RNN, LSTM

15기 분석 김상휘

목차

1. RNN 이란
2. RNN 종류
3. RNN 작동원리
4. RNN 을 이용한 예시
5. RNN 의 문제
6. LSTM 의 등장과 원리

1. RNN 이란

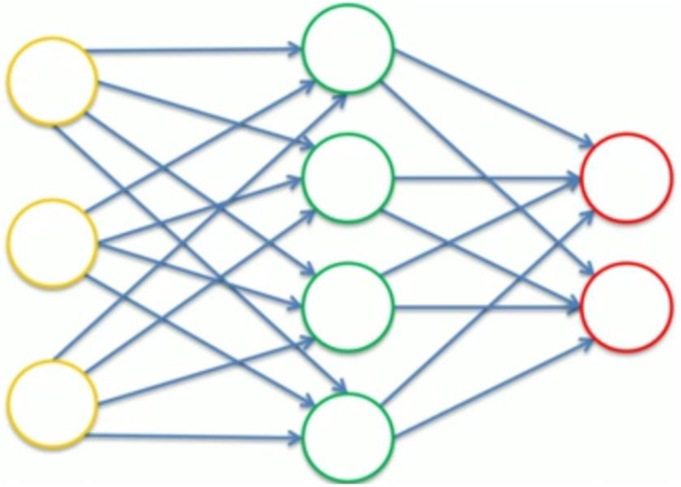
✓ **CNN** -> NN에서 **convolution**을 적용한 개념

✓ **RNN** -> NN에서 **recurrent**를 적용한 개념

* NN : Neural Network (신경망)

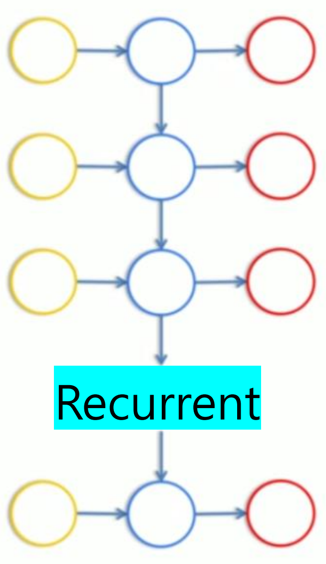
1. RNN 이란

NN
(Neural Network)

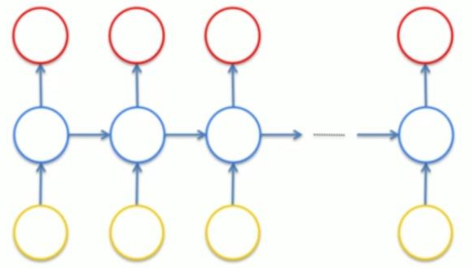


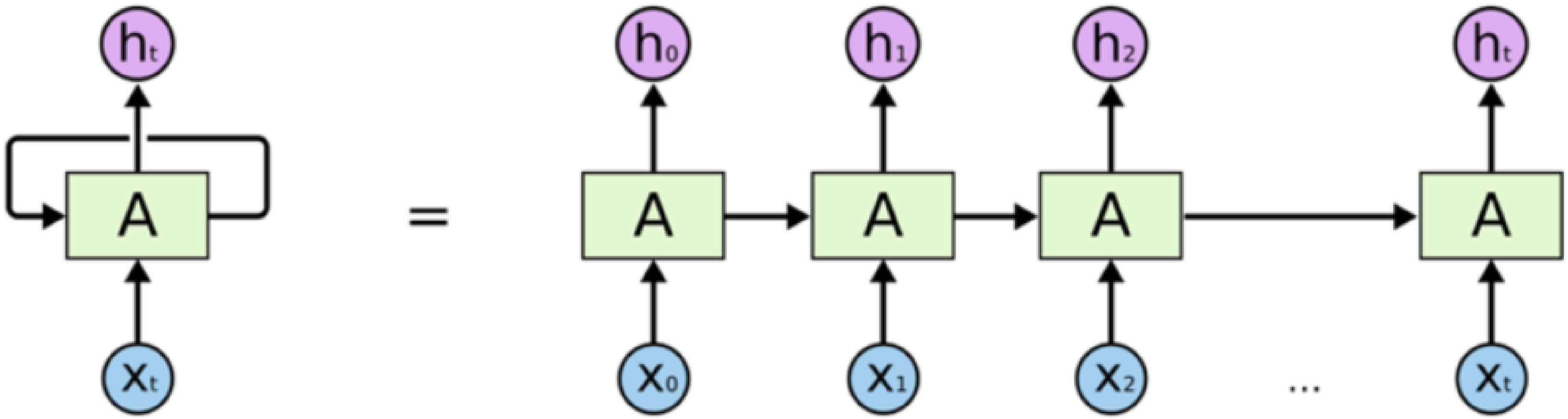
input layer output

RNN
(Recurrent Neural Network)

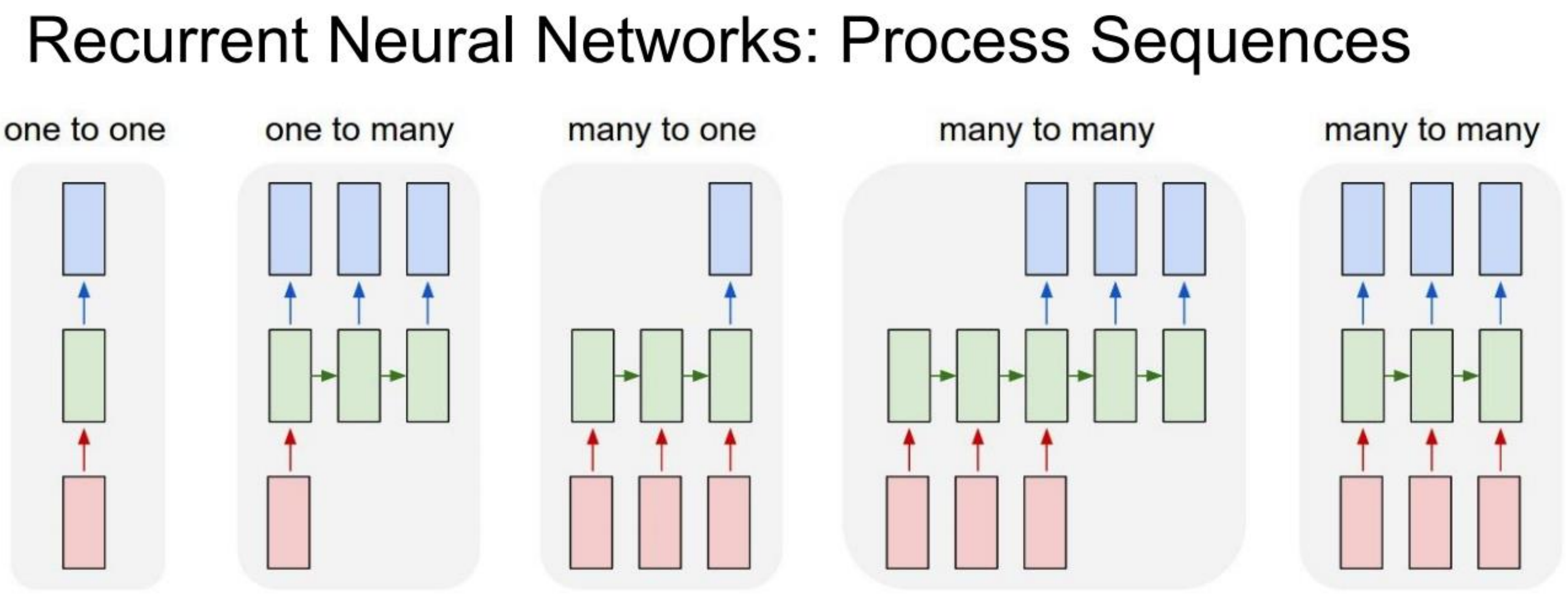


1. RNN 이란





2. RNN 종류

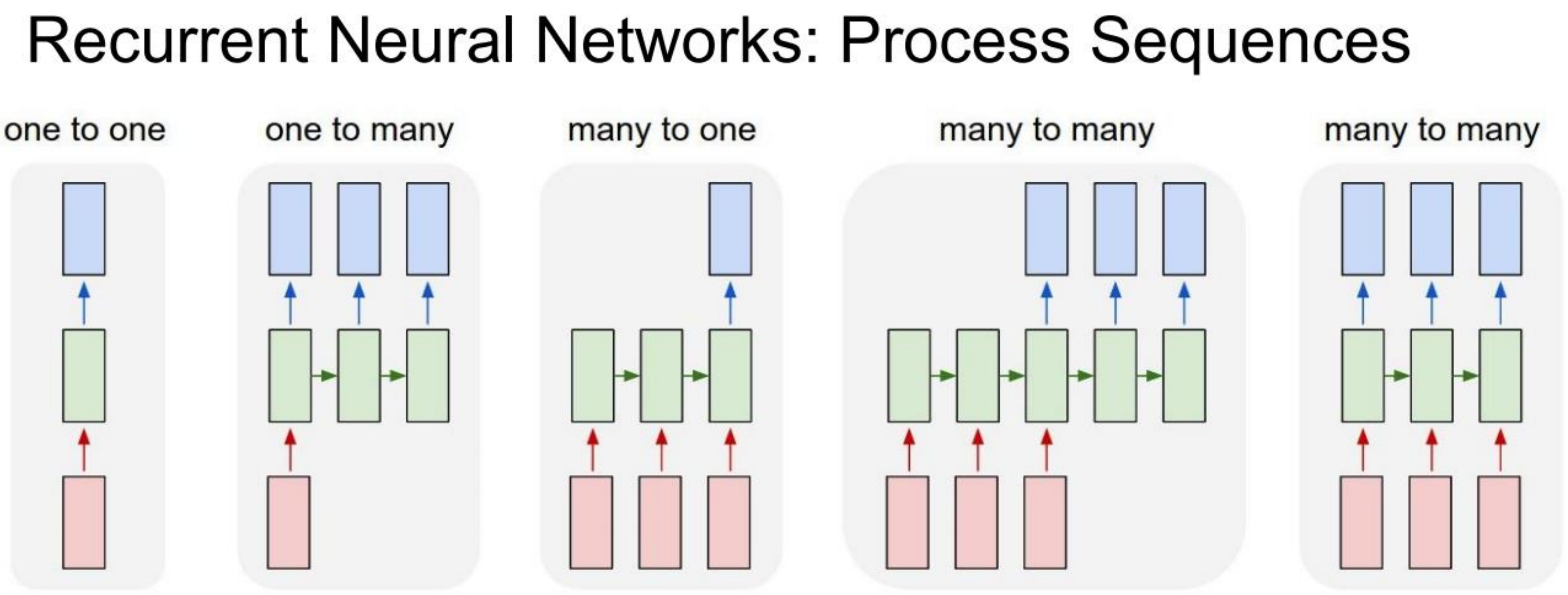


- one to many : Image Captioning (image -> sequence of words)
- many to one : Sentiment Classification (sequence of words -> sentiment)
- many to many : Machine Translation (seq of words -> seq of words)
- many to many : Video Classification on frame level

2. RNN 종류

Image Captioning

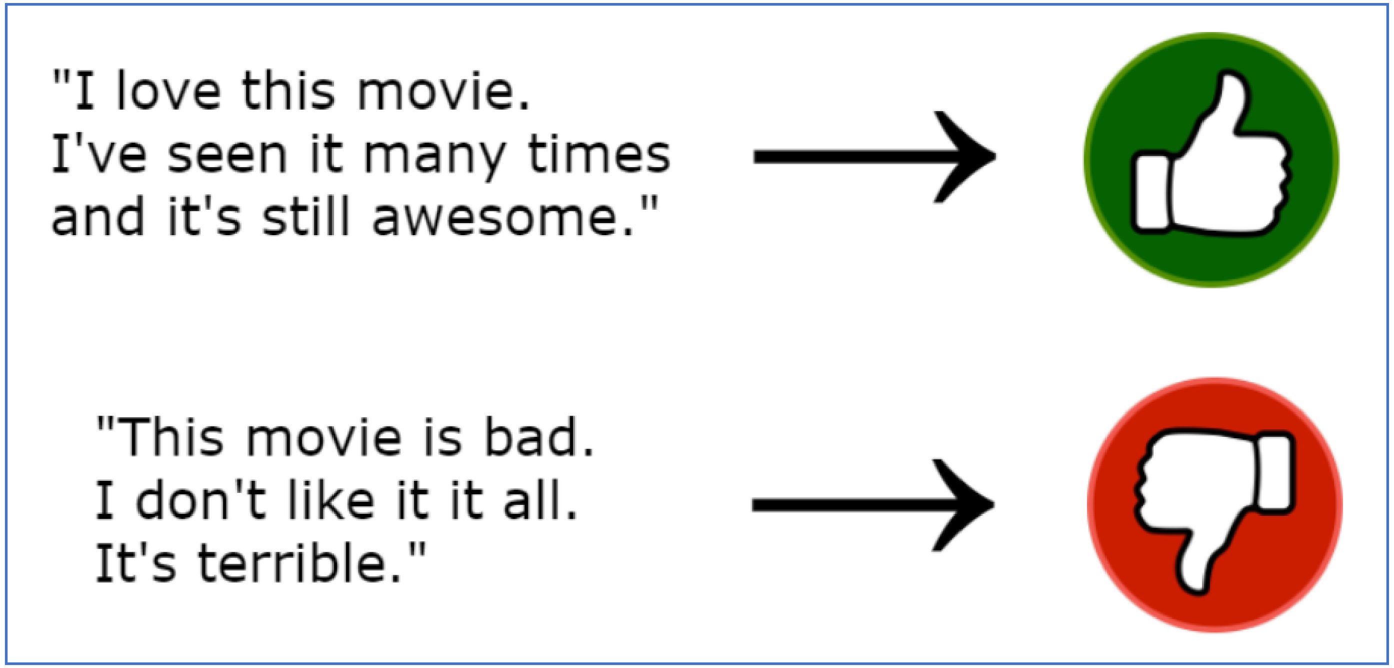
2. RNN 종류



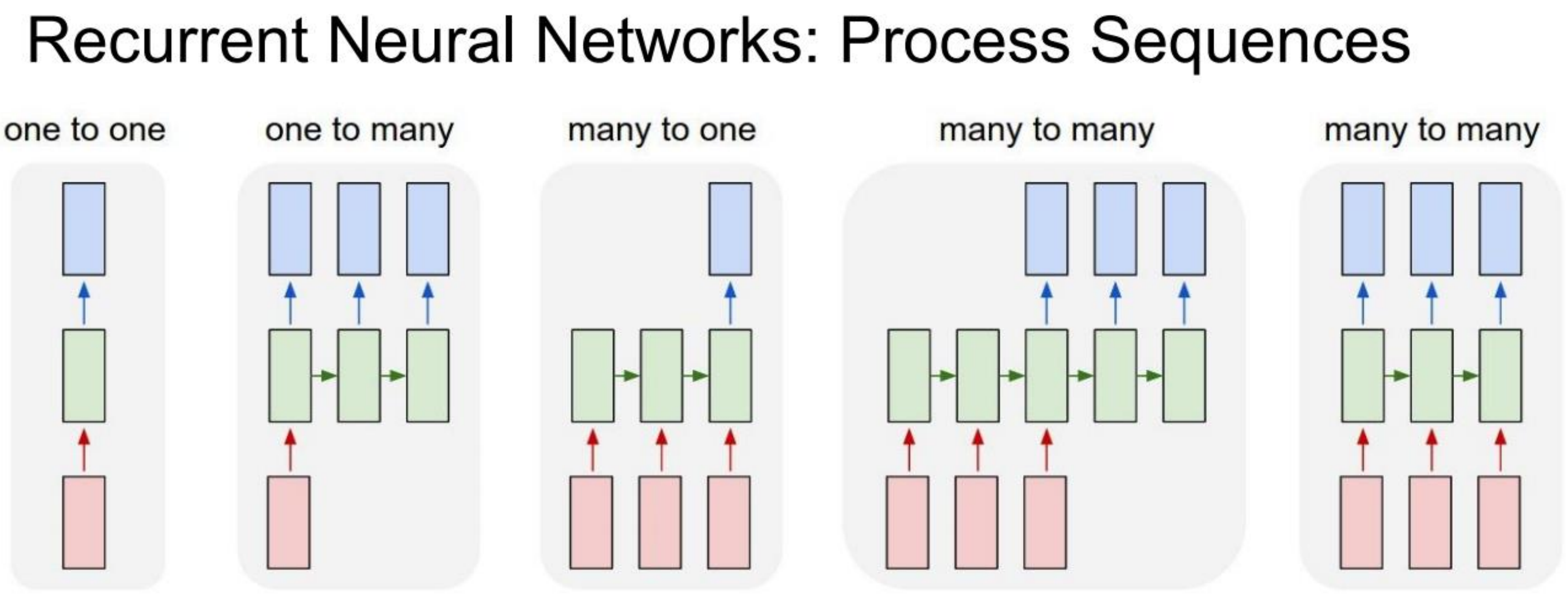
- **one to many** : Image Captioning (image -> sequence of words)
- **many to one** : Sentiment Classification (sequence of words -> sentiment)
- **many to many** : Machine Translation (seq of words -> seq of words)
- **many to many** : Video Classification on frame level

2. RNN 종류

Sentiment Classification



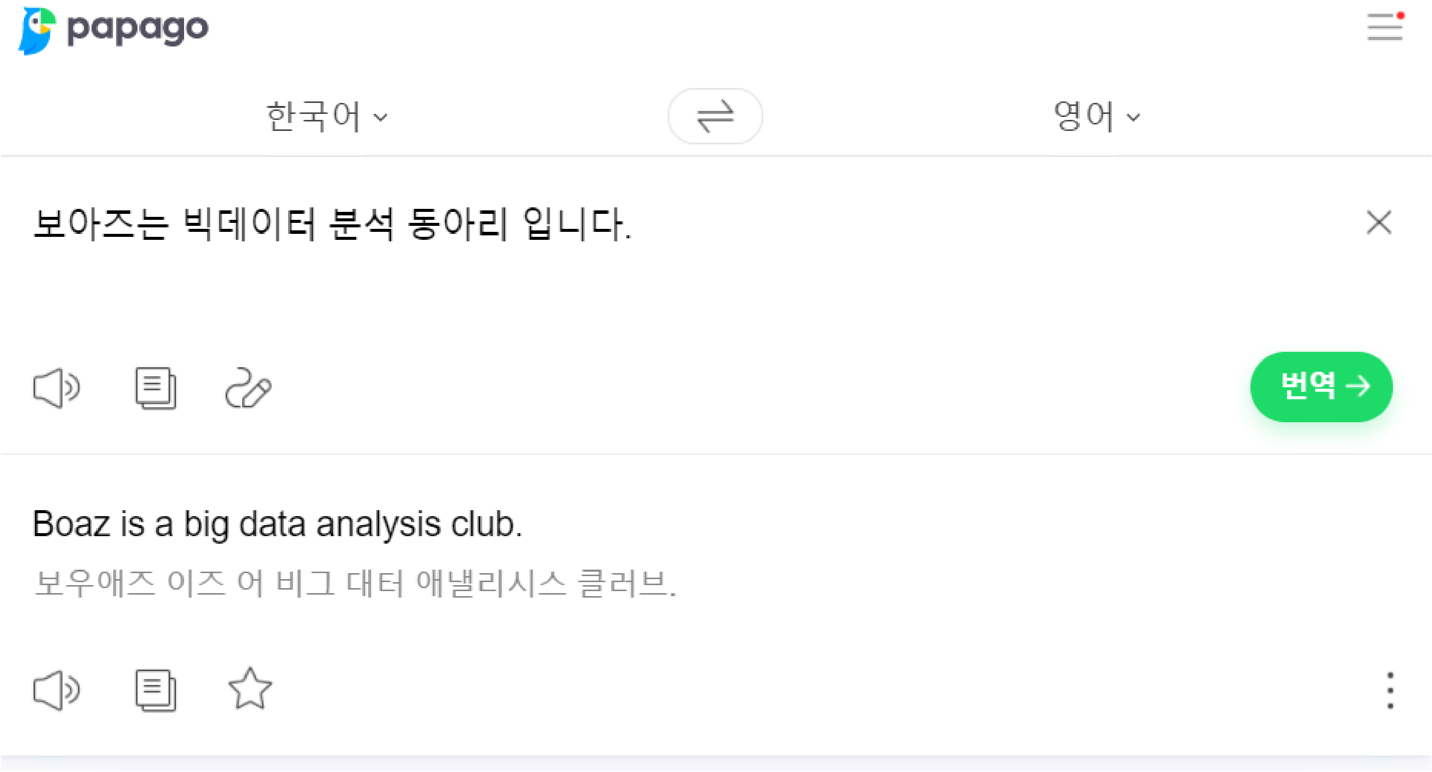
2. RNN 종류



- **one to many** : Image Captioning (image -> sequence of words)
- **many to one** : Sentiment Classification (sequence of words -> sentiment)
- **many to many** : Machine Translation (seq of words -> seq of words)
- **many to many** : Video Classification on frame level

2. RNN 종류

Machine Translation



3. RNN 작동 원리

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

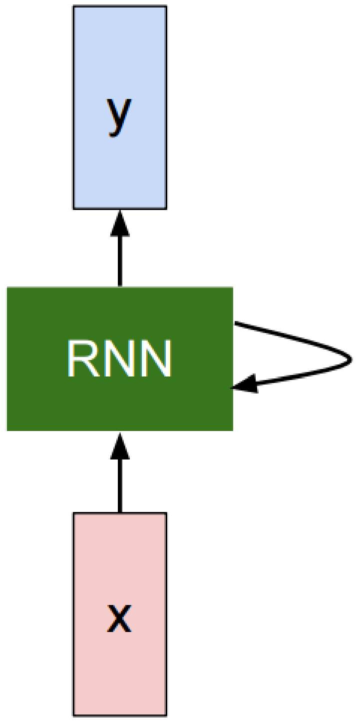
$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state

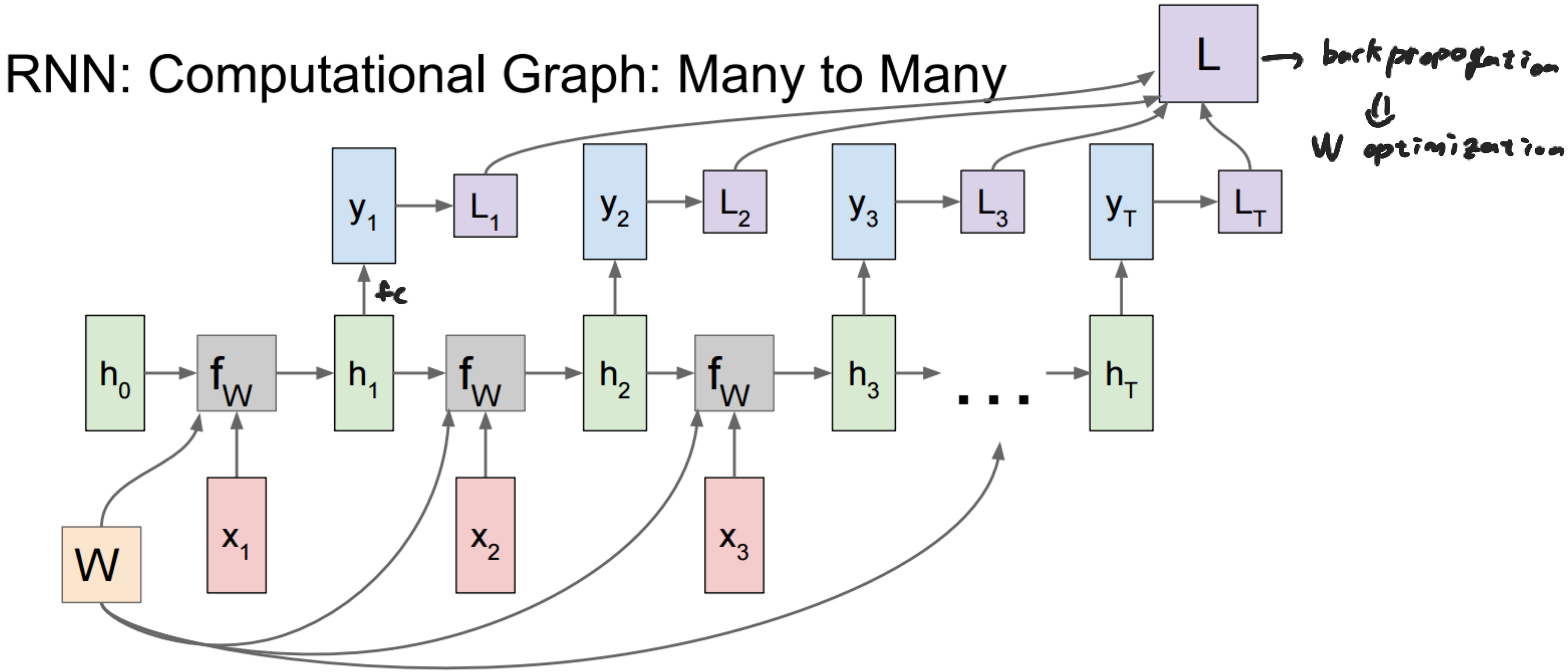
old state

input vector at some time step

some function with parameters W



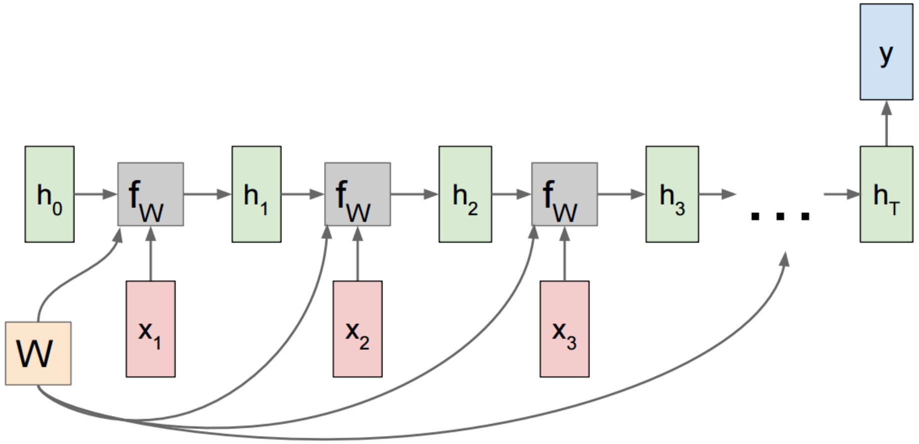
3. RNN 작동 원리



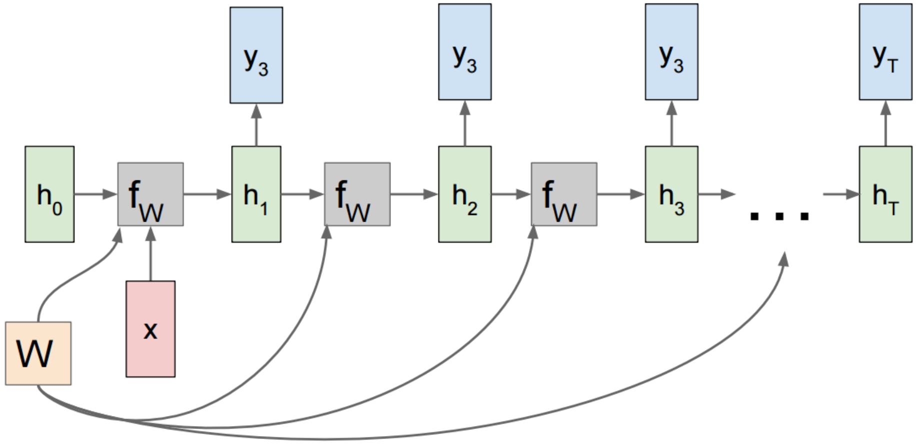
RNN: Computational Graph: Many to Many
L
back propagation
W optimization
y_1
L_1
y_2
L_2
y_3
L_3
y_T
L_T
fc
h_0
f_W
h_1
f_W
h_2
f_W
h_3
...
h_T
W
x_1
x_2
x_3

3. RNN 작동 원리

RNN: Computational Graph: Many to One

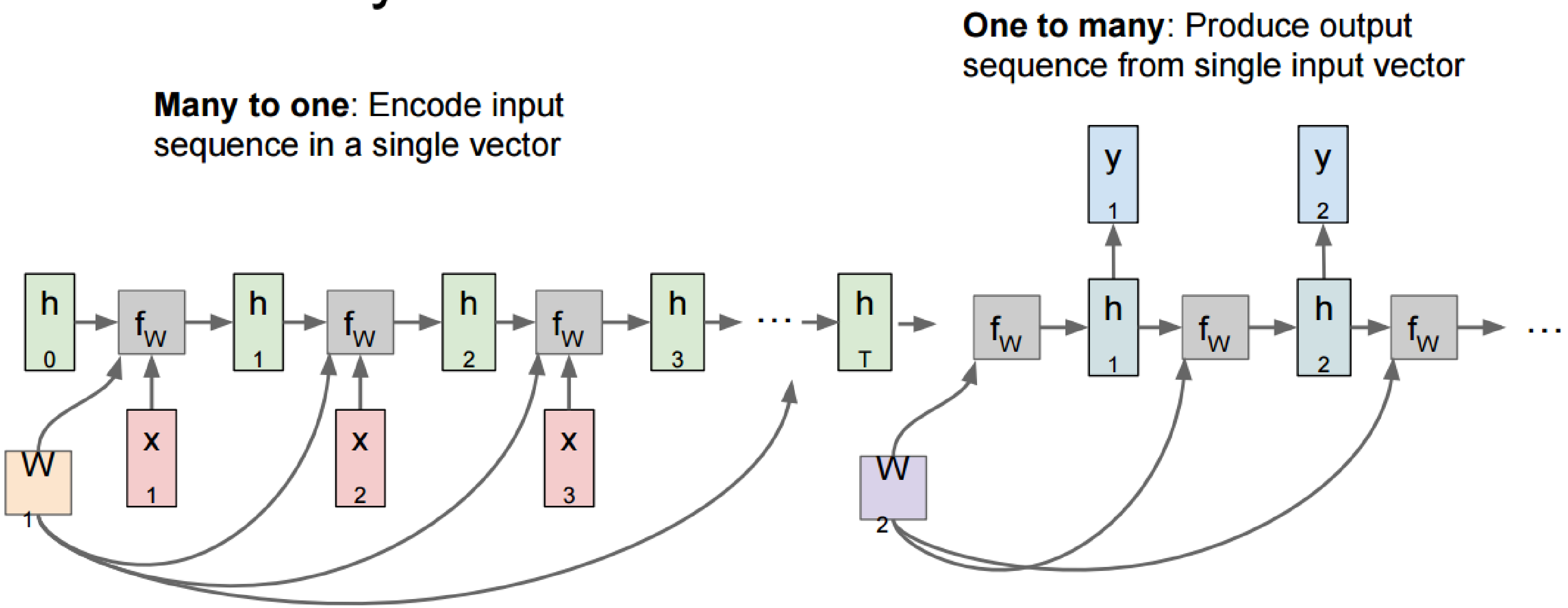


RNN: Computational Graph: One to Many



3. RNN 작동 원리

Sequence to Sequence: Many-to-one + one-to-many



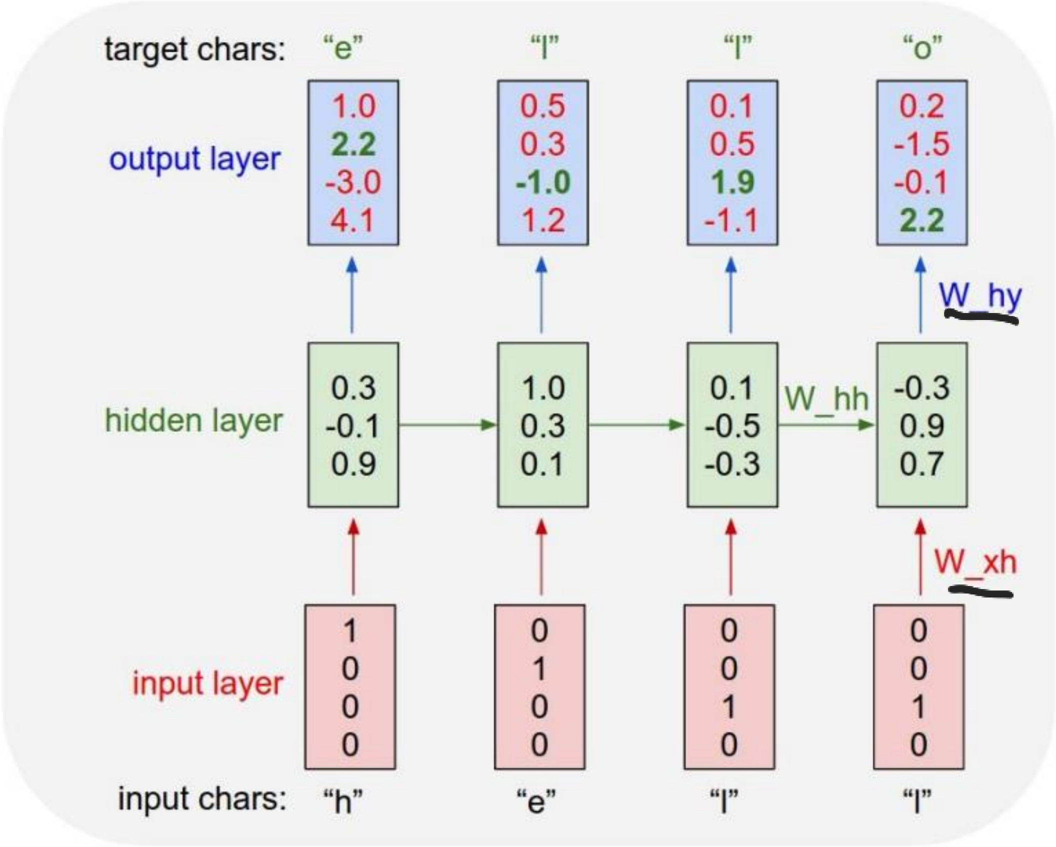
4. RNN 을 이용한 예시

4-1. Language Model

Example:
Character-level
Language Model

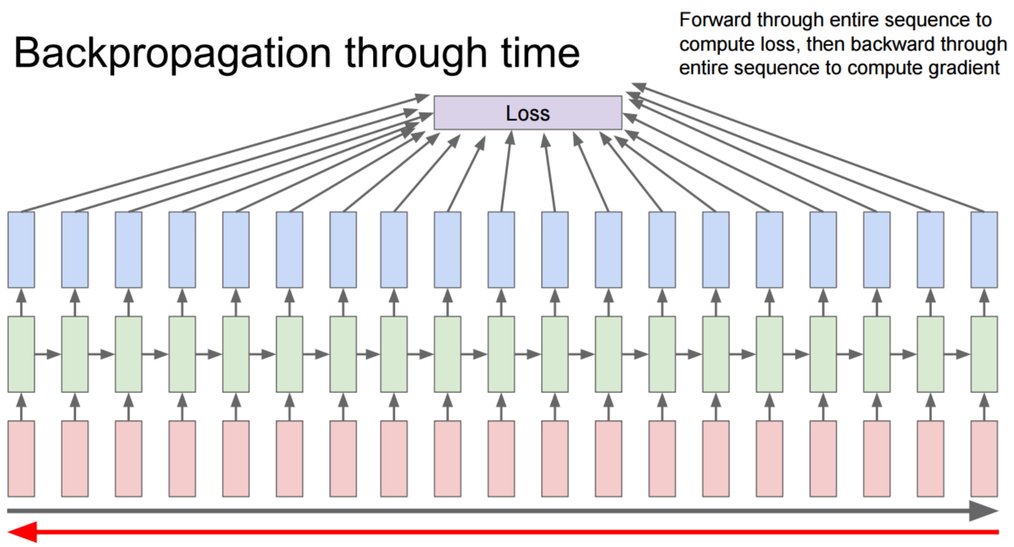
Vocabulary:
[h,e,l,o]

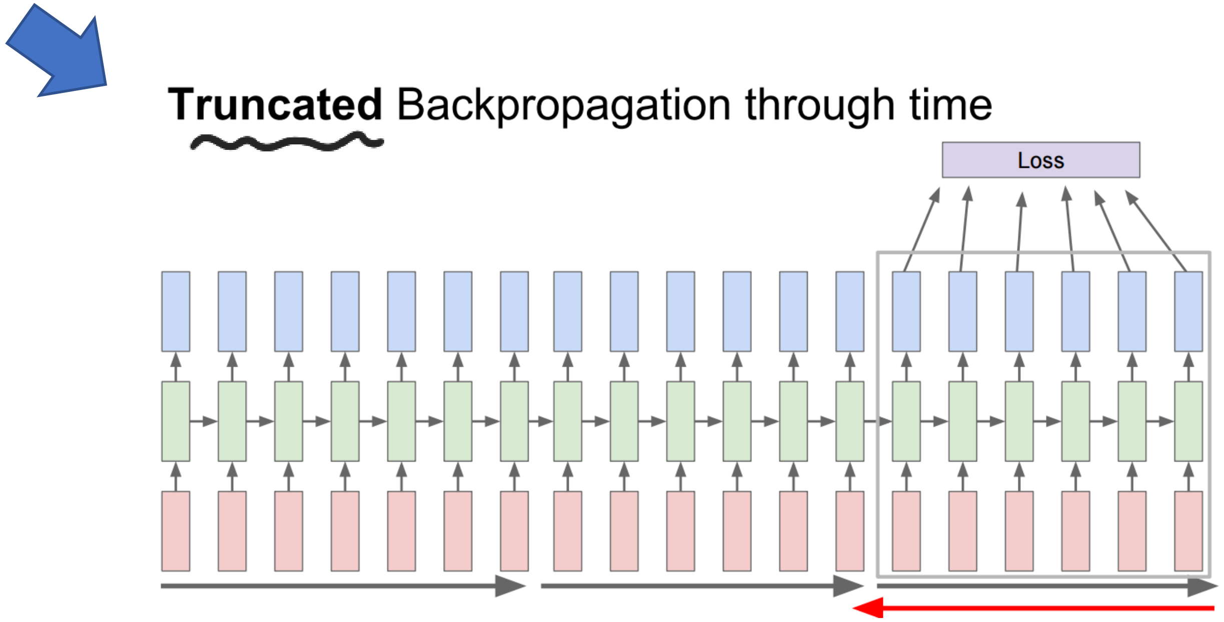
Example training
sequence:
“hello”



4. RNN 을 이용한 예시

4-1. Language Model



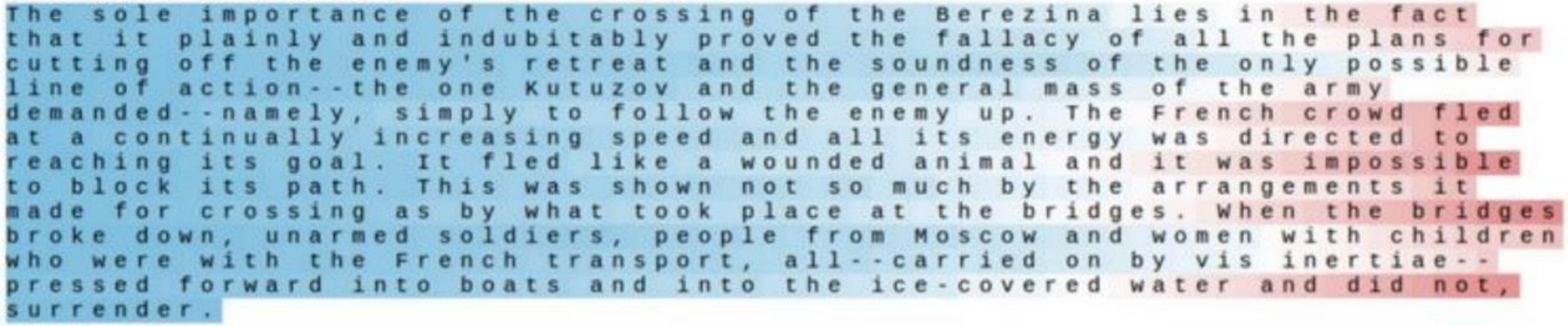


4. RNN 을 이용한 예시 (구조 또한 학습)

4-1. Language Model

Searching for interpretable cells

Cell sensitive to position in line:

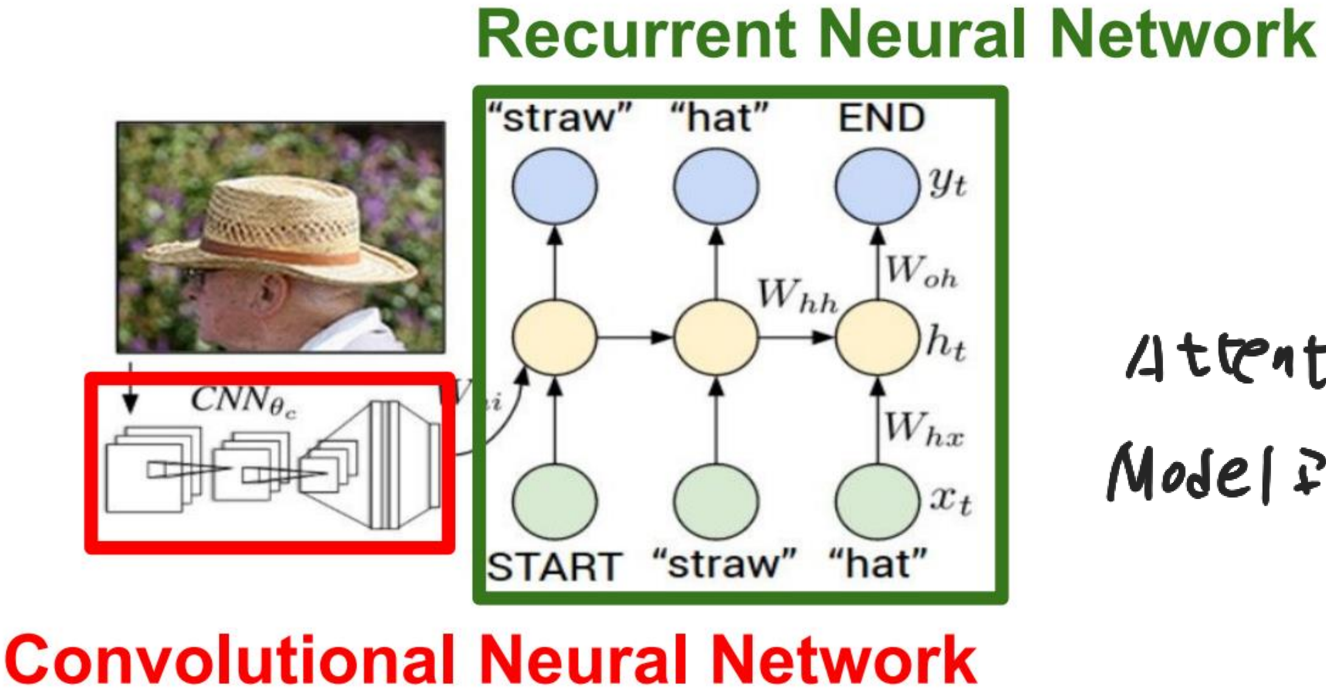


line length tracking cell

Karpathy, Johnson, and Fei-Fei: Visualizing and Understanding Recurrent Networks, ICLR Workshop 2016

4. RNN 을 이용한 예시

4-2. Image Captioning



Attention
Model과 연결

4. RNN 을 이용한 예시

4-2. Image Captioning

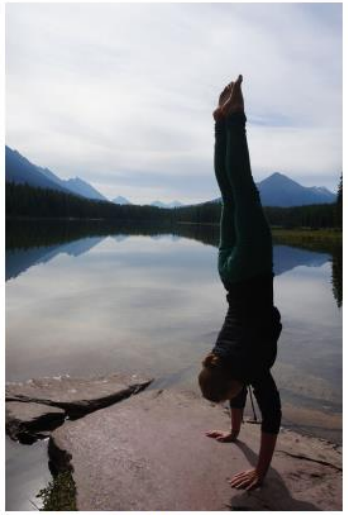
Attention RNN

⇒ 이미지설명 vector, 공간 vector

Image Captioning: Failure Cases

A woman is holding a cat in her hand

A person holding a computer mouse on a desk



A woman standing on a beach holding a surfboard

A bird is perched on a tree branch

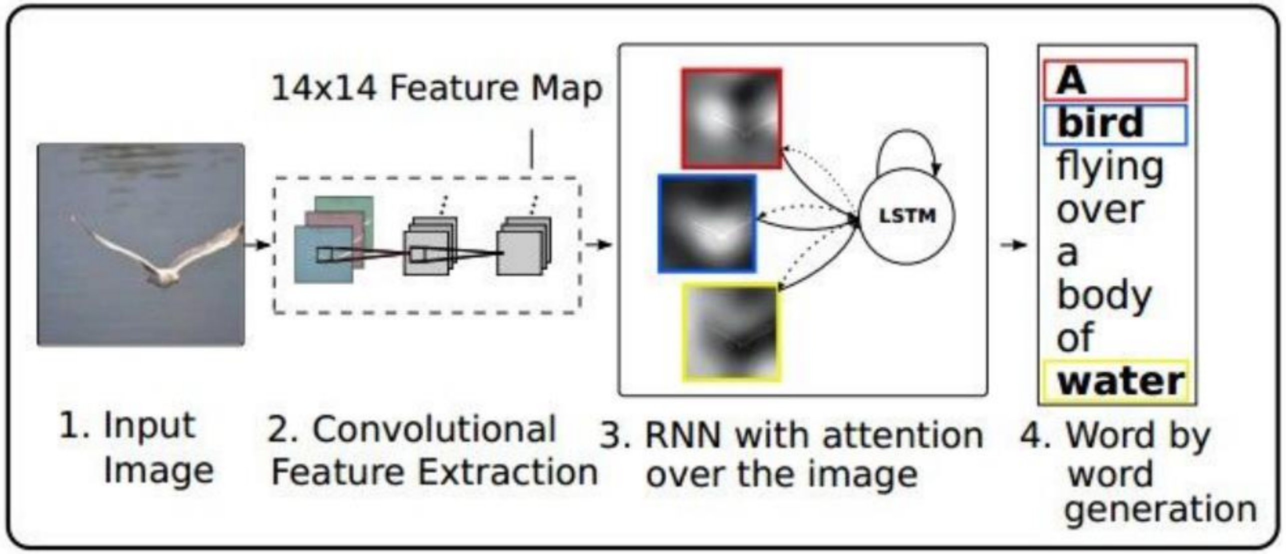
A man in a baseball uniform throwing a ball

4. RNN 을 이용한 예시

4-2. Image Captioning

Image Captioning with Attention

RNN focuses its attention at a different spatial location
when generating each word

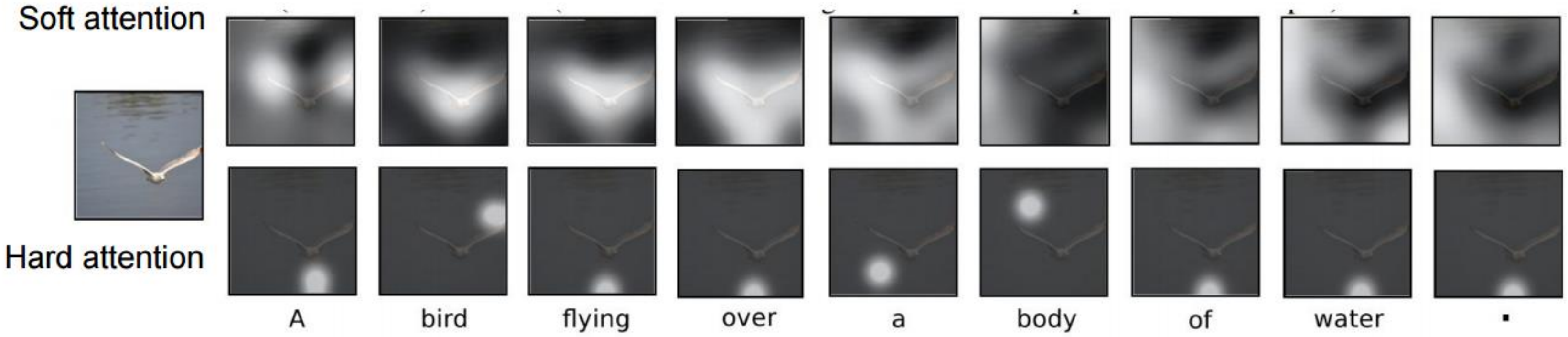


Xu et al, "Show, Attend, and Tell: Neural Image Caption Generation with Visual Attention", ICML 2015

4. RNN 을 이용한 예시

4-2. Image Captioning

Image Captioning with Attention

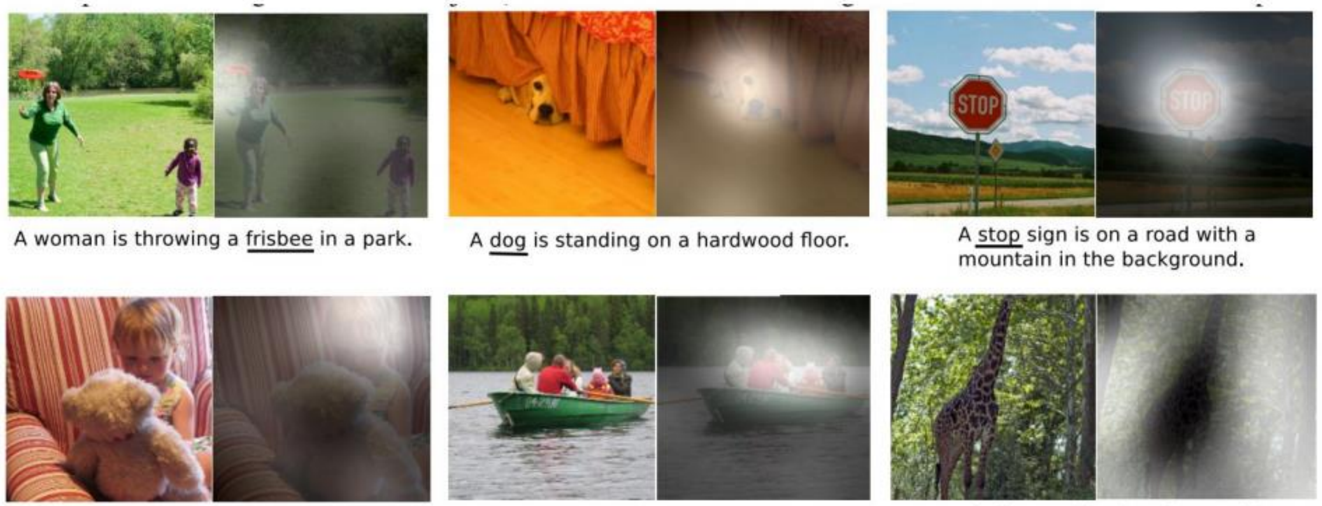


Xu et al, "Show, Attend, and Tell: Neural Image Caption Generation with Visual Attention", ICML 2015

4. RNN 을 이용한 예시

4-2. Image Captioning

Image Captioning with Attention



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.

Xu et al, "Show, Attend, and Tell: Neural Image Caption Generation with Visual Attention", ICML 2015

4. RNN 을 이용한 예시

4-3. Visual Question Answering (VQA)

Visual Question Answering

Q: What endangered animal is featured on the truck?
A: **A bald eagle.**
A: A sparrow.
A: A humming bird.
A: A raven.

Q: Where will the driver go if turning right?
A: **Onto 24 ¾ Rd.**
A: Onto 25 ¾ Rd.
A: Onto 23 ¾ Rd.
A: Onto Main Street.

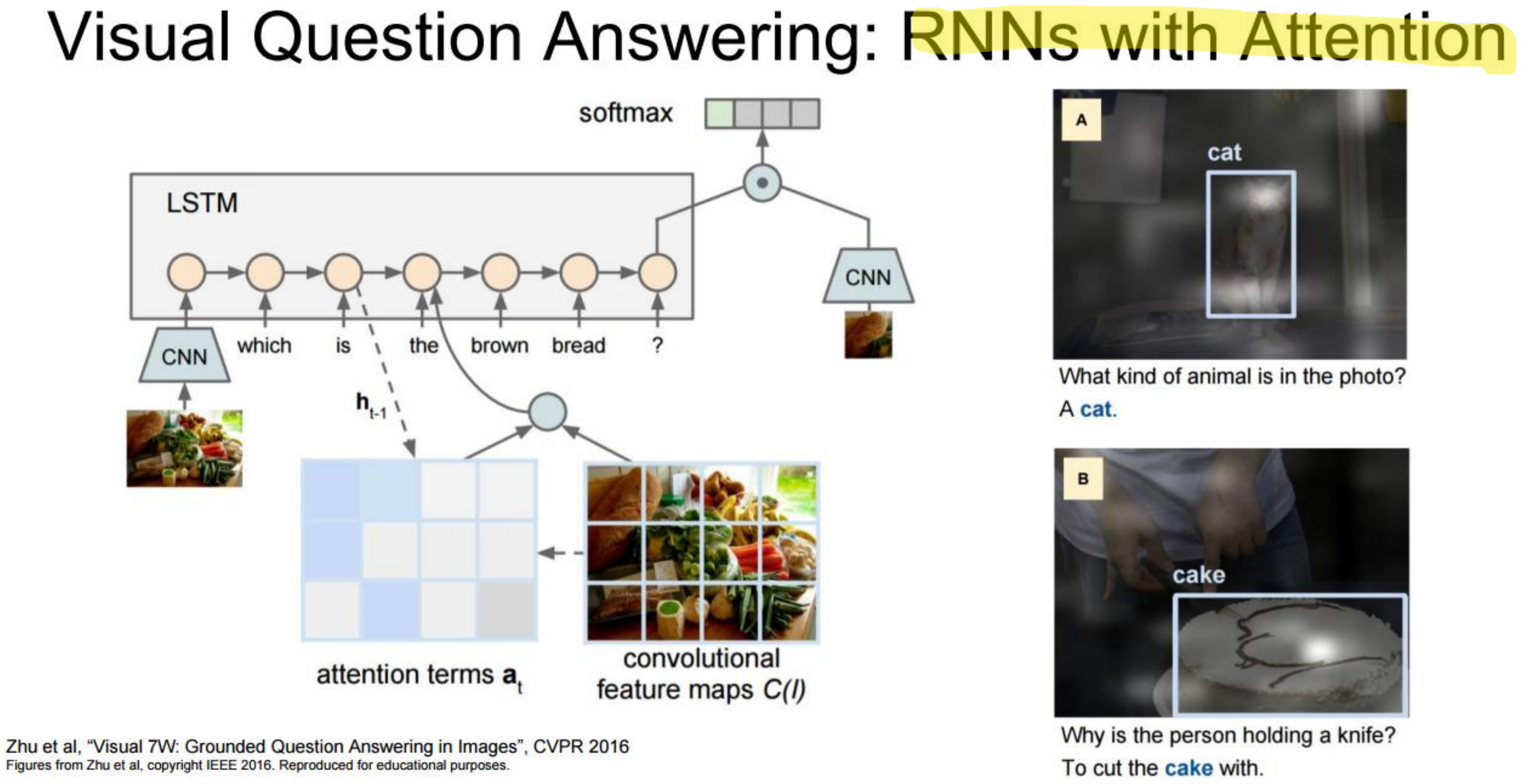
Q: When was the picture taken?
A: **During a wedding.**
A: During a bar mitzvah.
A: During a funeral.
A: During a Sunday church service

Q: Who is under the umbrella?
A: **Two women.**
A: A child.
A: An old man.
A: A husband and a wife.

Agrawal et al, “VQA: Visual Question Answering”, ICCV 2015
Zhu et al, “Visual 7W: Grounded Question Answering in Images”, CVPR 2016

4. RNN 을 이용한 예시

4-3. Visual Question Answering (VQA)



5. RNN 의 문제

Multilayer RNNs

$$\boxed{h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$h \in \mathbb{R}^n$ $\quad W^l \; [n \times 2n]$

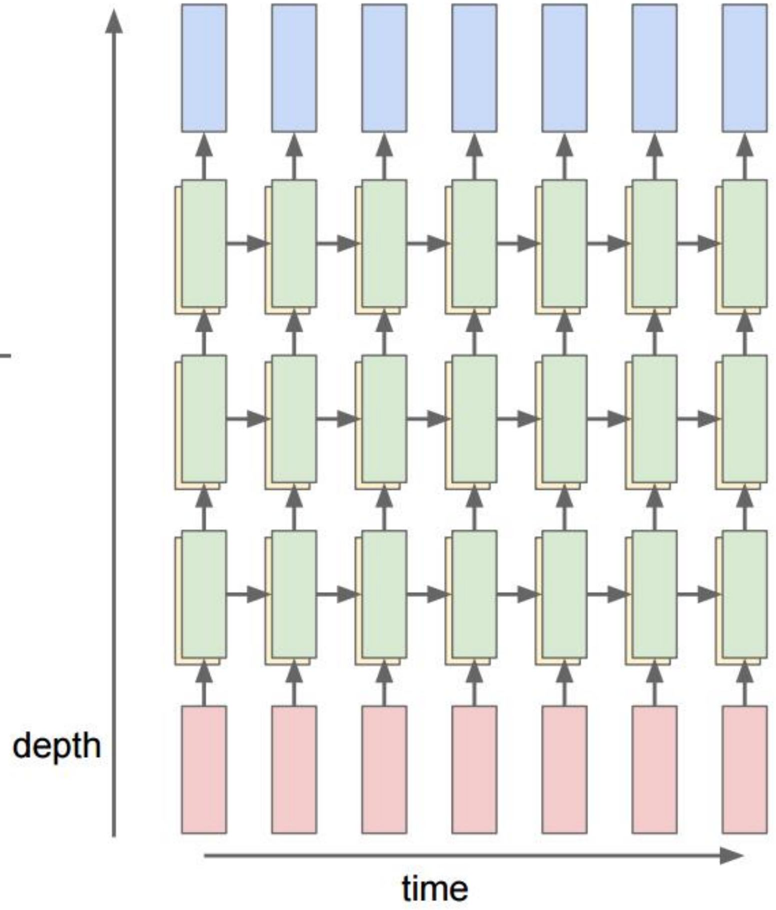
LSTM:

$W^l \; [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

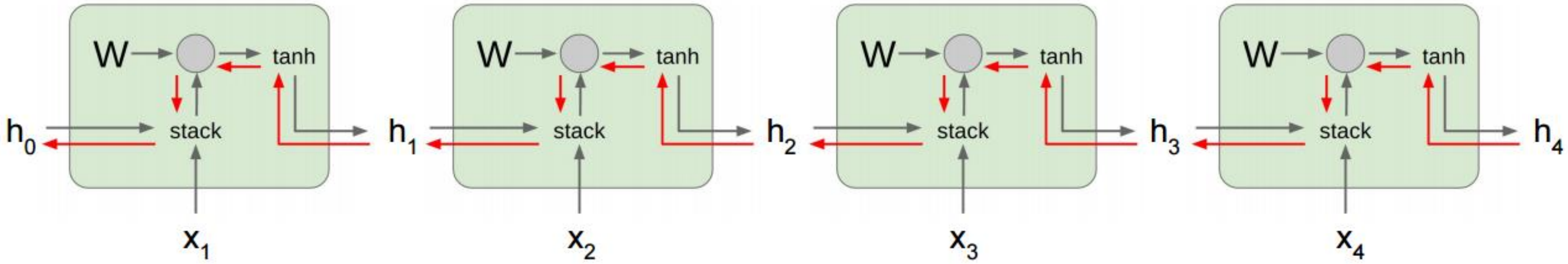
$$h_t^l = o \odot \tanh(c_t^l)$$



5. RNN 의 문제

Vanilla RNN Gradient Flow

Bengio et al, "Learning long-term dependencies with gradient descent is difficult", IEEE Transactions on Neural Networks, 1994
Pascanu et al, "On the difficulty of training recurrent neural networks", ICML 2013



Computing gradient of h_0 involves many factors of W (and repeated tanh)

Largest singular value > 1:
Exploding gradients

→ **Gradient clipping**: Scale gradient if its norm is too big

```
grad_norm = np.sum(grad * grad)
if grad_norm > threshold:
  grad *= (threshold / grad_norm)
```

Largest singular value < 1:
Vanishing gradients

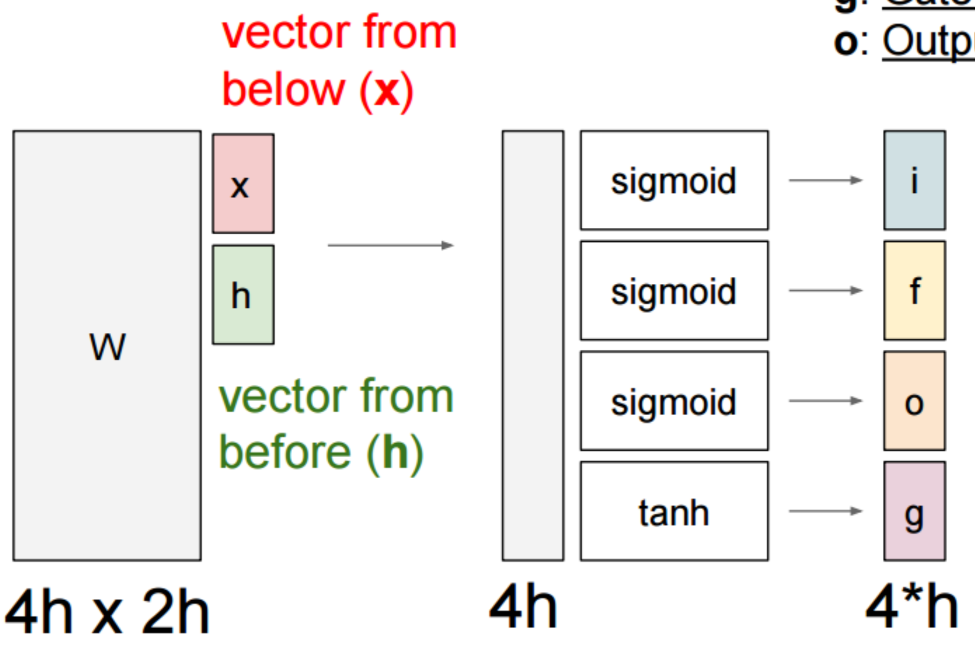
↳ 자체적 해결 불가

6. LSTM 의 등장과 원리

Long Short Term Memory (LSTM)

[Hochreiter et al., 1997]

f: Forget gate, Whether to erase cell
i: Input gate, whether to write to cell
g: Gate gate (?), How much to write to cell
o: Output gate, How much to reveal cell



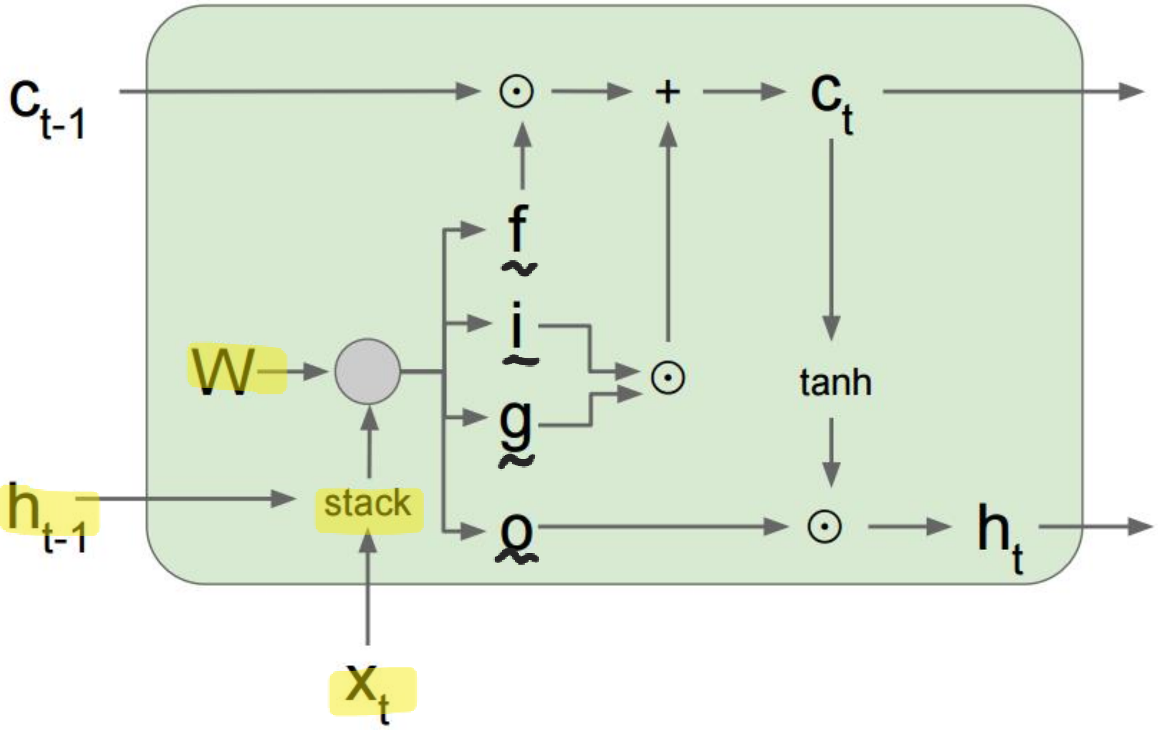
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

6. LSTM 의 등장과 원리

Long Short Term Memory (LSTM)
[Hochreiter et al., 1997]



$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

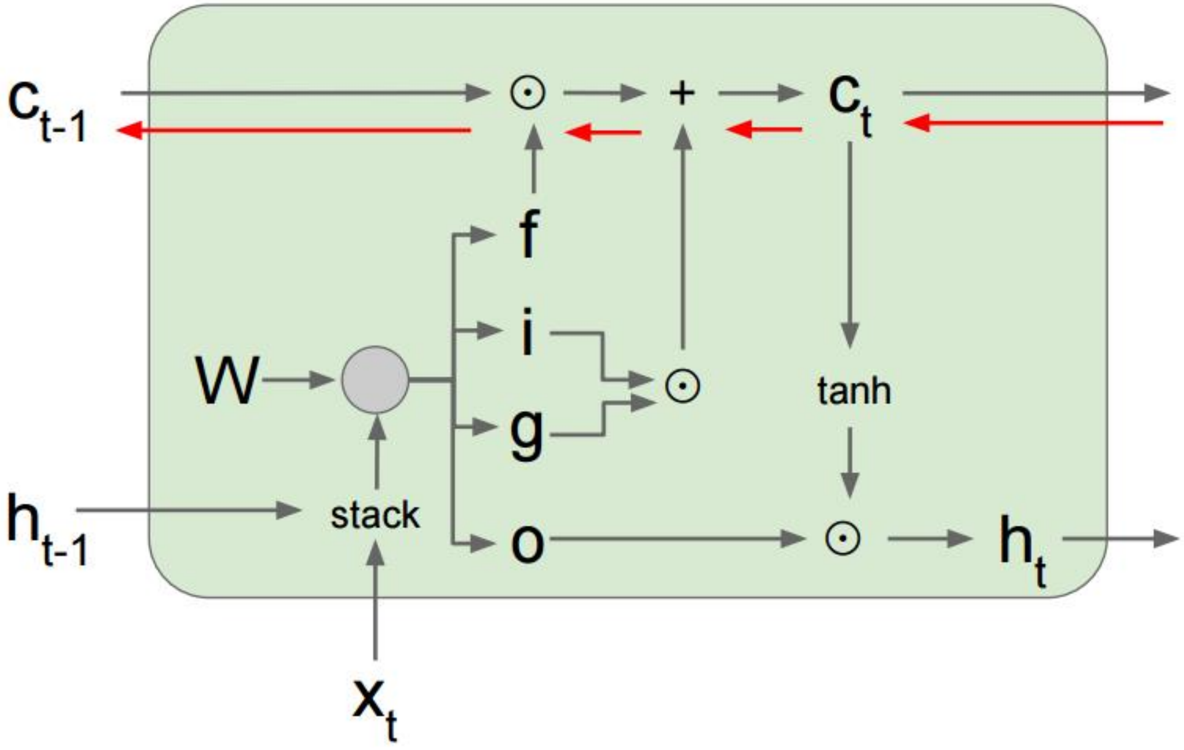
$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

6. LSTM 의 등장과 원리

Long Short Term Memory (LSTM): Gradient Flow

[Hochreiter et al., 1997]



Backpropagation from c_t to c_{t-1} only elementwise multiplication by f, no matrix multiply by W

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

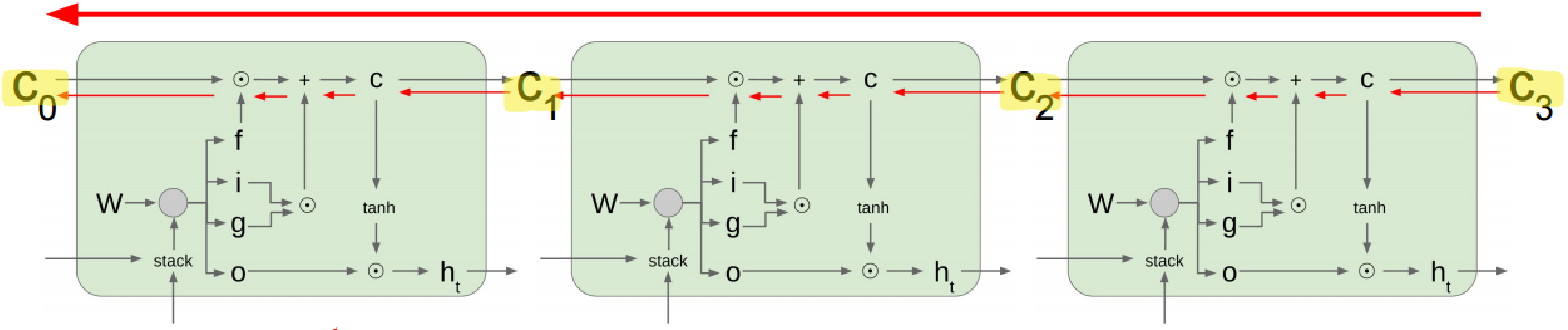
$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$

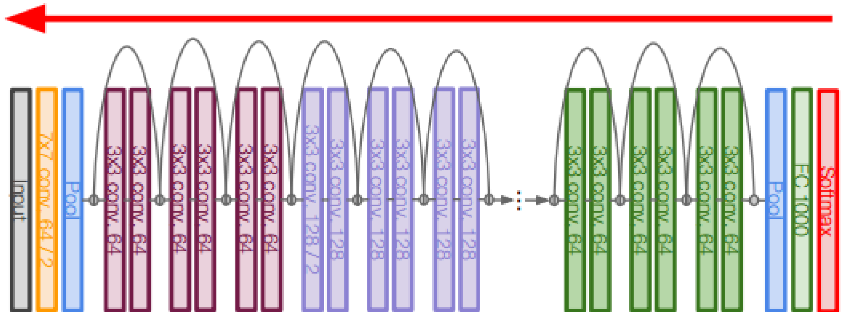
6. LSTM 의 등장과 원리

Long Short Term Memory (LSTM): Gradient Flow
[Hochreiter et al., 1997]

Uninterrupted gradient flow!



Similar to ResNet!



In between:
Highway Networks

$$g = T(x, W_T)$$
$$y = g \odot H(x, W_H) + (1 - g) \odot x$$

Srivastava et al, "Highway Networks", ICML DL Workshop 2015