

1. 시계열 문제

시간에 따라 변하는 데이터를 사용하여 추이를 분석하는 것
(추세 파악, 향후 전망 예측)

- 1) 불규칙 변동: 규칙성 없음, 예측 불가, 우연적
- 2) 추세 변동: 장기적인 변화 추세, 장기간에 걸쳐서 증가, 감소, 일정한 상태 유지, 단기간에 변동 찾기는 불가능
- 3) 순환 변동: 2~3년 정도의 일정 기간을 주기로 하여 순환적으로 나타남, 곡선 그리며 추세 변동
- 4) 계절 변동: 계절적 영향과 사회적 관습에 따라 1년 주기로 발생

2. AR, MA, ARMA, ARIMA

1) AR 모델

이전 관측 값이 이후 관측 값에 영향을 준다는 아이디어에 대한 모형으로 자기 회귀 모델이라고도 함

$$\underbrace{Z_t}_{\textcircled{1}} = \underbrace{\Phi_1 Z_{t-1} + \Phi_2 Z_{t-2} + \cdots + \Phi_p Z_{t-p}}_{\textcircled{2}} + \underbrace{a_t}_{\textcircled{3}}$$

2) MA 모델

트렌드가 변화

$$\underbrace{Z_t}_{\textcircled{1}} = \underbrace{\theta_1 a_{t-1} + \theta_2 a_{t-2} + \cdots + \theta_p a_{t-p}}_{\textcircled{2}} + \underbrace{a_t}_{\textcircled{3}}$$

3) ARMA 모델

AR과 MA 섞은 모델

$$Z_t = a + \Phi_1 Z_{t-1} + \cdots + \Phi_p Z_{t-p} + \theta_1 a_{t-1} + \cdots + \theta_q a_{t-q} + a_t$$

4) ARIMA 모델

모형은 자기 회귀와 이동 평균을 둘 다 고려하는 모형, 추세까지 고려

1. ARIMA() 함수를 호출하여 사용하는데, ARIMA(p,d,q) 함수에서 쓰는 파라미터는 다음과 같습니다.

- **p**: 자기 회귀 차수
- **d**: 차분 차수
- **q**: 이동 평균 차수

2. fit() 메서드를 호출하고 모델에 데이터를 적용하여 훈련시킵니다.

3. predict() 메서드를 호출하여 미래의 추세 및 동향에 대해 예측합니다.

3. 순환 신경망(RNN)

시간적으로 연속성이 있는 데이터 처리 목적

반복되는 순환 구조 가짐

1) 일대일

2) 일대다

3) 다대일

4) 다대다

5) 동기화 다대다

(1) RNN 계층과 셀

내장된 계층 및 셀 레벨의 API 제공

오직 하나의 단계만 처리

nn.RNNCell, nn.GRUCell, nn.LSTMCell

4. RNN 구조

입력층, 은닉층, 출력층, 가중치 3개

1) 은닉층

$$h_t = \tanh(\hat{y}_t)$$

$$\hat{y}_t = W_{hh} \times h_{t-1} + W_{xh} \times x_t$$

2) 출력층

$$\hat{y}_t = \text{softmax}(W_{hy} \times h_t)$$

3) 오차

각 단계마다 오차를 측정, 이후 평균 제곱 오차 use

4) 역전파

5. LSTM

1) 순전파

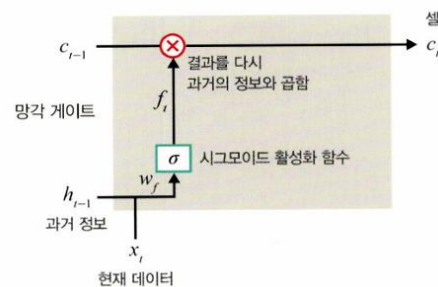
(1) 망각 게이트

과거 정보를 어느 정도 기억할지 결정

$$f_t = \sigma(w_f[h_{t-1}, x_t])$$

$$c_t = f_t \cdot c_{t-1}$$

▼ 그림 7-16 망각 게이트



(2) 입력 게이트

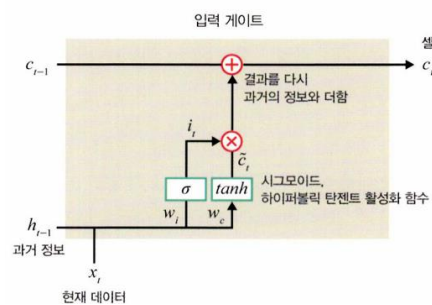
현재 정보를 기억하기 위해 만들어짐

$$i_t = \sigma(w_i[h_{t-1}, x_t])$$

$$\tilde{c}_t = \tanh(w_c[h_{t-1}, x_t])$$

$$c_t = c_{t-1} + i_t \cdot \tilde{c}_t$$

▼ 그림 7-17 입력 게이트



(3) 셀

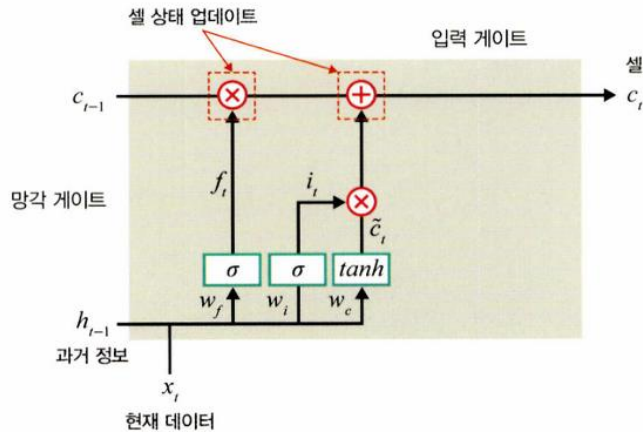
각 단계에 대한 은닉 노드

총합을 사용하여 셀 값 반영, 기울기 소멸 문제 해결

$$f_t = \sigma(w_f[h_{t-1}, x_t])$$

$$c_t = c_{t-1} + i_t \cdot \tilde{c}_t$$

▼ 그림 7-18 셀



(4) 출력 게이트

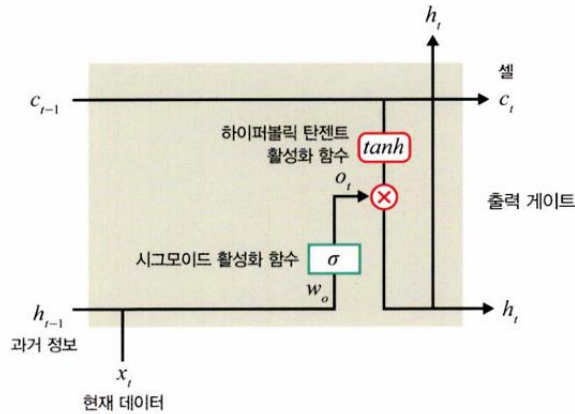
과거 정보와 현재 데이터를 사용하여 뉴런의 출력을 결정

$$o_t = \sigma(w_o [h_{t-1}, x_t])$$

$$h_t = o_t \cdot \tanh(c_{t-1})$$

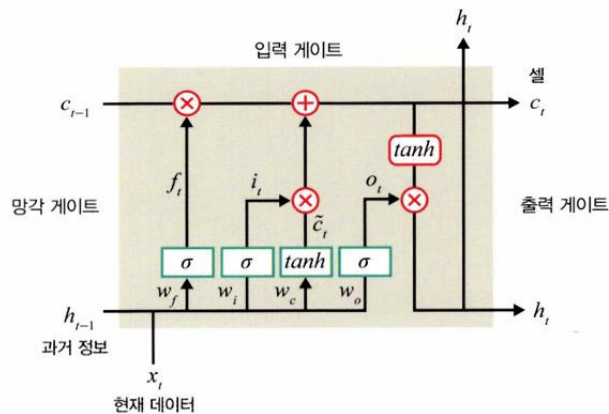
▼ 그림 7-19 출력 게이트

이전 은닉 상태 + 1번째 입력 = 다음 은닉 상태(=출력)

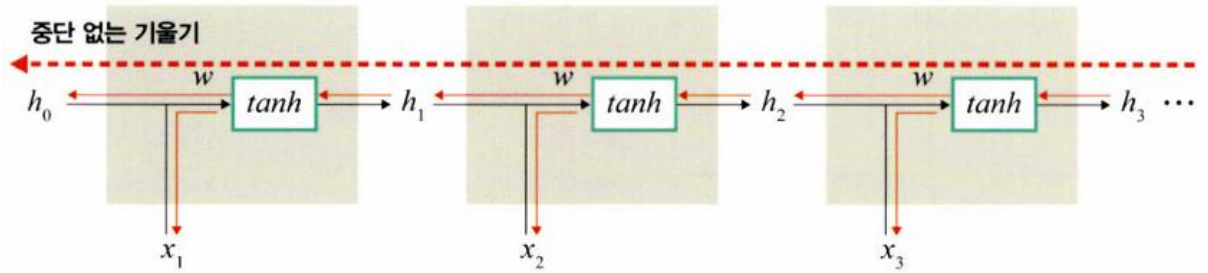


다음 그림은 망각 게이트, 입력 게이트, 출력 게이트를 모두 표현한 것입니다.

▼ 그림 7-20 LSTM 전체 게이트



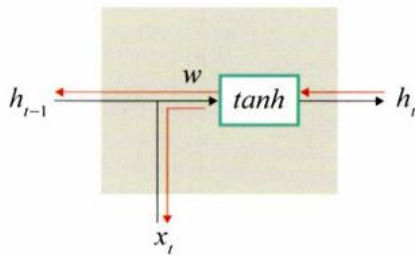
2) 역전파



다음은 역전파를 수행하기 위한 공식입니다.

$$\begin{aligned}
 t_t &= \tanh(w_{hh}h_{t-1} + w_{xh}x_t) \\
 &= \tanh\left((w_{hh} \quad w_{xh}) \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}\right) \\
 &= \tanh\left(w \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}\right)
 \end{aligned}$$

♥ 그림 7-22 입력층으로의 역전파



6. 게이트 순환 신경망 (GRU)

망각 게이트와 입력 게이트를 하나로 합친 것, 별도의 업데이트 게이트로 구성됨

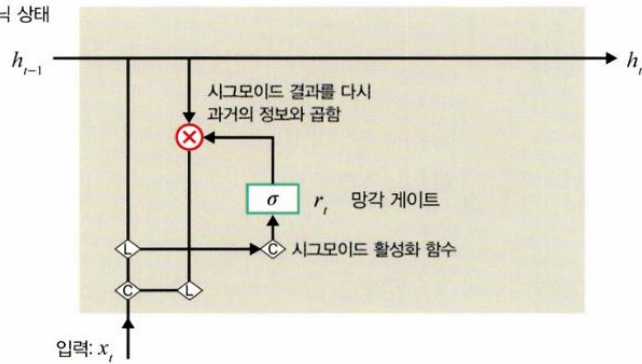
1) 망각 게이트

과거 정보를 적당히 초기화, 시그모이드 함수 출력으로 이용

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

▼ 그림 7-26 망각 게이트

은닉 상태

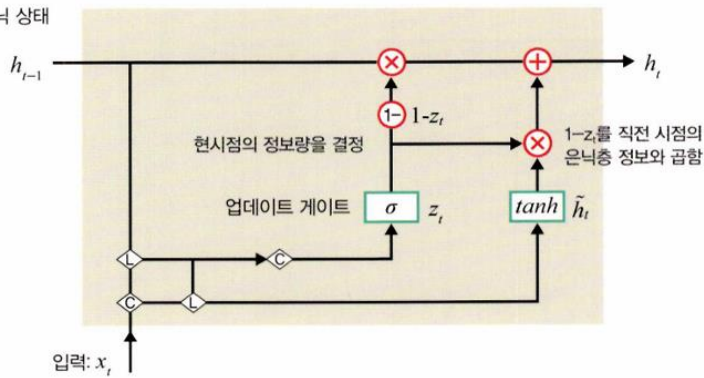


2) 업데이트 게이트

과거와 현재 정보의 최신화 비율을 결정하는 역할을 함

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

은닉 상태



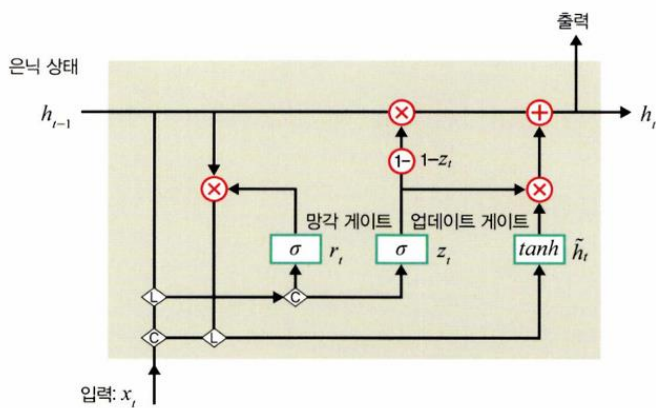
3) 후보군

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

4) 은닉층 계산

마지막으로 업데이트 게이트 결과와 후보군 결과 곱함

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$



7. 양방향 RNN

1) 양방향 RNN 구조

하나의 출력 값을 예측하는 데 메모리 셀 두 개를 사용

첫 번째 메모리 셀은 이전 시점의 은닉 상태를 전달받아 현재의 은닉 상태 계산

두 번째 메모리 셀은 다음 시점의 은닉 상태를 전달받아 현재의 은닉 상태 계산

그리고 이 값 두 개를 모두 출력층에서 출력 값을 예측하는 데 사용