5장. 딥러닝 – IV

4. 재귀 신경망

4.1 재귀 신경망

4.2 ReLU 활성화 함수를 사용하는 재귀 신경망

4.3 LSTM 재귀 신경망

4.4 GRU 재귀 신경망

4.5 재귀 신경망의 확장

4.6 재귀 신경망의 적용분야

4.1 재귀 신경망

❖ **재귀 신경망(Recurrent Neural Networks, RNN, 순환 신경망)**

- **서열 데이터**(Sequence data)
  - 음성, 자연어 문장, 동영상, 주가 변동 등의 데이터
  - 구성요소가 순차적으로 발생하거나 구성요소 간에 순서 존재
  - 이전 값들이 현재 값에 영향을 주는 경우

- 서열 데이터의 **분류, 예측**에서 **현재 시점**의 값과 **이전 시점의** 값들을 고려 필요

- 재귀 신경망은 서열 데이터의 학습 및 추론에 적합한 모델

- **기계 번역, 음성 인식, 필기체 인식, 영상 주석달기, 동영상에서 행동 인식, 작곡 및 작사** 등 다양한 응용 분야에서 활용

4.1.1 재귀 신경망의 구조와 동작

❖ **재귀 신경망의 구조**

- 기본적으로 **은닉층** 한 개와 **출력층**으로 구성입력의 일부로
- 과거의 정보를 반영하기 위해, **은닉층 또는 출력층의 값**을 입력의 일부로 **사용**

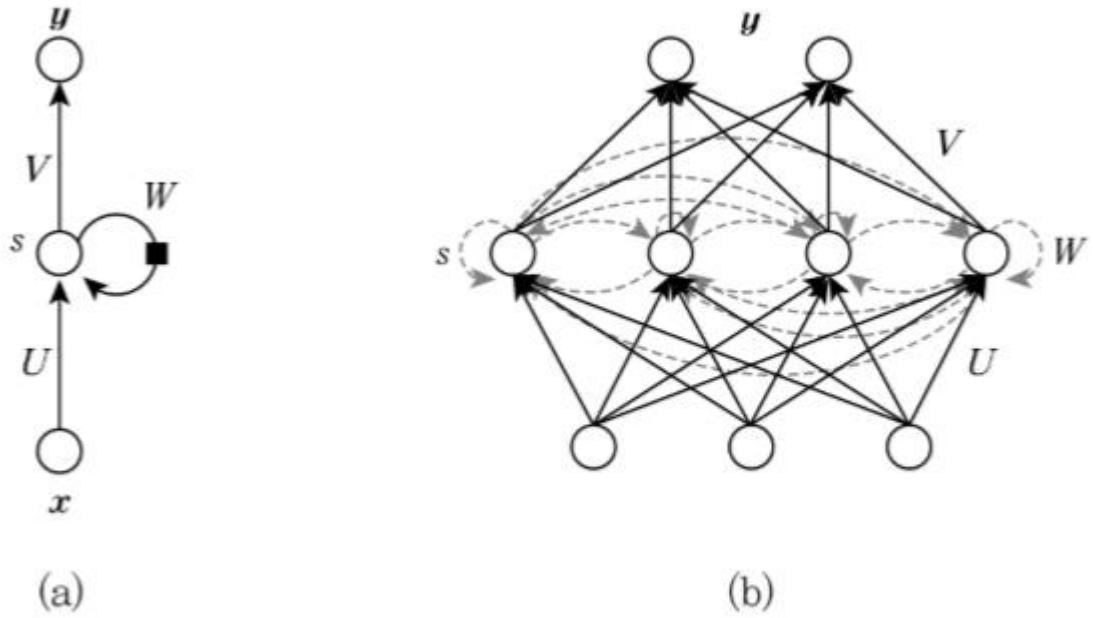


그림 5.52 RNN 모델 (a) RNN 모델의 형태 (b) RNN 모델의 실제 형태

재귀 신경망의 구조와 동작

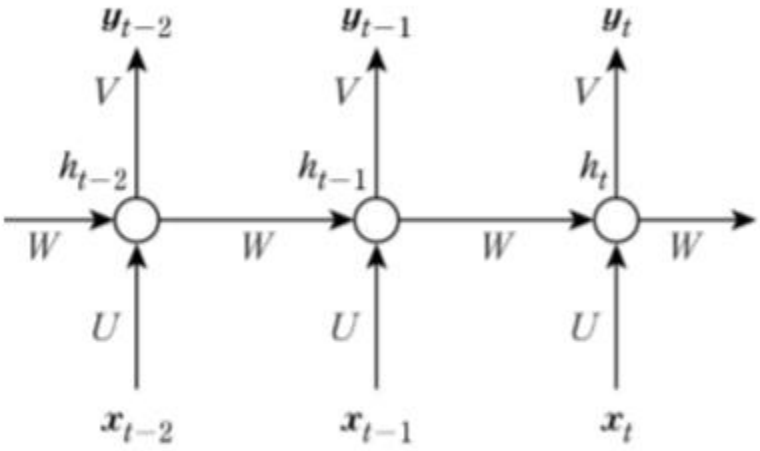
❖ 재귀 신경망의 동작

$$s_t = Ux_t + Wh_{t-1} + b_s$$

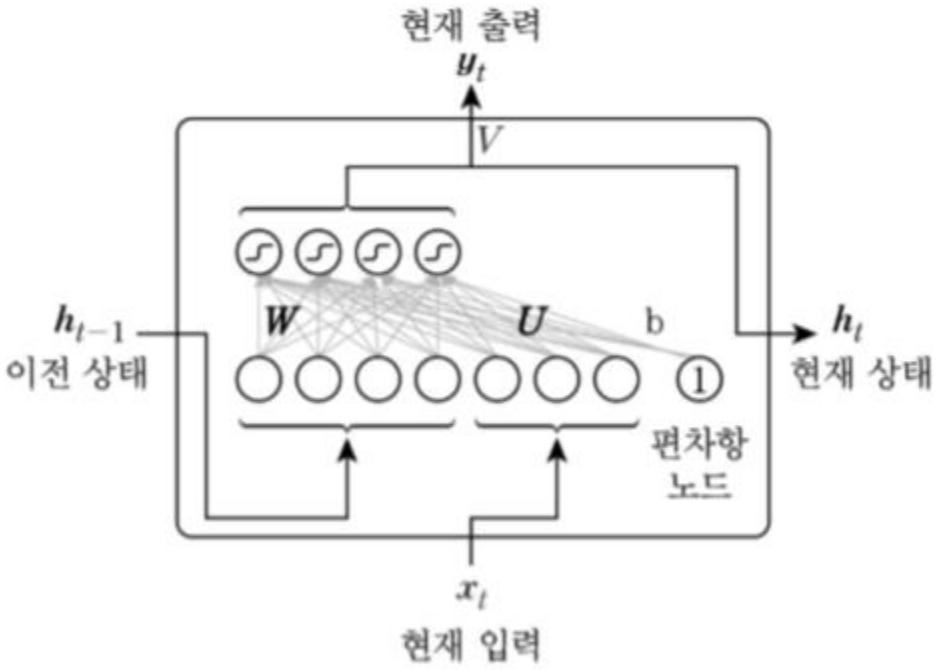
$$h_t = f(s_t)$$

$$z_t = Vh_t + b_z$$

$$y_t = g(z_t)$$



(a)

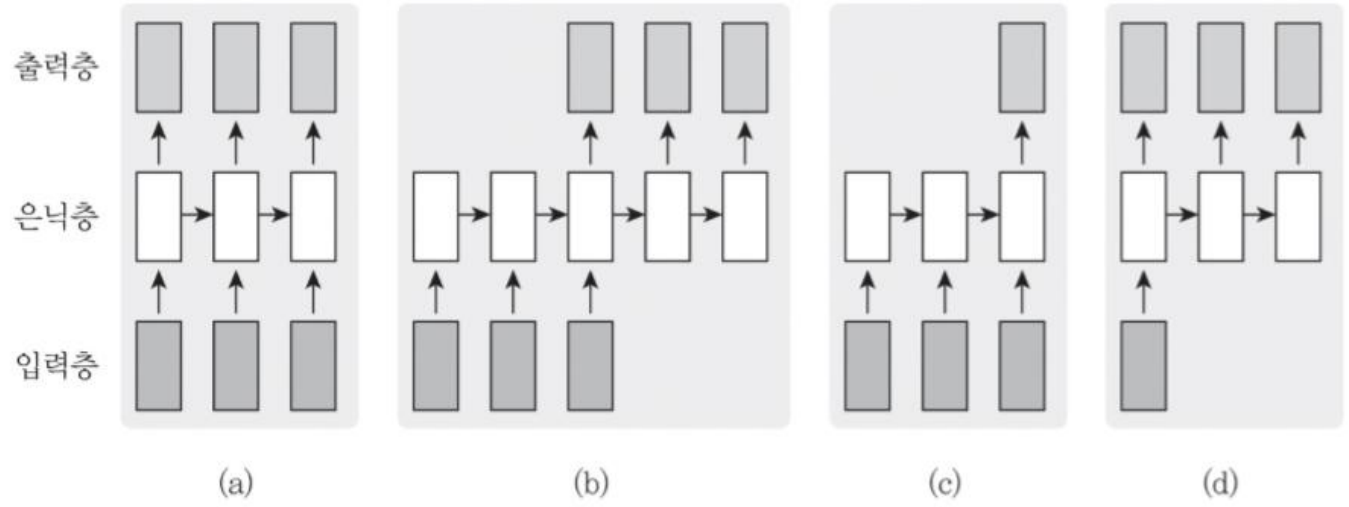


(b)

그림 5.53 RNN 모델 (a) RNN을 펼쳐놓은 형태 (b) RNN의 입출력 구조

재귀 신경망의 구조와 동작

❖ **재귀 신경망에서 입력과 출력의 대응 형태**



(a) 각 시점의 입력에 대한 출력이 학습 데이터에 지정
(b) 앞 시점에 입력이 끝나면서 출력값이 주어지는 상황
- **기계 번역**: '이것은 책이다' → 'This is a book'

(c) 일련의 데이터가 입력으로 주어진 다음, 마지막에 결과 값이 주어지는 상황
- **감성 분석**: '이 책은 내용이 알차게 구성되어 있다' → '긍정적'

(d) 하나의 입력에 대해 일련의 출력 이 나오는 것
- **영상 주석달기**: 영상 → 설명하는 문장

4.1.2 재귀 신경망의 학습

❖ **재귀 신경망의 학습 데이터 형태**

- 서열 데이터의 집합
- Ex. 문자열 'hello'의 학습 데이터 형태
  - h → e, e → l, l → l, l → o

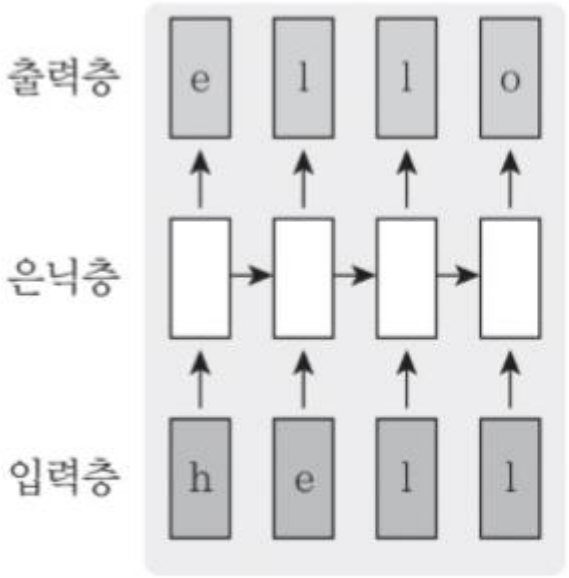


그림 5.55 RNN으로 'hello'를 학습할 때 학습 데이터

재귀 신경망의 학습

❖ **BPTT**(Back Propagation Through Time) **알고리즘**

- 과거 시간으로 오차를 전달하여 가중치 조정
- 과거 시점으로 오차를 전달할 때 각 가중치 는 동일하게 사용
- 학습을 할 때는, 각 시점에서의 그레디언트를 구한 다음, 그 평균값을 해당 변수에 대한 그레디언트로 사용

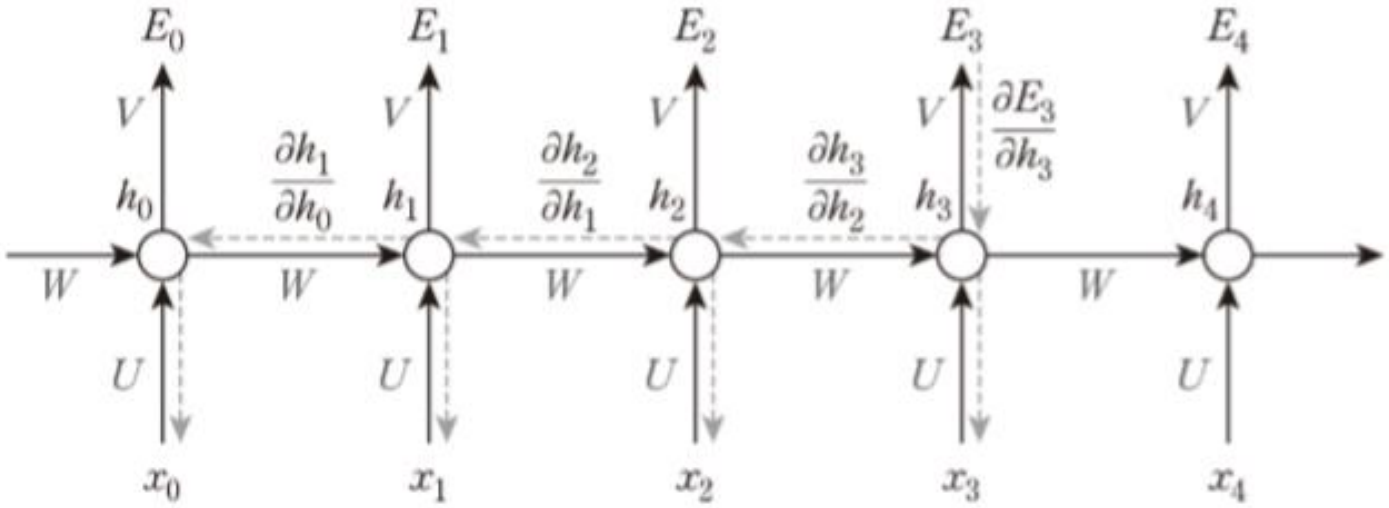


그림 5.56 $t = 3$에서 각 이전 시점으로 역전파되는 오차 정보의 흐름

재귀 신경망의 학습

❖ **BPTT 알고리즘 – Cont.**

- 목표 출력 서열
- RNN의 출력 서열

$$\boldsymbol{y}' = (\boldsymbol{y}_0', \boldsymbol{y}_1', \boldsymbol{y}_2', \cdots, \boldsymbol{y}_N')$$
$$\boldsymbol{y}_t = (y_{t1}, y_{t2}, \cdots, y_{tK})$$

- **오차 함수**
  - 분류

$$E_t(\boldsymbol{y}_t, \boldsymbol{y}_t') = -\sum_{i=1}^{K} y_{ti}' \log y_{ti}$$

  - 회귀

$$E_t(\boldsymbol{y}_t, \boldsymbol{y}_t') = \frac{1}{K}\sum_{k=1}^{K} (y_{ti}' - y_{ti})^2$$

- **목적함수**

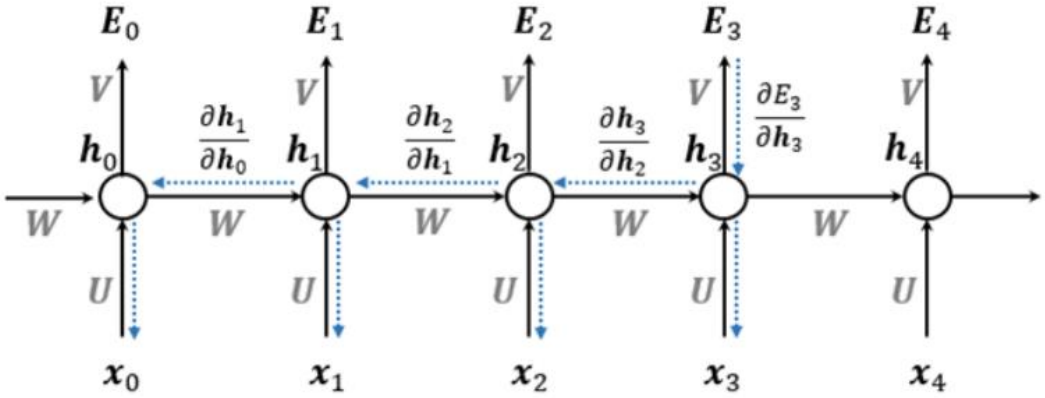
$$E(\boldsymbol{y}, \boldsymbol{y}') = \sum_t E_t(\boldsymbol{y}_t, \boldsymbol{y}_t')$$

- **목적 함수의 그레디언트**

$$\frac{\partial E}{\partial W} = \sum_i \frac{E_t(\boldsymbol{y}_t, \boldsymbol{y}_t')}{\partial W}$$

재귀 신경망의 학습

❖ **BPTT 알고리즘 – Cont.**



- $t = 3$ 일 때 W 에 대한 그레디언트

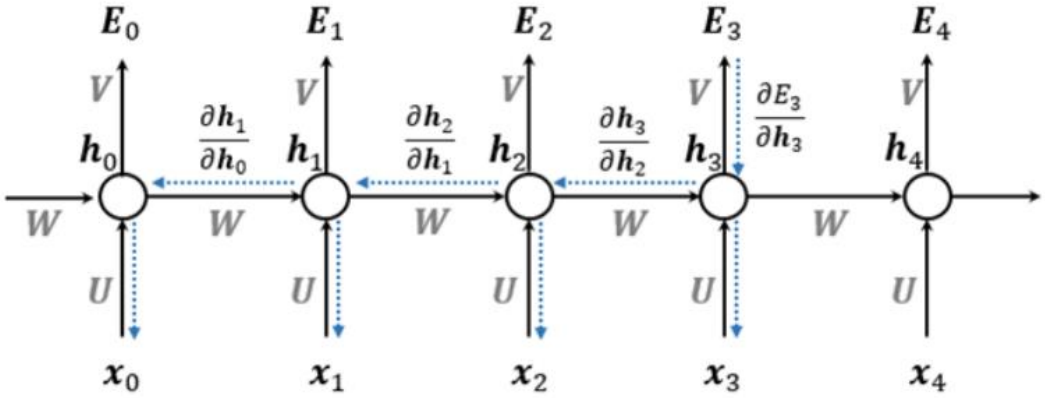
$$\frac{\partial E_3}{\partial W} = \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial h_3}\frac{\partial h_3}{\partial W}$$

$$\frac{\partial E_3}{\partial W} = \sum_{k=0}^{3} \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial h_3}\frac{\partial h_3}{\partial h_k}\frac{\partial h_k}{\partial W}$$

$$= \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial h_3}\left[\frac{\partial h_3}{\partial h_3}\frac{\partial h_3}{\partial W} + \frac{\partial h_3}{\partial h_2}\frac{\partial h_2}{\partial W} + \frac{\partial h_3}{\partial h_1}\frac{\partial h_1}{\partial W} + \frac{\partial h_3}{\partial h_0}\frac{\partial h_0}{\partial W}\right]$$

재귀 신경망의 학습

❖ BPTT 알고리즘 – Cont.



- $t = 3$일 때 V 에 대한 그레디언트

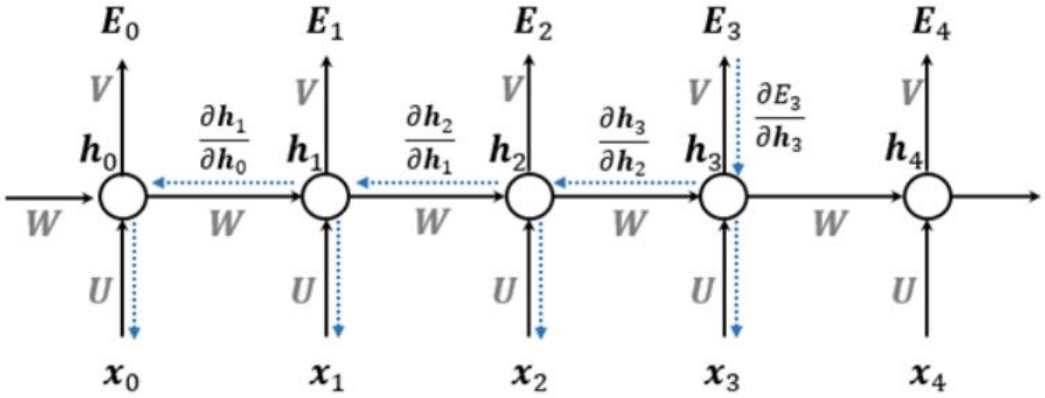
$$\frac{\partial E_3}{\partial V} = \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial V} = \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial z_3}\frac{\partial z_3}{\partial V}$$

- $t = 3$ 일 때 U 에 대한 그레디언트

$$\frac{\partial E_3}{\partial U} = \sum_{k=0}^{3}\frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial h_3}\frac{\partial h_3}{\partial h_k}\frac{\partial h_k}{\partial U}$$

4.1.3 재귀 신경망의 기울기 소멸과 폭발

❖ 그레디언트



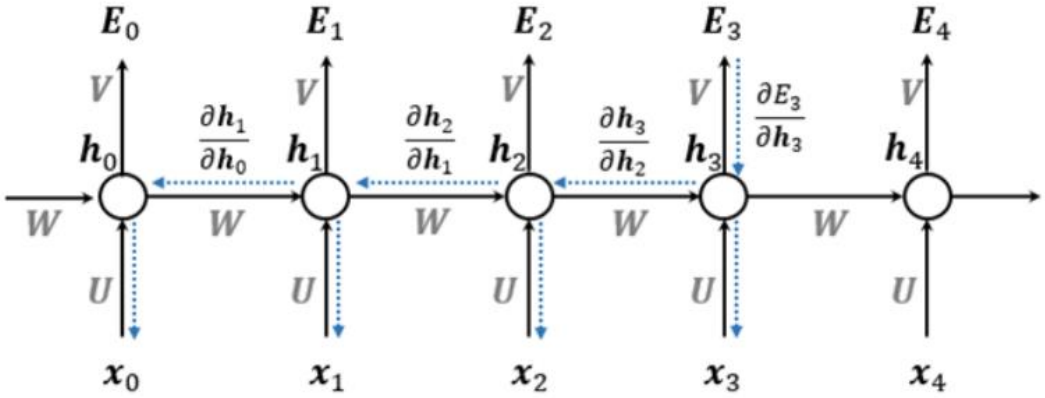
$$\frac{\partial E_3}{\partial W} = \sum_{k=0}^{3} \frac{\partial E_3}{\partial y_3} \frac{\partial y_3}{\partial h_3} \frac{\partial h_3}{\partial h_k} \frac{\partial h_k}{\partial W} = \sum_{k=0}^{3} \frac{\partial E_3}{\partial y_3} \frac{\partial y_3}{\partial h_3} \left(\prod_{j=k+1}^{3} \frac{\partial h_j}{\partial h_{j-1}} \right) \frac{\partial h_k}{\partial W}$$

$$h_j = f(s_j) = f(Ux_j + Wh_{j-1} + b_s)$$

$$\frac{\partial h_j}{\partial h_{j-1}} = W^{\top} diag(f'(s_j))$$

재귀 신경망의 기울기 소멸과 폭발

❖ 그레디언트 – cont.



$$\prod_{j=1}^{3} \frac{\partial \boldsymbol{h}_j}{\partial \boldsymbol{h}_{j-1}} = W^{\top} diag(f'(\boldsymbol{s}_3)) W^{\top} diag(f'(\boldsymbol{s}_2)) W^{\top} diag(f'(\boldsymbol{s}_1))$$

- $t = 100$이면, W^{T}의 100 거듭제곱 포함
- $f(x) = x$ 라고 가정할 때
  - $f'(x) = 1 \rightarrow \ diag(f'(s)) = I$

$$\prod_{j=1}^{100} \frac{\partial \boldsymbol{h}_j}{\partial \boldsymbol{h}_{j-1}} = (W^{\top})^{100}$$

재귀 신경망의 기울기 소멸과 폭발

❖ 그레디언트 – cont.

- 행렬 W의 고유값 분해

$$W = Q\Lambda Q^{-1}$$

Λ : 고유값을 대각 원소로 갖는 대각 행렬

Q : 고유벡터를 열벡터로 갖는 행렬

$$(W^\top)^{100} = (Q^{-1})^\top \Lambda^{100} Q^\top$$

- ex. $\Lambda = \begin{pmatrix} -0.6 & 0 \\ 0 & 1.8 \end{pmatrix}$ $\Lambda^3 = \begin{pmatrix} (-0.6)^3 & 0 \\ 0 & 1.8^3 \end{pmatrix} = \begin{pmatrix} -0.216 & 0 \\ 0 & 3.375 \end{pmatrix}$

$$\Lambda^{10} = \begin{pmatrix} 0.006 & 0 \\ 0 & 357.046 \end{pmatrix}$$

- |고유값| < 1이면, 기울기 소멸
- |고유값| > 1이면, 기울기 폭발

재귀 신경망의 기울기 소멸과 폭발

❖ **기울기 소멸 문제(Vanishing gradient problem)**

- 오차 정보를 역전파 시키는 과정에서 그레디언트가 급격히 영벡터에 가까워져서 학습이 되지 않는 현상
- 일부 가중치 성분에서만 발생 가능 ⇨ 문제 발생 파악 곤란

❖ **기울기 폭발 문제(Exploding gradient problem)**

- 학습과정에 그레디언트가 급격히 커지는 현상
- 일부 성분에서의 기울기 폭발현상은 다른 성분에 바로 파급
  ⇨ 문제 발생 확인 용이

4.1.4 재귀 신경망의 기울기 소멸과 폭발 조건

❖ **재귀 신경망의 기울기 소멸과 폭발 조건**

- λ_1 : 가중치 행렬 W의 최대 고유값

- $\lambda_1 < 1/\gamma$ 일 때
  - **기울기 소멸 문제 반드시 발생**
  - 활성화 함수가 $\tanh(\,)$일 때, $\gamma = 1$
  - 활성화 함수가 $\sigma(\,)$일 때, $\gamma = 1/4$

- $\lambda_1 > 1/\gamma$ 일 때
  - **기울기 폭발 문제 발생 가능**

4.1.5 기울기 폭발 문제의 대응 방법

❖ **기울기 폭발 문제의 대응 방법**

- **RMSprop 방법 사용**
  - 최근 그레디언트들의 크기의 평균에 해당 하는 값으로 나누어 사용
  - 그레디언트의 갑작스러운 큰 변화를 방지하는 효과
  - 기울기 소멸 문제뿐만 아니라 기울기 폭발 문제 완화

- **단기 BPTT(truncated BPTT) 사용**
  - 오차정보를 최근 몇 단계까지만 역전파
  - 가중치 행렬 W이 거듭제곱되는 횟수 제한

- **그레디언트 최대값 고정 방법 사용**
  - 그레디언트가 일정한 임계값 이상이 되면 임계값 으로 고정
  - $\|\nabla f\| > \theta$ 이면

$$\nabla f \leftarrow \theta \frac{\nabla f}{\|\nabla f\|}$$

4.2 LeRU 활성화 함수를 사용하는 재귀 신경망

❖ LeRU를 사용하는 재귀 신경망

- **IRNN**
  - 은닉층에서 은닉층으로의 가중치를 나타내는 행렬 W를 **항등행렬** I로 초기화한 후에 학습
  - 기존의 RNN 모델보다 높은 성능 개선 가능

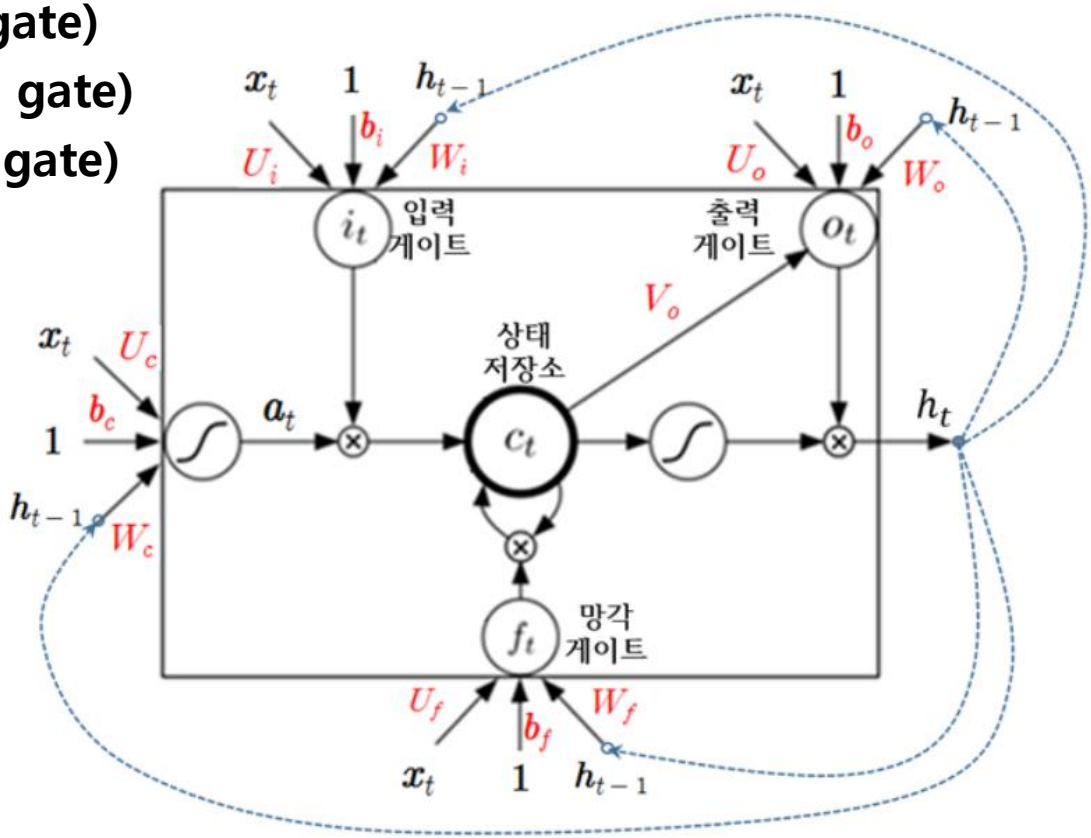
- **np-RNN**
  - 은닉층에서 은닉층으로의 가중치를 고유값의 하나는 1이고 나머지는 1보다 작은 값은 갖는 **양의 준정부호 행렬**(positive semi-definite matrix)로 초기화

- **uRNN**
  - 은닉층의 가중치 W가 모든 고유값의 크기가 1인 **유니타리 행렬**(unitary matrix)이 되도록 하면서 학습

4.3 LSTM 재귀 신경망

❖ **LSTM(Long Short Time Memory) RNN**

- 역전파되는 그레디언트가 쉽게 소멸되는 현상을 완화시키는 RNN 모델
- 각 은닉 노드가 **상태 저장소(memory cell)와 저장, 출력, 망각**을 **조절**하는 **게이트(gate)** 포함
  - **입력 게이트(input gate)**
  - **출력 게이트(output gate)**
  - **망각 게이트(forget gate)**



4.3.1 LSTM 재귀 신경망의 구조

❖ **일반 재귀 신경망의 입력에 대한 민감도**

- 시점 $t = 1$에서의 입력에 대한 **시점별 민감도**를 **노드의 진하기**로 보인 것
- 시간이 진행됨에 따라 새로운 입력이 은닉 상태에 반영되기 때문에 과거의 기억은 점차 사라짐

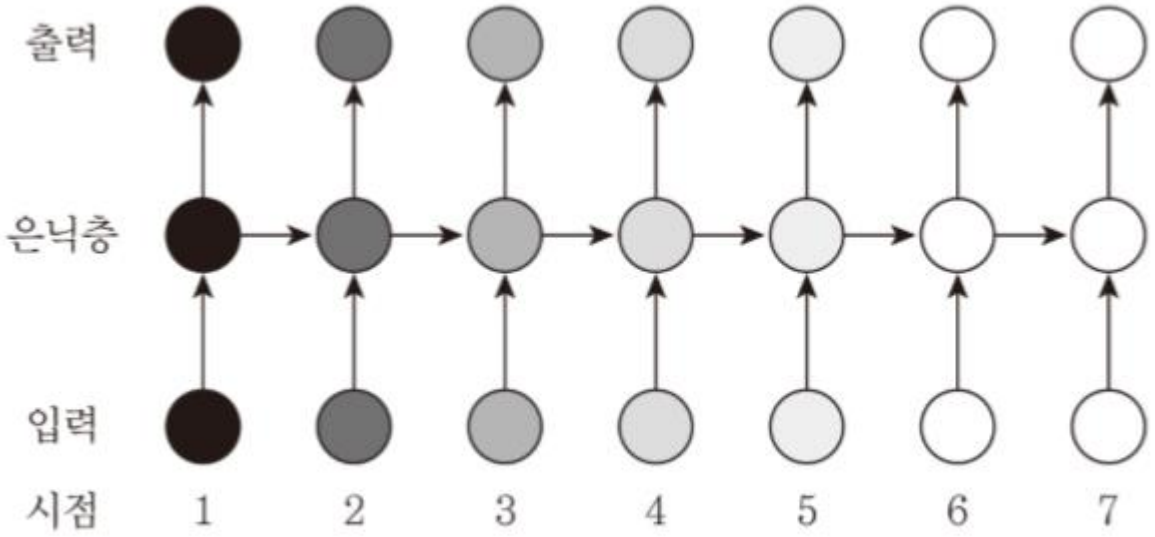


그림 5.57 재귀 신경망에서 시점 1의 입력에 대한 그레디언트의 민감도

LSTM 재귀 신경망의 구조

❖ **게이트 장착을 통한 민감도 조절**

- 게이트의 조작을 통해 먼 시점까지 영향 전파 가능
- LSTM은 게이트의 동작을 학습을 통해 결정

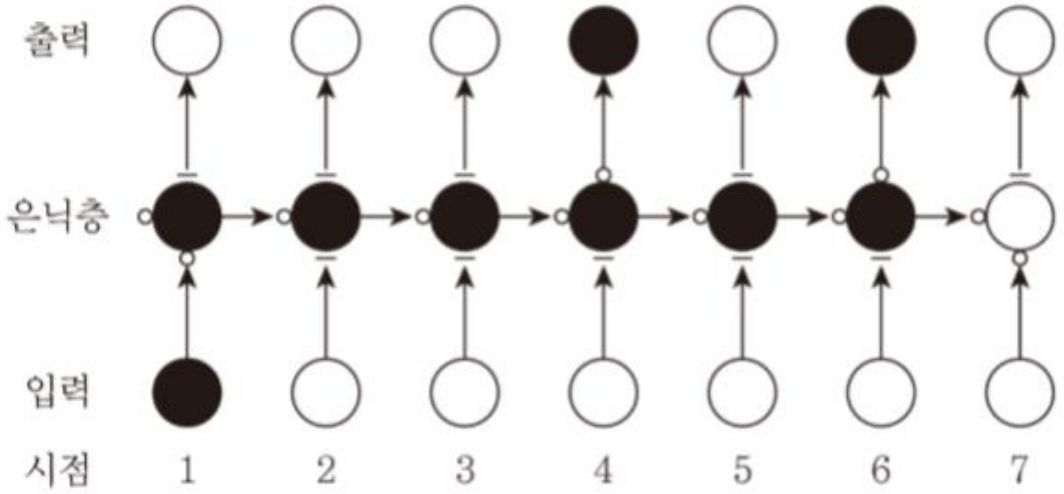
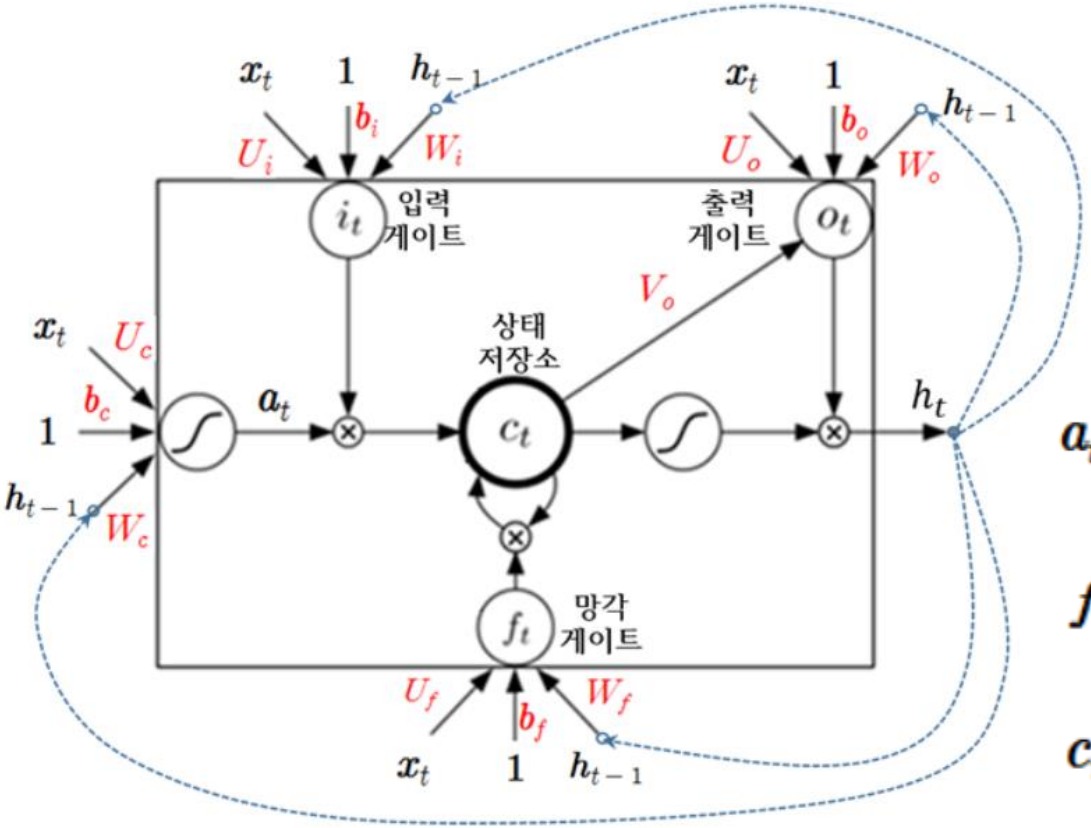


그림 5.58 게이트의 열림-닫힘에 의한 시점 1의 입력에 대한 그레디언트의 민감도

노드의 왼쪽에는 망각 게이트, 아래쪽에는 입력 게이트, 위쪽에는 출력 게이트가 위치하고, ○는 게이트의 열림, - 또는 |는 게이트의 닫힘을 나타낸다.

4.3.2 LSTM 재귀 신경망의 동작

❖ LSTM 재귀 신경망의 동작



$$\boldsymbol{i}_t = \sigma(U_i \boldsymbol{x}_t + W_i \boldsymbol{h}_{t-1} + \boldsymbol{b}_i)$$

$$\boldsymbol{a}_t = \tanh(U_c \boldsymbol{x}_t + W_c \boldsymbol{h}_{t-1} + \boldsymbol{b}_c)$$

$$\boldsymbol{f}_t = \sigma(U_f \boldsymbol{x}_t + W_f \boldsymbol{h}_{t-1} + \boldsymbol{b}_f)$$

$$\boldsymbol{c}_t = \boldsymbol{i}_t \circ \boldsymbol{a}_t + \boldsymbol{f}_t \circ \boldsymbol{c}_{t-1}$$

$$\boldsymbol{o}_t = \sigma(U_o \boldsymbol{x}_t + W_o \boldsymbol{h}_{t-1} + V_o \boldsymbol{c}_{t-1} + \boldsymbol{b}_o)$$

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

LSTM 재귀 신경망의 동작

알고리즘 5.3 LSTM 재귀 신경망의 실행

입력 : 서열 데이터 $(\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T)$

가중치 $W_c, U_c, W_i, U_i, W_f, U_f, W_o, U_o, V_0$

편차항 $\boldsymbol{b}_c, \boldsymbol{b}_i, \boldsymbol{b}_f, \boldsymbol{b}_o$

출력 : LSTM의 출력 $(\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_T)$

1. $\boldsymbol{h}_0 \leftarrow 0$
2. $\boldsymbol{c}_0 \leftarrow 0$
3. for $t = 1$ to T
4. $\quad \boldsymbol{i}_t \leftarrow \sigma(U_i \boldsymbol{x}_t + W_i \boldsymbol{h}_{t-1} + \boldsymbol{b}_i)$
5. $\quad \boldsymbol{a}_t \leftarrow \tanh(U_c \boldsymbol{x}_t + W_c \boldsymbol{h}_{t-1} + \boldsymbol{b}_c)$
6. $\quad \boldsymbol{f}_t \leftarrow \sigma(U_f \boldsymbol{x}_t + W_f \boldsymbol{h}_{t-1} + \boldsymbol{b}_f)$
7. $\quad \boldsymbol{c}_t \leftarrow \boldsymbol{i}_t \circ \boldsymbol{a}_t + \boldsymbol{f}_t \circ \boldsymbol{c}_{t-1}$
8. $\quad \boldsymbol{o}_t \leftarrow \sigma(U_o \boldsymbol{x}_t + W_o \boldsymbol{h}_{t-1} + V_o \boldsymbol{c}_{t-1} + \boldsymbol{b}_o)$
9. $\quad \boldsymbol{h}_t \leftarrow \boldsymbol{o}_t \circ \tanh(\boldsymbol{c}_t)$

4.3.3 LSTM 재귀 신경망의 학습

❖ **LSTM 재귀 신경망의 학습**

- BPTT 사용
- 게이트와 상태 저장소에 대한 오차 함수 $\boldsymbol{E}$의 그레디언트

$$\frac{\partial E}{\partial c_t^k} = \frac{\partial E}{\partial h_t^k}\frac{\partial h_t^k}{\partial c_t^k} = \frac{\partial E}{\partial h_t^k} o_t^k \left(1 - \tanh^2(c_t^k)\right)$$

$$\frac{\partial E}{\partial i_t^k} = \frac{\partial E}{\partial c_t^k}\frac{\partial c_t^k}{\partial i_t^k} = \frac{\partial E}{\partial c_t^k} a_t^k$$

$$\frac{\partial E}{\partial f_t^k} = \frac{\partial E}{\partial c_t^k}\frac{\partial c_t^k}{\partial f_t^k} = \frac{\partial E}{\partial c_t^k} c_{t-1}^k$$

$$\frac{\partial E}{\partial a_t^k} = \frac{\partial E}{\partial c_t^k}\frac{\partial c_t^k}{\partial a_t^k} = \frac{\partial E}{\partial c_t^k} i_t^k$$

$$\frac{\partial E}{\partial o_t^k} = \frac{\partial E}{\partial h_t^k}\frac{\partial h_t^k}{\partial o_t^k} = \frac{\partial E}{\partial h_t^k} tanh(c_t^k)$$

LSTM 재귀 신경망의 학습

❖ LSTM 재귀 신경망의 학습 – cont.

- 시점 $t-1$로 역전파되는 그레디언트

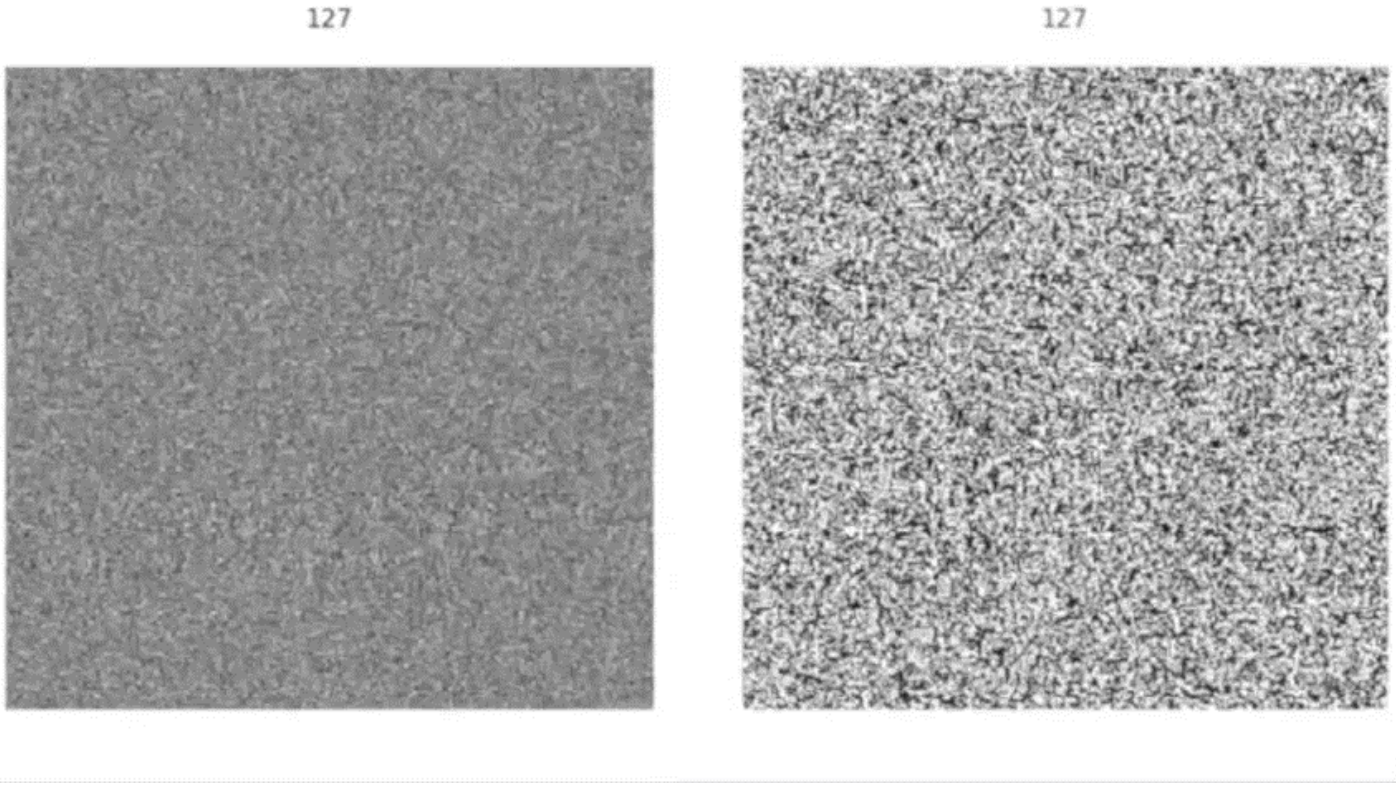
$$\frac{\partial E}{\partial c_{t-1}^k} = \frac{\partial E}{\partial c_t^k}\frac{\partial c_t^k}{\partial c_{t-1}^k} = \frac{\partial E}{\partial c_t^k}\frac{\partial\left(i_t^k a_t^k + f_t^k c_{t-1}^k\right)}{\partial c_{t-1}^k} = \frac{\partial E}{\partial c_t^k} f_t^k$$

- p 단계 과거로 전달되는 그레디언트

$$\frac{\partial E}{\partial c_{t-p}^k} = \frac{\partial E}{\partial c_t^k}\prod_{n=t-p}^{t} f_n^k$$

LSTM 재귀 신경망의 학습

❖ 기존 재귀 신경망과 LSTM 재귀 신경망
- 시간에 따른 그레디언트 전달 비교

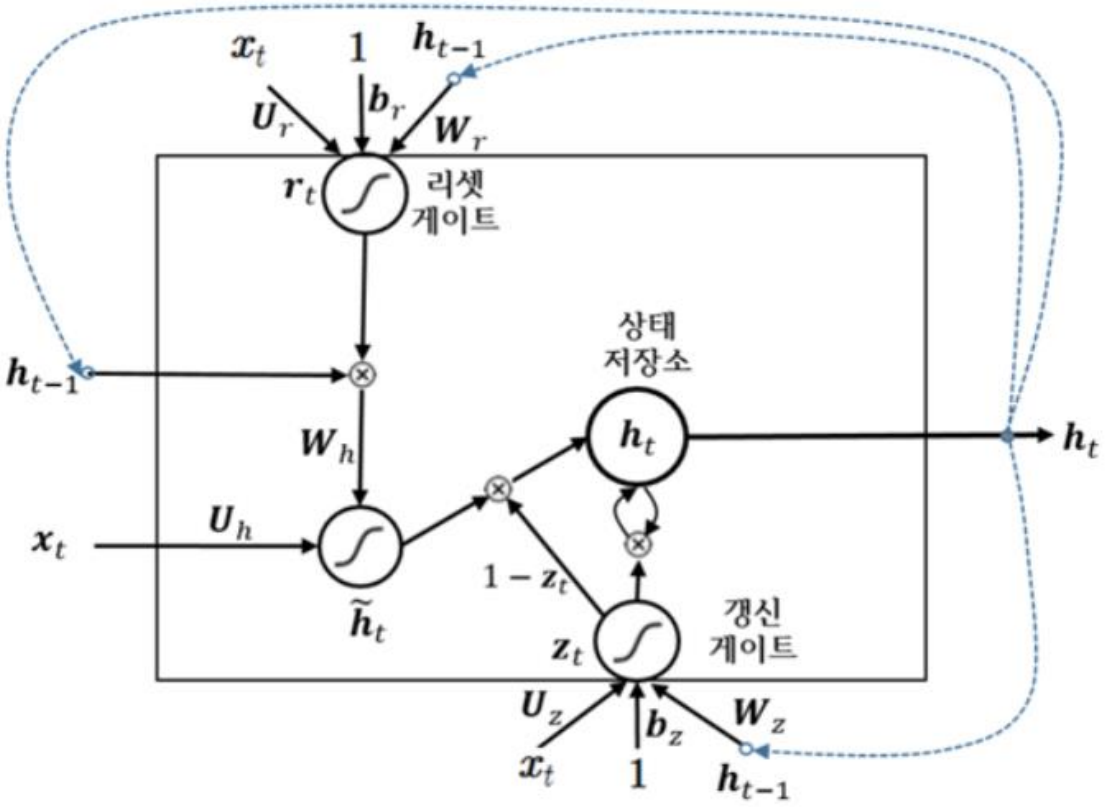


https://imgur.com/gallery/vaNahKE

4.4 GRU 재귀 신경망

❖ **GRU(gated recurrent unit) 재귀 신경망**

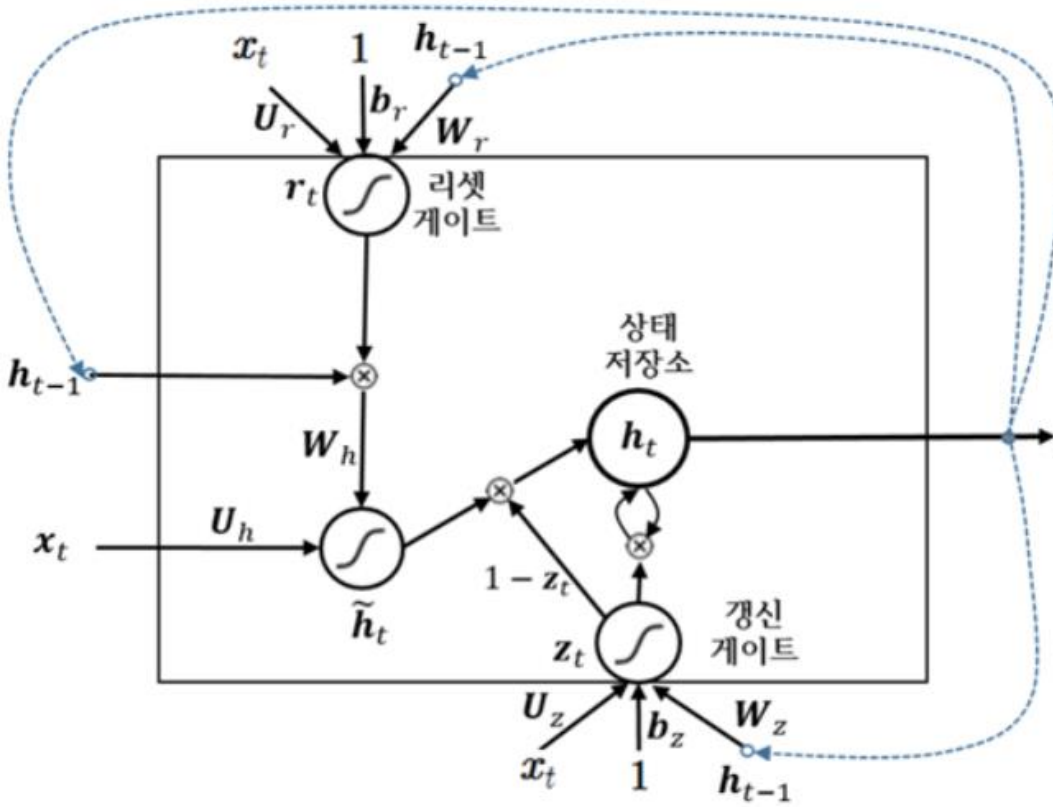
- **상태 저장소**(memory cell)과 **리셋 게이트**(reset gate) 및 **갱신 게이트**(update gate) 포함 모델



GRU 재귀 신경망

❖ **GRU 재귀 신경망의 동작**

$$\boldsymbol{r}_t = \sigma(U_r \boldsymbol{x}_t + W_r \boldsymbol{h}_{t-1} + \boldsymbol{b}_r)$$



$$\boldsymbol{h}_t = \boldsymbol{z}_t \circ \boldsymbol{h}_{t-1} + (1 - \boldsymbol{z}_t) \circ \tilde{\boldsymbol{h}}_t$$

$$\boldsymbol{z}_t = \sigma(U_z \boldsymbol{x}_t + W_z \boldsymbol{h}_{t-1} + \boldsymbol{b}_z)$$

$$\tilde{\boldsymbol{h}}_t = \tanh(U_h \boldsymbol{x}_t + W_h (\boldsymbol{r}_t \circ \boldsymbol{h}_{t-1}))$$

GRU 재귀 신경망

❖ **GRU 재귀 신경망의 학습**

- BPTT 알고리즘 사용
- LSTM 재귀 신경망의 파라미터 개수의 약 ¾ 파라미터 포함
- LSTM 재귀 신경망과 유사한 성능

4.5 재귀 신경망의 확장

❖ 양방향 재귀 신경망(Bidirectional RNN)

- 시점 t의 출력이 이전 시점의 입력값과 은닉층의 값들 뿐만 아니라 이후 시점의 입력값과 은닉층의 값들에도 영향을 받도록 한 모델

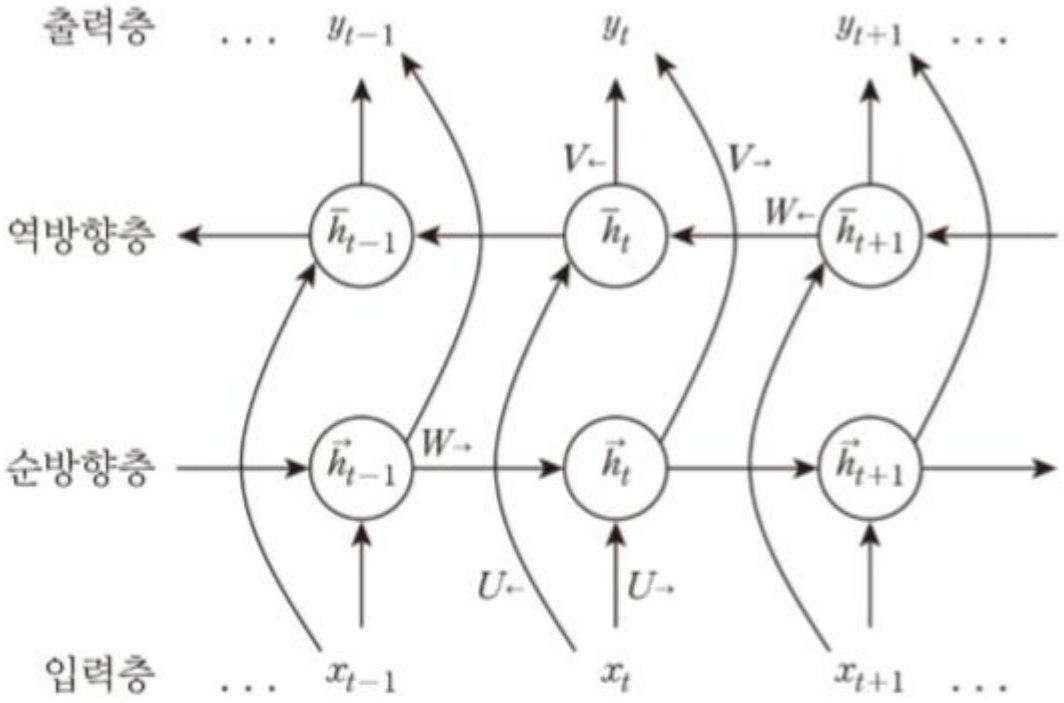


그림 5.61 양방향 재귀 신경망

$$\boldsymbol{y}_t = f\left(V_{\rightarrow}\vec{\boldsymbol{h}}_t + V_{\leftarrow}\overleftarrow{\boldsymbol{h}}_t + \boldsymbol{b}_o\right)$$

$$\overleftarrow{\boldsymbol{h}}_t = \sigma\left(U_{\leftarrow}\boldsymbol{x}_t + W_{\leftarrow}\overleftarrow{\boldsymbol{h}}_{t+1} + \boldsymbol{b}_{\leftarrow}\right)$$

$$\vec{\boldsymbol{h}}_t = \sigma\left(U_{\rightarrow}\boldsymbol{x}_t + W_{\rightarrow}\vec{\boldsymbol{h}}_{t-1} + \boldsymbol{b}_{\rightarrow}\right)$$

재귀 신경망의 확장

❖ 딥러닝 재귀 신경망(Deep RNN)

- 여러 개의 재귀 신경망을 쌓아서 아래층의 출력을 바로 위층의 입력으로 받아들이도록 만든 모델

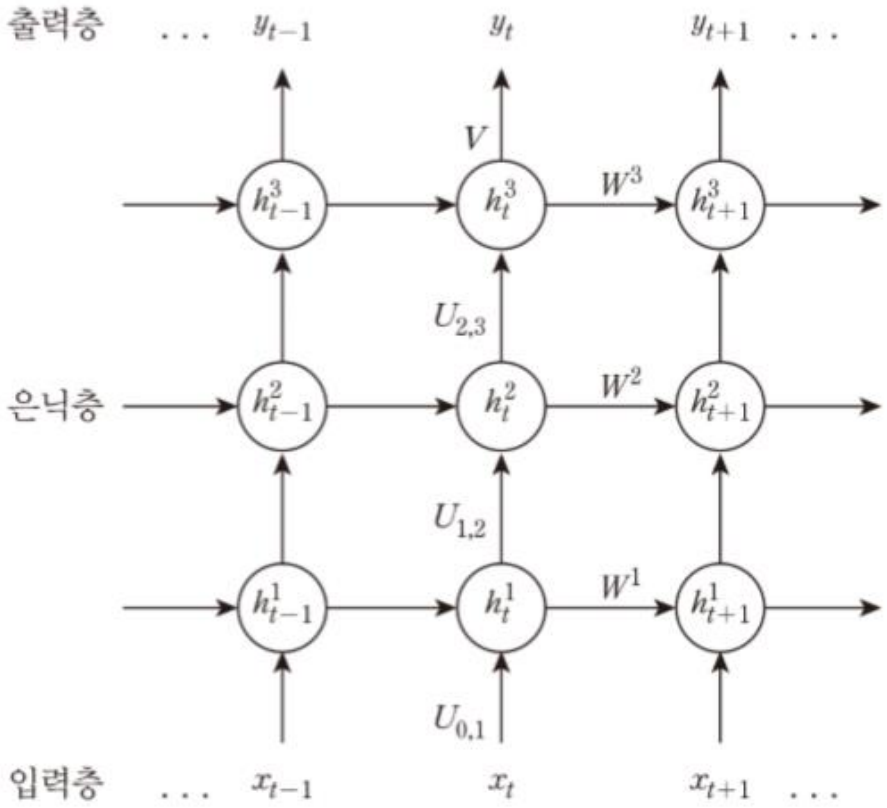


그림 5.62 딥러닝 재귀 신경망

for $l = 1$ to L

$\boldsymbol{h}_0^l \leftarrow \mathbf{0}$

for $t = 1$ to T

$\boldsymbol{h}_t^1 \leftarrow \sigma(U_{0,1}\boldsymbol{x}_t + W^1\boldsymbol{h}_{t-1}^1 + \boldsymbol{b}_1)$

for $l = 2$ to L

$\boldsymbol{h}_t^l \leftarrow \sigma(U_{l-1,l}\boldsymbol{h}_t^{l-1} + W^l\boldsymbol{h}_{t-1}^l + \boldsymbol{b}_l)$

$\boldsymbol{y}_t \leftarrow f(V\boldsymbol{h}_t^L + \boldsymbol{b}_o)$

재귀 신경망의 확장

❖ 딥러닝 양방향 재귀 신경망(Deep bidirectional RNN)

- 양방향 재귀 신경망과 딥러닝 재귀 신경망을 결합한 형태

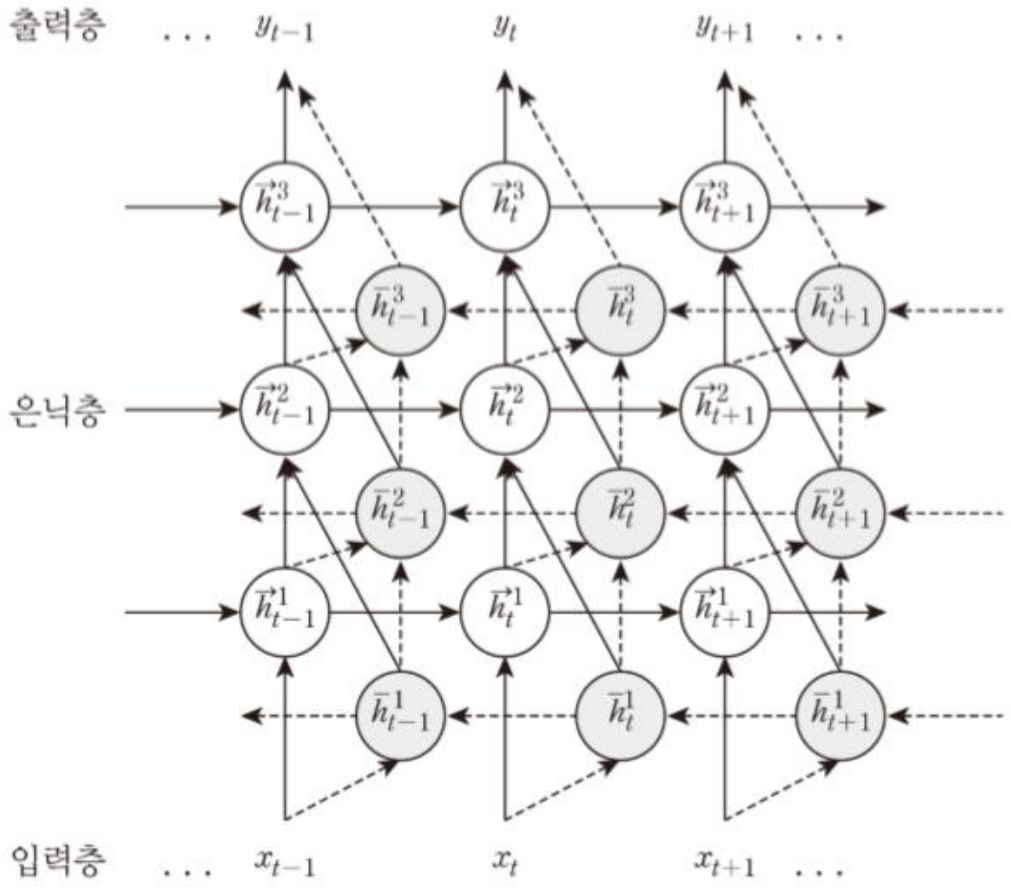


그림 5.63 딥러닝 양방향 재귀 신경망

$$\boldsymbol{y}_t = g\left(V_{\rightarrow}\overrightarrow{\boldsymbol{h}}_t^{L} + V_{\leftarrow}\overleftarrow{\boldsymbol{h}}_t^{L} + \boldsymbol{b}_o\right)$$

$$\overrightarrow{\boldsymbol{h}}_t^{l} = f\left(\overrightarrow{U}_{\rightarrow}^{l}\,\overrightarrow{\boldsymbol{h}}_t^{l-1} + \overrightarrow{U}_{\leftarrow}^{l}\,\overleftarrow{\boldsymbol{h}}_t^{l-1} + W_{\rightarrow}^{l}\,\overrightarrow{\boldsymbol{h}}_{t-1}^{l} + \boldsymbol{b}_{\rightarrow}^{l}\right)$$

$$\overleftarrow{\boldsymbol{h}}_t^{l} = f\left(\overleftarrow{U}_{\rightarrow}^{l}\,\overrightarrow{\boldsymbol{h}}_t^{l-1} + \overleftarrow{U}_{\leftarrow}^{l}\,\overleftarrow{\boldsymbol{h}}_t^{l-1} + W_{\leftarrow}^{l}\,\overleftarrow{\boldsymbol{h}}_{t+1}^{l} + \boldsymbol{b}_{\leftarrow}^{l}\right)$$

$$\overrightarrow{\boldsymbol{h}}_t^{1} = f\left(U_{\rightarrow}^{1}\,\boldsymbol{x}_t + W_{\rightarrow}^{1}\,\overrightarrow{\boldsymbol{h}}_{t-1}^{1} + \boldsymbol{b}_{\rightarrow}^{1}\right)$$

$$\overleftarrow{\boldsymbol{h}}_t^{1} = f\left(U_{\leftarrow}^{1}\,\boldsymbol{x}_t + W_{\leftarrow}^{1}\,\overleftarrow{\boldsymbol{h}}_{t+1}^{1} + \boldsymbol{b}_{\leftarrow}^{1}\right)$$

재귀 신경망의 확장

❖ **잔차연결 딥러닝 재귀 신경망(Deep RNN with Residual Connection)**

- 딥러닝 재귀 신경망에 층을 건너 뛸 수 있는 지름길 연결(skip connection)을 추가한 모델

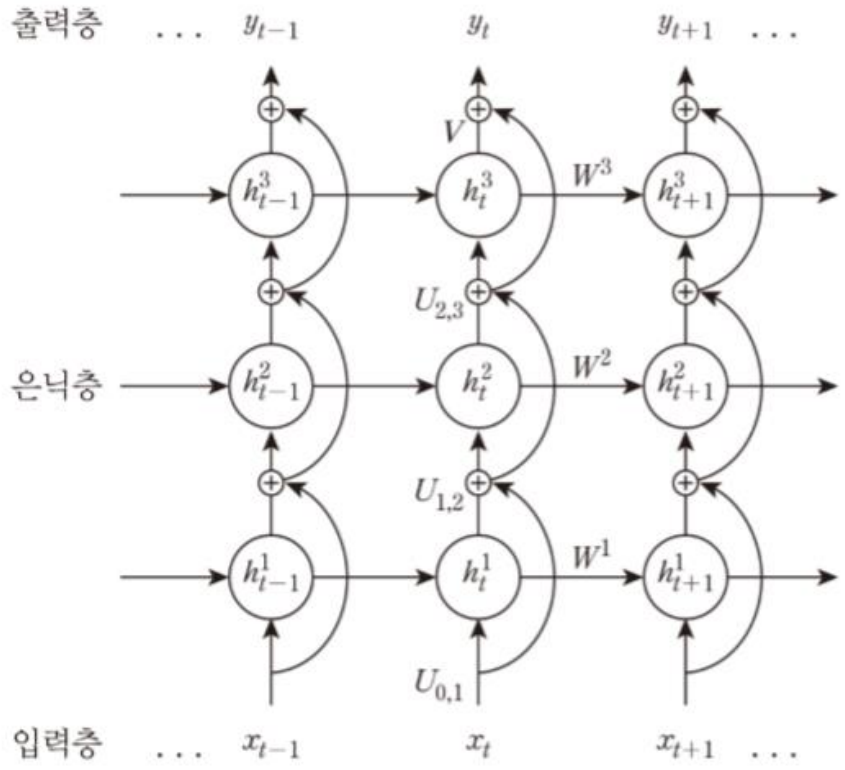


그림 5.64 잔차연결 딥러닝 재귀 신경망

$$\boldsymbol{h}_t^1 \leftarrow \sigma(U_{0,1}\boldsymbol{x}_t + W^1\boldsymbol{h}_{t-1}^1 + \boldsymbol{b}_1) + \boldsymbol{x}_t$$

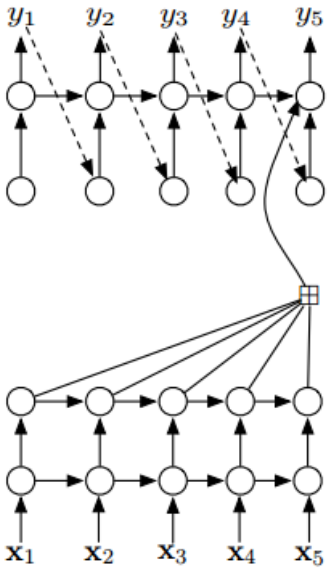
$$\boldsymbol{h}_t^l \leftarrow \sigma(U_{l-1,l}\boldsymbol{h}_t^{l-1} + W^l\boldsymbol{h}_{t-1}^l + \boldsymbol{b}_l) + \boldsymbol{h}_{l-1}$$

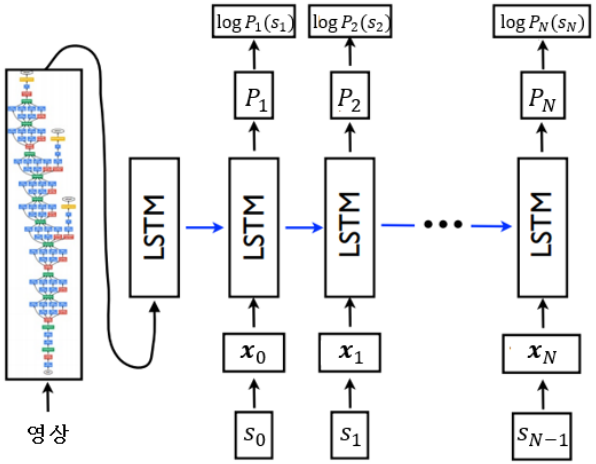
4.6 재귀 신경망의 적용 분야

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.
```

❖ **적용분야**

- **자연어 처리**
  - 언어 모델(language model)
  - 문장 생성
  - 문서 생성
  - 기계 번역(machine translation)
- **음성 인식**
- **영상 주석달기**(Image captioning)





"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."