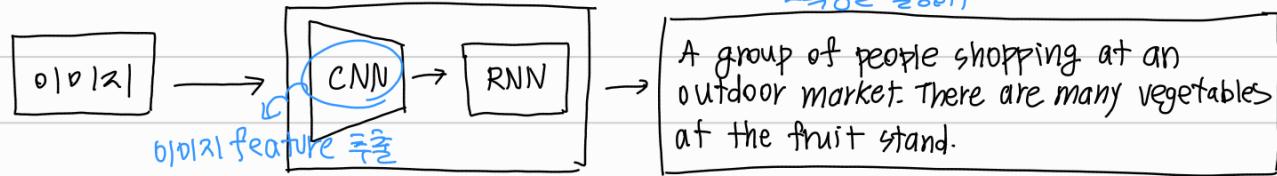


동빈나님의 유튜브 강의를 보고 공부하면서 정리한 내용입니다  
문제가 될 시 삭제하겠습니다  
(출처:<https://youtu.be/yfsFW-mfOEY?feature=shared>)

# 동빈나 - "Show and Tell: A Neural Image Caption Generator"

- \* 이미지 캡션 생성 (Image caption Generation) 이란?
  - 이미지를 설명하는 문장을 생성하는 기술 분야를 의미
  - 대표적인 모델로는 오늘 소개하는 Neural Image Captioning (NIC)가 있음
 

↳ CNN 네트워크를 이용해 이미지의 특징을 추출한 뒤에 RNN을 거쳐 문장을 생성할 수 있음  
↑ 특징을 활용해



⇨ 이미지를 설명하는 문장을 만들

- \* 이미지 캡션을 생성하는 "이미지를 번역"하는 문제로 보기

- 입력: 이미지 I
- 출력: 목표 문장  $S = \{s_1, s_2, \dots, s_n\}$

I를 설명하는 적절한 문장



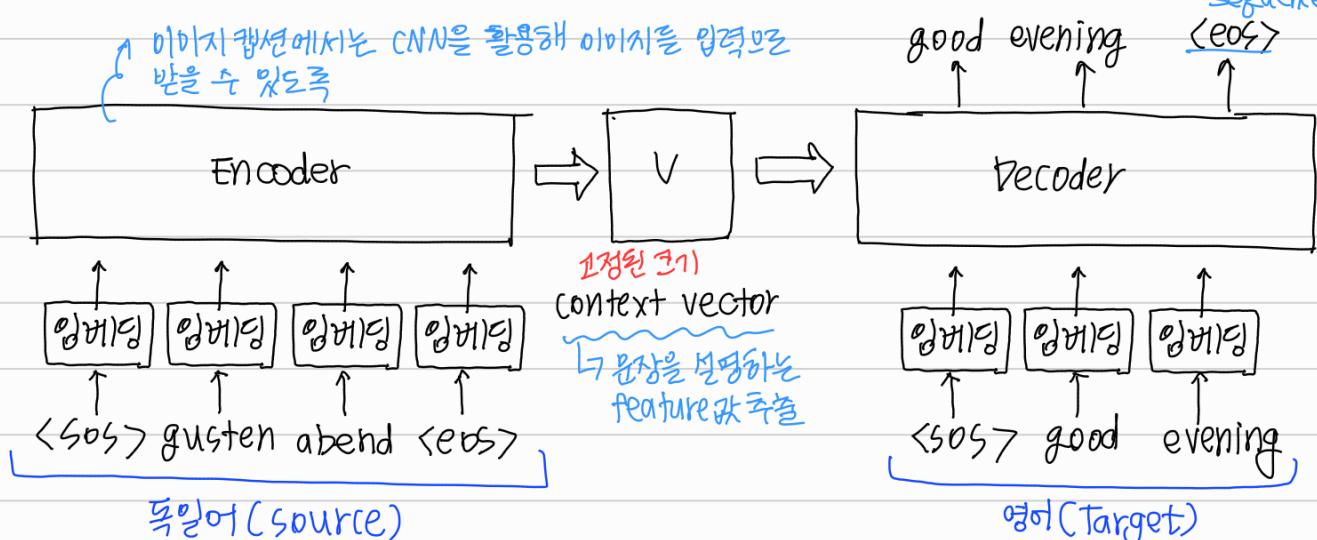
- 따라서 기능도 (likelihood)  $P(S|I)$ 를 최대화 (maximization) 하는 문제로 정의할 수 있음
 

이미지 I가 들어왔을 때  
적절한 문장의 확률값을 오델링 ⇒ 어떤 문장이 더 높은 확률이 적절한지 예측

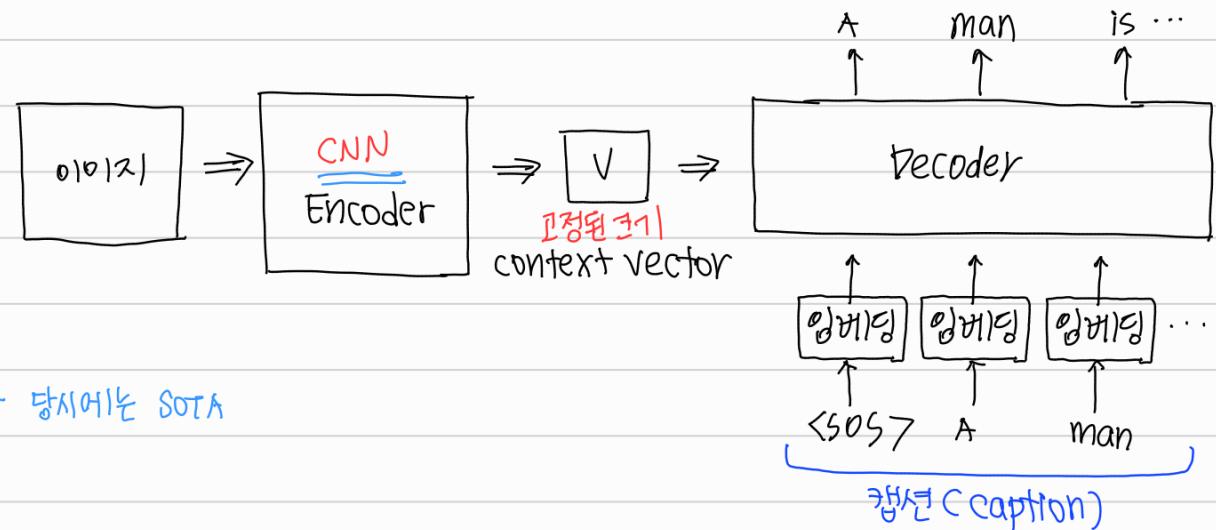
- \* 이미지 캡션 생성과 기계번역의 공통점

- 기계번역에서는  $P(T|S)$ 를 최대화 (maximizing) 함

↳ 소스문장 (source sentence)을 대표하는 하나의 벡터 (context vector)를 이용  
end of sequence



• 기계 번역 작업에서의 인코더(Encoder)를 CNN으로 대체하여 이미지 캡션을 생성할 수 있음



- 당시에는 SOTA

## \* 공식 (Formulation)

• 하나의 이미지를 설명(description)으로 번역(translation)하는 작업에 배우할 수 있음

$$\theta^* = \arg \max_{\theta} \sum_{(I, S)} \log p(S | I; \theta)$$

↗ 일반 기계번역에 사용되는  
objective f과 비슷  
이미지

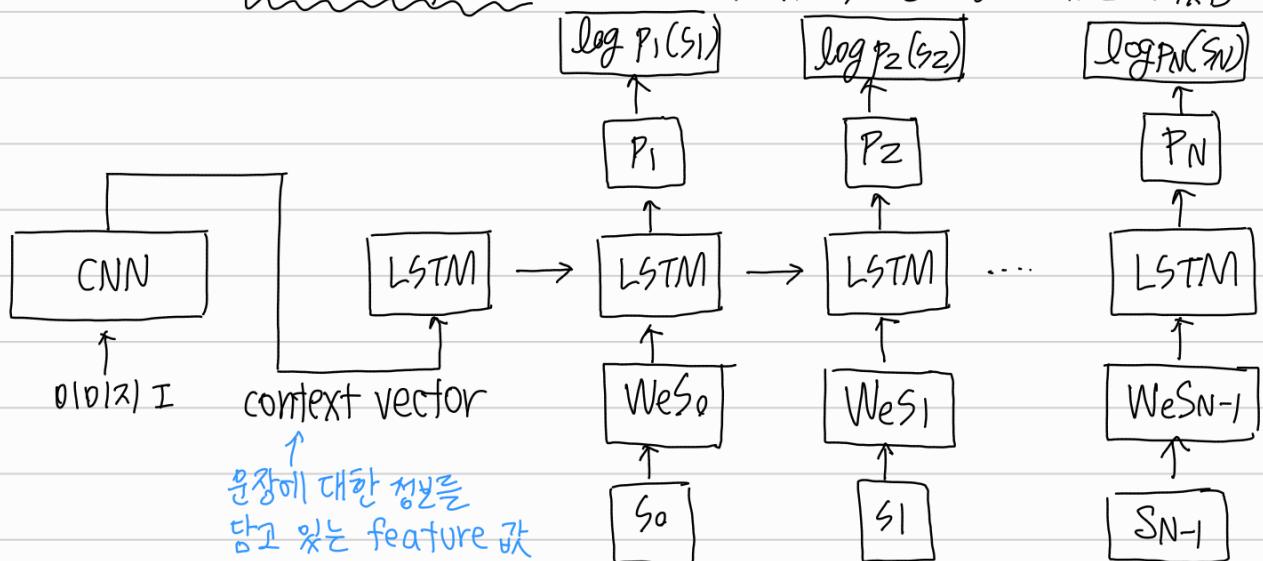
• 연쇄법칙(chain-rule)을 이용해 다음의 식으로 전개할 수 있음

$$\log p(S | I) = \sum_{t=0}^N \log p(S_t | I, S_0, \dots, S_{t-1})$$

↑  
시작점

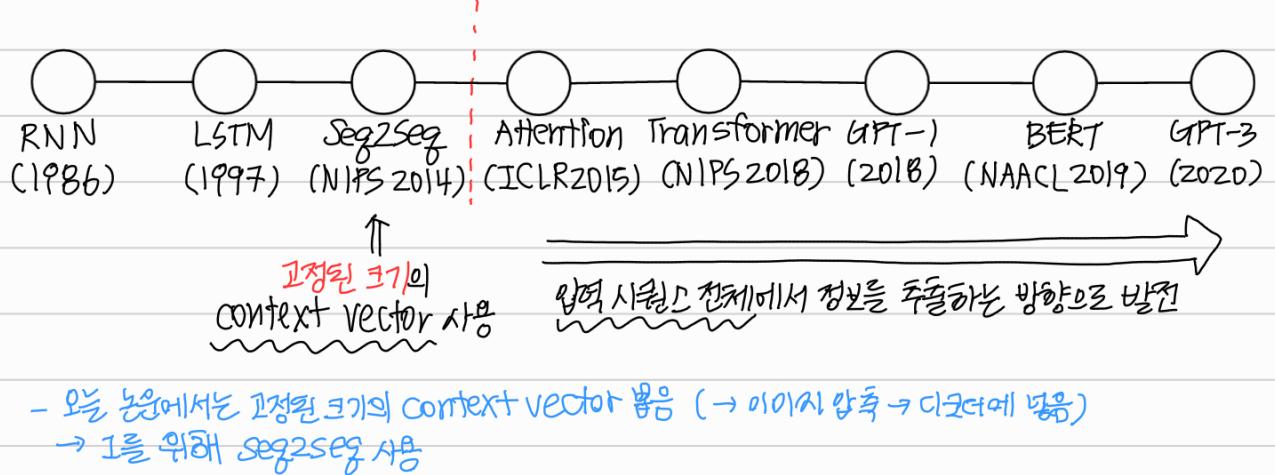
## \* NIC (Neural Image Caption) 아키텍처

• 하나의 이미지를 설명(description)으로 번역(translation)하는 작업에 배우할 수 있음



## \* [배경지식] 딥러닝 기반의 언어 모델 발전 과정

- 딥러닝 기반의 언어 모델의 발전 과정은 다음과 같음
  - 오늘 소개하는 논문에서는 LSTM을 활용한 Seq2Seq와 유사한 아키텍처를 사용
  - 2021년 기준으로 최신 고성능 모델들은 Transformer 아키텍처를 기반으로 하고 있음



## \* RNN 자체의 알아보기

### • 입력 (input)

$x_t$  = 각각의 입력 단어

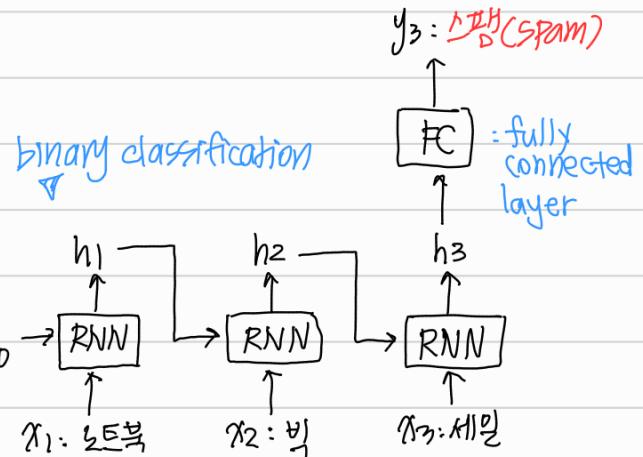
### • 하든 상태 (hidden state)

$$h_t = \text{sigmoid}(W^{hx}x_t + W^{hh}h_{t-1})$$

### • 출력 (output)

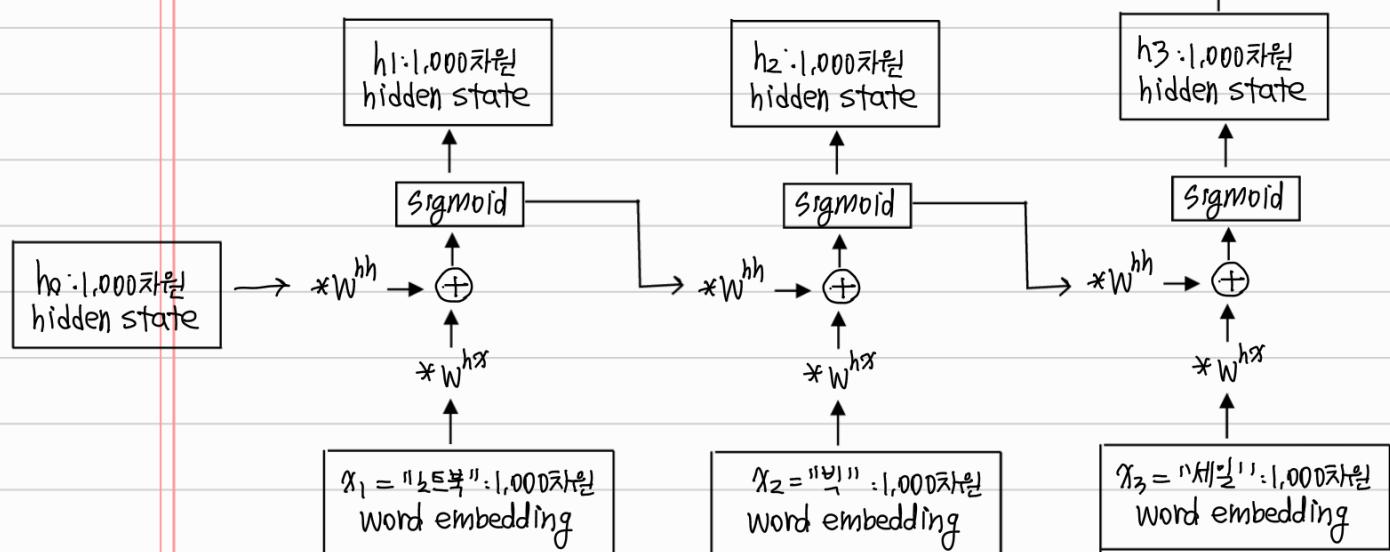
$$y_t = W^{yh}h_t$$

각각 차리 ✓



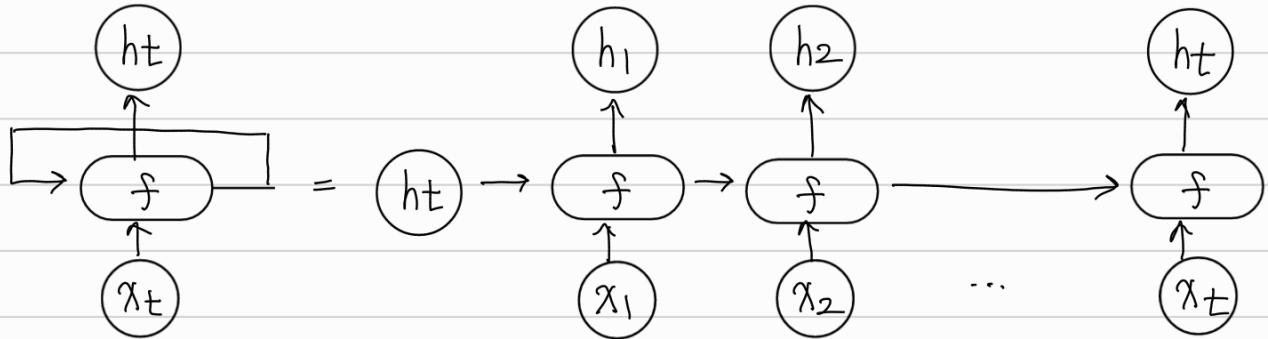
• 전체 예시 그림에서  $W^{hx}, W^{hh}, W^{yh}$ 는 하나의 파라미터를 공유함

hidden state: 이전까지 입력으로 들어왔던 모든 단어들을 포함적으로 가지고 있는 feature vector

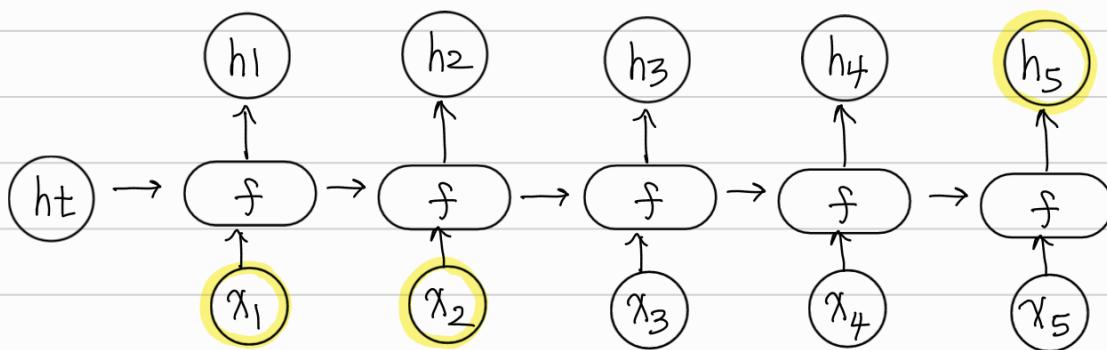


## \* RNN의 한계점

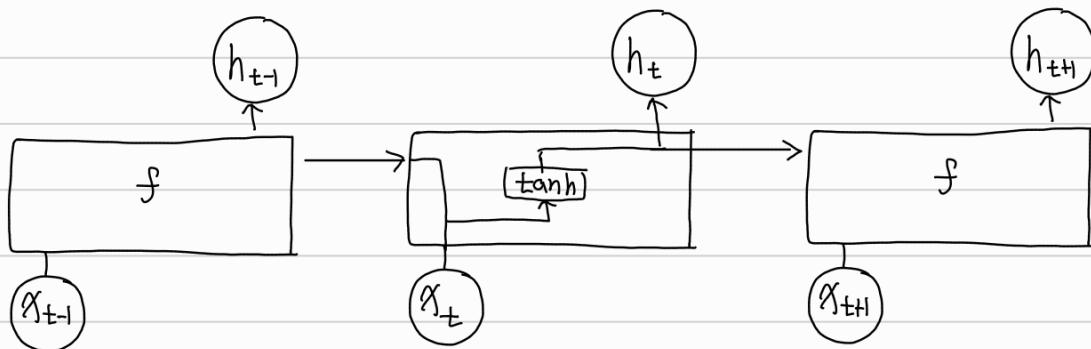
- 이론적으로 RNN을 이용하여 긴 길이의 순차적인 데이터를 효과적으로 처리할 수 있음



- 실제로는 토큰(token) 사이의 거리가 먼 경우 연속적인 정보가 잘 전달되지 않을 수 있음

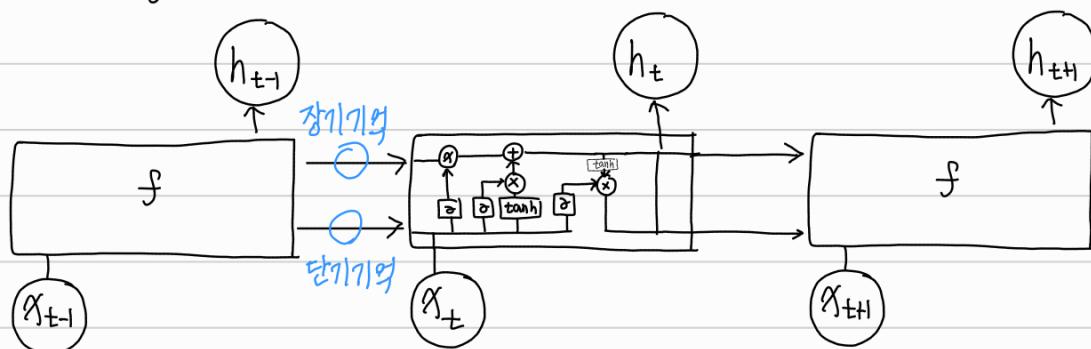


## \* RNN(Recurrent Neural Network) 아키텍처



문제개선  
↓

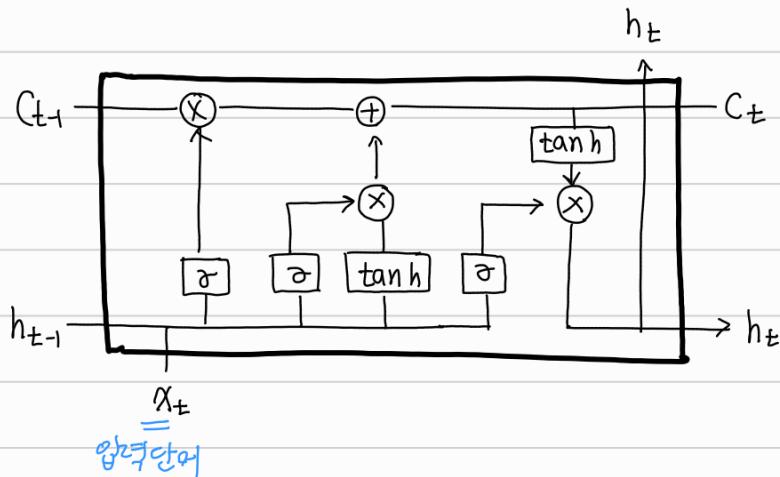
## \* LSTM(Long Short-Term Memory) 아키텍처



## \* LSTM 핵심 아이디어: 두 개의 상태 정보

- LSTM은 RNN과는 다르게 **두 가지의 상태 정보**를 저장하고 처리함.

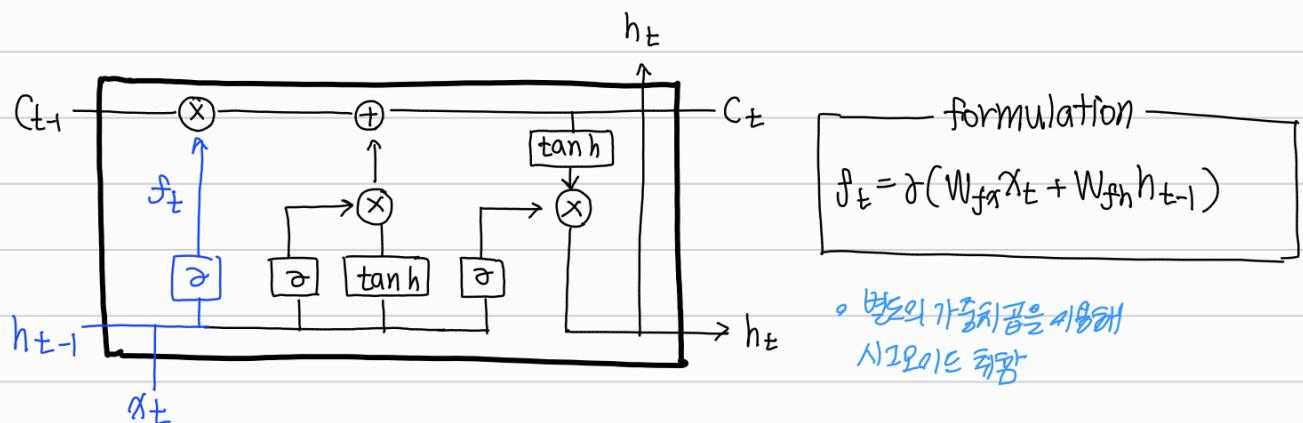
- 장기 기억: Cell State 가 갖도록 함
- 단기 기억: Hidden State가 갖도록 함



## \* LSTM 핵심 아이디어: 게이트(Gates)

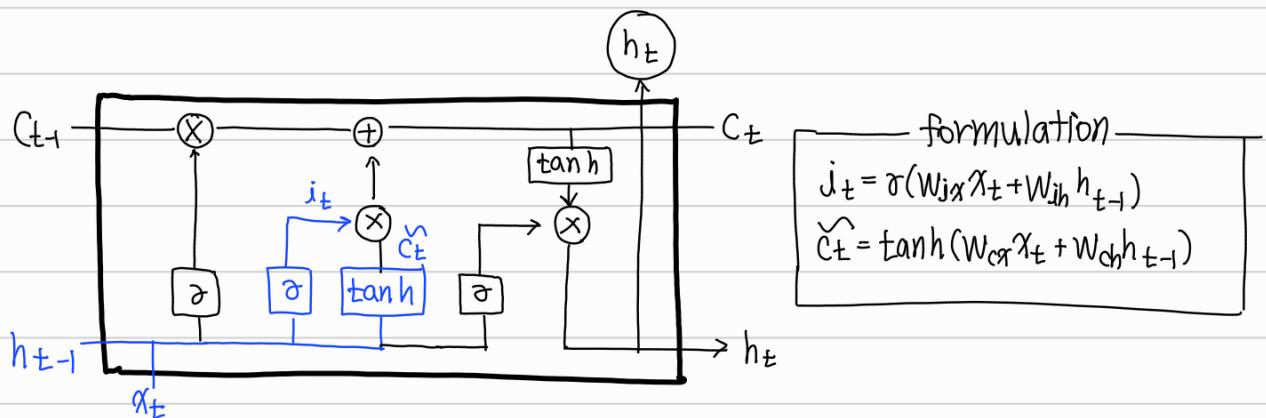
- Forget Gate는 어떠한 정보를 잊게 만들지 결정하는 레이어임

↳ 오래된 정보 중에서 필요 없는 정보는 잊게 됨

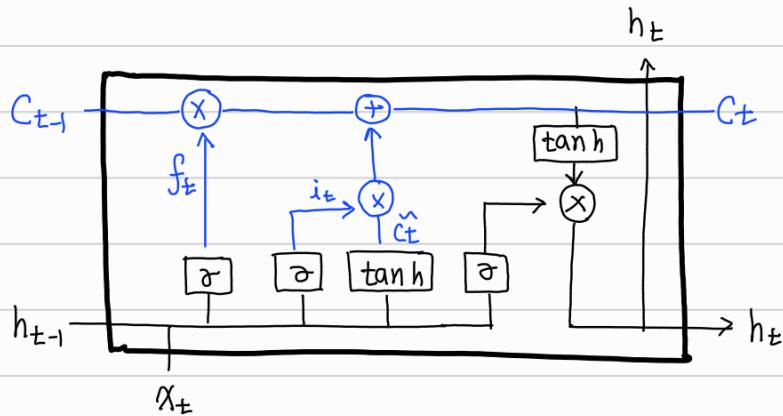


- Input Gate는 새로운 정보를 장기 기억(Cell State)에 반영하는 역할을 수행함

↳ 새롭게 특징한 정보를 기억하도록 만듬



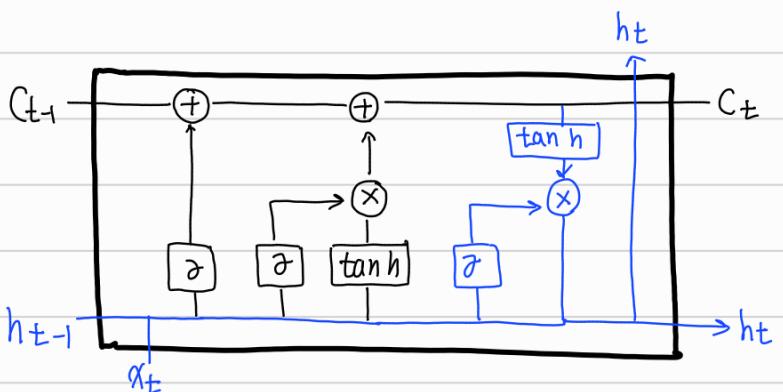
- 장기 기억(Cell State)은 Forget Gate와 Input Gate를 이용하여 업데이트됨



$$\boxed{\text{Formulation}}$$

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

- Output Gate는 장기 기억과 현자(Hidden State)를 이용해 단기 기억(Hidden State)을 생성함.



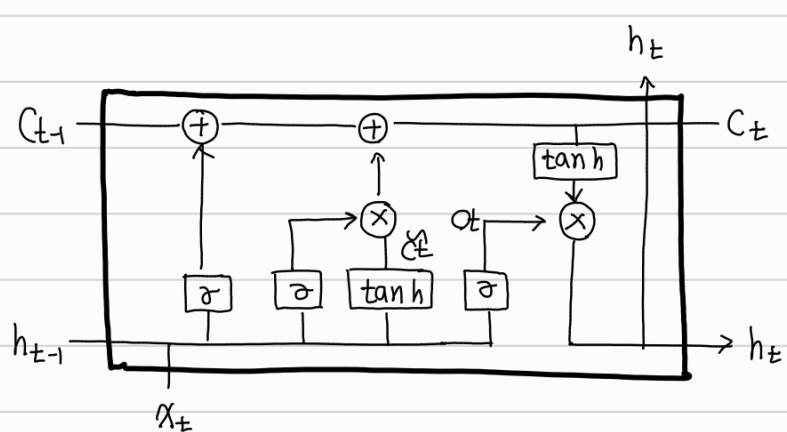
$$\boxed{\text{formulation}}$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1})$$

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

### \* LSTM 전체 공식

- LSTM 전체 공식은 다음과 같습니다.
  - 공식에 등장하는 모든 가중치 (weight)는 공유됨.



$$\boxed{\text{Formulation}}$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1})$$

$$j_t = \sigma(W_{jx}x_t + W_{jh}h_{t-1})$$

$$\tilde{c}_t = \tanh(W_{cx}x_t + W_{ch}h_{t-1})$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1})$$

$$C_t = f_t * C_{t-1} + j_t * \tilde{c}_t$$

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

## \* 생성 결과(Generation Results)

- 대표적인 평가 지표를 이용해 NIC를 이용해 생성된 결과를 평가한 것은 다음과 같음.

기존에 제안된  
다른 메타드에  
비해 성능이 좋음

Metric	BLEU-4	METEOR	CIDER
NIC	<b>27.7</b>	<b>23.7</b>	<b>85.5</b>
Random	4.6	9.0	5.1
Nearest Neighbor	9.9	15.7	36.5
Human	21.7	25.2	85.4

[Table] Scores on the MSCOCO development set.

Approach	PASCAL (xfer)	Flickr 30k	Flickr 8k	SBU
Im2Text [24]				11
TreeTalk [18]				19
BabyTalk [16]	25		48	
Tri5Sem [11]		55	58	
m-RNN [21]		56	51	
MNLN [14] <sup>5</sup>				
SOTA	25	56	58	19
NIC	<b>59</b>	<b>66</b>	<b>63</b>	<b>28</b>
Human	69	68	70	

[Table] BLEU-1 scores. Authors only report previous work results when available.

사람에 \* 사람에  
필적할 만큼 X  
CBLU SCORE는  
정확성↑

## HUMAN Evaluation

• 사람에 1점(worst)부터 4점(best)까지의 점수로 평가한 결과는 다음과 같음

- 실제 정답(best)에 비하면 매우 정수가 낮음
- 그래도 이전까지의 모델보다 성능이 뛰어남

Metric	BLEU-4	METEOR	CIDER
NIC	<b>27.7</b>	<b>23.7</b>	<b>85.5</b>
Random	4.6	9.0	5.1
Nearest Neighbor	9.9	15.7	36.5
Human	21.7	25.2	85.4

[Table] Scores on the MSCOCO development set.

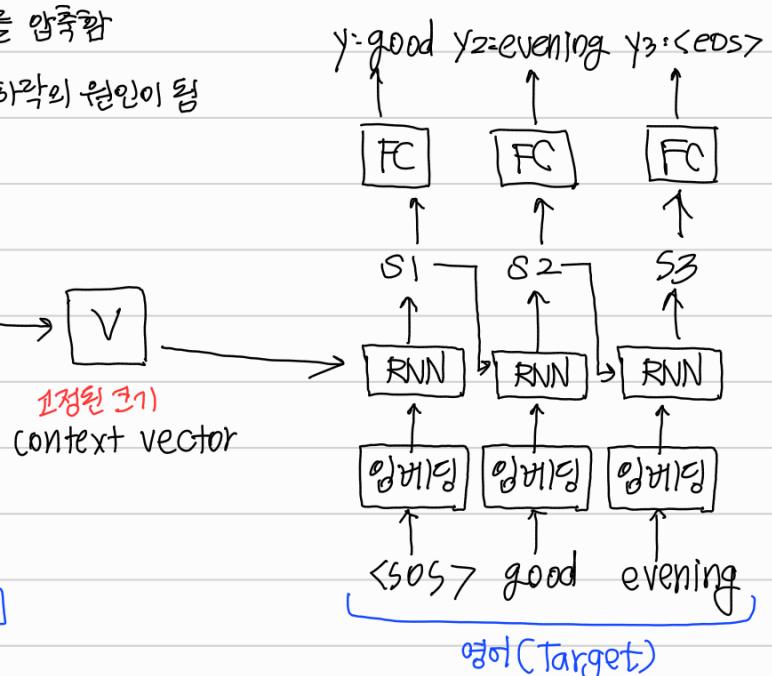
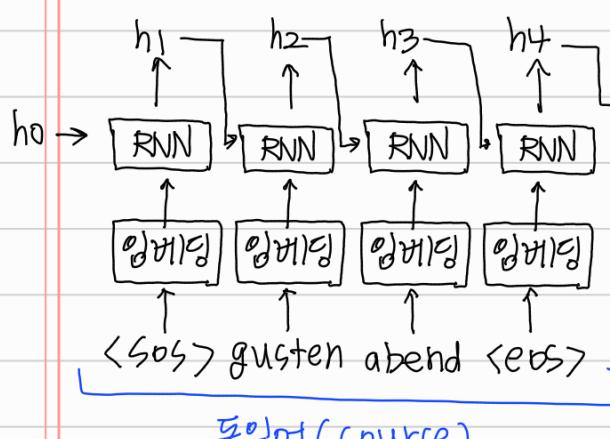
Approach	PASCAL (xfer)	Flickr 30k	Flickr 8k	SBU
Im2Text [24]				11
TreeTalk [18]				19
BabyTalk [16]	25		48	
Tri5Sem [11]		55	58	
m-RNN [21]		56	51	
MNLN [14] <sup>5</sup>				
SOTA	25	56	58	19
NIC	<b>59</b>	<b>66</b>	<b>63</b>	<b>28</b>
Human	69	68	70	

[Table] BLEU-1 scores. Authors only report previous work results when available.

## \* [후속 연구] 기존 seq2seq 모델들의 한계점

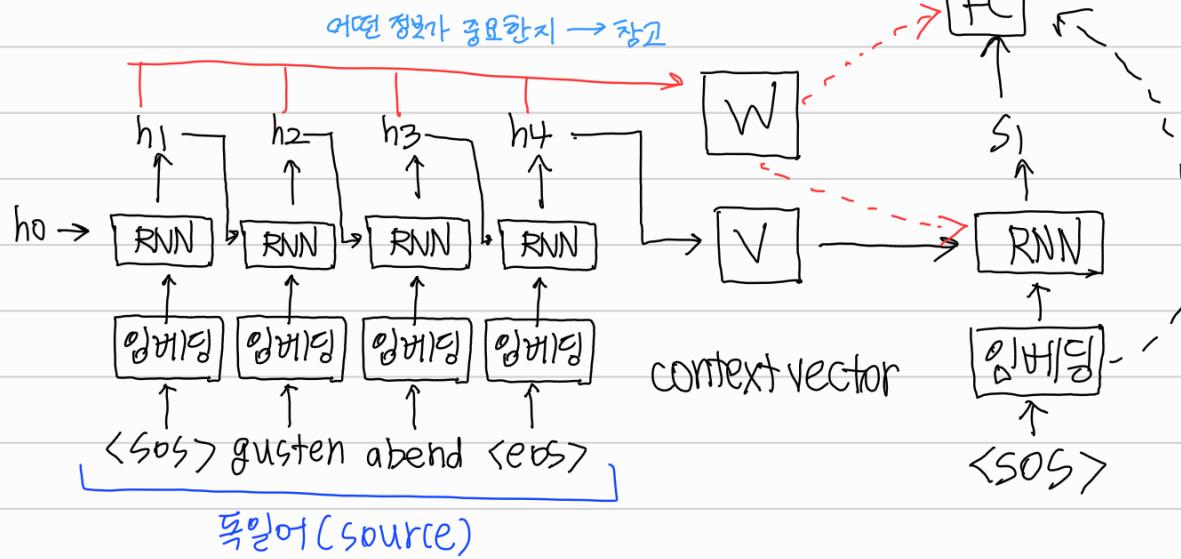
- context vector V에 소스 문장의 정보를 압축함

↳ 병목(bottleneck)이 발생하여 성능 하락의 원인이 됨



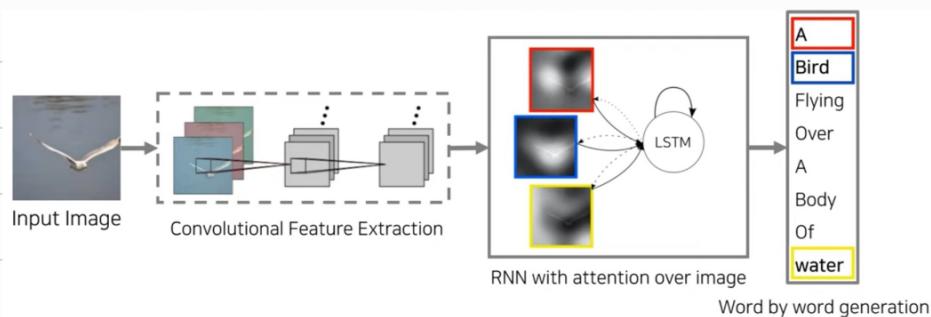
## \* [후속연구] Seq2Seq with Attention

- Seq2Seq 모델에 어텐션(attention) 매커니즘을 사용함
  - 디코터는 인코터의 모든 출력(outputs)을 참고함

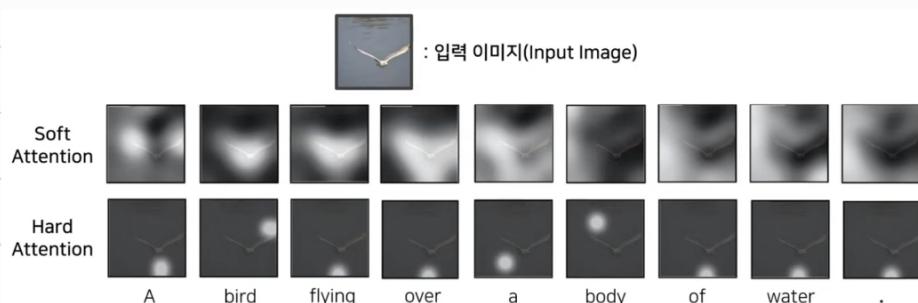


## \* [후속연구] Neural Image Caption Generation with Visual Attention (ICML 2015)

- 본 논문에서는 Neural Image Caption (NIC) 네트워크에 어텐션(attention) 기법을 적용함
  - RNN을 사용할 때 이미지 전체에 대한 어텐션(attention) 정보를 활용함. 어텐션 → 성능↑



- 차례대로 단어(word)를 하나씩 생성할 때의 어텐션(attention)을 시각화한 것은 다음과 같음  
(어떤 정보를 참고했는지)
  - 본 논문에서는 ① 소프트(soft) 어텐션과 하드(hard) 어텐션을 제안함



- 소프트(80%) 어扪션의 다양한 예시를 확인해 봅



A woman is throwing a frisbee in a park.



A dog is standing on a hardwood floor.



A stop sign is on a road with a mountain in the background.



A little girl sitting on a bed with a teddy bear.



A group of people sitting on a boat in the water.



A giraffe standing in a forest with trees in the background.