

Week4-2

1. 학습정리

1. RNN

1. 가장 큰 특징

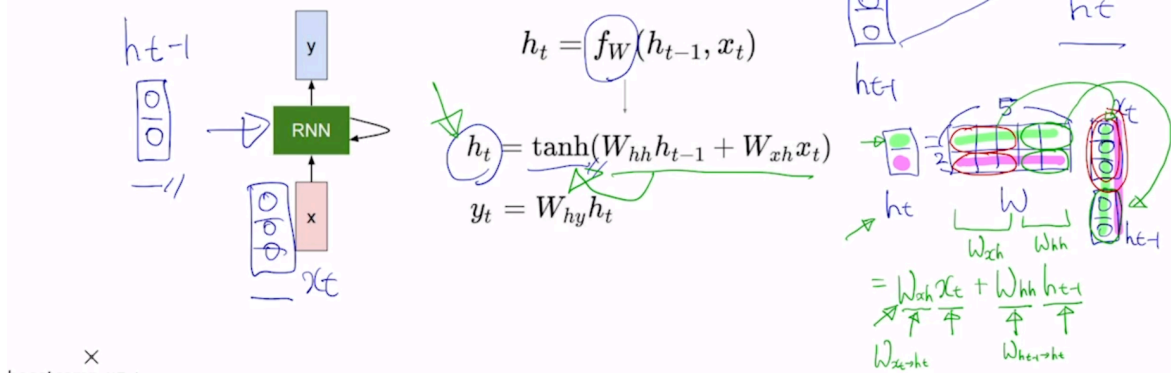
1. 파라미터 W 는 모든 타임스텝에서 동일한 값을 공유

Recurrent Neural Network

Vanilla RNN

- How to calculate the hidden state of RNNs

- The state consists of a single "hidden" vector \mathbf{h}



2. \mathbf{h}_t 계산법

1. 입력벡터 \mathbf{x}_t 와 이 전 단계의 히든벡터 \mathbf{h}_{t-1} 을 붙이고 가중치 W 와 내적

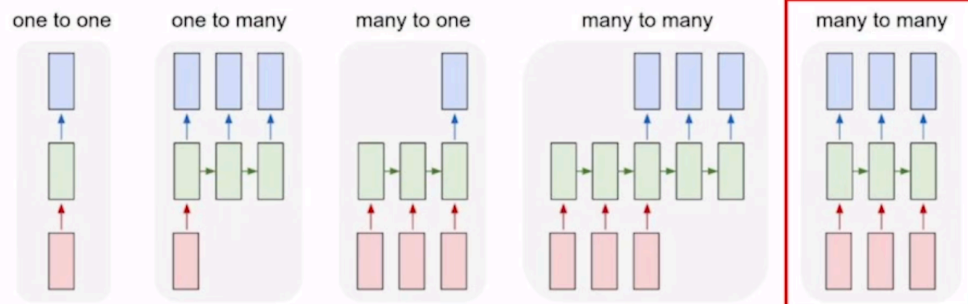
3. RNN의 타입

Types of RNNs

Typ

- Sequence-to-sequence

- Video classification on frame level



1. 원투원

1. 시퀀스 데이터가 아닌것

2. 원 투 매니

1. 예 : 이미지 캡셔닝

1. 사진데이터에 대한 설명을 붙이는 것

2. RNN 구조에서는 매 타임스텝마다 입력이 주어졌는데 이 경우 추가적으로 주어질 입력이 없음
 1. 모든 값이 0으로 채워진 입력을 줌
3. 매니 투 원
 1. 예 : 감정분석
 2. 마지막 타임스텝에서 나온 히든벡터를 가지고 아웃풋 레이어를 적용
4. 매니 투 매니 1
 1. 예 : 머신번역
 2. 마지막 타임스텝 까지 읽은 후 한글 번역을 순차적으로 생성
5. 매니 투 매니 2
 1. 입력이 주어질 때 마다 예측
 2. 실시간성이 요구되는 것에 사용
4. RNN 역전파 BPTT
 1. 한 번에 몇 개의 타임스텝만 고려하도록 잘라서 학습
 2. 학습된 내용은 히든 벡터 내의 특정 위치가 기억하고 있음
 1. 예를 들면 벡터 내 어떤 위치는 괄호를 열고 닫는 것을 기억
 3. 문제점
 1. 배니싱 그래디언트, 그래디언트 폭발
5. LSTM
 1. RNN의 그래디언트 배니싱, 폭발 문제를 해결
 2. 롱 텀 디펜던시 문제를 개선한 모델
 3. 기존 RNN과의 입출력 차이

$$h_t = f_w(x_t, h_{t-1})$$

$$\{C_t, h_t\} = \text{LSTM}(x_t, C_{t-1}, h_{t-1})$$

4. 셀 스테이트 벡터
 1. 여러가지 필요로 하는 완전한 정보를 담고있는 벡터
5. 히든 스테이트 벡터
 1. 셀 스테이트 벡터를 가공해서 해당 타임스텝에서 노출할 정보만 남긴 벡터
 2. 해당 타임스텝에서 아웃풋 레이어의 입력벡터로 사용
6. 포갯게이트
 1. 셀 스테이트 벡터에서 무엇을 얼마나 잊을지를 퍼센트로 저장한 벡터
 2. 셀 스테이트 벡터와 곱해짐
7. 게이트게이트
 1. 인풋 게이트를 통과한 벡터를 셀에 얼마나 쓸지
8. 셀 스테이트 벡터는 포갯게이트를 통과한 벡터와 게이트게이트를 통과한 벡터의 합으로 갱신됨
9. 아웃게이트
 1. 히든 스테이트 벡터를 만듦
6. GRU

1. LSTM을 경량화해서 적은 메모리, 빠른 계산시간
2. 포갯게이트 대신 1-인풋게이트에 해당하는 값을 사용
 1. 인풋게이트가 커질수록 포갯게이트에 해당하는 값이 작아짐
 2. 이전 히든 스테이트와 현재 h_t 의 틸다를 가중평균으로 더함
3. 역전파
 1. 곱셈이 아니라 더해서 원하는 정보를 만들기 때문에 그래디언트 배니싱 문제가 없음
7. GRU 랭귀지 모델 실습
 1. 티쳐 소싱
 1. 처음은 모델이 초기화 상태라 학습이 잘 되지 않음
 2. 그래서 처음 입력 전 타임스텝에서 나온 아웃풋을 입력으로 주지 않고 원래의 인풋을 입력으로 줌
 2. 양방향 RNN
 1. 타임스텝 반대로 돌면서 새로운 특징 발견 가능
 2. 아웃풋레이어에 순방향 역방향이 컨кат 되기 때문에 사이즈가 달라짐

Summary on RNN/LSTM/GRU

- RNNs allow a lot of **flexibility** in architecture design
- Vanilla RNNs are **simple** but don't work very well
- Backward flow of gradients in RNN can **explode or vanish**
- Common to use LSTM or GRU: their additive interactions **improve gradient flow**

2. 피어세션

1. 퍼더퀘스천

1. BPTT 이외에 RNN/LSTM/GRU의 구조를 유지하면서 gradient vanishing/exploding 문제를 완화할 수 있는 방법
 1. <https://newsight.tistory.com/94>
2. RNN/LSTM/GRU 기반의 Language Model에서 초반 time step의 정보를 전달하기 어려운 점을 완화할 수 있는 방법이
 1. 1D 컨볼루션 레이어와, 로 입력 데이터 압축

2.