RNN 첫걸음

시퀀스(sequence) 데이터 : 소리, 문자열, 주가 등 순차적으로 들어있는 데이터

- 시퀀스 데이터는 독립동등분포(i.i.d.) 가정을 잘 위배하기 때문에 순서를 바꾸거나 과거정보에 손실이 발생하면 데이터의 확률분포도 바뀜.

  ex) 개가 사람을 물었다 ≠ 사람이 개를 물었다.

- 이전 시퀀스의 정보를 가지고 앞으로 발생할 데이터의 확률분포를 다루기 위해 조건부확률을 이용할 수 있음.

$$
\begin{aligned}
P(X_1, \ldots, X_t) &= P(X_t | X_1, \ldots, X_{t-1}) P(X_1, \ldots, X_{t-1}) \\
&= P(X_t | X_1, \ldots, X_{t-1}) P(X_{t-1} | X_1, \ldots, X_{t-2}) \times \\
&\qquad \times P(X_1, \ldots, X_{t-2}) \\
&= \prod_{s=1}^{t} P(X_s | X_{s-1}, \ldots, X_1)
\end{aligned}
$$

⇒ 초기시점인 X_1 에서부터 바로 직전 과거의 정보인 X_{s-1} 까지의 정보를 사용하여 현재시점인 X_s 를 모델링하는 조건부확률

요 기호는 $s = 1, \ldots, t$ 까지 모두 곱하라는 기호입니다

(위 식은 과거의 모든 정보를 사용하지만, 꼭 모든 과거정보들이 필요하진 않음)

- 시퀀스 데이터를 다루기 위해선 길이가 가변적인 데이터를 다룰 수 있는 모델이 필요.

$$X_t \sim P(X_t | \boxed{X_{t-1}, \ldots, X_1})$$

$$X_{t+1} \sim P(X_{t+1} | \boxed{X_t, X_{t-1}, \ldots, X_1})$$

조건부에 들어가는 데이터 길이는 가변적입니다

$$X_t \sim P(X_t | \boxed{X_{t-1}, \ldots}, X_1)$$

$$X_{t+1} \sim P(X_{t+1} | \boxed{X_t, X_{t-1}}, \ldots, X_1)$$

고정된 길이 τ 만큼의 시퀀스만 사용하는 경우 $AR(\tau)$ (Autoregressive Model) 자기회귀모델이라고 부릅니다

$$X_t \sim P(X_t | X_{t-1}, \boxed{\ldots, X_1}) \rightarrow H_t$$

$$X_{t+1} \sim P(X_{t+1} | X_t, \boxed{X_{t-1}, \ldots, X_1}) \rightarrow H_{t+1}$$

또 다른 방법은 바로 이전 정보를 제외한 나머지 정보들을 H_t 라는 잠재변수로 인코딩해서 활용하는 잠재 AR 모델입니다

$$X_t \sim P(X_t | X_{t-1}, \boxed{H_t})$$

$$X_{t+1} \sim P(X_{t+1} | X_t, \boxed{H_{t+1}})$$

$$H_t = \mathrm{Net}_\theta(H_{t-1}, X_{t-1})$$

잠재변수 H_t 를 신경망을 통해 반복해서 사용하여 시퀀스 데이터의 패턴을 학습하는 모델이 RNN 입니다

RNN모델 이해하기.

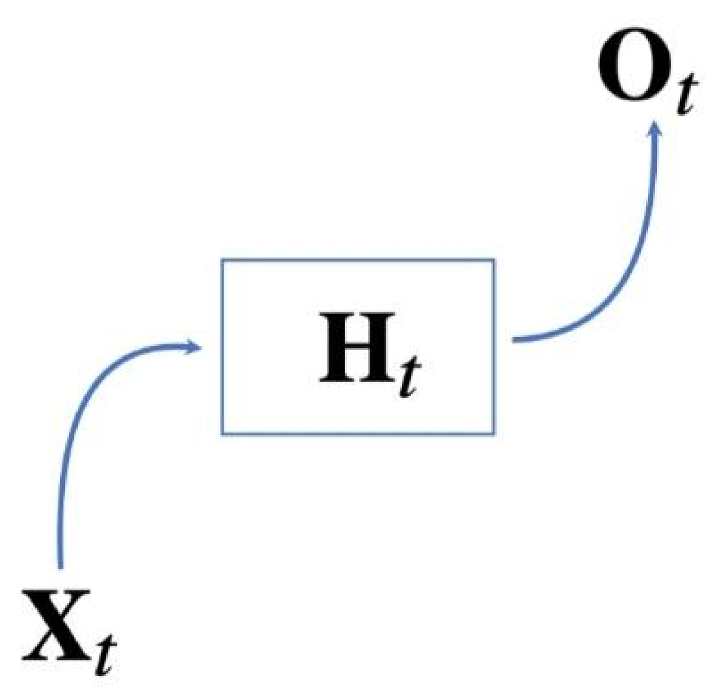
· 가장 기본적인 RNN은 MLP와 유사한 모양이지만, 과거의 정보를 다루지 않음.

이 모델은 과거의 정보를 다룰 수 없습니다

$$\mathbf{O}_t = \mathbf{H}_t\mathbf{W}^{(2)} + \mathbf{b}^{(2)}$$

$$\mathbf{H}_t = \sigma(\mathbf{X}_t\mathbf{W}^{(1)} + \mathbf{b}^{(1)})$$

잠재변수 활성화함수 가중치행렬 bias

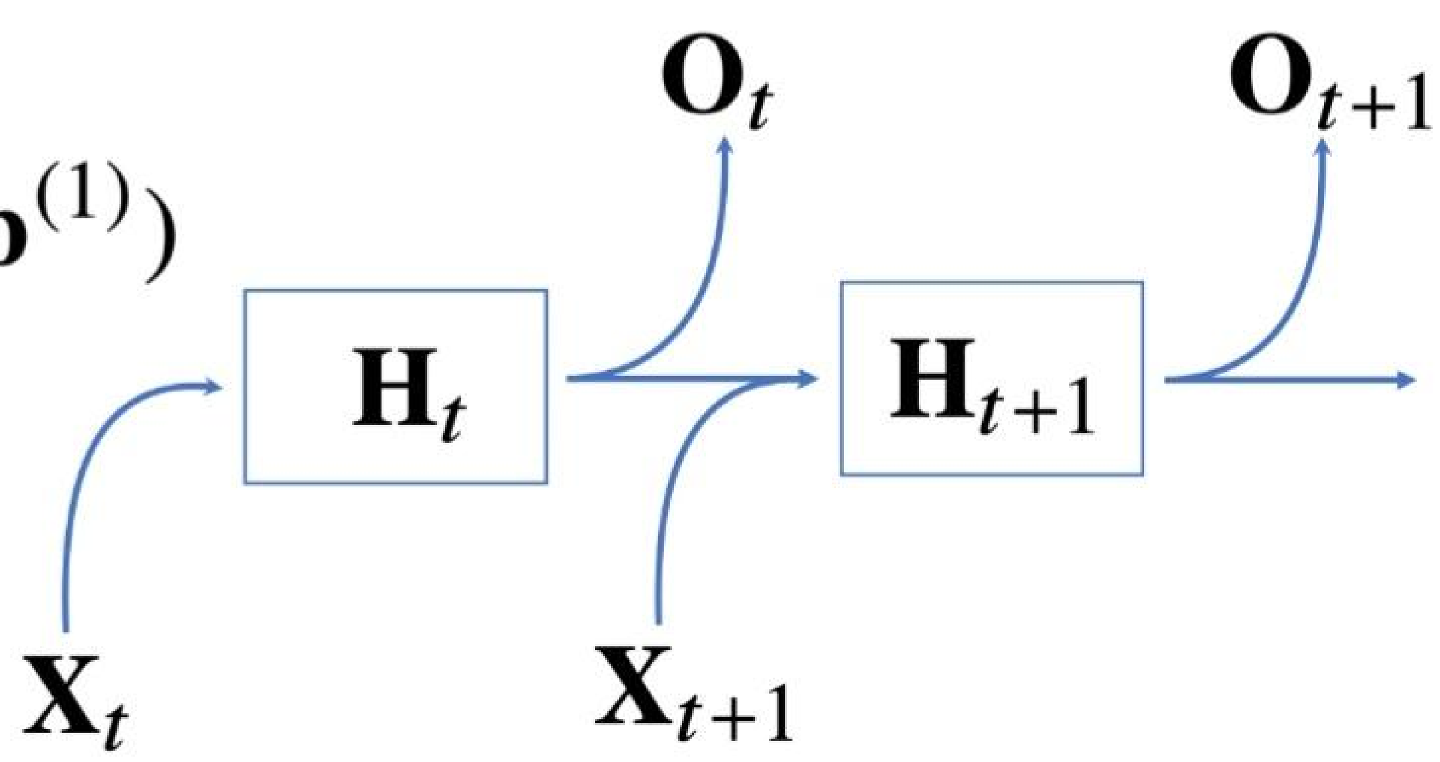


· RNN은 이전순서의 잠재변수와 현재의 입력을 활용하여 모델링 함.

$$\mathbf{O}_t = \mathbf{H}_t\mathbf{W}^{(2)} + \mathbf{b}^{(2)}$$

$$\mathbf{H}_t = \sigma(\mathbf{X}_t\mathbf{W}_X^{(1)} + \mathbf{H}_{t-1}\boxed{\mathbf{W}_H^{(1)}} + \mathbf{b}^{(1)})$$

잠재변수인 $\mathbf{H}_t$ 를 복제해서 다음 순서의
잠재변수를 인코딩하는데 사용합니다

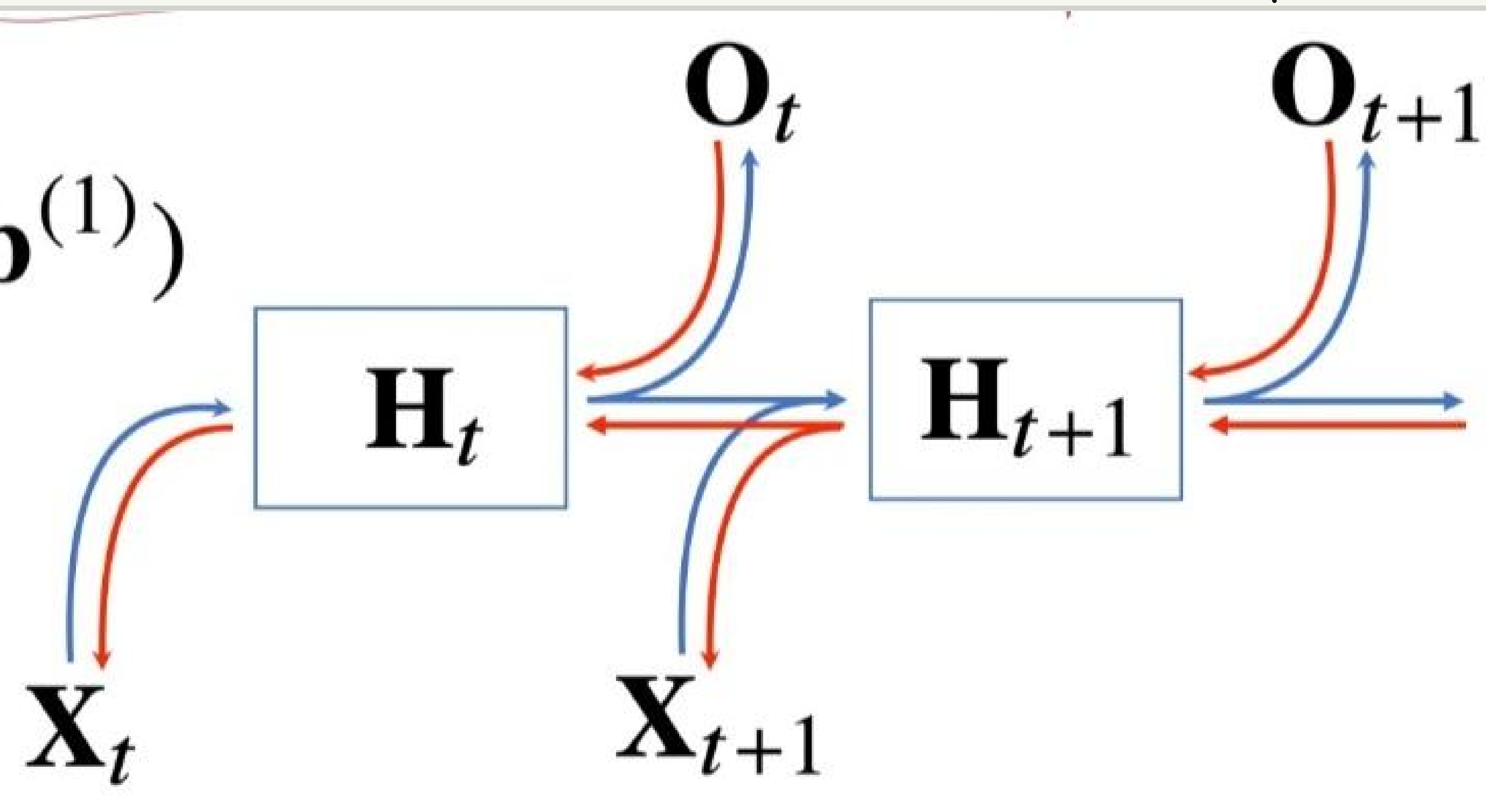


· RNN 역전파는 잠재변수의 연결그래프에 따라 순차적으로 계산함.

$$\mathbf{O}_t = \mathbf{H}_t\mathbf{W}^{(2)} + \mathbf{b}^{(2)}$$

$$\mathbf{H}_t = \sigma(\mathbf{X}_t\mathbf{W}_X^{(1)} + \mathbf{H}_{t-1}\mathbf{W}_H^{(1)} + \mathbf{b}^{(1)})$$

이를 Backpropagation Through Time
(BPTT) 이라 하며 RNN 의 역전파 방법이다



BPTT

- BPTT 를 통해 RNN 의 가중치행렬의 미분을 계산해보면 아래와 같이 미분의 곱으로 이루어진 항이 계산됩니다

$h_t = f(x_t, h_{t-1}, w_h)$ and $o_t = g(h_t, w_o)$.

$$L(x, y, w_h, w_o) = \sum_{t=1}^{T} \ell(y_t, o_t)$$

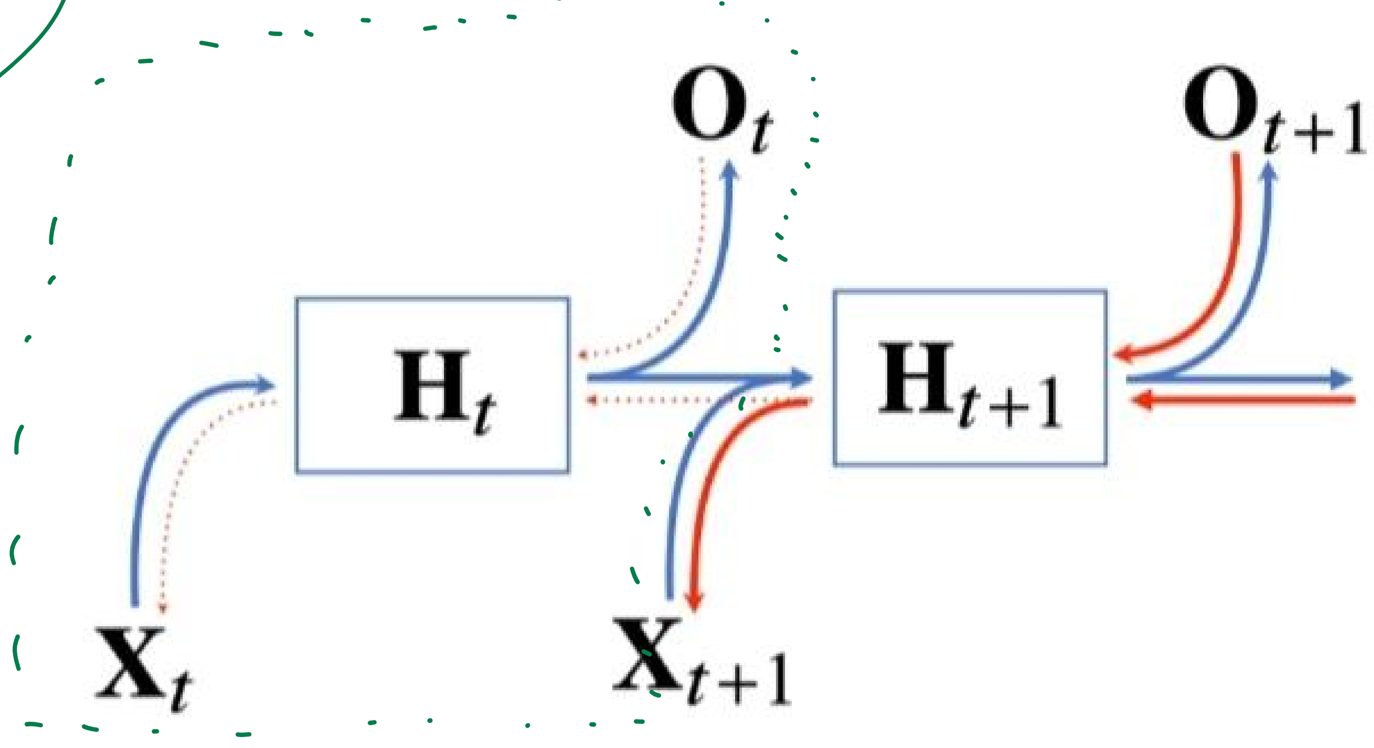
$$\partial_{w_h} L(x, y, w_h, w_o) = \sum_{t=1}^{T} \partial_{w_h} \ell(y_t, o_t) = \sum_{t=1}^{T} \partial_{o_t} \ell(y_t, o_t) \partial_{h_t} g(h_t, w_h) [\partial_{w_h} h_t]$$

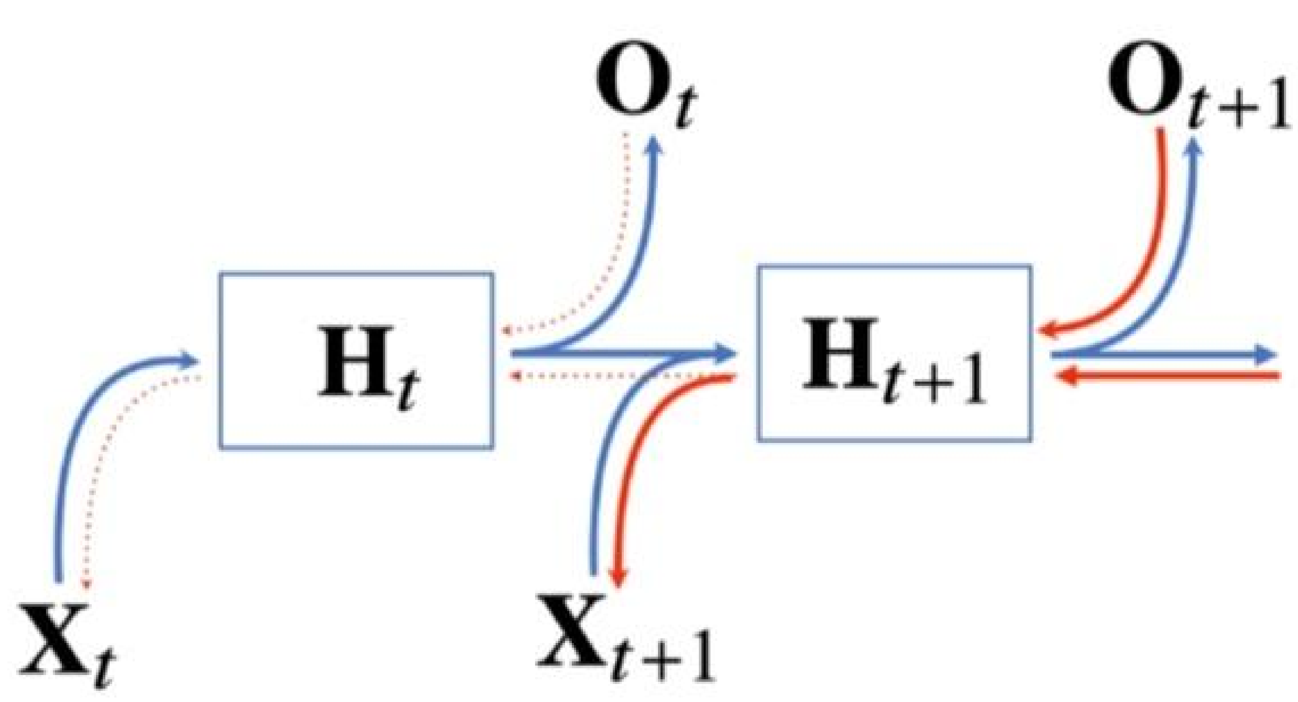
시퀀스 길이가 길어질수록
이 항은 불안정해지기 쉽습니다

$$\partial_{w_h} h_t = \partial_{w_h} f(x_t, h_{t-1}, w_h) + \sum_{i=1}^{t-1} \boxed{\left(\prod_{j=i+1}^{t} \partial_{h_{j-1}} f(x_j, h_{j-1}, w_h) \right)} \partial_{w_h} f(x_i, h_{i-1}, w_h)$$

- 시퀀스 길이가 길어지는 경우 BPTT 를 통한 역전파 알고리즘의 계산이 불안정해지므로 길이를 끊는 것이 필요합니다

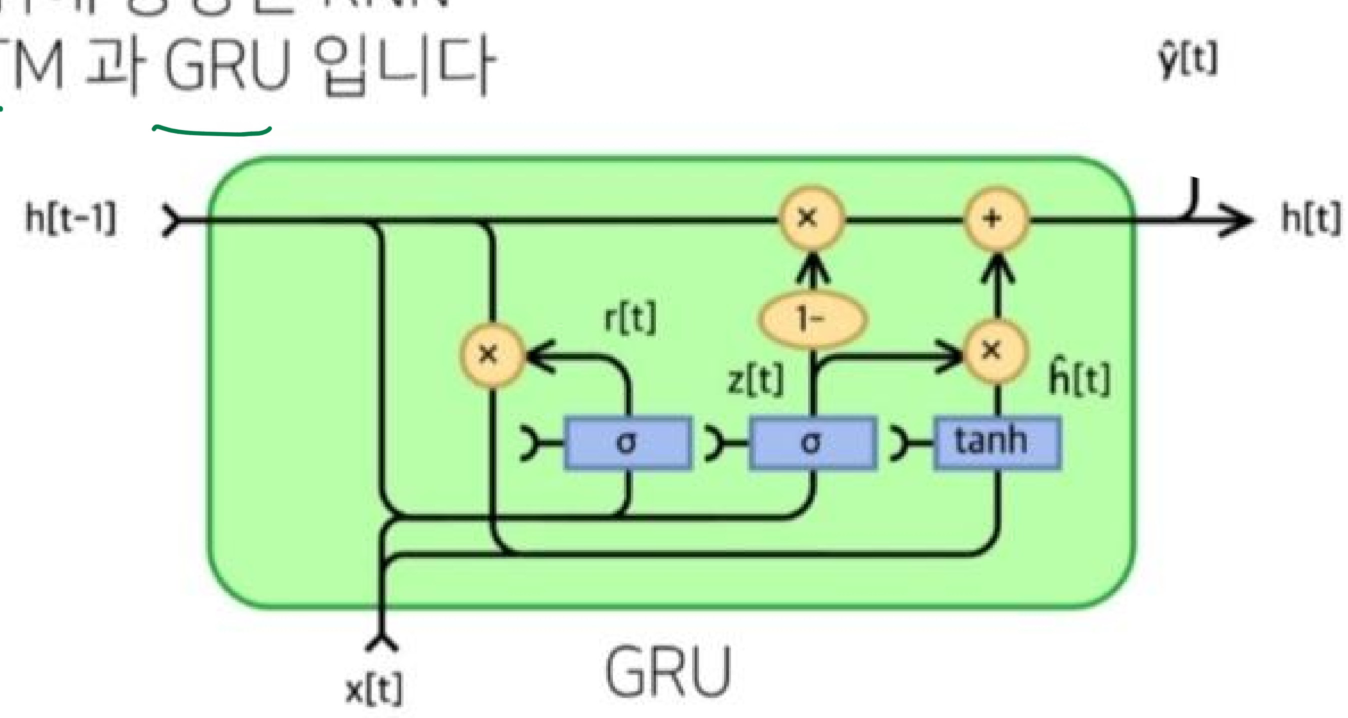
이를 truncated BPTT 라 부릅니다



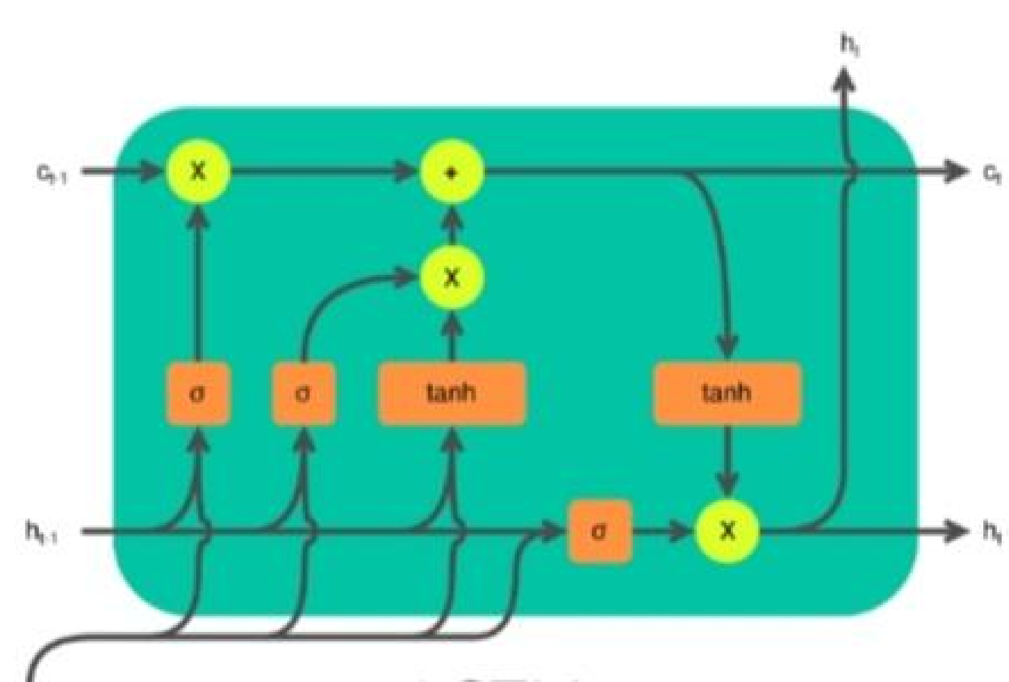


- 이런 문제들 때문에 Vanilla RNN 은 길이가 긴 시퀀스를 처리하는데 문제가 있습니다

이를 해결하기 위해 등장한 RNN
네트워크가 LSTM 과 GRU 입니다



GRU



LSTM