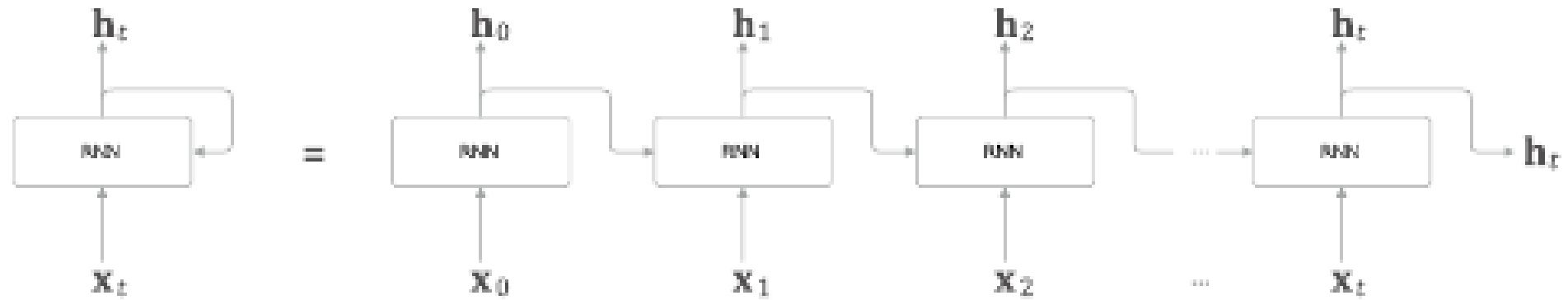


6. Gate add RNN

곰탱이
wkdnffla3@gmail.com

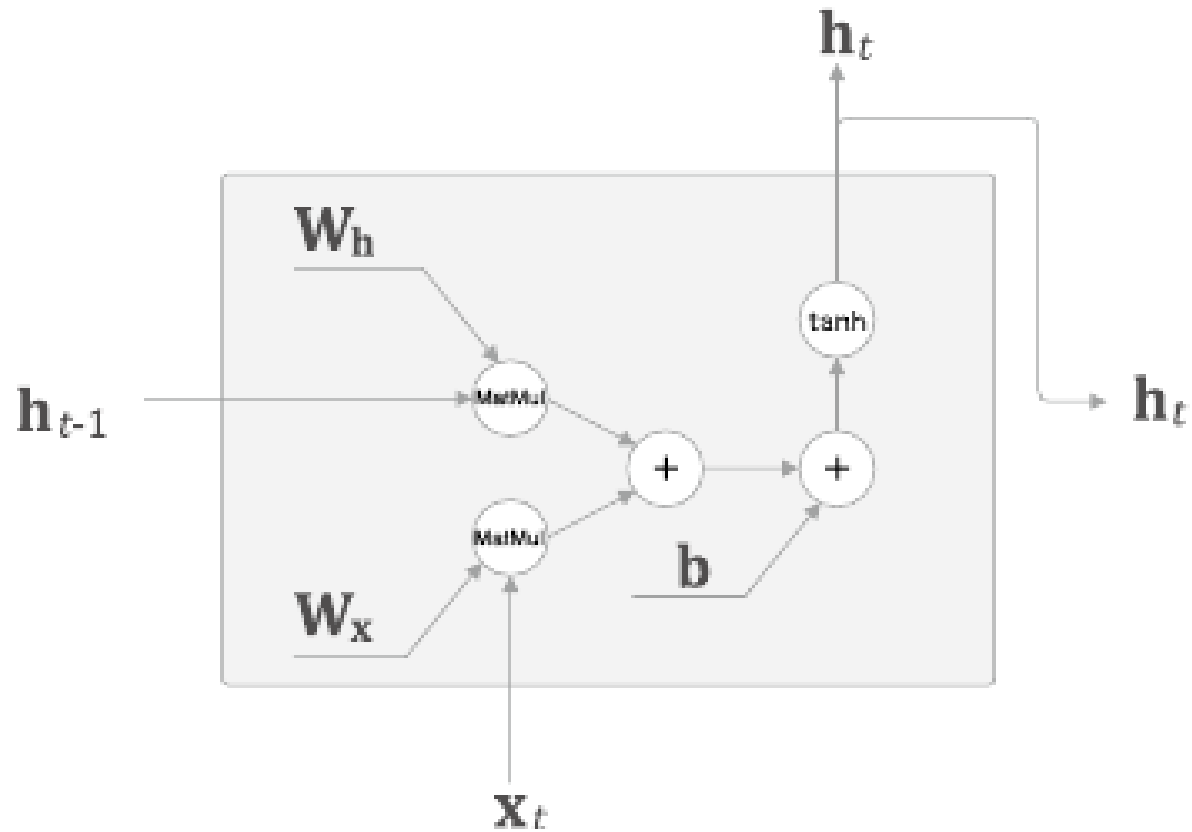
1. RNN 복습

그림 6-1 RNN 계층: 순환을 펼치기 전과 후



1. RNN 복습

그림 6-2 RNN 계층의 계산 그래프(MatMul 노드는 행렬 곱을 나타냄)



1. RNN의 문제점

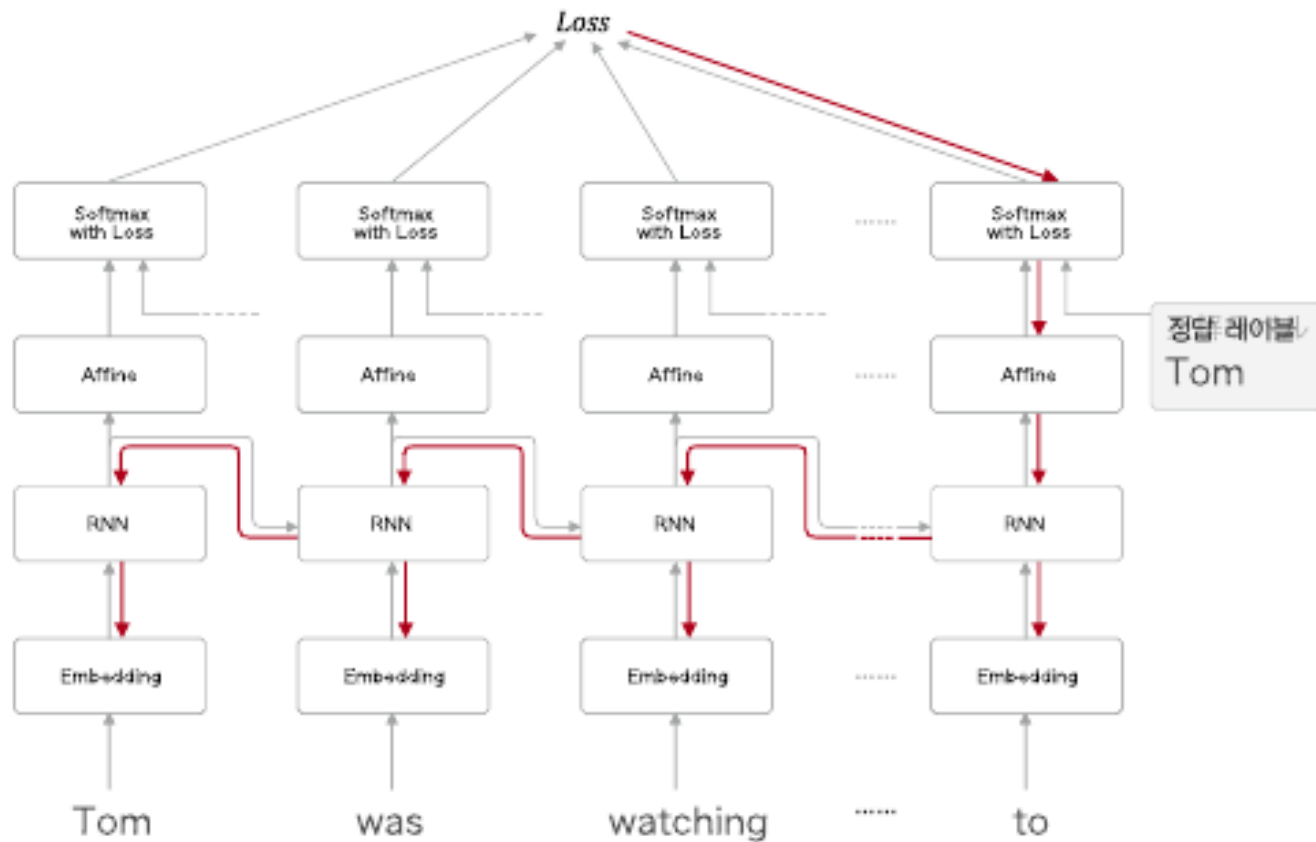
- 언어 모델은 주어진 단어들을 기초로 다음에 출현할 단어를 예측 하는 일을 합니다.

그림 6-3 “?”에 들어갈 단어는?: (어느 정도의) 장기 기억이 필요한 문제의 예

Tom was watching TV in his room. Mary came into the room. Mary said hi to ?

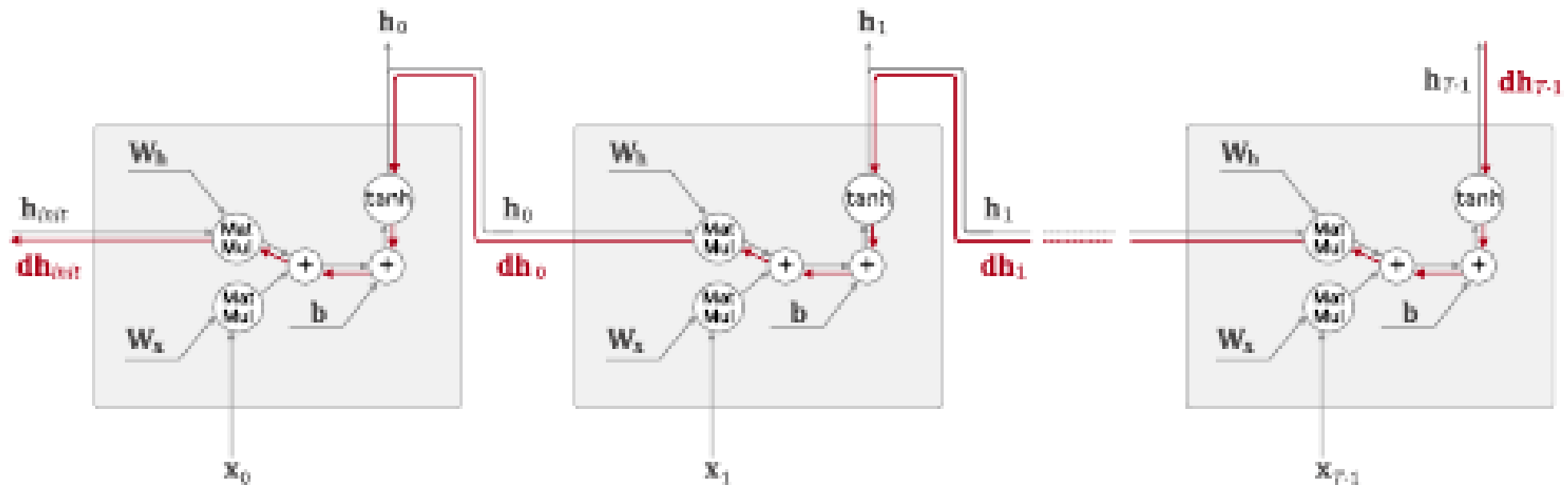
1. RNN의 문제점

그림 6-4 정답 레이블이 "Tom"임을 학습할 때의 기울기 흐름



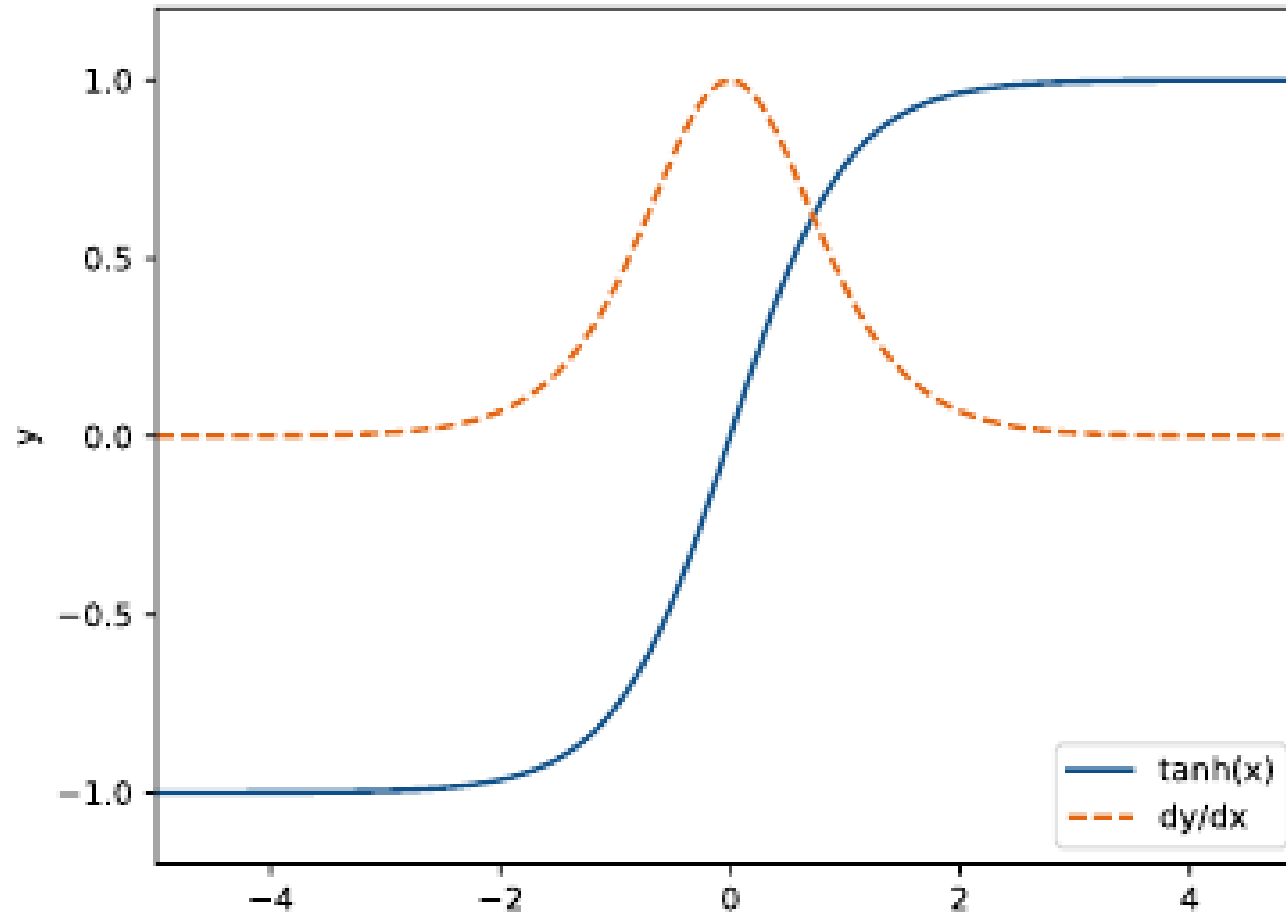
1. RNN의 문제점

그림 6-5 RNN 계층에서 시간 방향으로의 기울기 전파



1. RNN의 문제점 - 기울기 소실

그림 6-6 $y = \tanh(x)$ 의 그래프(점선은 미분)



1. RNN의 문제점 - 기울기 폭발

그림 6-7 RNN 계층의 행렬 곱에만 주목했을 때의 역전파의 기울기

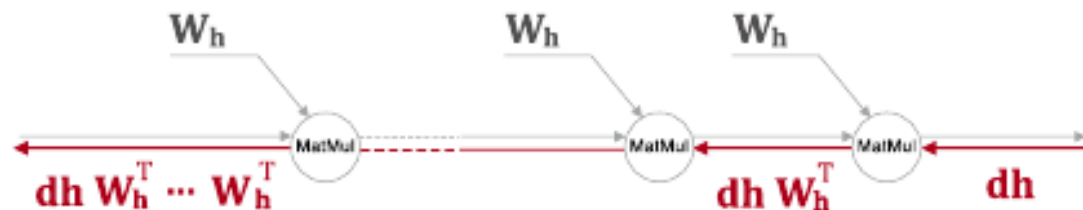
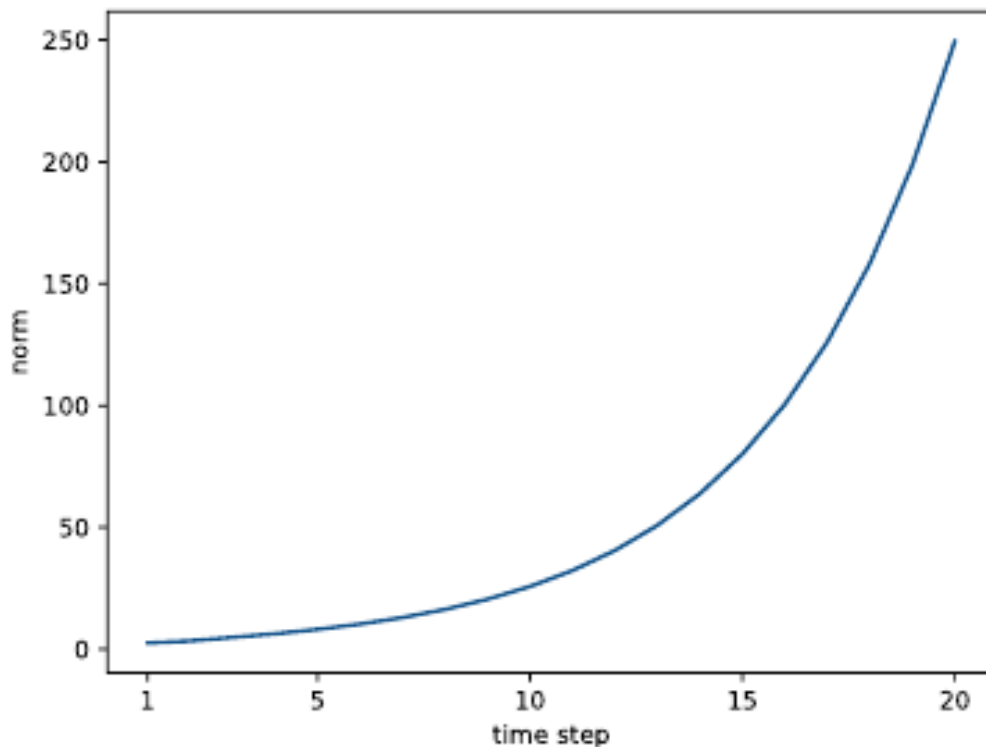


그림 6-8 기울기 dh 는 시간 크기에 비례하여 지수적으로 증가한다.



1. RNN의 문제점 – 기울기 폭발 대책

if $\|\hat{\mathbf{g}}\| \geq \text{threshold}$:

$$\hat{\mathbf{g}} = \frac{\text{threshold}}{\|\hat{\mathbf{g}}\|} \hat{\mathbf{g}}$$

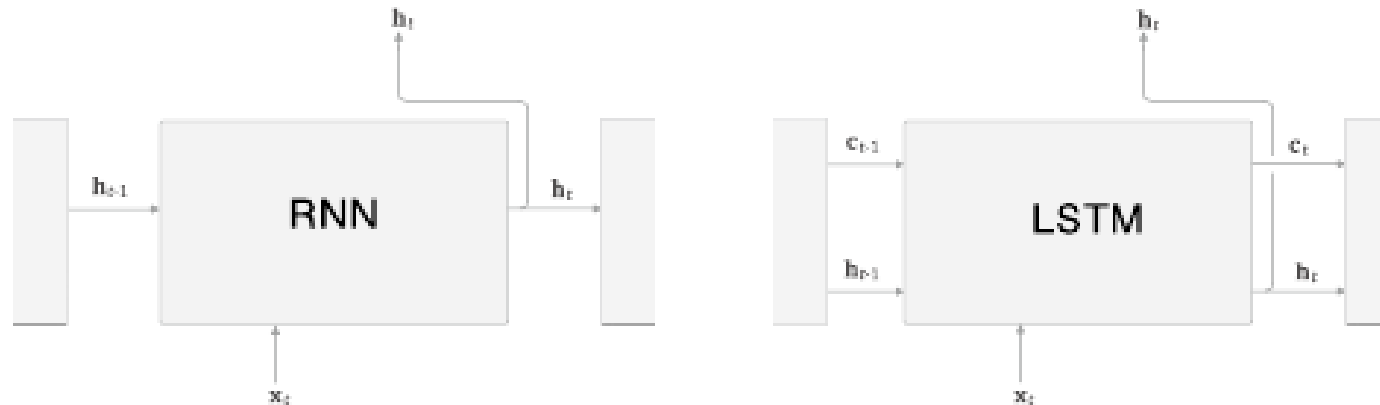
$\hat{\mathbf{g}}$: 가중치에 대한 모든 기울기를 총합

1. RNN의 문제점 – 기울기 소실 대책

- 기울기 소실 문제를 해결하려면 RNN 계층의 구조를 바꿔야 한다.
- 게이트를 추가시키는 방법이 존재.
- LSTM과 GRU가 있다.

2. LSTM

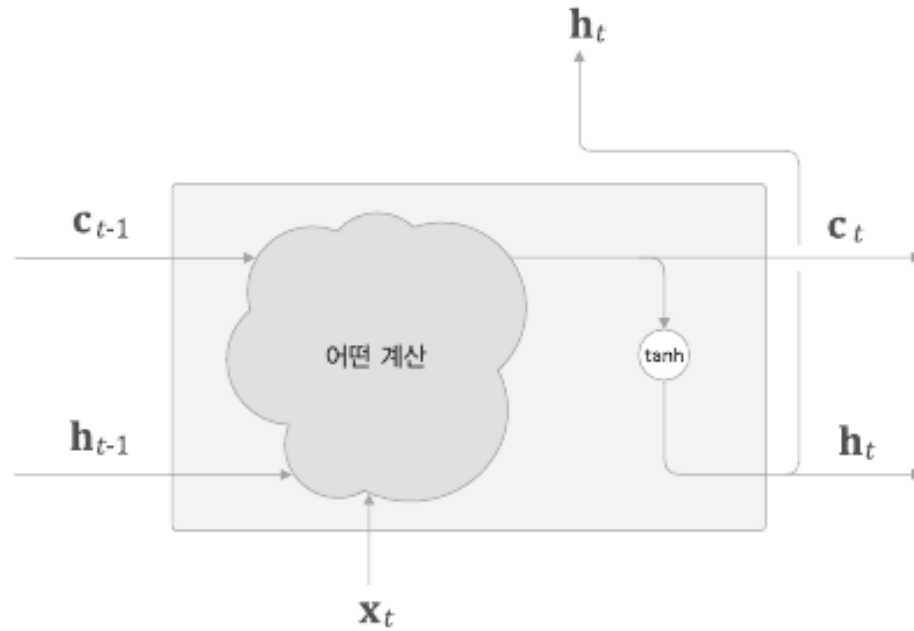
그림 6-11 RNN 계층과 LSTM 계층 비교



- C라는 경로가 존재
- 기억셀 or 셀 이라고 하며 LSTM 전용의 기억 메커니즘
- 데이터를 자기 자신으로만 주고 받는다는 점이 특징

2. LSTM

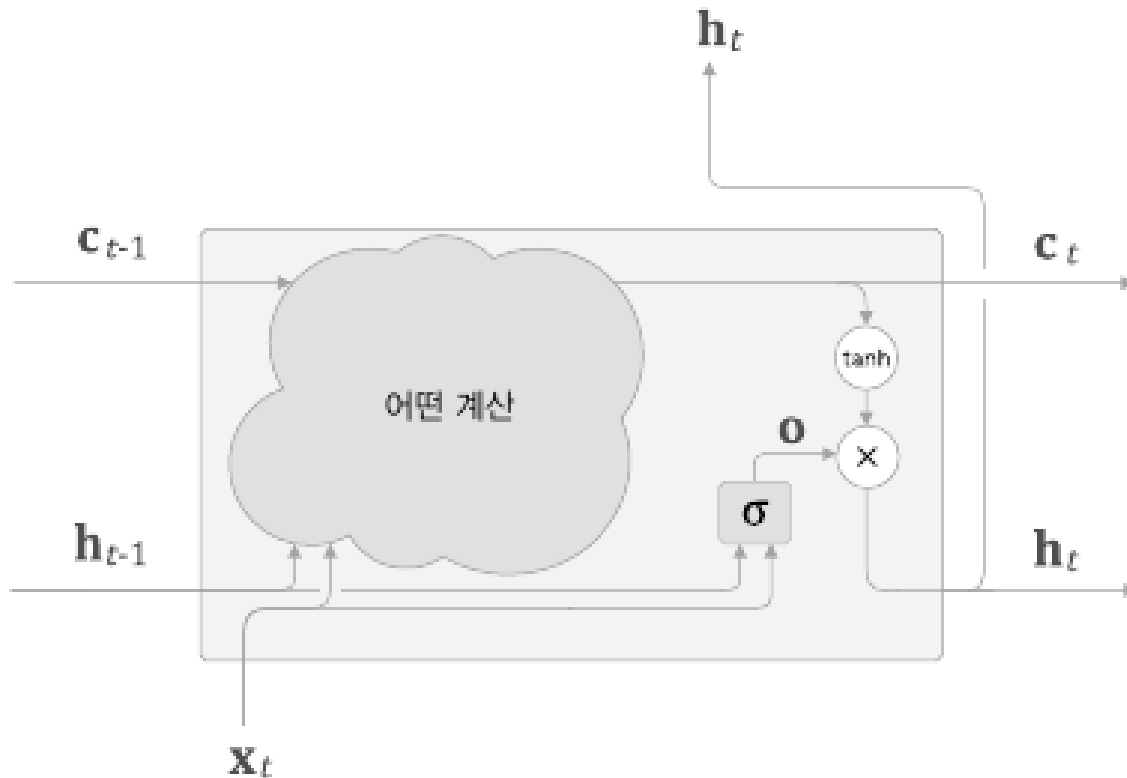
그림 6-12 기억 셀 c_t 를 바탕으로 은닉 상태 h_t 를 계산하는 LSTM 계층



- c 에는 과거부터 현재 t 까지의 모든 정보가 저장되어있다.
- 이것을 바탕으로 \tanh 이용해 h (은닉상태)를 출력합니다.

2. LSTM – output 게이트

그림 6-15 output 게이트 추가



$$\mathbf{o} = \sigma(\mathbf{x}_t \mathbf{W}_x^{(o)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(o)} + \mathbf{b}^{(o)})$$

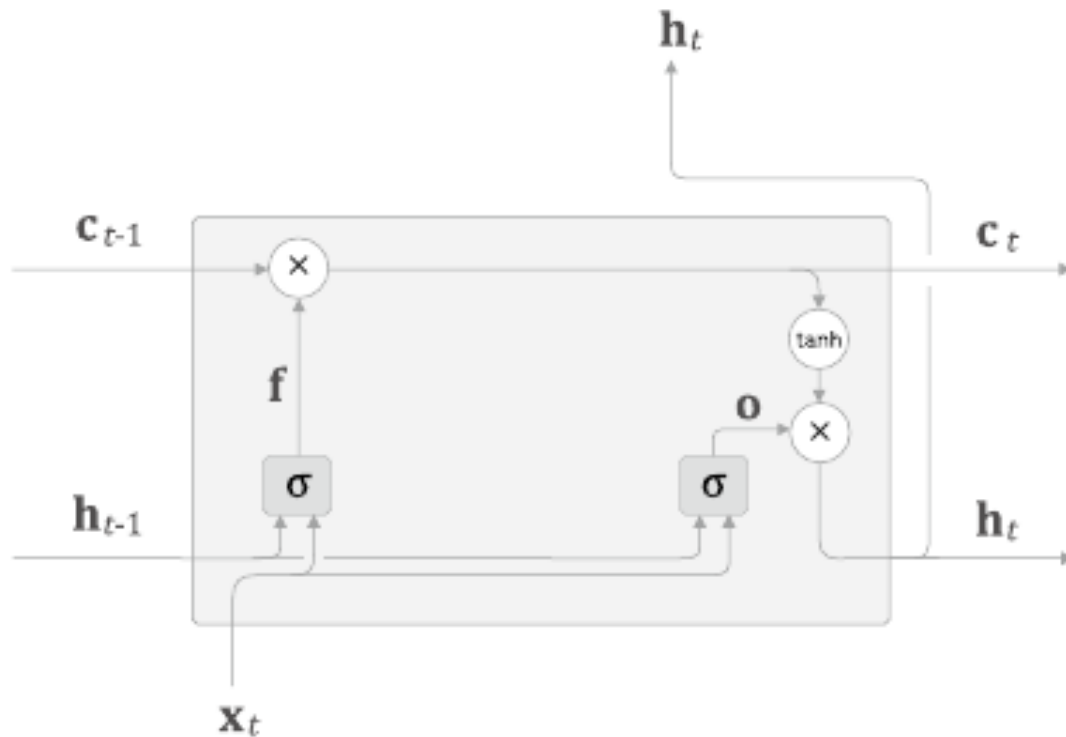
[식 6.1]

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

[식 6.2]

2. LSTM – forget 게이트

그림 6-16 forget 게이트 추가

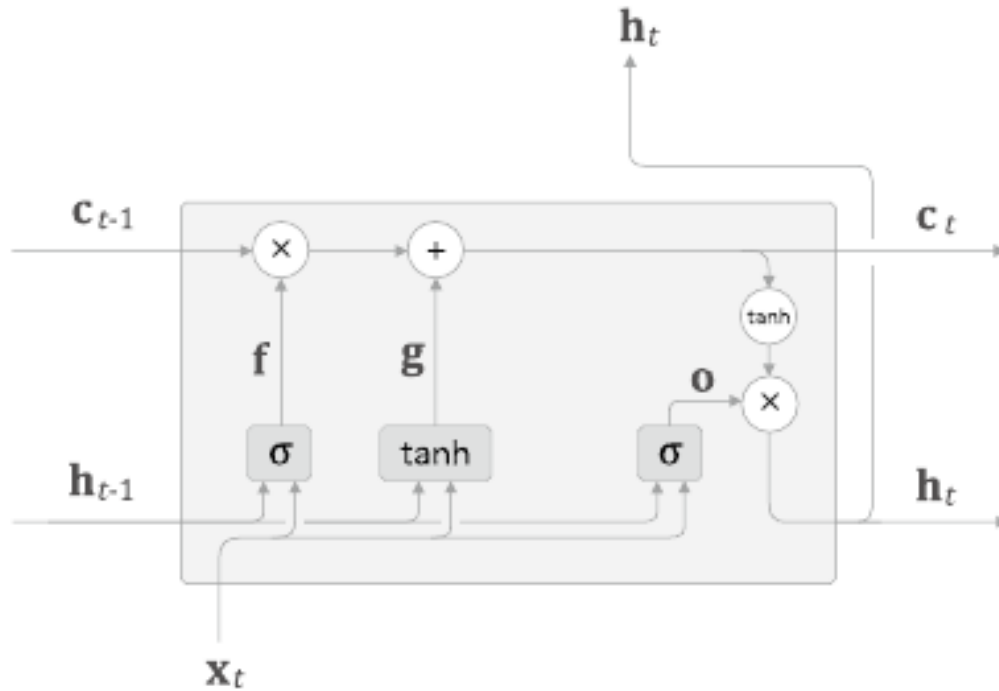


$$\mathbf{f} = \sigma(\mathbf{x}_t \mathbf{W}_x^{(f)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(f)} + \mathbf{b}^{(f)})$$

[식 6.3]

2. LSTM – 새로운 기억 셀

그림 6-17 새로운 기억 셀에 필요한 정보를 추가

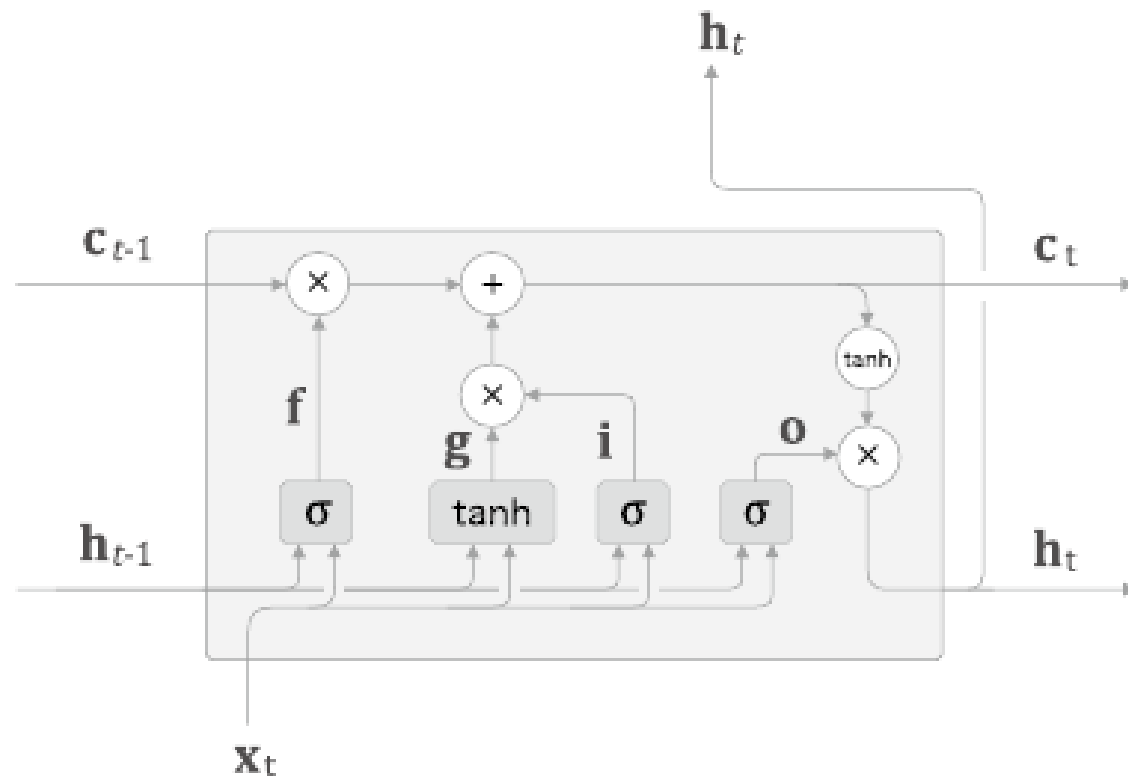


$$g = \tanh(x_t W_x^{(g)} + h_{t-1} W_h^{(g)} + b^{(g)})$$

[식 6.4]

2. LSTM – input 게이트

그림 6-18 input 게이트 추가

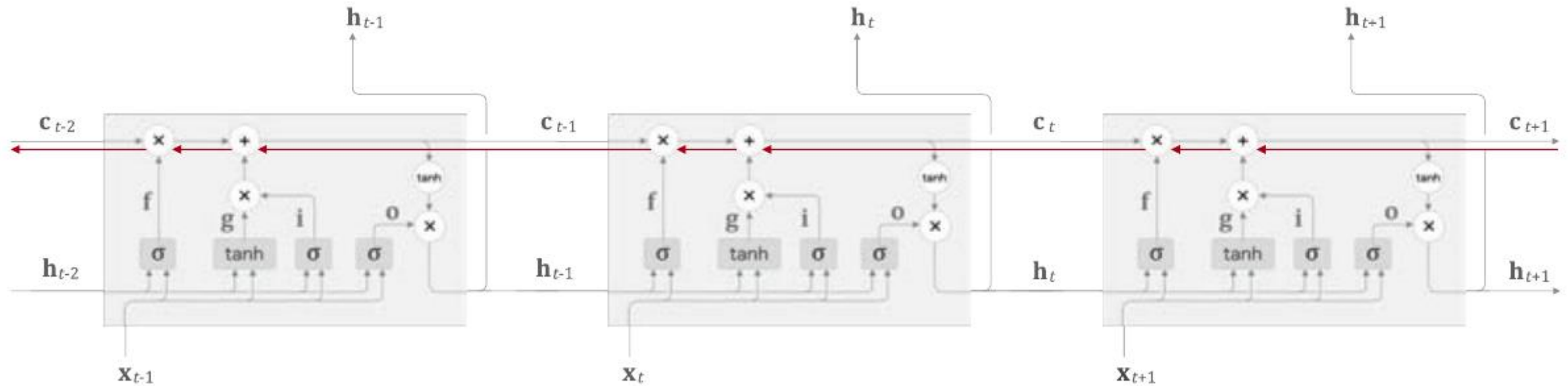


$$\mathbf{i} = \sigma(\mathbf{x}_t \mathbf{W}_x^{(i)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(i)} + \mathbf{b}^{(i)})$$

[식 6.5]

2. LSTM – LSTM의 기울기 흐름

그림 6-19 기억 셀의 역전파



3. LSTM 구현

$$\begin{aligned}\mathbf{f} &= \sigma(\mathbf{x}_t \mathbf{W}_x^{(f)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(f)} + \mathbf{b}^{(f)}) \\ \mathbf{g} &= \tanh(\mathbf{x}_t \mathbf{W}_x^{(g)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(g)} + \mathbf{b}^{(g)}) \\ \mathbf{i} &= \sigma(\mathbf{x}_t \mathbf{W}_x^{(i)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(i)} + \mathbf{b}^{(i)}) \\ \mathbf{o} &= \sigma(\mathbf{x}_t \mathbf{W}_x^{(o)} + \mathbf{h}_{t-1} \mathbf{W}_h^{(o)} + \mathbf{b}^{(o)})\end{aligned}$$

[식 6.6]

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

[식 6.7]

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

[식 6.8]

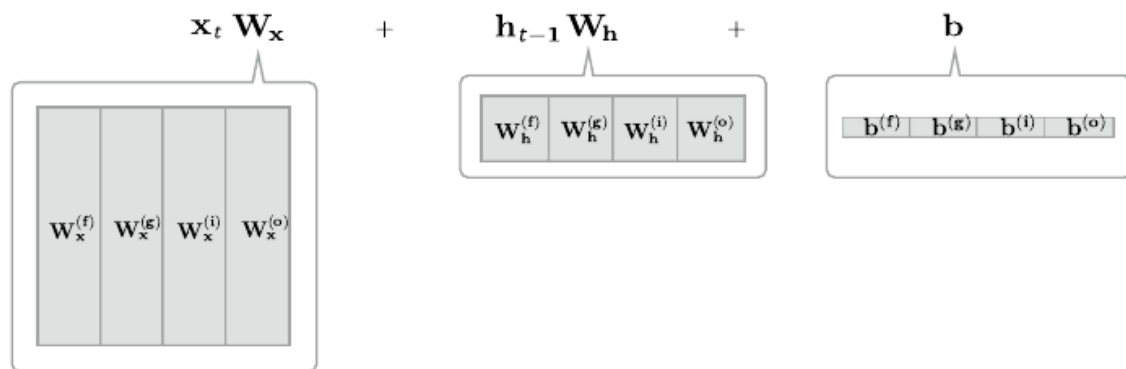
3. LSTM 구현

그림 6-20 각 식의 가중치들을 모아 4개의 식을 단 한 번의 아핀 변환으로 계산

$$\begin{aligned} x_t W_x^{(f)} + h_{t-1} W_h^{(f)} + b^{(f)} \\ x_t W_x^{(g)} + h_{t-1} W_h^{(g)} + b^{(g)} \\ x_t W_x^{(i)} + h_{t-1} W_h^{(i)} + b^{(i)} \\ x_t W_x^{(o)} + h_{t-1} W_h^{(o)} + b^{(o)} \end{aligned}$$

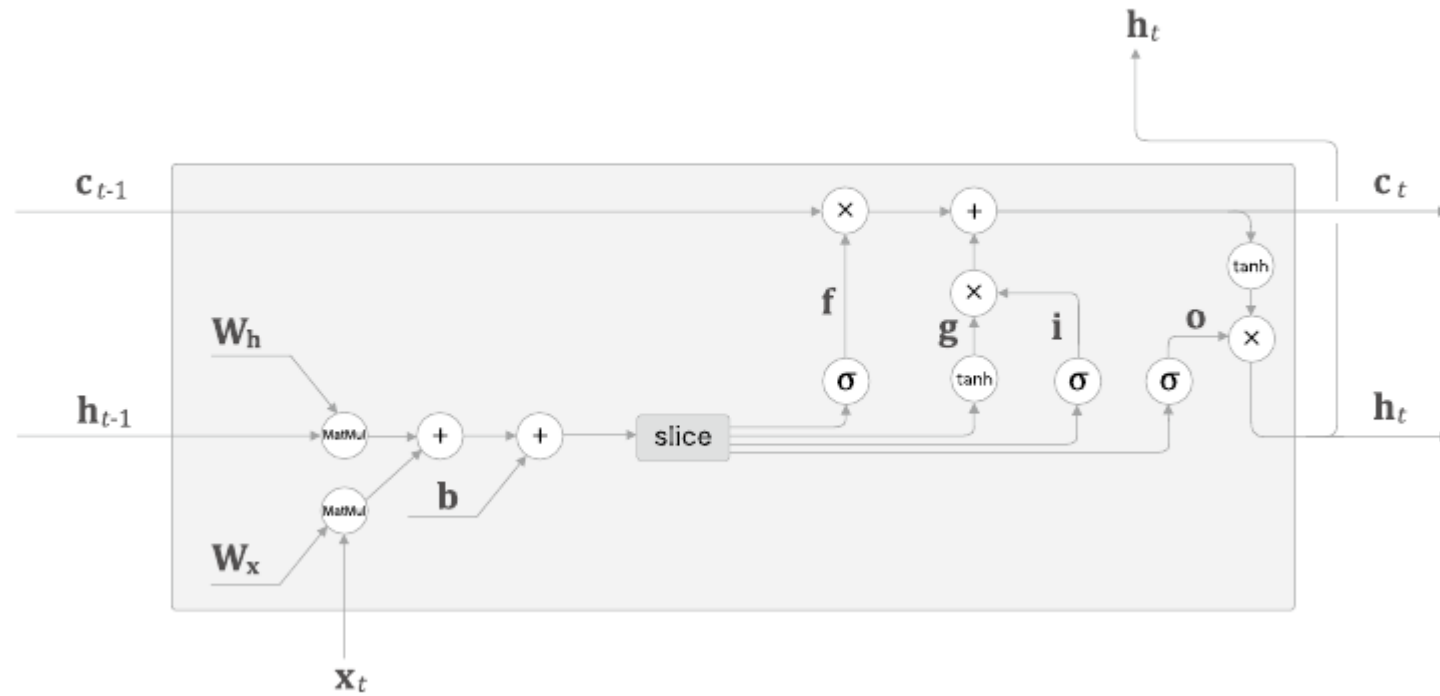


$$x_t [W_x^{(f)} W_x^{(g)} W_x^{(i)} W_x^{(o)}] + h_{t-1} [W_h^{(f)} W_h^{(g)} W_h^{(i)} W_h^{(o)}] + [b^{(f)} b^{(g)} b^{(i)} b^{(o)}]$$



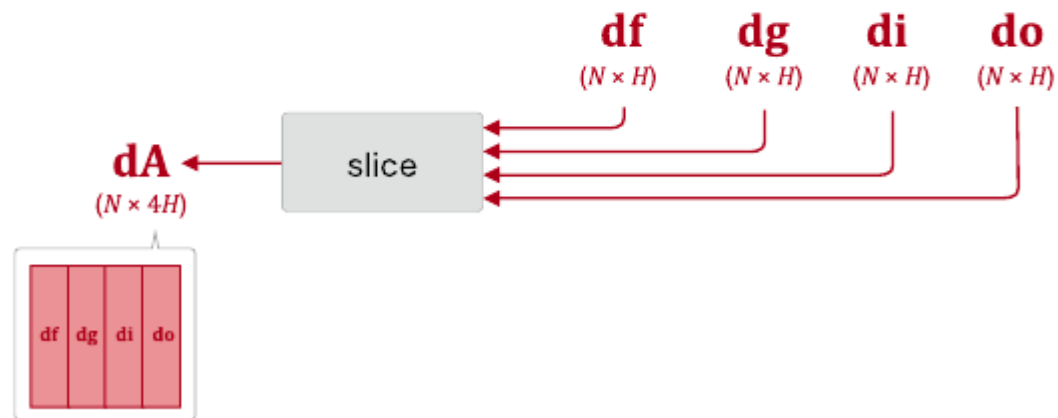
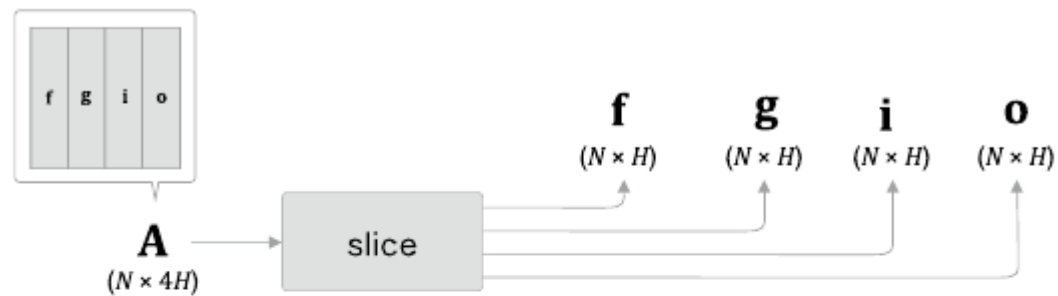
3. LSTM 구현

그림 6-21 4개분의 가중치를 모아 아핀 변환을 수행하는 LSTM의 계산 그래프



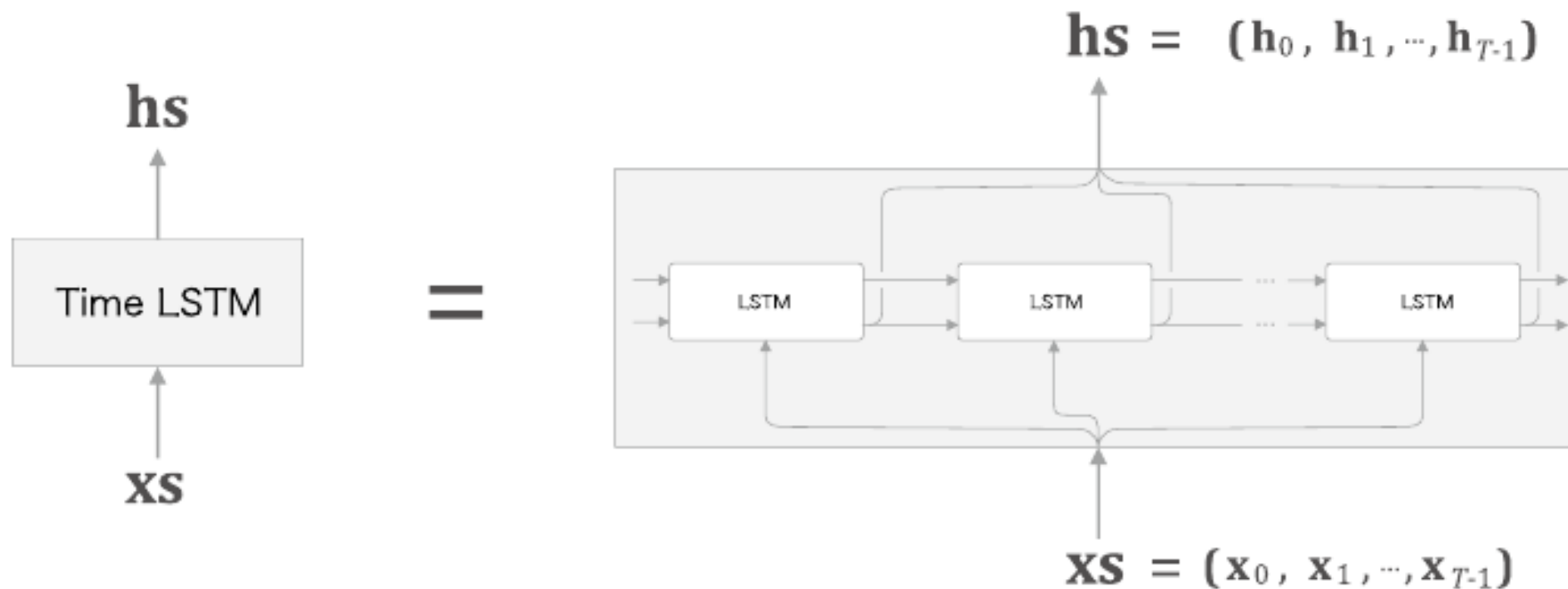
3. LSTM 구현

그림 6-23 slice 노드의 순전파(위)와 역전파(아래)



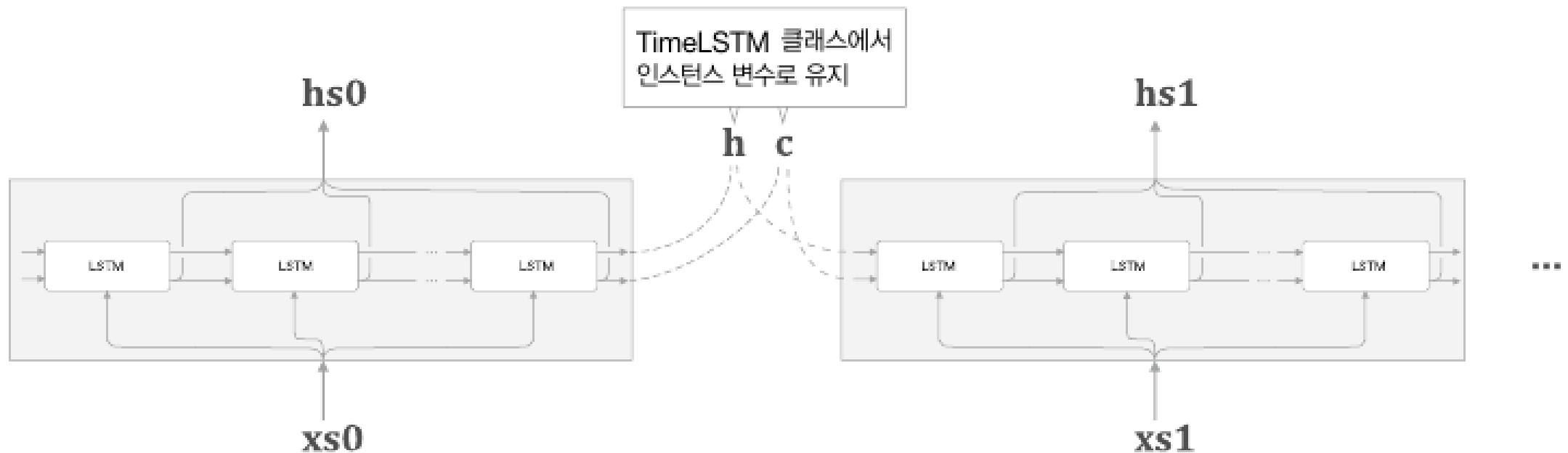
4. Time LSTM 구현

그림 6-24 Time LSTM의 입출력



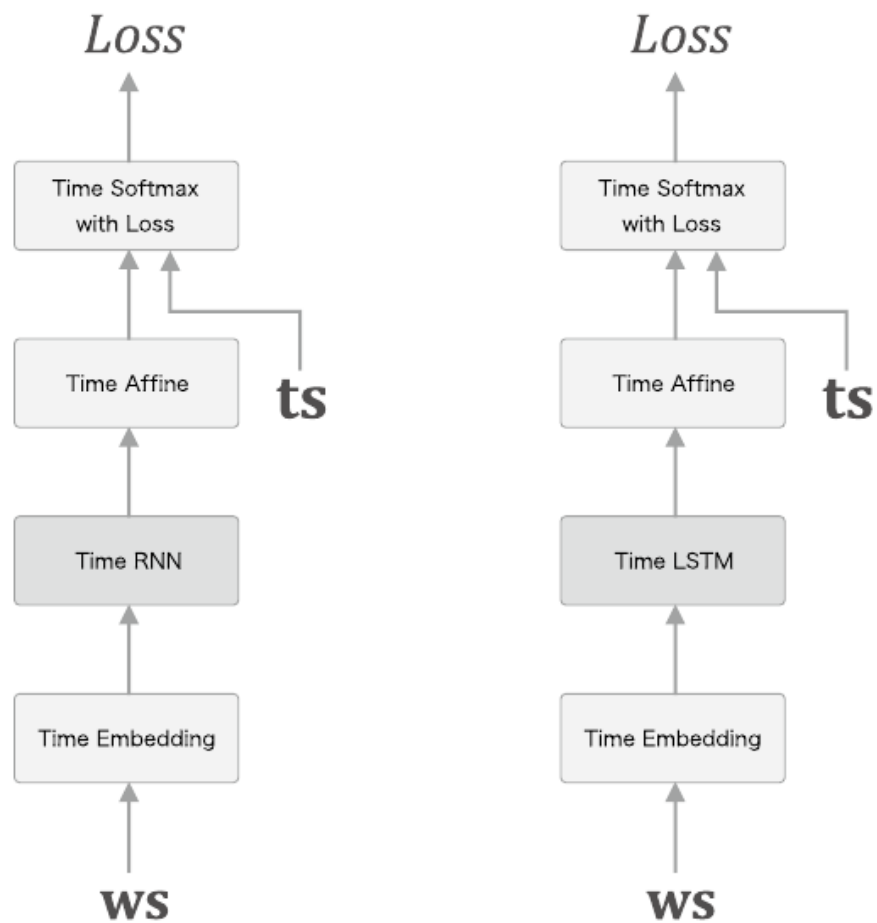
4. Time LSTM 구현

그림 6-25 Time LSTM 역전파의 입출력



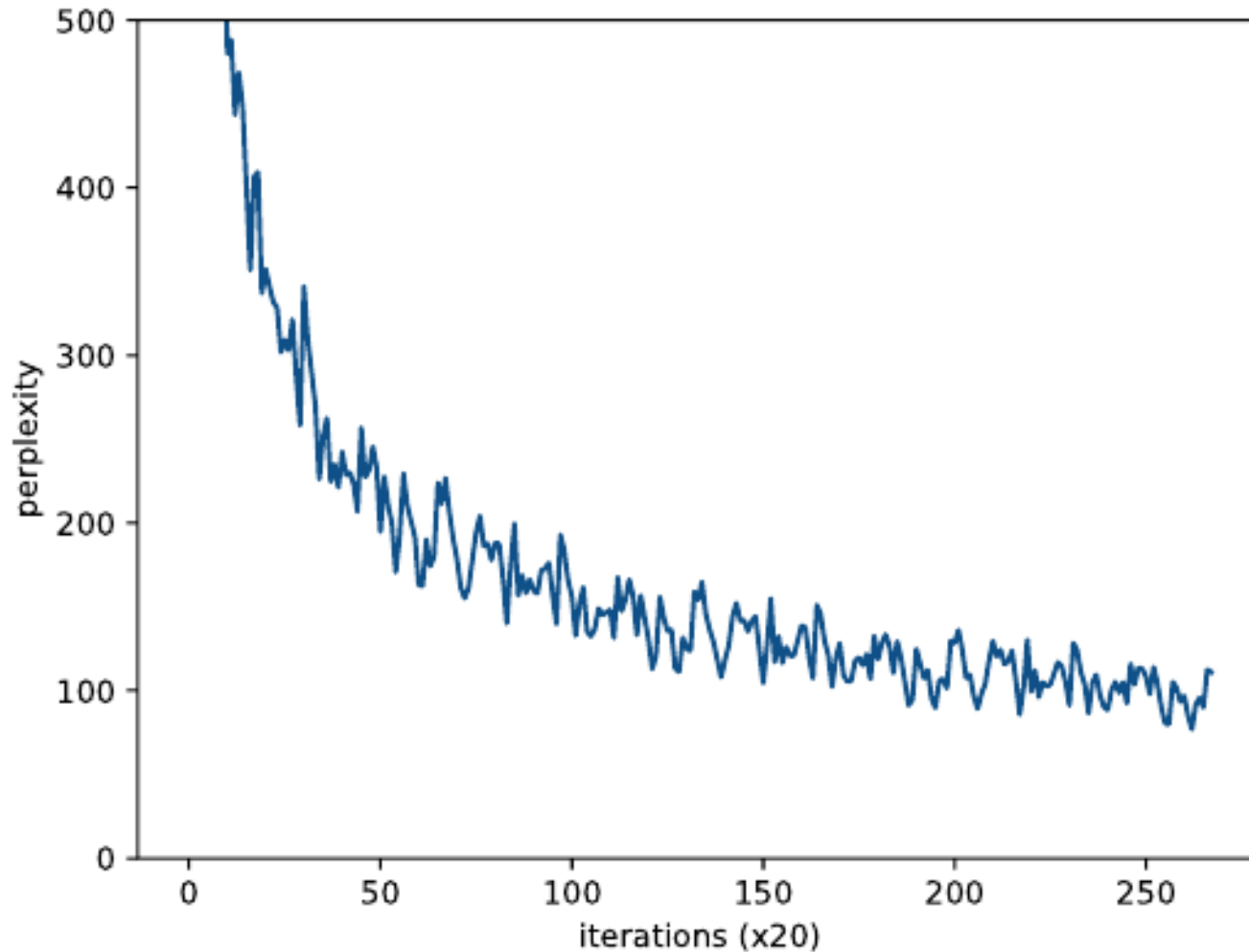
5. LSTM을 사용한 언어모델

그림 6-26 언어 모델의 신경망 구성(왼쪽은 앞 장에서 작성한 Time RNN을 이용한 모델, 오른쪽은 이번 장에서 작성하는 Time LSTM을 이용한 모델)



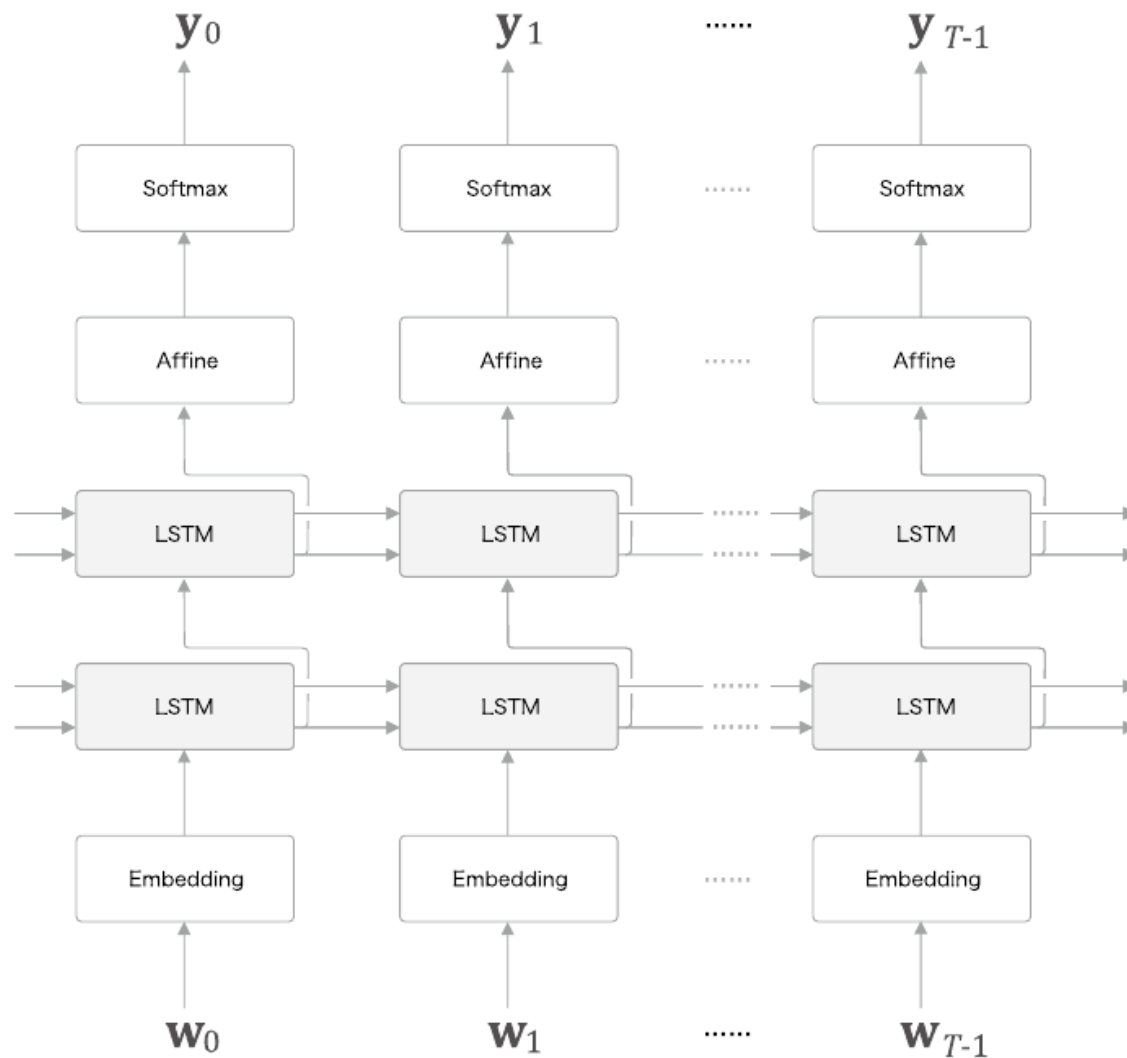
5. LSTM을 사용한 언어모델 - 결과

그림 6-28 퍼플렉서티 추이(훈련 데이터를 대상으로 20번째 반복마다 평가한 결과)



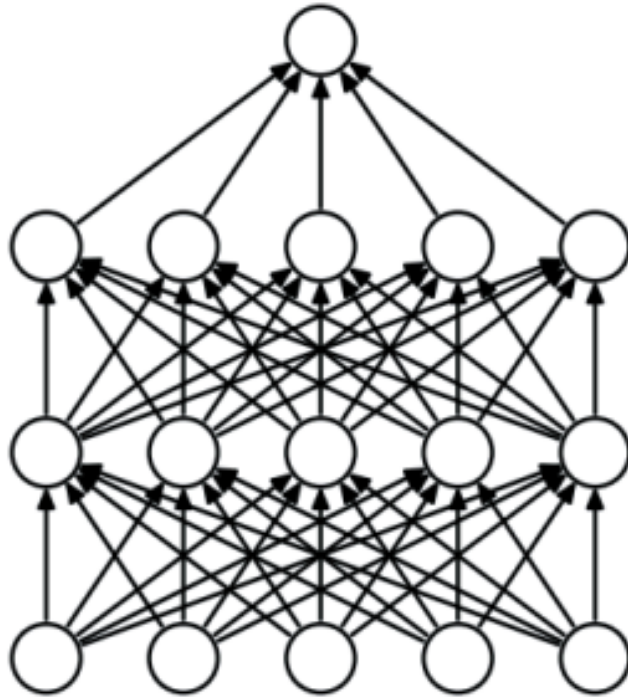
6. RNNLM 추가 개선 – LSTM 계층 다층화

그림 6-29 LSTM 계층을 2층으로 쌓은 RNNLM

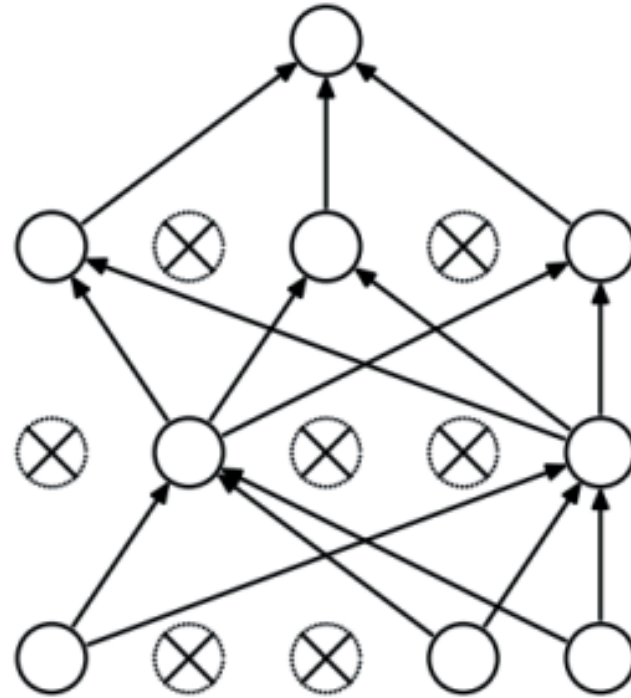


6. RNNLM 추가 개선 – 드롭아웃에 의한 과적합 억제

그림 6-30 드롭아웃 개념도(문헌 [9]에서 인용): 왼쪽이 일반적인 신경망, 오른쪽이 드롭아웃을 적용한 신경망



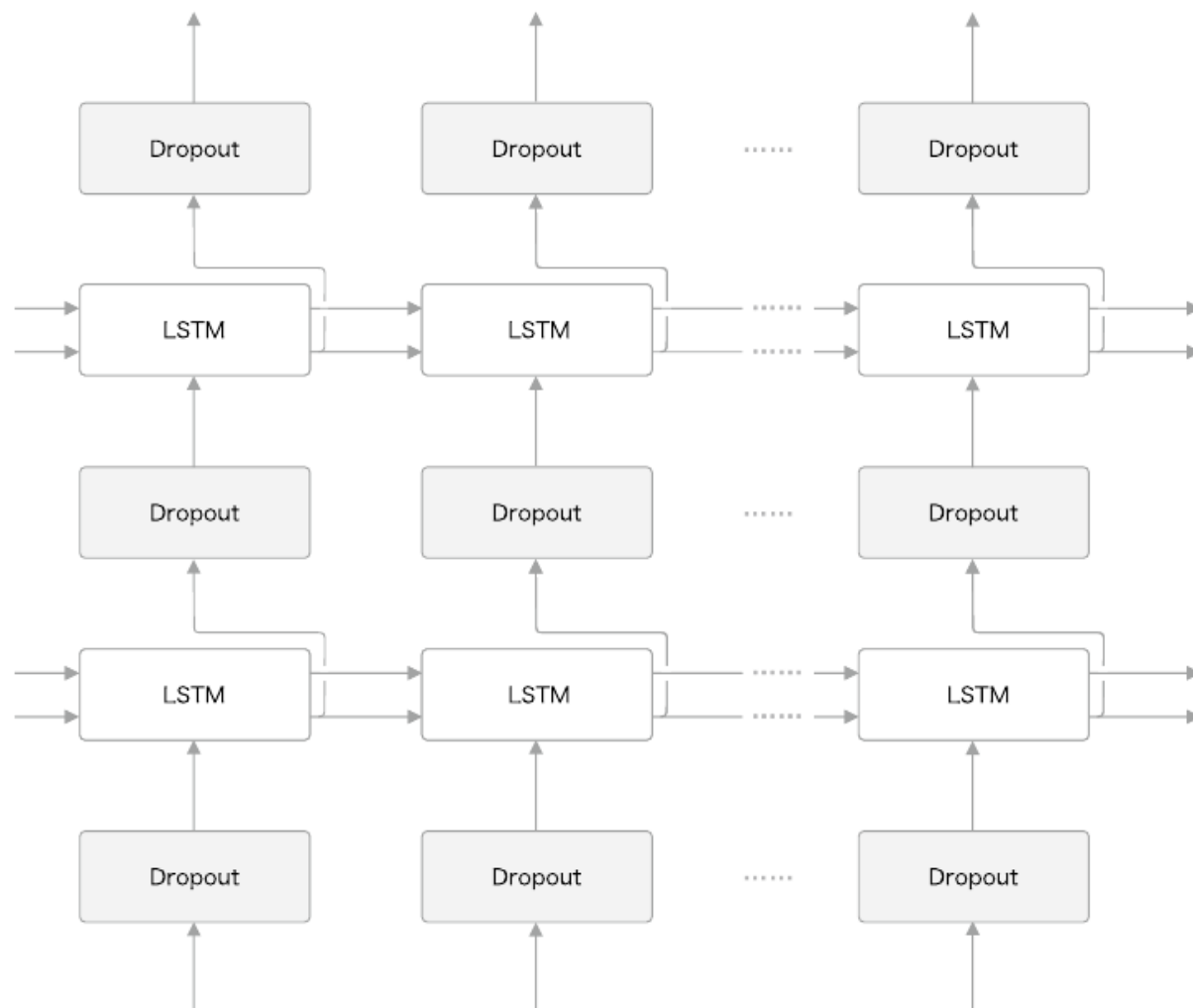
(a) 일반적인 신경망



(b) 드롭아웃을 적용한 모습

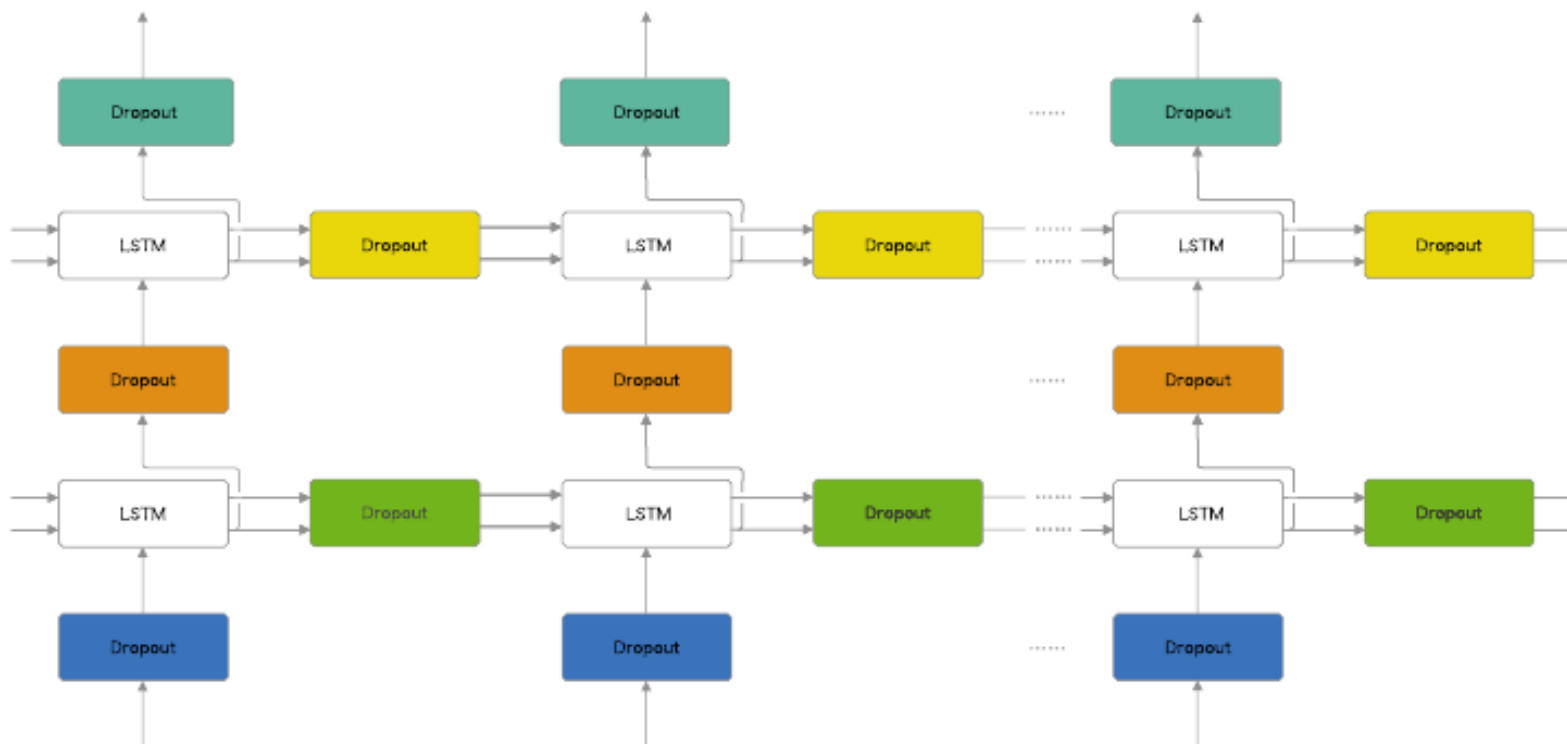
6. RNNLM 추가 개선 – 드롭아웃에 의한 과적합 억제

그림 6-33 좋은 예: 드롭아웃 계층을 깊이 방향(상하 방향)으로 삽입



6. RNNLM 추가 개선 – 드롭아웃에 의한 과적합 억제

그림 6-34 변형 드롭아웃의 예: 색이 같은 드롭아웃끼리는 같은 마스크를 이용한다. 이처럼 같은 계층에 적용되는 드롭아웃끼리는 공통의 마스크를 이용함으로써 시간 방향 드롭아웃도 효과적으로 작동할 수 있다.



6. RNNLM 추가 개선 – 가중치 공유

그림 6-36 BetterRnnlm 클래스의 신경망 구성

