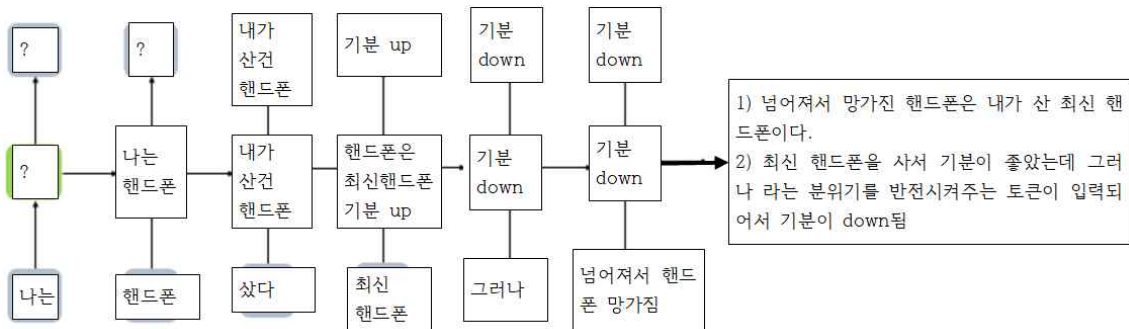
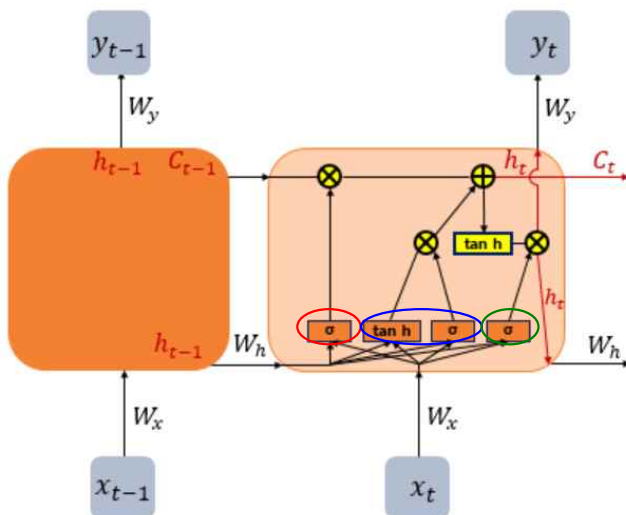


LSTM은 Long short term Memory의 약자로 장단기 메모리이다. 기존 RNN은 충분한 기억력을 가지고 있지 못해 뒤로 갈수록 정보가 손실된다. 아래 이미지를 통해 예를 들겠다.



만약 위처럼 Sequence Length가 짧다면 RNN도 충분히 기억할 능력이 된다. 하지만 만약에 문장이 “나는 핸드폰을 샀다. 그 핸드폰은 최신형 핸드폰이다. 그러나 나는 넘어져서 핸드폰을 망가졌다. 너무 슬펐다. 그래서 대리점에 가서 수리 관련 문의를 했다. 대리점에서는 고객 과실로 인한 파손은 수리가 불가능하다고 했다. 하지만 유상 수리일 경우 잠시 사용을 위한 임대 시스템이 존재한다. 따라서 나는 대리점에서 **핸드폰**을 받고 집에 왔다.”처럼 아주 긴 문장이 있다고 가정하자. 그렇다면 마지막 “핸드폰”은 “새로산 핸드폰”인가 아니면 “대리점에서 임대해준 핸드폰”인가에 대해 RNN은 기억하지 못한다. 이를 장기 의존성 문제 즉 Gradient Vanishing 문제이다.

아래 이미지는 LSTM의 기본적인 구조이다.



기존 RNN과 비교해 보았을 때 $C_t = [C_1, C_2, C_3, \dots, C_t]$ 라는 새로운 길이 생긴 것을 확인할 수 있다. 이것을 셀 상태라고 불리우며 셀 상태를 구하기 위해 삭제 게이트(빨간 원), 입력 게이트(파란 원) 출력 게이트(초록 원)가 존재한다.

삭제 게이트와 입력 게이트를 통해 장기 상태를 기억하고 출력 게이트는 단기 상태를 의미한다. 각 게이트들을 계산하는 수식이 존재하긴 하지만 컴퓨터가 알아서 연산을 해주고 복잡하기 때문에 Manual로 연산하는 과정은 생략 하도록 한다. 코드를 통해서 RNN과의 차이를 확인하도록 하겠다.