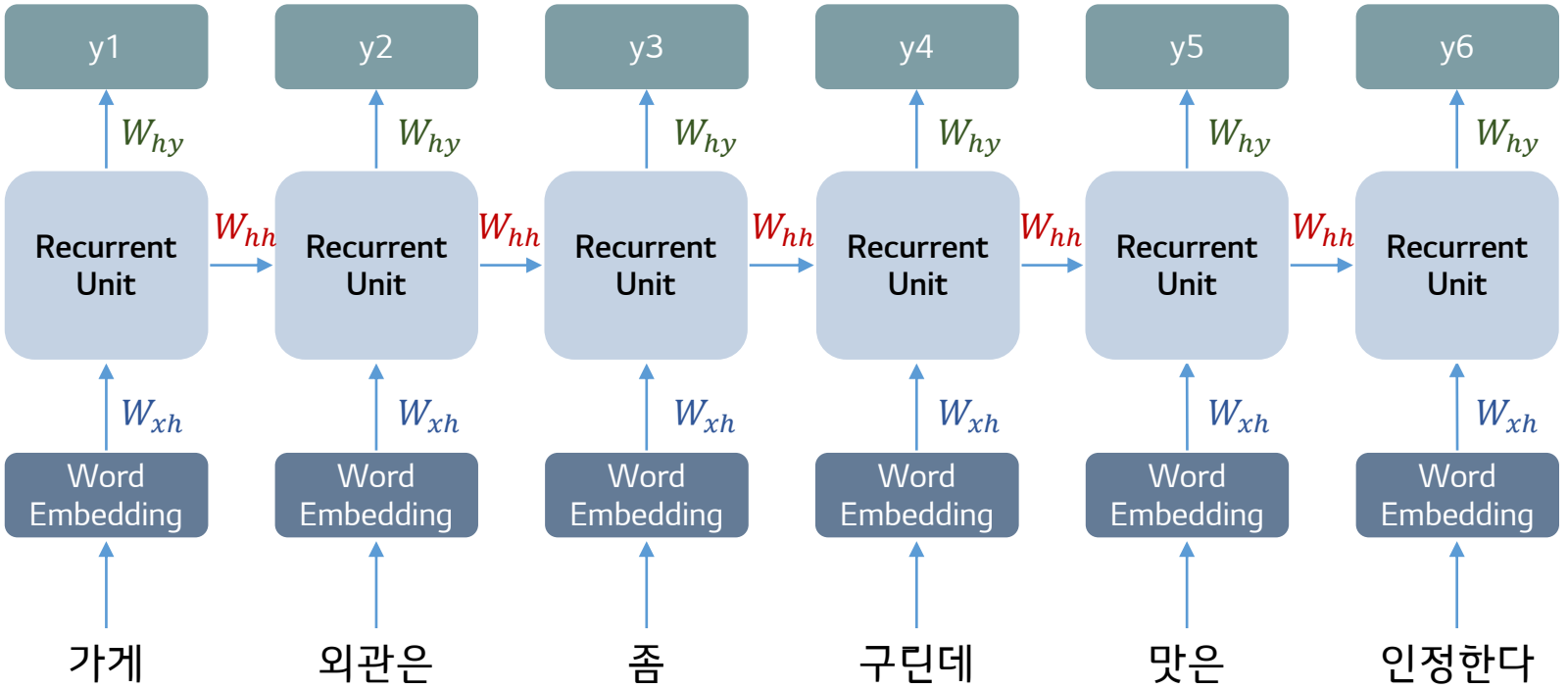
RNN(Recurrent Neural Network)

- 순차적으로 들어오는 Sequential data에 적합
- 매 시점의 데이터를 누적한 정보를 이용해 모델링

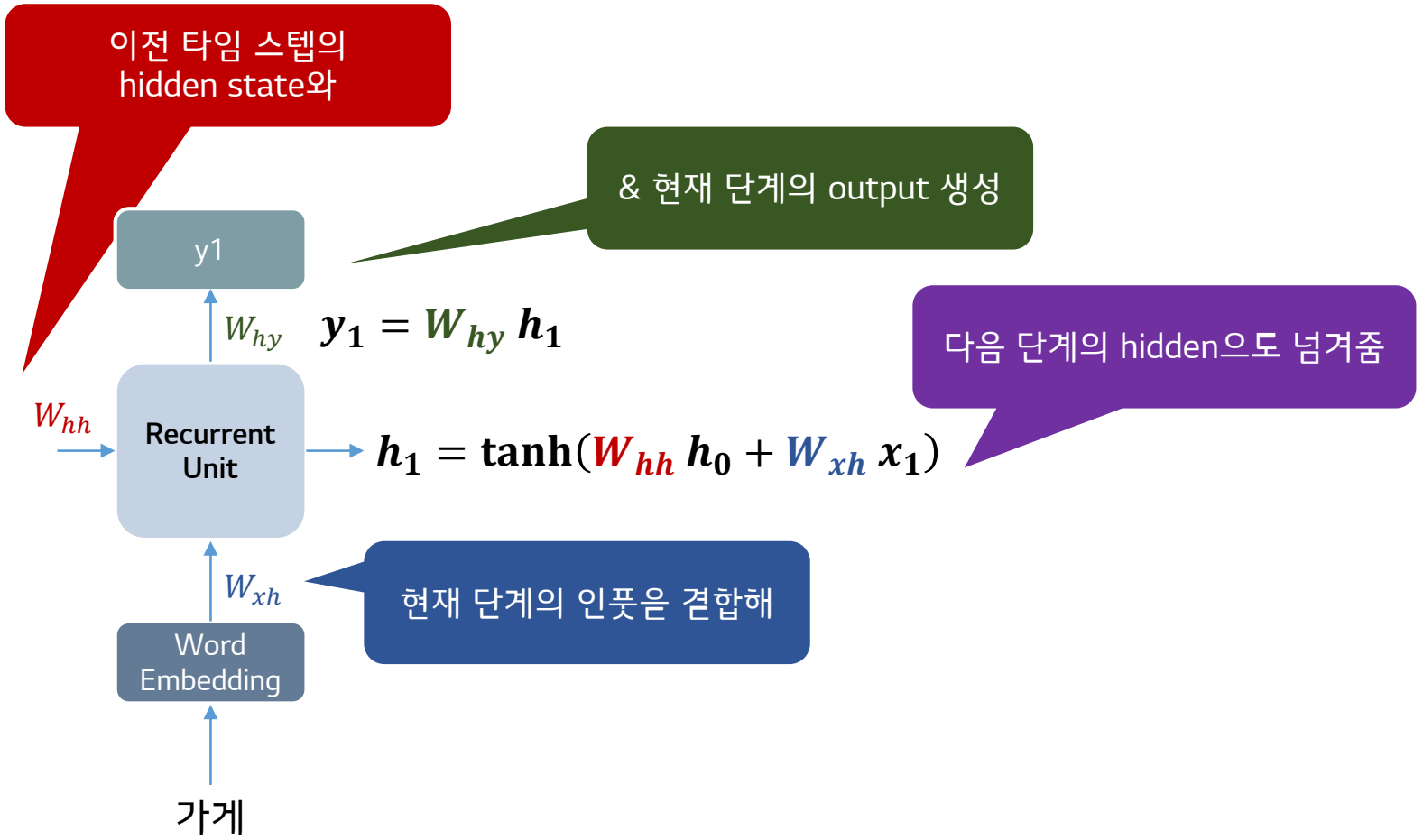


RNN(Recurrent Neural Network)

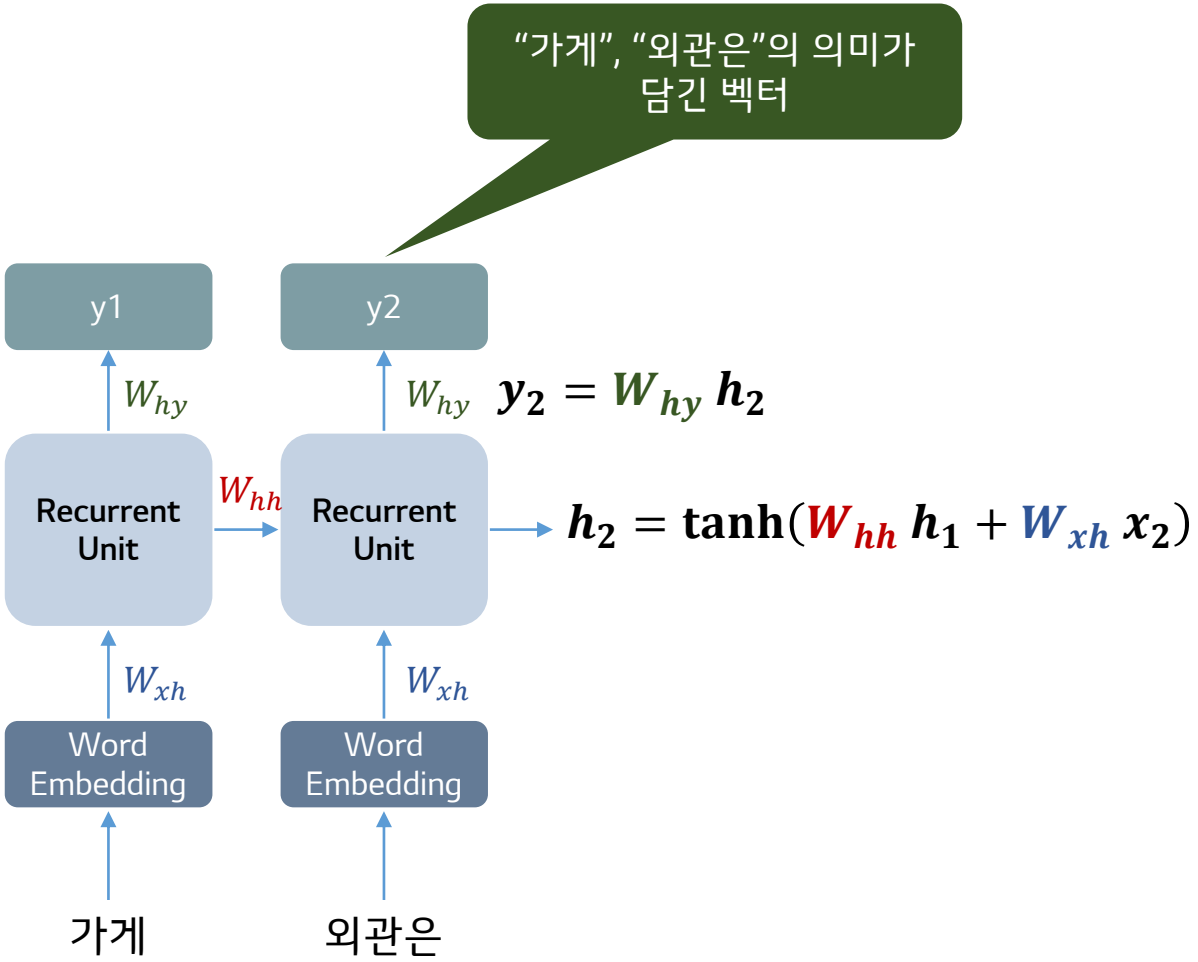
$$h_t = \tanh(W_{hh}\, h_{t-1} + W_{xh}\, x_t)$$
$$y_t = W_{hy}\, h_t$$



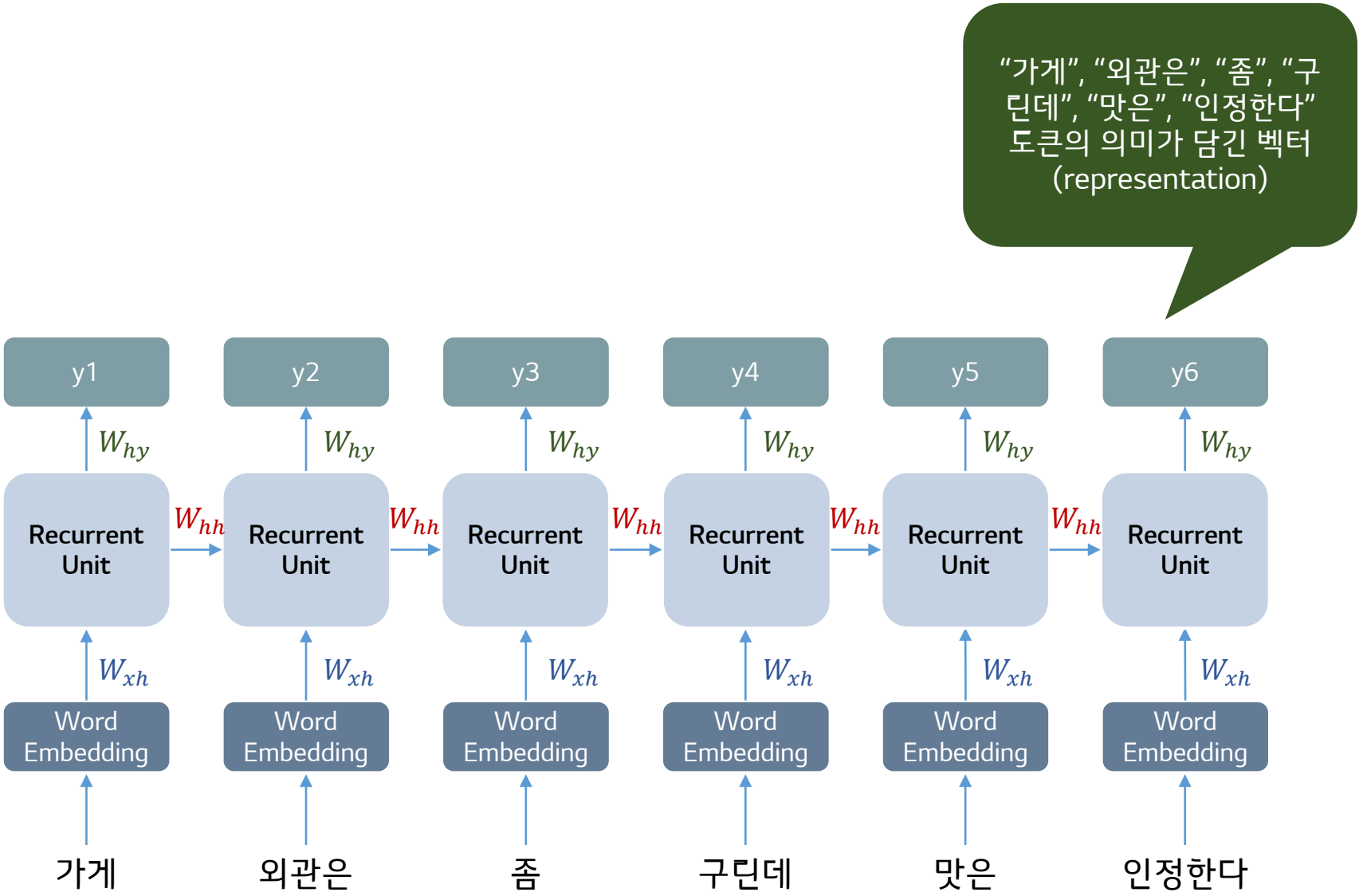
RNN(Recurrent Neural Network)



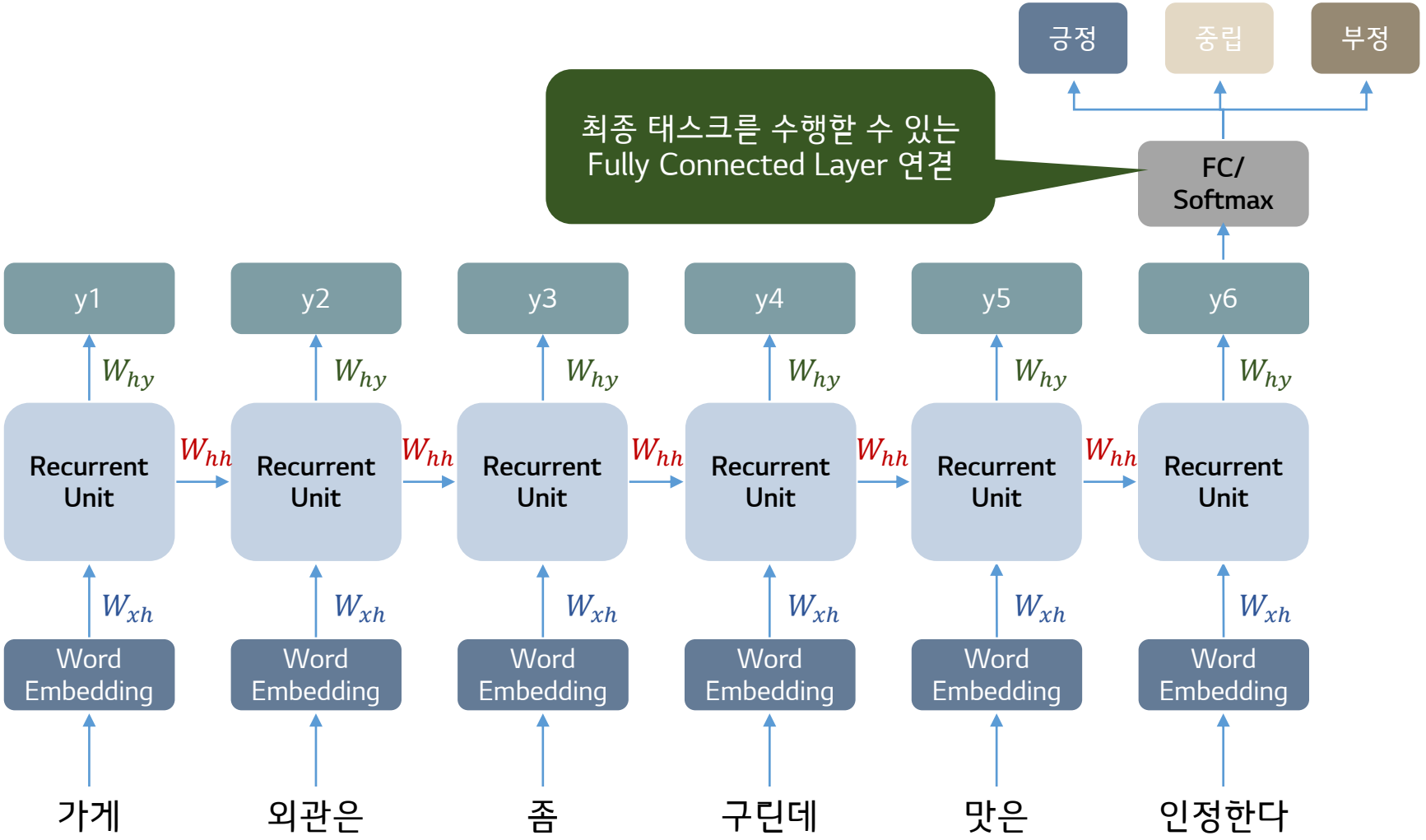
RNN(Recurrent Neural Network)



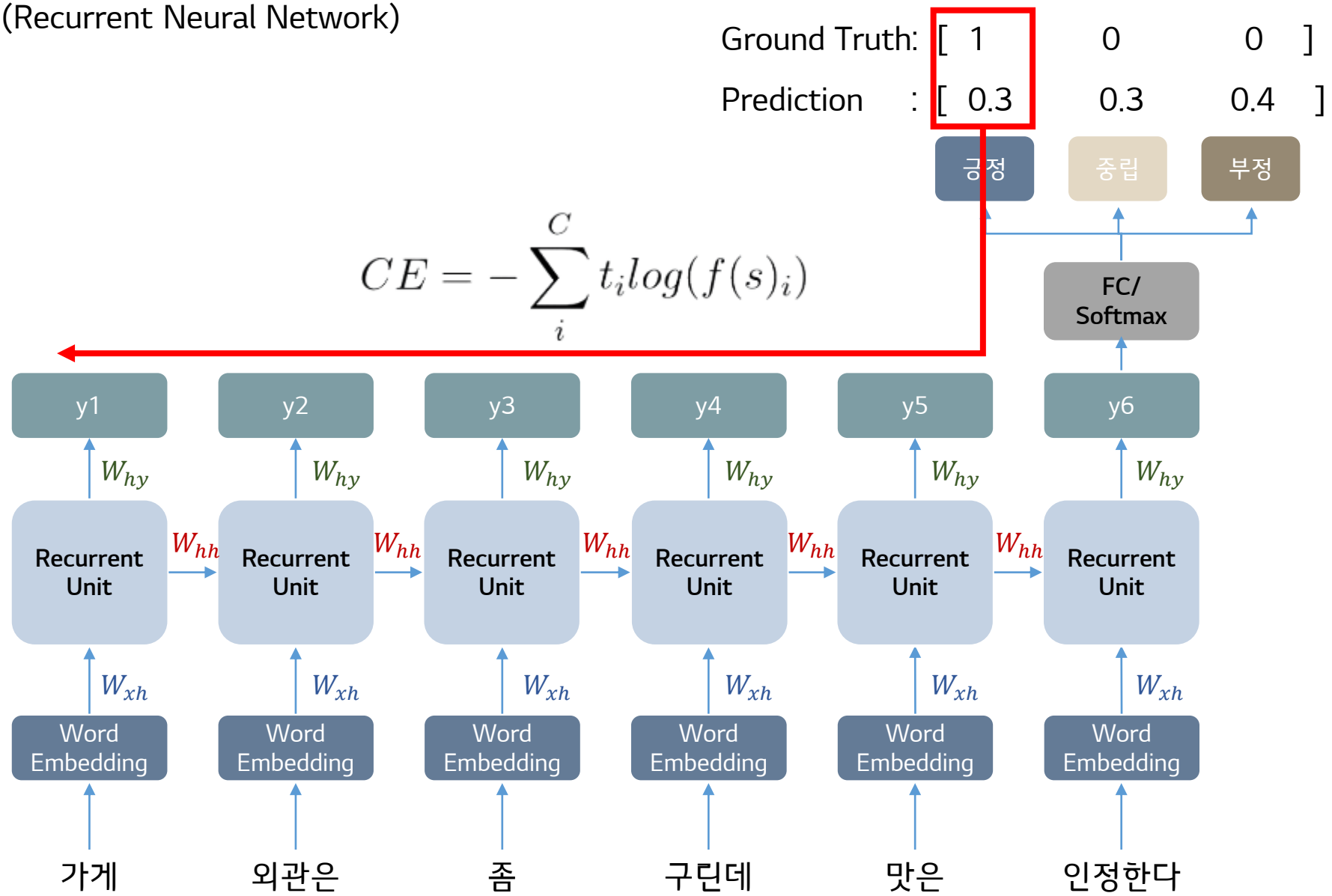
RNN(Recurrent Neural Network)



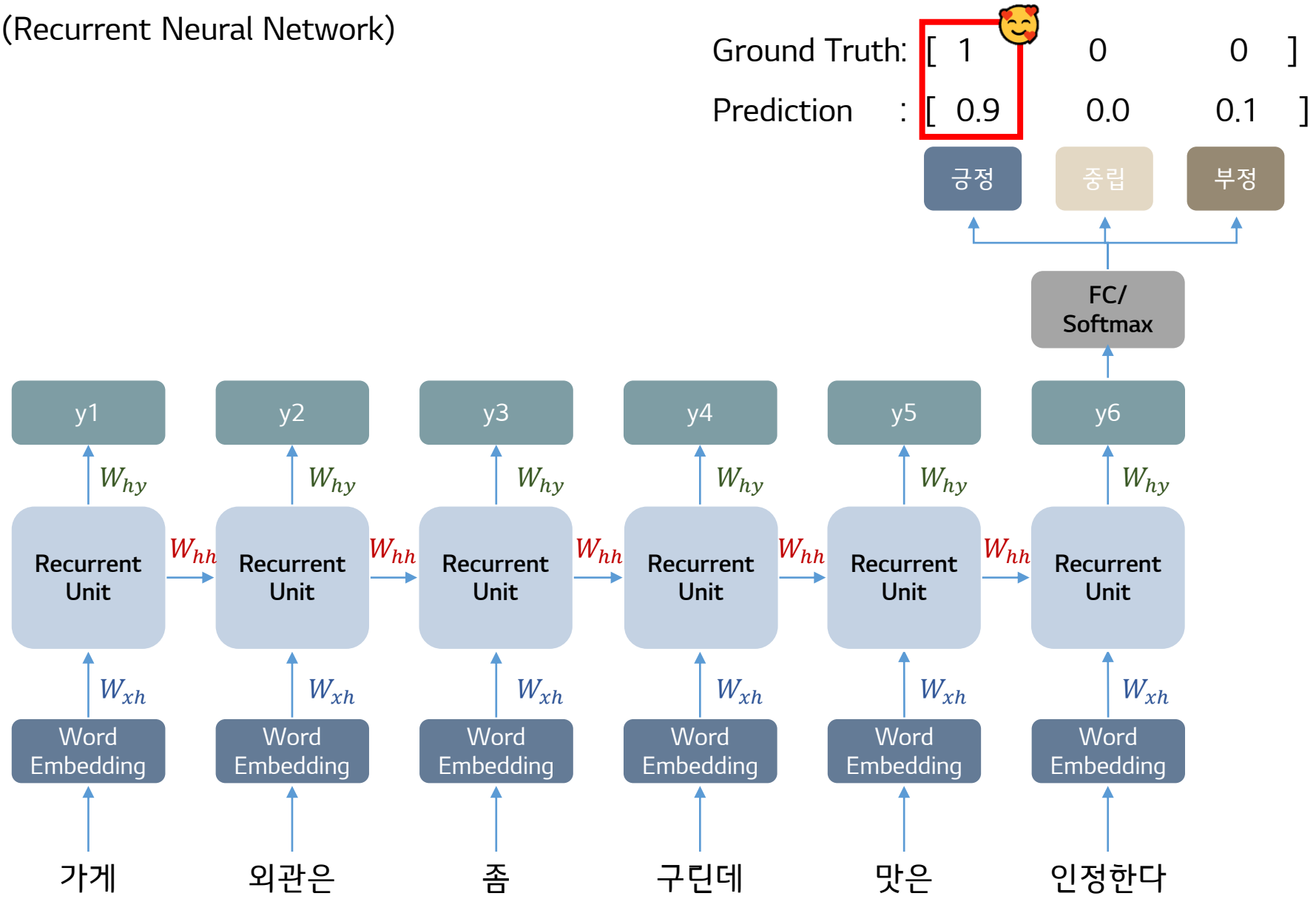
RNN(Recurrent Neural Network)



RNN(Recurrent Neural Network)



RNN(Recurrent Neural Network)



RNN(Recurrent Neural Network)

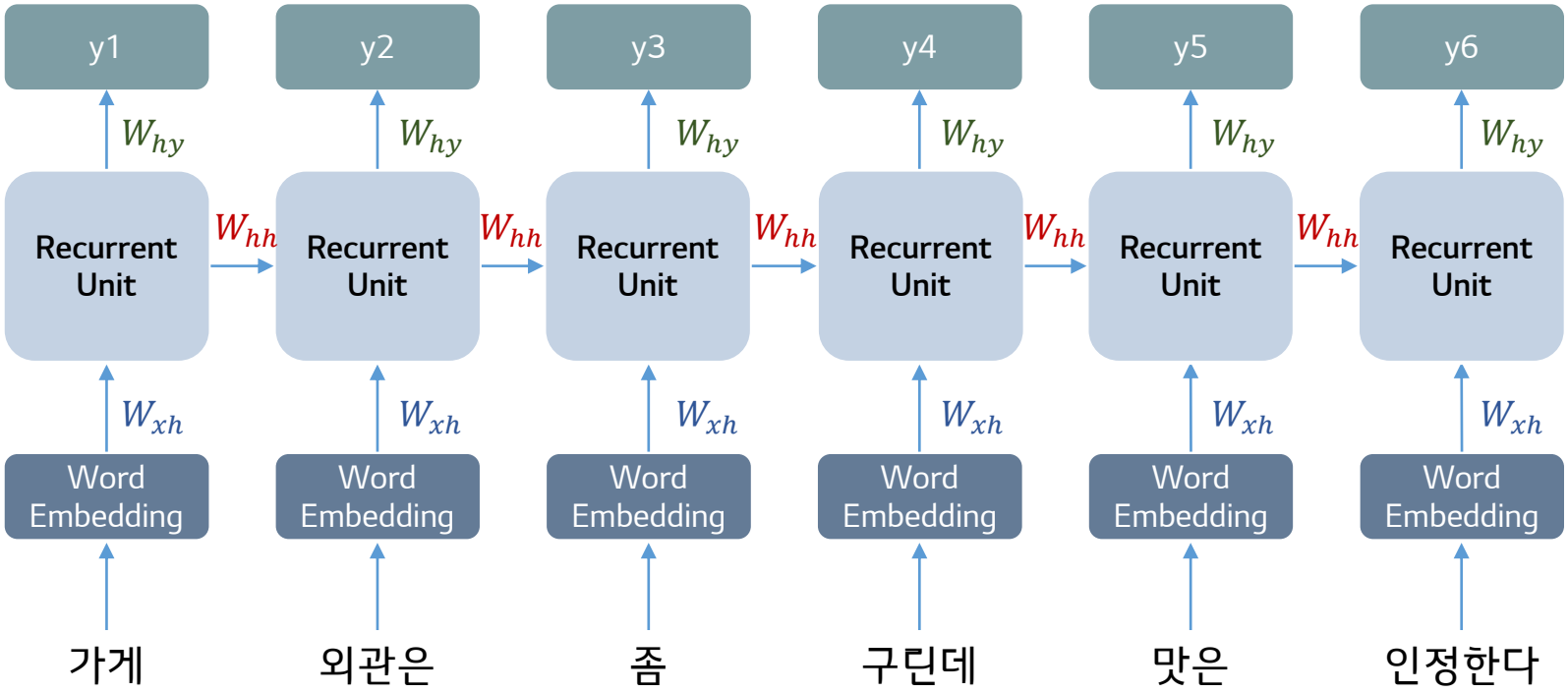
이전 타임 스텝의 hidden state와

현재 단계의 인풋은 결합해

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

$$y_t = W_{hy}\, h_t$$

현재 단계의 output은 생성해가는 과정

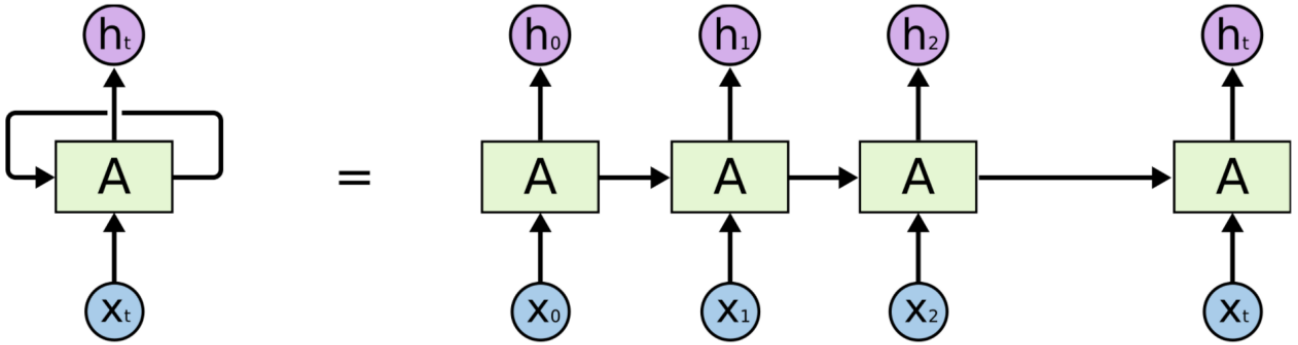


RNN(Recurrent Neural Network) 특징

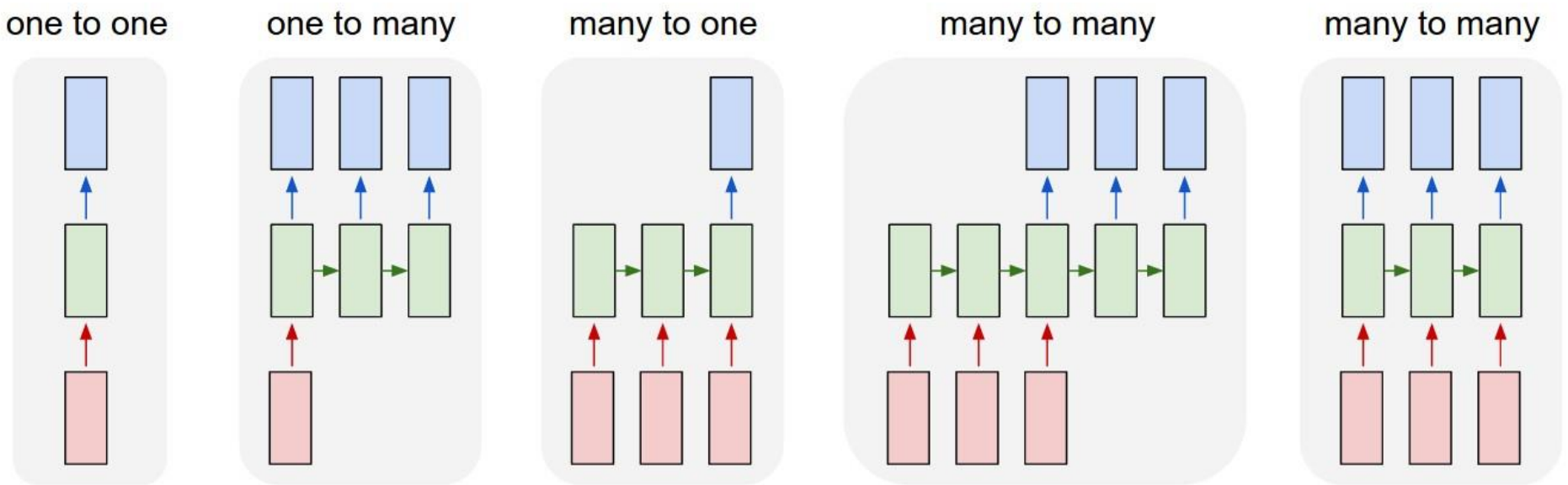
- Shared representation : 각각의 타임 스텝에서 같은 파라메터를 공유한다
  -> 긴 입력이 들어와도 모델 크기가 증가하지 않는다는 장점

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

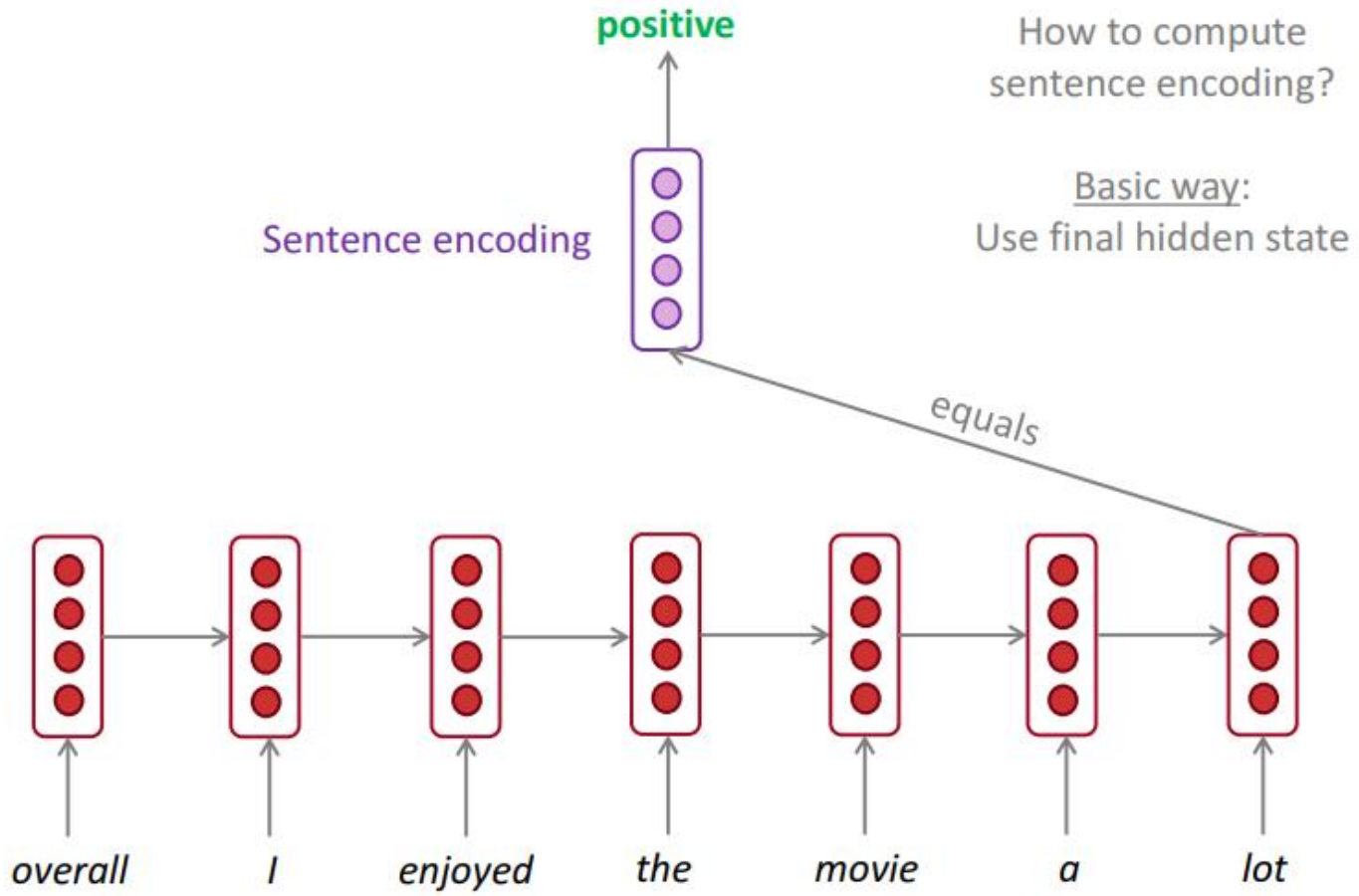
- 입력/출력의 유연성 : 임의의 길이의 순차적인(sequential) 입력을 처리할 수 있다



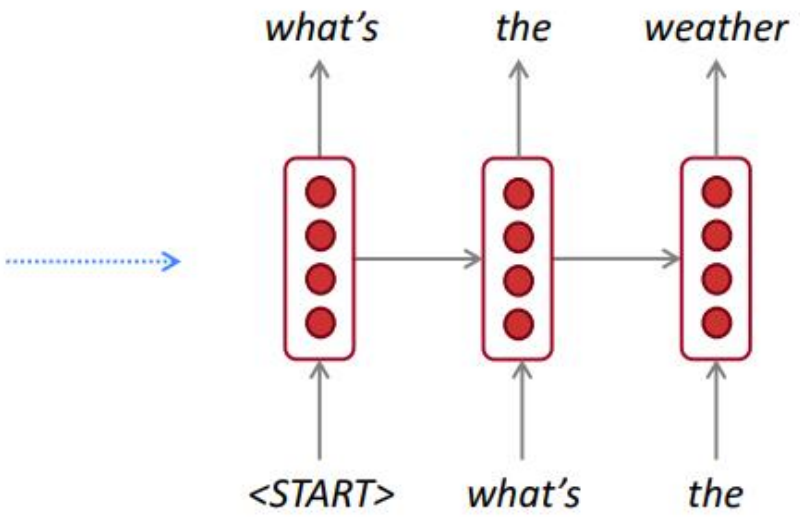
입력/출력의 유연성 -> 다양한 모델 디자인이 가능.



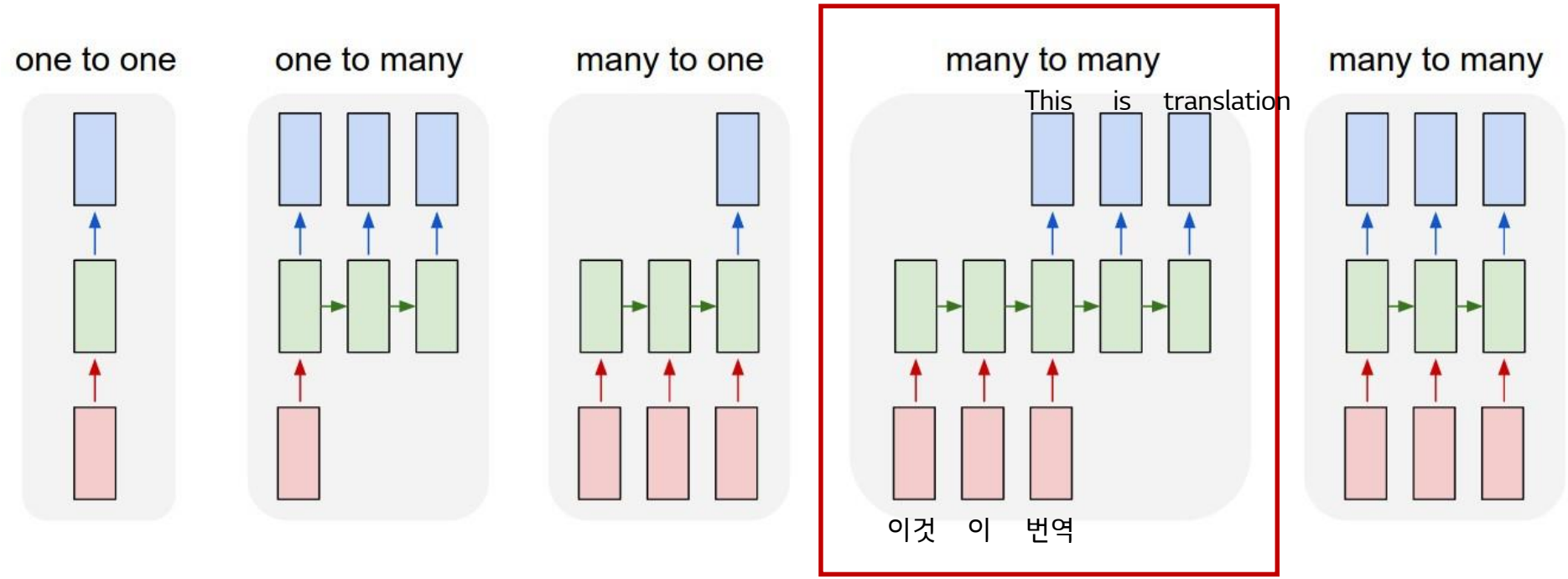
예시 1) 문장 분듀 - many to one



예시 2) 음성 인식 - many to many



Quiz) 번역 태스크에 RNN은 사용할 수 있을까요?



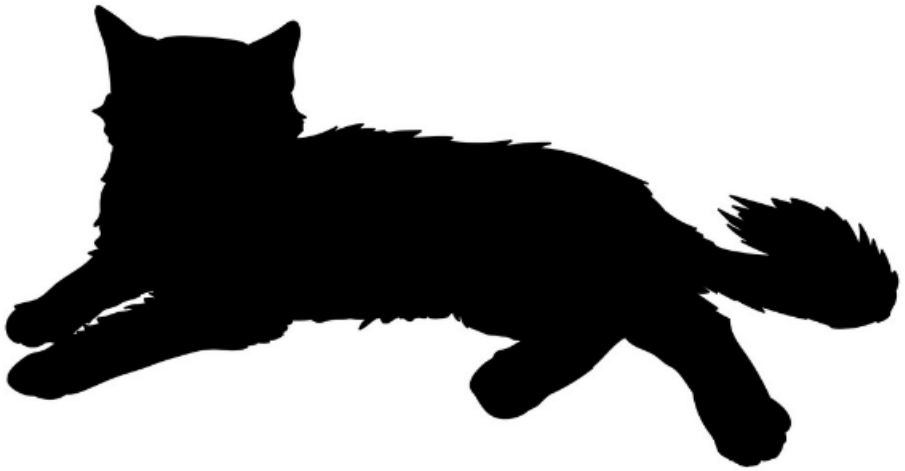
이전 step이 모두 계산되어야 현 step은 계산할 수 있기 때문에 다소 “느리다”

(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

- 언어에서 Long term dependency?

The cat, which already ate pear, apple, banana, tuna can, and ... , was full.

The cats, which already ate pear, apple, banana, tuna can, and ... , were full.



(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

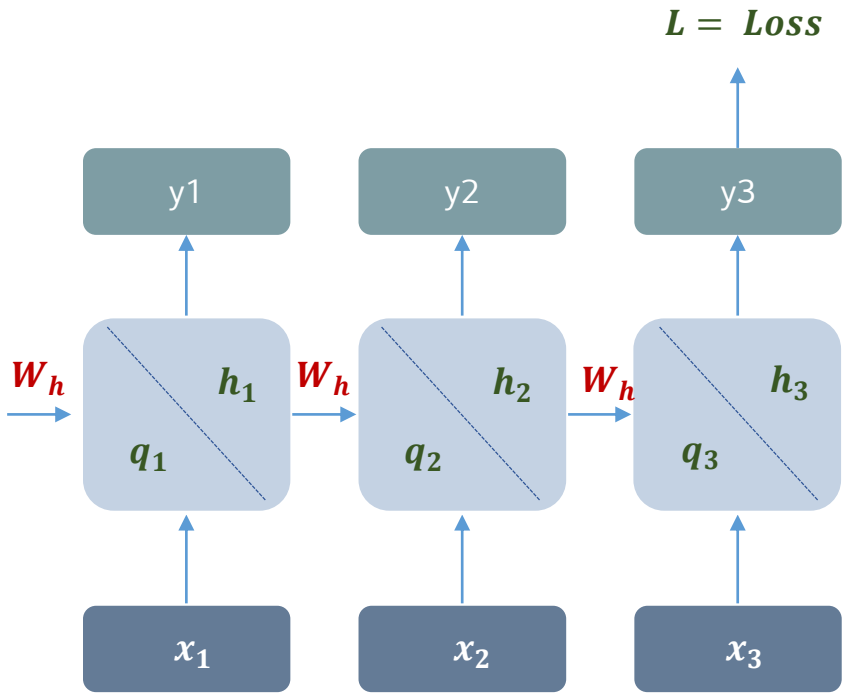
- Back-propagation throughout time

activation function

$$\mathbf{h}_t = \tanh(\underbrace{\mathbf{h}_{t-1}\mathbf{W}_h + \mathbf{x}_t\mathbf{W}_x + \mathbf{b}}_{q_t})$$

$$\mathbf{o}_t = \mathbf{c} + \mathbf{h}_t\mathbf{V}$$

$$\mathbf{y}_t = \mathrm{softmax}(\mathbf{o}_t)$$



(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

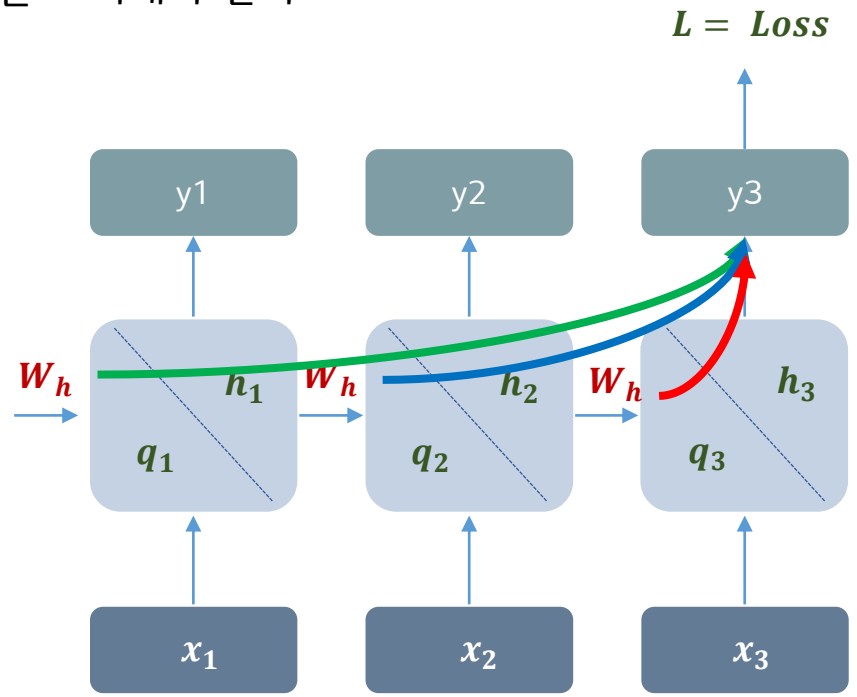
- Back-propagation throughout time

  : W가 거쳐간 모든 타임스텝의 경로에 대해 gradient를 고려해야 한다.

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial W} +$$

$$\frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial h_2}\frac{\partial h_2}{\partial q_2}\frac{\partial q_2}{\partial W} +$$

$$\frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial h_2}\frac{\partial h_2}{\partial q_2}\frac{\partial q_2}{\partial h_1}\frac{\partial h_1}{\partial q_1}\frac{\partial q_1}{\partial W}$$



(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

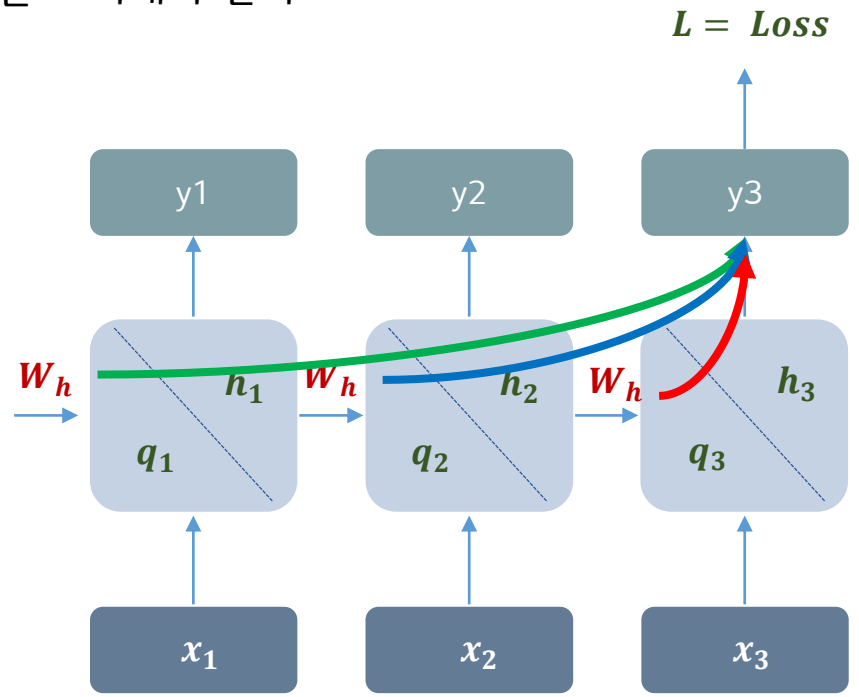
- Back-propagation throughout time

  : W가 거쳐간 모든 타임스텝의 경로에 대해 gradient를 고려해야 한다.

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial W} +$$

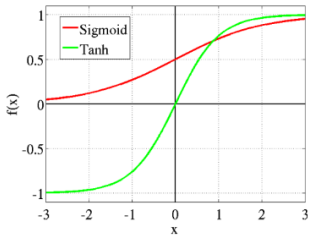
$$\frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial h_2}\frac{\partial h_2}{\partial q_2}\frac{\partial q_2}{\partial W} +$$

$$\frac{\partial L}{\partial h_3}\frac{\partial h_3}{\partial q_3}\frac{\partial q_3}{\partial h_2}\frac{\partial h_2}{\partial q_2}\frac{\partial q_2}{\partial h_1}\frac{\partial h_1}{\partial q_1}\frac{\partial q_1}{\partial W}$$



예 : Sigmoid 의 도함수 값 ∈ (0,0.25]<1

0.25*0.25*0.25 = 0.015625
조금만 많아져도 0에 수렴해 감



(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

왜 vanishing gradient가 문제가 될까?

- gradient의 크기가 거의 0이 되며, n이 커질 수록 t, t+n 타임스텝 사이의 의존성이 감소
- 즉, 현재 타임스텝의 output은 이전 타임스텝의 내용을 다 잊어버림

  → long-term dependency

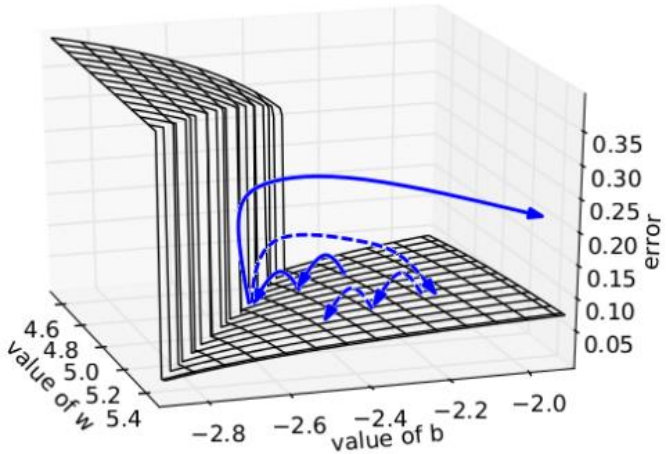
(실제로는) 많은 step 전의 정보를 활용할 수 있는 모델이 아니다 >> long-term dependency problem

반대로 1보다 큰 미분 값이 계속 곱해지며 exploding gradient 문제가 발생할 수 있음

- Gradient Clipping
  - gradient의 norm이 어떤 상한을 넘지 않도록 유지함으로써,
    parameter들을 너무 큰 폭으로 update실선하지 않도록 방지점선

Algorithm 1 Pseudo-code for norm clipping the gradients whenever they explode

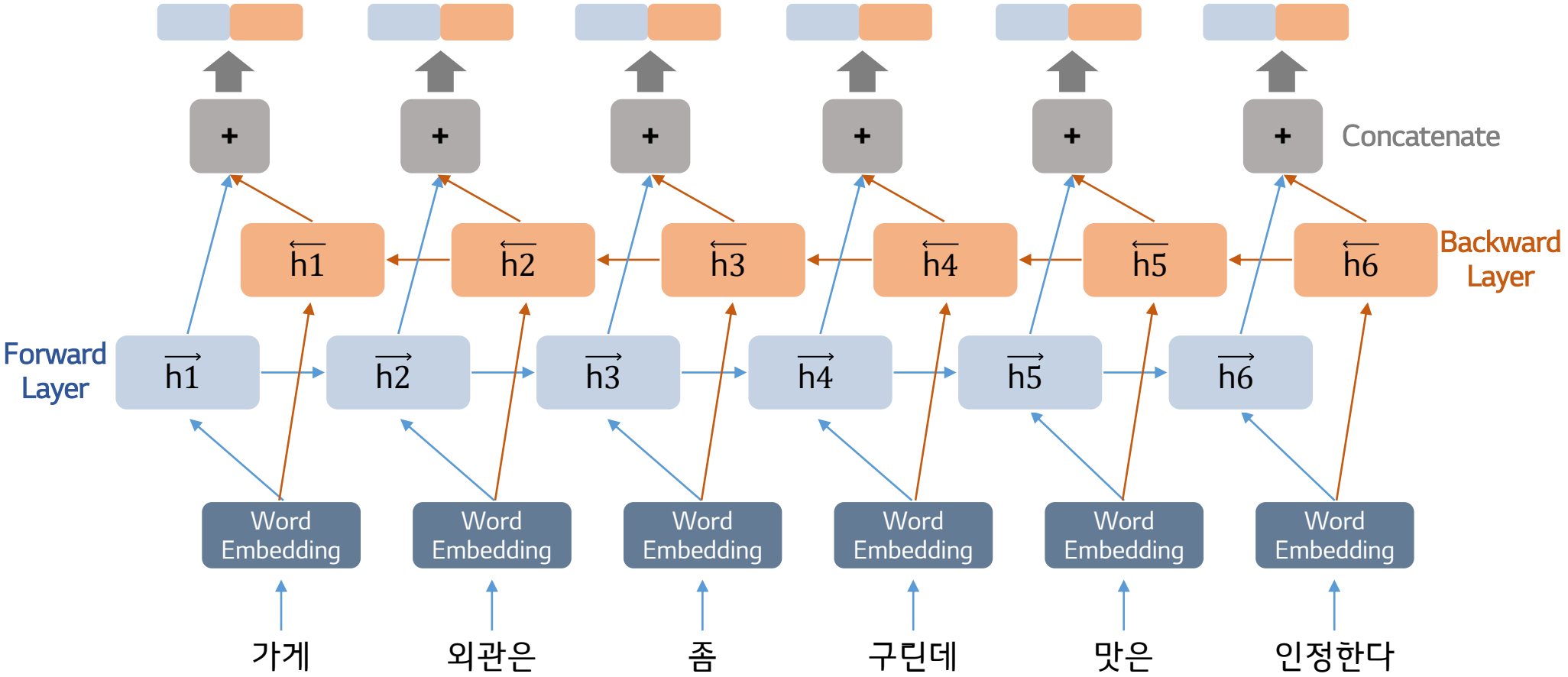
$\hat{\mathbf{g}} \leftarrow \frac{\partial \mathcal{E}}{\partial \theta}$
if $\|\hat{\mathbf{g}}\| \geq threshold$ **then**
  $\hat{\mathbf{g}} \leftarrow \frac{threshold}{\|\hat{\mathbf{g}}\|}\hat{\mathbf{g}}$
end if



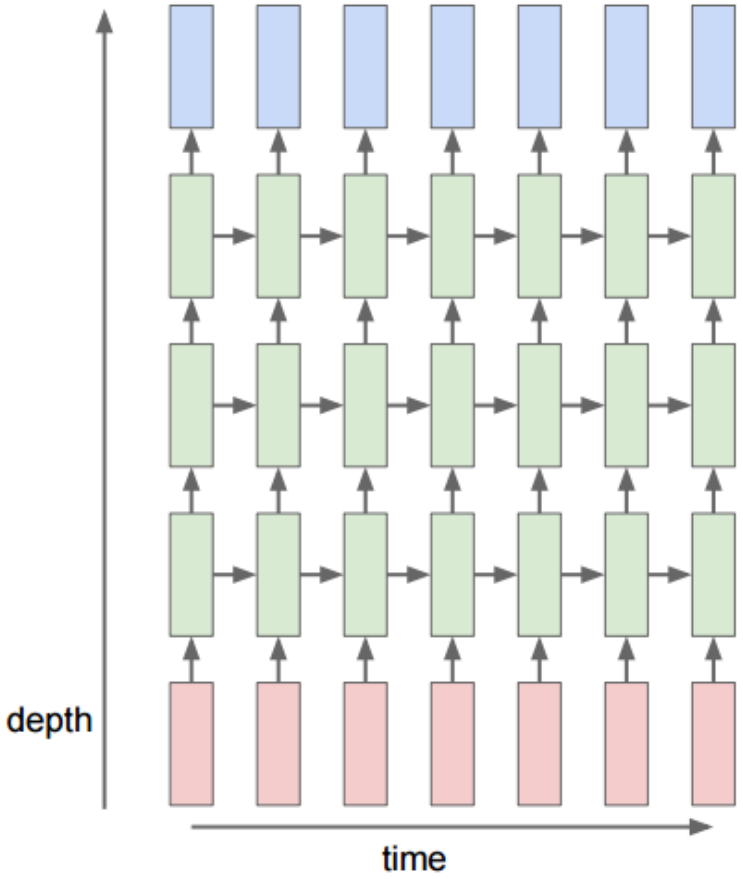
- Exploding gradient는 gradient clipping같은 테크닉으로 비교적 쉽게 해결 가능
  - Vanishing gradient 문제는 좀더 까다로움 -> 새로운 셀 구조 등장!

Bidirectional RNN : 양 방향에서 오는 문맥 정보 환용하기

도큰은 정 방향으도 처리하는 forward layer에서 나온 hidden 과
역방향으도 처리하는 backward layer에서 나온 hidden 은 concat해서 환용하는 방법



Multi-layer RNN : 깊은 레이어를 가지는 RNN



$$h_t^{layer} = tanh\left(W^{layer}\begin{pmatrix} h_t^{layer-1} \\ h_{t-1}^{layer} \end{pmatrix}\right)$$

- 이전 레이어에서 넘어온 정보를 이용해 RNN을 깊게 쌓을 수 있음.
- 일반적으로 언어 모델에서는 layer을 4층 이상 깊게 쌓지는 않음.
- Over-fitting을 방지하기 위해 레이어 사이에 dropout을 추가해 주는 것이 좋음.


갬성

RNN은 이용한 감성 분듀 실습

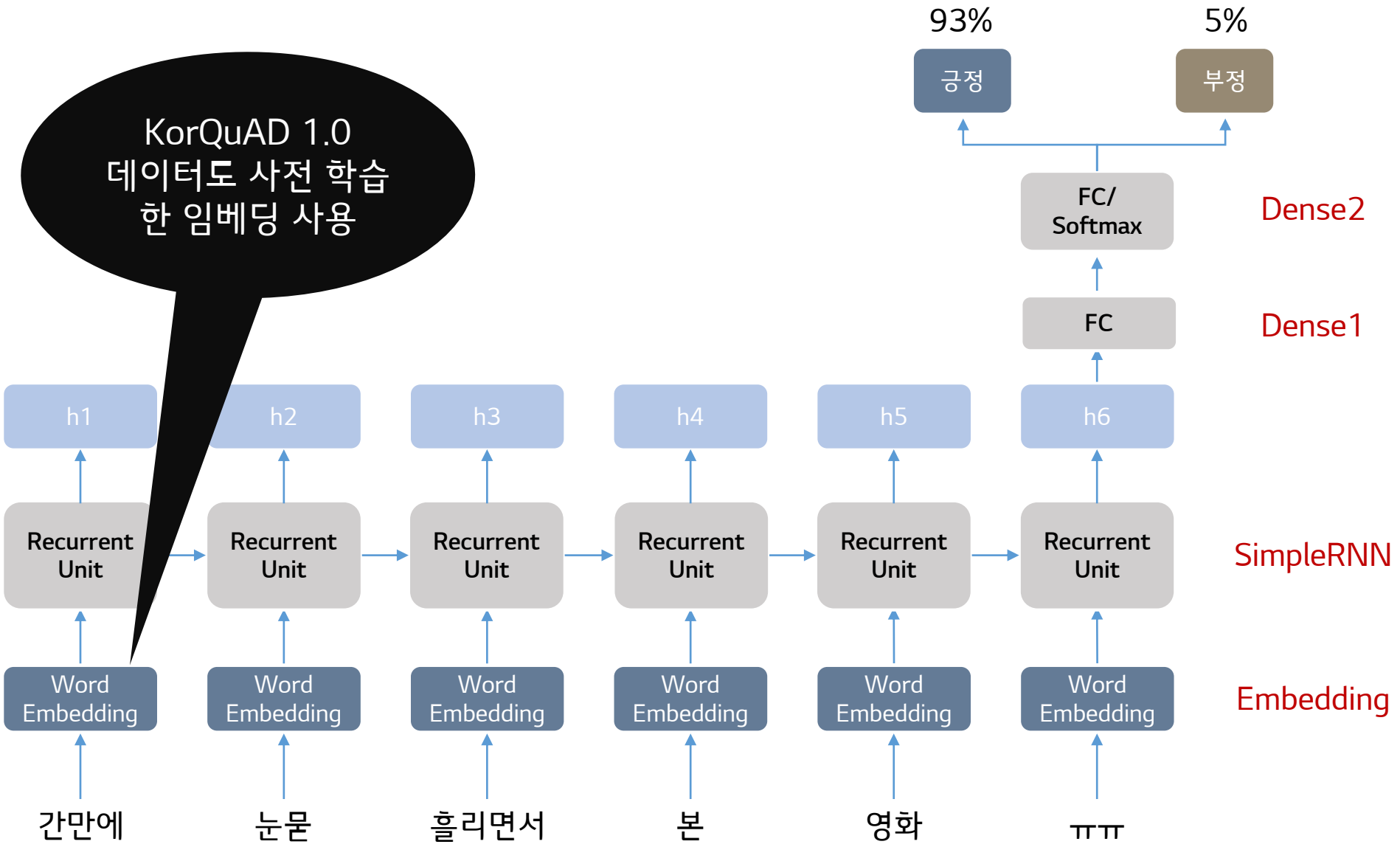
실습 2_RNN_basic.ipynb

- **데이터** : 네이버 영화리뷰 (https://github.com/e9t/nsmc)

- 네이버 영화 평점 크롤링 데이터 20만 건
- 평점 1-4 댓글 = 부정
- 평점 9-10 댓글 = 긍정 으도 라벨링

- **학습 목표** :
  - NLU의 전체 FLOW (토크나이징 -> 인코딩 -> 임베딩 -> 모델링)을 이해하고 구현한다.
  - TensorFlow에서 RNN layer을 사용해 감성분류 모델을 구축할 수 있다.
  - 기 학습된 Word Embedding을 불러와 학습에 활용할 수 있다.
  - 모델을 컴파일하고 분석, 평가하는 딥러닝 전체적인 플로우를 실행할 수 있다.

CBOW도 학습해둔 Word Embedding 환용하기



CBOW도 학습해둔 Word Embedding 환용하기

- 사전 학습된 임베딩은...

```
사과 -> 반박 , 표명 , 비난 , 해명 , 경고 , 답변 , 애도 , 항의
취업 -> 미연 , 냉전시대 , 우주정거장 , 므스티슬라비츠 , 교비 , 플래너스 , 홀수 , 야누코비치
밥 -> 목 , 술 , 결혼식 , 눈물 , 세스 , 공경 , 숲 , 고함
사랑 -> 스타일 , 웃음 , 질주 , 존경 , 컨셉 , 전설 , 꿈 , 미츠
수학 -> 화학 , 생리 , 해부학 , 광학 , 도덕 , 생태 , 천문학 , 살아나는데다
운동 -> 정책 , 개혁 , 인권 , 정변 , 행정 , 통일 , 세력 , 3.1
가수 -> 아티스트 , 배우 , 댄스 , 힙합 , 아이돌 , 팝 , 여자 , 컨트리
회사 -> 모터 , 요르단 , 호부 , 시라쿠사 , 고모 , 디자이너 , 사이토 , 혁명가
자신감 -> 유치하고 , 교훈 , 제물 , 명성 , 불안감 , 결실 , 책임감 , 아쉬움
```

- 위키백과에 대해 사전 학습된 단어사전 -> 문어체
- 네이버 영화 리뷰 -> 문어체

- (Quiz) 감성분석 태스크에서 새로 나오는 단어들은 어떻게 할까요?

CBOW도 학습해둔 Word Embedding 환용하기