

# Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

Kyunghyun Cho et al

---

Choe Haein

2022.08.07

University Of Seoul

# Table of Contents

1. RNN

2. LSTM

3. GRU

4. After this paper

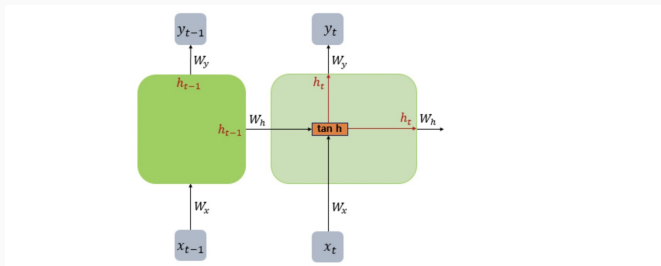
5. Reference

RNN

---

# Vanilla RNN

- RNN은 은닉층(메모리 cell)의 데이터를 다음 시점으로 보낸다
- 과거 정보를 반영하는 방식으로 시계열 데이터, 순차 데이터 입력
- 단어 단위로 순차데이터를 입력받아 기계번역 분야에 활용됨



입력되는 단어가 많아지면 역전파 과정에서의 계산량 때문에 학습에 문제가 발생할 수 있음

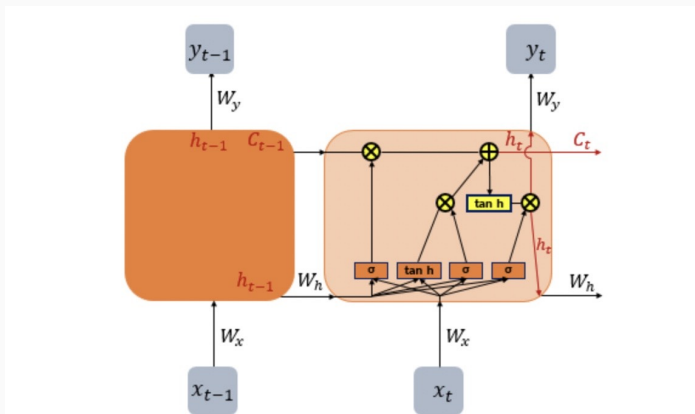
- Vanishing Gradient (기울기 소실) : Derivatives are less than 1
- Exploding Gradient (기울기 폭주) : Derivatives are greater than 1

-> 너무 오래 전 정보에 대해서 효과적이지 못함 (초기 은닉층의 영향력 감소하게 됨)

# LSTM

---

이전 시점의 정보를 얼마나 기억할 지를 결정하는 구조를 추가하였음

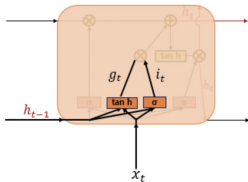


- Input Gate : 현재 정보를 보존하는 게이트
- Forget Gate : 현재 시점의 입력과 이전 시점의 은닉층을 받아 삭제할 정보의 비율을 결정함
- Cell State : Input Gate 와 Forget Gate 의 결과값을 곱하여 계산
- Output Gate : Forget Gate를 통과한 값과 Cell State 값이 현재 시점 은닉층 결정



# LSTM

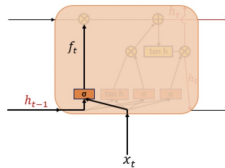
(1) 입력 게이트



$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$$

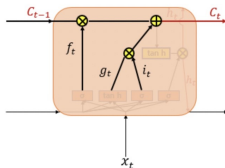
$$g_t = \tanh(W_{xg}x_t + W_{hg}h_{t-1} + b_g)$$

(2) 삭제 게이트



$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$

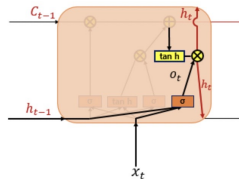
(3) 셀 상태



$$C_t = f_t \odot C_{t-1} + i_t \odot g_t$$

entrywise production

(4) 출력 게이트와 은닉 상태



$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$

$$h_t = o_t \odot \tanh(c_t)$$

GRU

---

LSTM 의 Input Gate 와 Forget Gate 를 합쳐 구조를 단순화시킴



After this paper

---

- RNN 구조에서 벗어나서, Long-Term Dependency 를 해결할 수 있는 새로운 아키텍처가 존재함
- 현재 주류가 되는 GPT, BERT 같은 알고리즘은 트랜스포머 아키텍처를 활용하여 우수한 성능을 내고 있음
- 그래서 그것도 시간 남으면 해보면 좋을것같다..

## Reference

---

- [딥러닝 기계 번역] Seq2Seq: Sequence to Sequence Learning with Neural Networks
- [Paper Review] Learning Phrase Representations using RNN Encoder-Decoder
- [DSBA]CS224N-08.Machine Translation, Seq2Seq, Attention
- [논문 리뷰] Learning Phrase Representation using RNN Encoder-Decoder for Statistical Machine Translation
- 풀잇스쿨 NLP Bootcamp
- 딥러닝을 이용한 자연어 처리 입문