Lec. 15-1 Long Short Term Memory

cell-gate, input-gate, output-gate

Will Cover

1. Introduction
   - Long-term dependency problem (gradient V/E)
   - introduction to 3 gates
2. LSTM forward computation flow
   - what is calculated at each gate
   - summarized behavior table
3. LSTM BPTT flow
   - what to update ?
   - how cell-state is safe from BPTT ?
4. Quick LSTM example (Tensorflow)
   - Tensorflow Time-Series Tutorial

2. LSTM forward computation flow

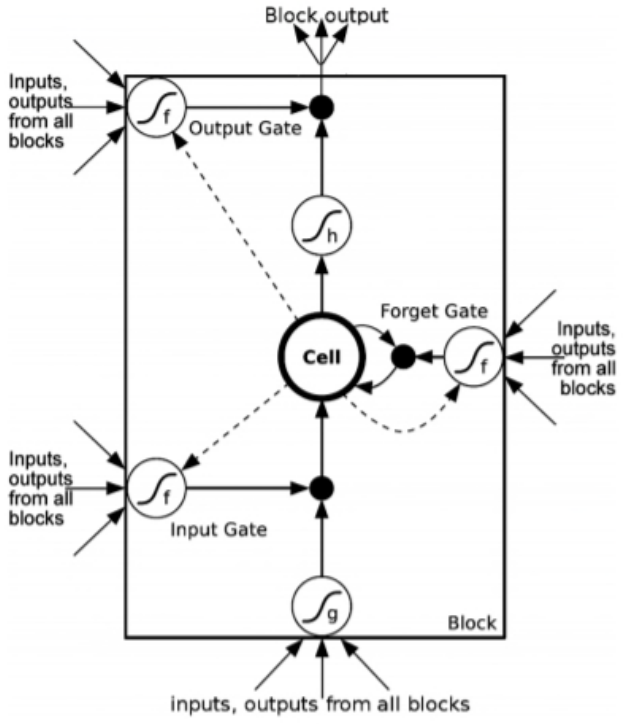
Summarized behavior table

앞서 본 각 게이트의 역할을 생각하며 살펴보면

f, i 의 조합에 따라 해당 셀의 동작을 다음과 같이 해석 할 수 있다.

Long-Term Short Term Memory

Replace each single unit in an RNN by a memory block -

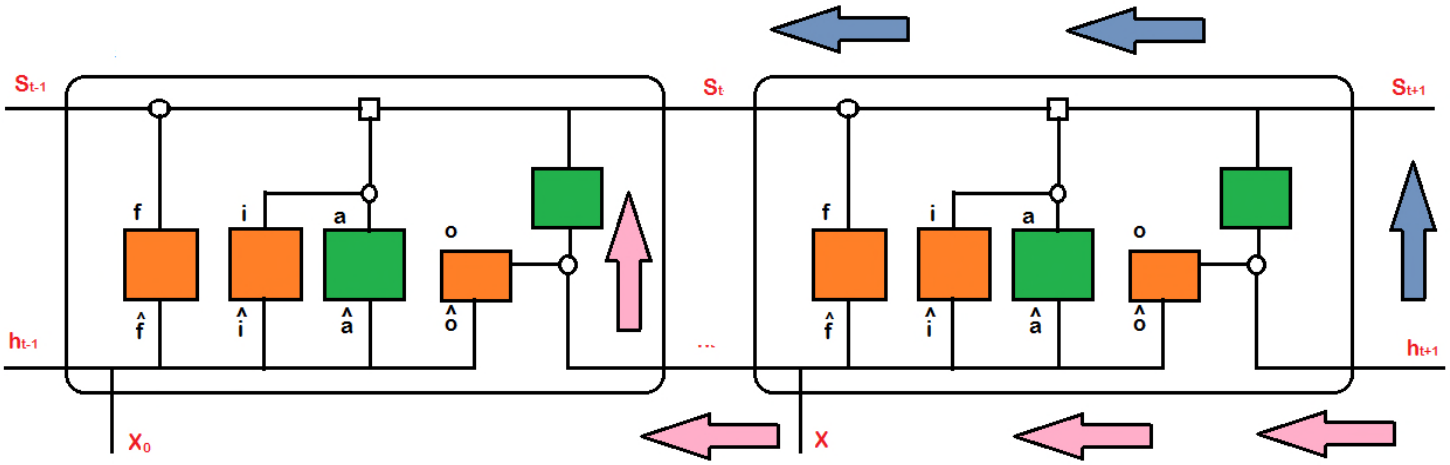


$$c_{t+1} = c_t \cdot \text{forget gate} + \text{new input} \cdot \text{input gate}$$

- $i = 0, f = 1 \Rightarrow$ remember the previous value
- $i = 1, f = 1 \Rightarrow$ add to the previous value
- $i = 0, f = 0 \Rightarrow$ erase the value
- $i = 1, f = 0 \Rightarrow$ overwrite the value

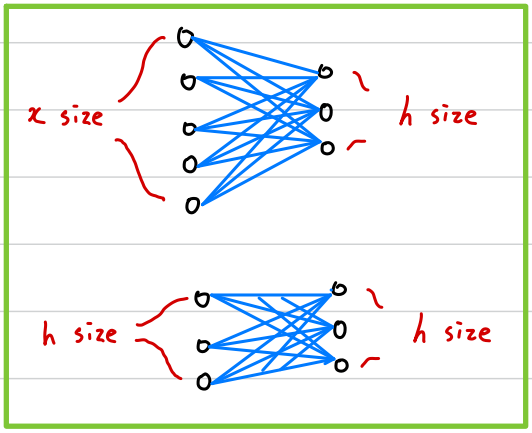
Setting $i = 0, f = 1$ gives the reasonable "default" behavior of just remembering things.

3. LSTM BPTT flow



1. Loss → h_{t+1} → S_{t+1} → S_t
2. Loss → h_{t+1} → o_{t+1} → h_t → S_t

우리가 "갱신" 해야 하는 것.



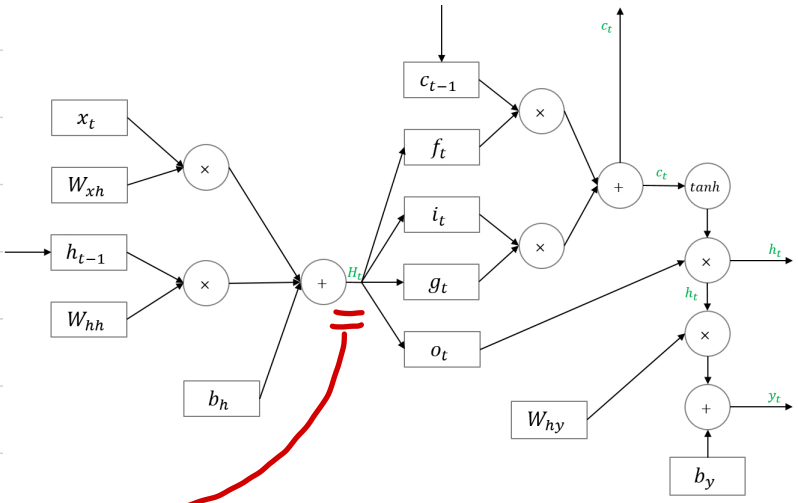
← 꼴 4 세트

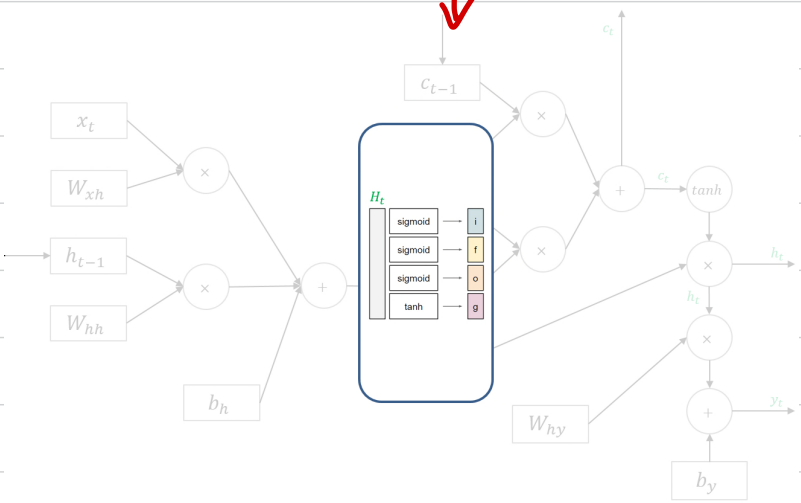
f, i, 2, o

$$\underset{1\times 5}{(\ x\ )} \times \underset{5\times 3}{\left(W_x^f\right)} = \underset{1\times 3}{(\ h\ )}$$

$$\underset{1\times 3}{(\ h\ )} \times \underset{3\times 3}{\left(W_h^f\right)} = \underset{1\times 3}{(\ h\ )}$$

이 가중치 행렬은 이런식으로도 생각할 수 있는데,





W_{xh} : 5 x 12

W_{hh} : 3 x 12 로 생긴

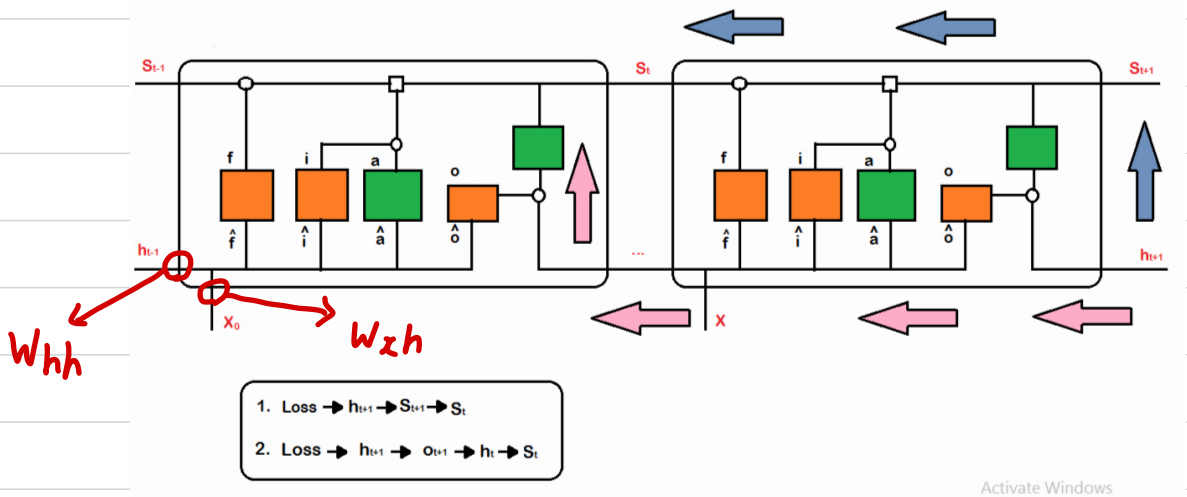
두행렬로 부터 (1, 12)

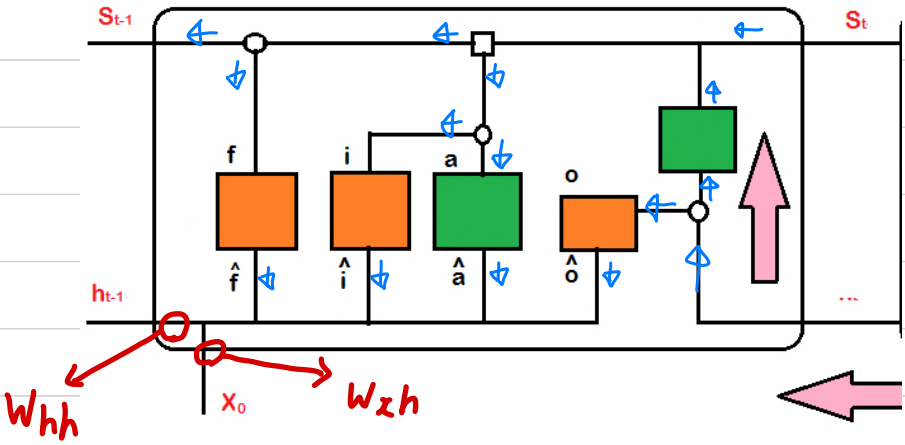
결과 행렬을 얻고,

f	i	a	o

이렇게 나눠 적용.

이러한 관점에서 볼때,





1. Loss → h_{t+1} → S_{t+1} → S_t
2. Loss → h_{t+1} → o_{t+1} → h_t → S_t

역전파 방향이 위처럼 되긴한데, $\tanh'$, sig' 를 되돌아가며 발생하는 기울기 소실 위험은 여전히 존재한다.

그럼 tanh, sigmoid 를 역으로 통과하기 전, 그 입구까지 전달되는 기울기를 살펴보면

$\overline{s^t}$ 의 역할이 중요함을 알 수 있다.

만약 $\overline{s^{t-1}} = \alpha \cdot \overline{s^t}$ 꼴의 관계라면

RNN과 마찬가지로 소실 / 폭발 이 그대로 일어날 확률이 높다.

s^{t-1} 에서 s^t 가 계산되는 순전파식을 살펴보면,

$$s^t = f^t \otimes s^{t-1} + i^t \otimes g^t$$

$\frac{d}{ds^{t-1}}$

$$\frac{ds^t}{ds^{t-1}} = \frac{d}{ds^{t-1}}\left(f^t \otimes s^{t-1}\right) + \frac{d}{ds^{t-1}}\left(i^t \otimes g^t\right)$$

0

↓

핵심은 이 녀석인데,

"아다마르 곱이라" 라는 말만 있고 이게 그래서 왜 기울기 위험이 없는지 도저히 모르겠어요 ....

$$\frac{\partial c_t}{\partial c_{t-1}} = \frac{\partial}{\partial c_{t-1}}[c_{t-1} \otimes f_t \oplus \tilde{c}_t \otimes i_t]$$

$$= \frac{\partial}{\partial c_{t-1}}[c_{t-1} \otimes f_t] + \frac{\partial}{\partial c_{t-1}}[\tilde{c}_t \otimes i_t]$$

$$= \frac{\partial f_t}{\partial c_{t-1}} \cdot c_{t-1} + \frac{\partial c_{t-1}}{\partial c_{t-1}} \cdot f_t + \frac{\partial i_t}{\partial c_{t-1}} \cdot \tilde{c}_t + \frac{\partial \tilde{c}_t}{\partial c_{t-1}} \cdot i_t$$

↳ 찾은 식인데 왜 + 로 풀리는 지 설명 가능하신 분 BBQ 올리브 후라이드 깊트 바로 전달합니다.

Long-Term Short Term Memory

- In each step, we have a vector of memory cells **c**, a vector of hidden units **h**, and vectors of input, output, and forget gates **i**, **o**, and **f**.
- There's a full set of connections from all the inputs and hiddens to the input and all of the gates:

$$\begin{pmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \mathbf{g}_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \mathbf{W} \begin{pmatrix} \mathbf{y}_t \\ \mathbf{h}_{t-1} \end{pmatrix}$$

$$\mathbf{c}_t = \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \mathbf{g}_t$$

$$\mathbf{h}_t = \mathbf{o}_t \circ \tanh(\mathbf{c}_t)$$

- Exercise: show that if $\mathbf{f}_{t+1} = 1$, $\mathbf{i}_{t+1} = 0$, and $\mathbf{o}_t = 0$, the gradients for the memory cell get passed through unmodified, i.e.

$$\overline{\mathbf{c}_t} = \overline{\mathbf{c}_{t+1}}.$$

↳ 이런 얘기도 있긴 합니다.

그러나 특정 성격을 만족하는 데이터에 대해서만 $\overline{C_t} = \overline{C_{t+1}}$ 임을 설명할 뿐

Cell-State = free pass BP 를 설명할 수는 없네요...

4. Quick LSTM example (Tensorflow)

https://tykimos.github.io/2017/04/09/RNN_Layer_Talk/

살펴볼것 : Units, input shape, return sequence, return states, stateful