Week3-4

1. **학습정리**
    1. 시퀀스 데이터
        1. 순차적으로 들어오는 데이터 (문자열 등)
        2. 시계열 데이터 : 시간 순서대로 들어오는 데이터
        3. 독립동등분포(i.i.d.) 가정을 위배하기 쉬워서 순서를 바꾸거나 과거 정보를 손실하면 데이터 확률 분포도 바뀜
        4. 과거의 데이터로 미래의 확률분포를 예상하기 위해 조건부 확률 사용

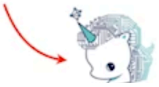
시퀀스 데이터를 어떻게 다루나요?

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

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

이전에 배운 베이즈 법칙을 사용합니다

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



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

5. 시퀀스 데이터를 다루기 위해서는 가변 길이 데이터를 다룰 수 있는 모델 필요

2. RNN

- 가장 기본적인 RNN 모형은 MLP 와 유사한 모양입니다

$\mathbf{W}^{(1)}, \mathbf{W}^{(2)}$ 은 시퀀스와 상관없이 불변인 행렬입니다

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

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

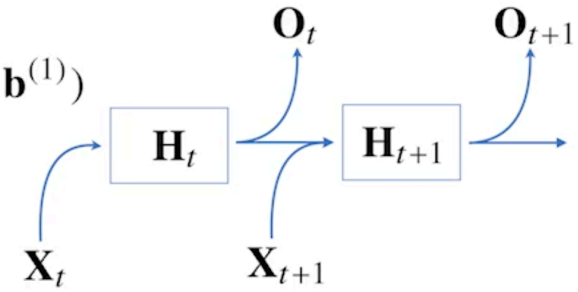
잠재변수 활성화함수 가중치행렬 bias

Recurrent Neural Network 을 이해하기

- 가장 기본적인 RNN 모형은 MLP 와 유사한 모양입니다
- RNN 은 이전 순서의 잠재변수와 현재의 입력을 활용하여 모델링합니다

$$\mathbf{O}_t = \mathbf{H}_t \mathbf{W}^{(2)} + \mathbf{b}^{(2)}$$
$$\mathbf{H}_t = \sigma(\mathbf{X}_t \mathbf{W}_X^{(1)} + \mathbf{H}_{t-1} \mathbf{W}_H^{(1)} + \mathbf{b}^{(1)})$$

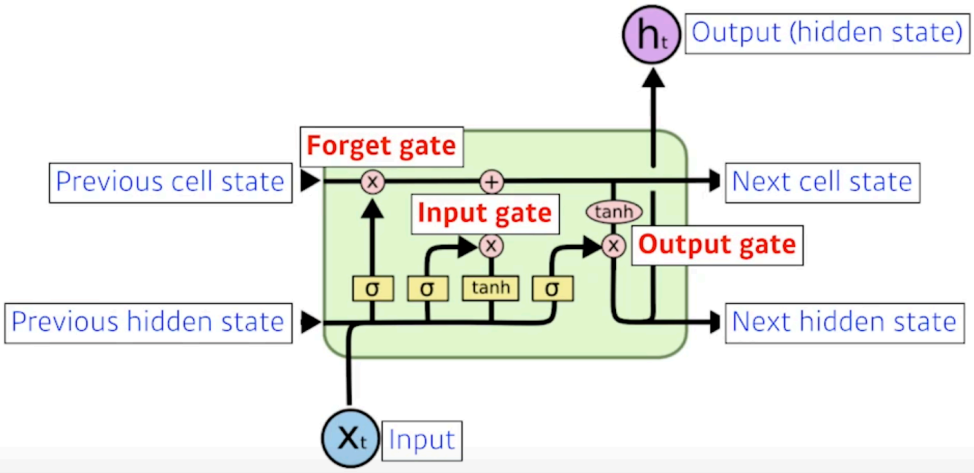
잠재변수인 $\mathbf{H}_t$ 를 복제해서 다음 순서의 잠재변수를 인코딩하는데 사용합니다



1. 기본적인 RNN 은 MLP와 비슷
    1. 하지만 MLP는 그 순간의 입력만 고려하기 때문에 과거의 정보를 다룰 수 없음
    2. RNN은 이전 시점의 잠재변수와 곱해질 새로운 가중치 행렬 도입
2. BPTT (시간에 따른 역전파)
    1. 모든 t 시점의 손실함수를 계산해 전달할 그래디언트 계산
    2. 모든 잠재변수에 대한 미분텀이 곱해지게 됨
    3. 시퀀스 길이가 길어질 수록 곱해지는 텀들이 불안정해짐
        1. 1보다 크면 굉장히 크게 곱해지고
        2. 1보다 작게 되면 굉장히 작아지게 됨
    4. 일반적인 BPTT를 모든 시점에 적용하면 학습이 굉장히 불안정해짐
3. 기울기 소실
    1. 위의 문제점에 의해 기울기가 0으로 가게 됨
    2. 이렇게 되면 과거 정보를 유실하게 됨
    3. 해결책 : LSTM, GRU

3. **Sequential Models - RNN**
    1. LSTM

Long Short Term Memory

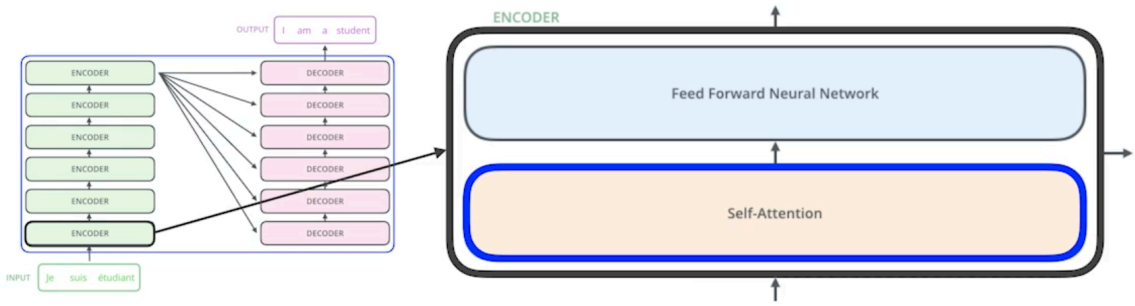


1. 롱텀 디펜던시를 반영가능
2. 프리비어스 히든 스테이트
    1. 이전의 출력값
3. 프리비어스 셀 스테이트
    1. 현재까지 들어온 정보를 요약한 것
4. 포겟게이트
    1. 이전 셀스테이트에서 나온 정보를 현재 입력과 이전 히든 스테이트의 값으로 버릴지 말지 판단
5. 인풋게이트
    1. 어떤 정보를 셀스테이트에 올릴지 정함
6. 아웃풋 게이트
    1. 아웃풋 할 걸 정함

2. GRU
    1. 셀스테이트 없이 히든 스테이트만 사용
    2. 두 개의 게이트로 LSTM과 굉장히 비슷한 역할
    3. 적은 파라미터로 동일한 아웃풋을 내므로 일반화가 좋아 LSTM 보다 성능 좋음
    4. 트랜스포머가 더 좋음

4. 트랜스포머
    1. 재귀적인 구조가 아닌 어텐션 구조 기반
    2. 시퀀스 투 시퀀스
    3. 인코더는 단어가 몇 개던 재귀적으로 돌지 않고 단 한번에 처리 가능
    4. 이해할 것
        1. 인코더가 어떻게 한 번에 처리하는지
        2. 인코더와 디코더 사이 정보교환
        3. 디코더가 어떻게 단어를 생성하는지
    5. 구조

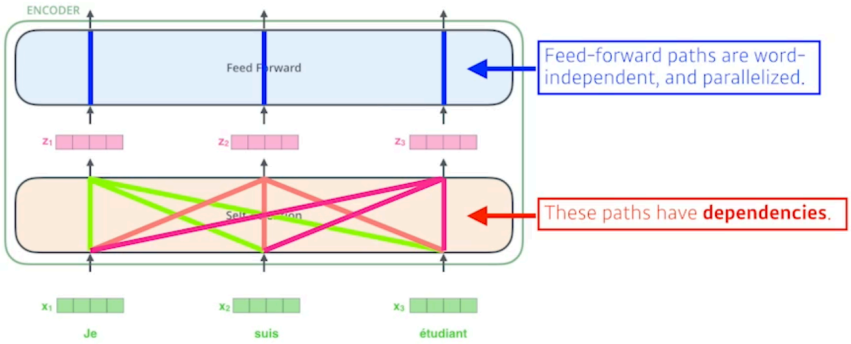
Transformer



- The Self-Attention in both encoder and decoder is the cornerstone of Transformer.

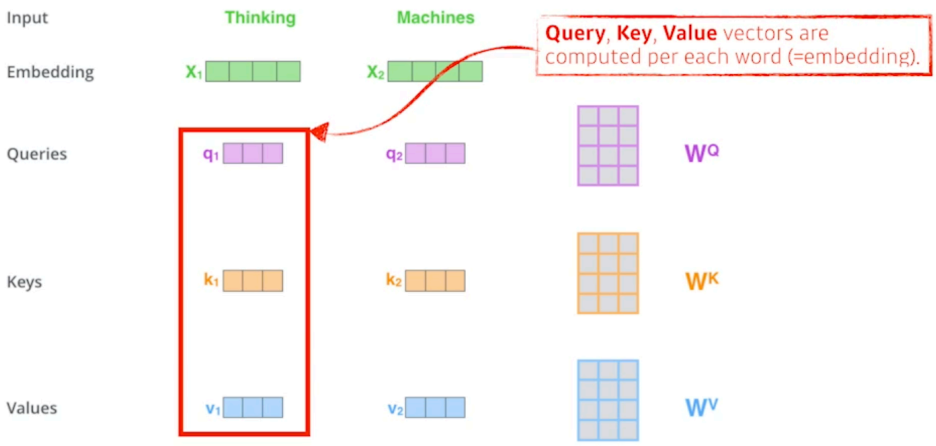
6. 셀프어텐션

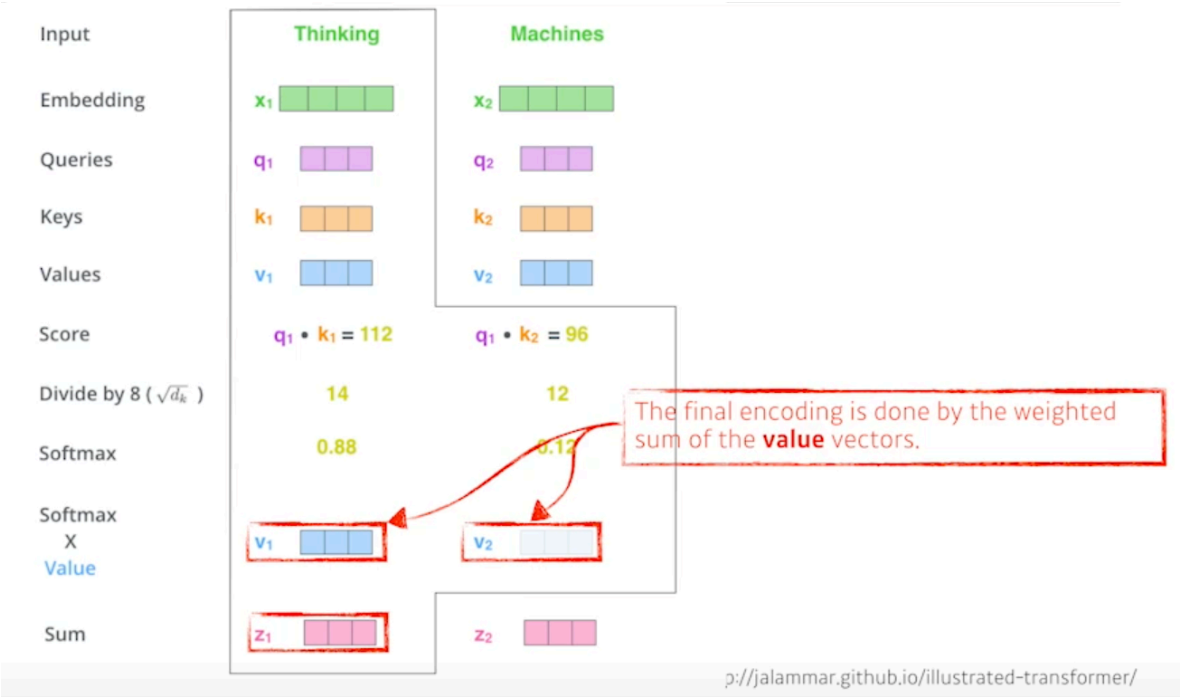
Transformer



- Then, Transformer encodes each word to feature vectors with Self-Attention.

Transformer





$$\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) V = Z$$

- Calculating Q, K, and V from X in a matrix form.

1. 각 단어마다 특정 숫자의 벡터로 변환
2. 셀프 어텐션의 결과는 다른 단어 벡터도 영향을 끼침
3. 피드 포워드는 각 단어 스스로만 영향끼침
4. 각 단어를 그 자체로만 이해하지 않고 문장속의 다른 단어들과의 관계를 이해
5. 각 단어를 인코딩할 때 다른 단어와의 관계성을 이해
6. 셀프어텐션은 각 단어마다 세 가지 벡터를 만들어냄
7. 임베팅 벡터 x1 을 새로운 단어 벡터로 바꿀 것임
8. 스코어벡터
    1. 인코딩 하고자 하는 벡터의 쿼리벡터와 자신 포함 나머지 모든 n개의 단어에 대한 키 벡터를 구해서 내적
    2. i 번째 단어가 나머지 n개의 단어와 얼마나 관계가 있는지를 파악
    3. 어떤 단어들과 인터렉션이 많아야 하는지 알 수 있음

4. 소프트맥스 해서 어텐션웨이트 구할 수 있음
5. 이것이 밸류 벡터의 웨이트가 됨

9. 최종벡터
    1. 밸류 벡터와 어텐션웨이트를 웨이티드 썸 하면 하나의 단어에 대한 인코딩된 벡터가 나옴
    2. 웨이티드 썸을 하는 것이기 때문에 차원이 달라도 됨
    3. 인코딩 된 벡터의 차원은 밸류벡터와 같음
10. 쿼리벡터와 키 벡터는 내적해야해서 항상 차원이 같아야 함
11. 잘 되는 이유
    1. 같은 입력이라도 다른 입력들이 달라지면 출력이 달라짐
    2. 따라서 더 많은 걸 표현할 수 있음
12. 단점
    1. n 개의 단어를 한 번에 처리해야 하기 때문에 n이 너무 커지면 하드웨어 한계로 처리하지 못할 수 있음
    2. 계산하는데 n제곱만큼 소요됨
    3. rnn은 n개의 입력이면 n 번 하면 되므로 하드웨어 제약 x
13. 멀티 헤디드 어텐션
    1. 하나의 입력에 대해 쿼리 키 밸류를 벡터를 하나 말고 여러개 만듦
    2. 하나의 임베딩된 입력당 n개의 인코딩된 벡터가 나옴
    3. 나온 여러개의 임베디드 벡터를 다시 원래의 디멘전으로 줄여줌
14. 포지셔널 인코딩
    1. 지금까지의 인코딩은 단어의 순서에 대한 데이터를 포함하지 않음
    2. 주어진 입력에 미리 정해진 어떤 값을 더함

7. 디코더
    1. 인코더는 디코더로 키와 밸류를 보냄

2. **피어세션**
    1. 강의 리뷰
        1. 키워드
            1. RNN 수식
            2. AI Math
            3. CNN 필터 크기
        2. 퍼더 퀘스천
    2. RNN 기초 다지기
    3. 데이터셋
        1. 주제 선정
            1. 디즈니 캐릭터
        2. 각자 모을 캐릭터 이미지 선정
            1. 나 : 인어공주
        3. 모을 이미지 수 결정
            1. 캐릭터당 200개