NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau, KyungHyun Cho Yoshua Bengio

목차

(0) ABSTRACT

- **Neural machine translation**

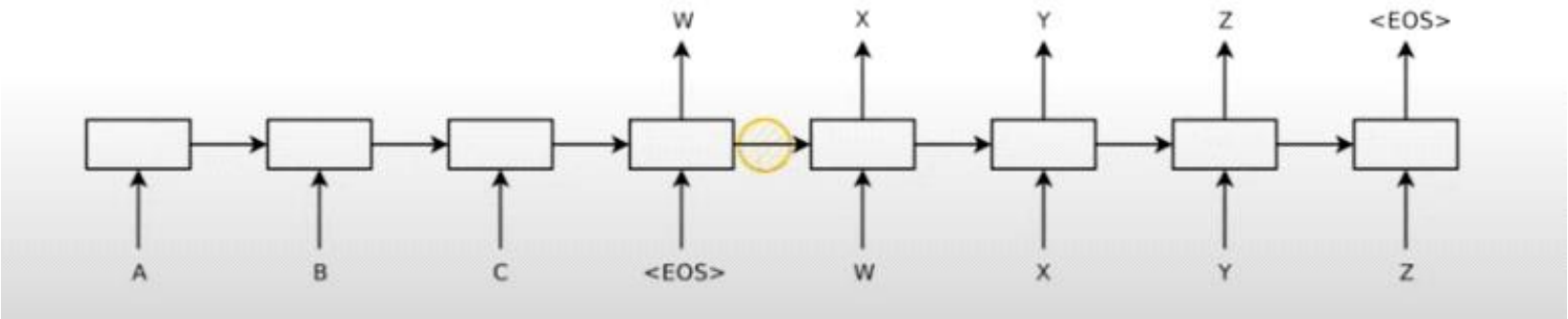
- Phrase-based Machine Translation에서 발전
- 대부분 encoder-decoder형태

해당 논문

고정벡터 사용이 encoder-decoder bottleneck 문제

- 길이가 긴 source sentence에 대한 충분한 정보를 담을 수 없음
- 충분한 성능 향상의 방해 요소

=> 입력 시퀀스를 벡터시퀀스로 인코딩후, 매 디코딩 step마다 벡터 시퀀스의 subset을 adaptive하게 선택



[Encoder-Decoder Architecture]

참고 : https://www.youtube.com/watch?v=S2msiG9g7Us

(1) INTRODUCTION

- **Neural machine translation**

- Encoder-Decoder
  - encodes a source sen- tence into a fixed-length vector.
  - A decoder then outputs a translation from the encoded vector.
  - encoder and the decoder for a language pair, is jointly trained to maximize the probability of a correct translation given a source sentence.

  => A potential issue with this encoder–decoder approach is that a neural network needs to be able to compress all the necessary information of a source sentence into a fixed-length vector.
  showed that indeed the performance of a basic encoder–decoder deteriorates rapidly as the length of an input sentence increases.

  모든 정보를 고정길이 벡터로 압축이 되어야함.

해결을 위해
we introduce an extension to the encoder–decoder model which learns to align and translate jointly.
기본 인코더와 디코더가 전체 입력 문장을 단일 고정길이 벡터로 인코딩하려고 시도하지않고,
변환하려는 동안 adaptively 하게 , 벡터 부분집합을 선택함. → 긴 문장에 더 잘 대처하는 모델임.

(2) BACKGROUND

- **Neural machine translation**

$$h_t = f\left(x_t, h_{t-1}\right)$$

$$c = q\left(\left\{h_1, \cdots, h_{T_x}\right\}\right),$$

[식-1], [식-2]

(2014)
RNN을 이용한 encoder-decoder 모델
Encoder는 입력으로
문장 x=(x1,x2,...,xTx) 을 고정된 길이의 벡터 c로 변환하게 된다.
RNN을 이용하는 경우를 [식-1], [식-2]통해서 벡터 c를 생성한다.

[식-1]에서 ht h_t 는 time t 에서의 hidden state를 의미한다.

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t \mid \left\{y_1, \cdots, y_{t-1}\right\}, c),$$

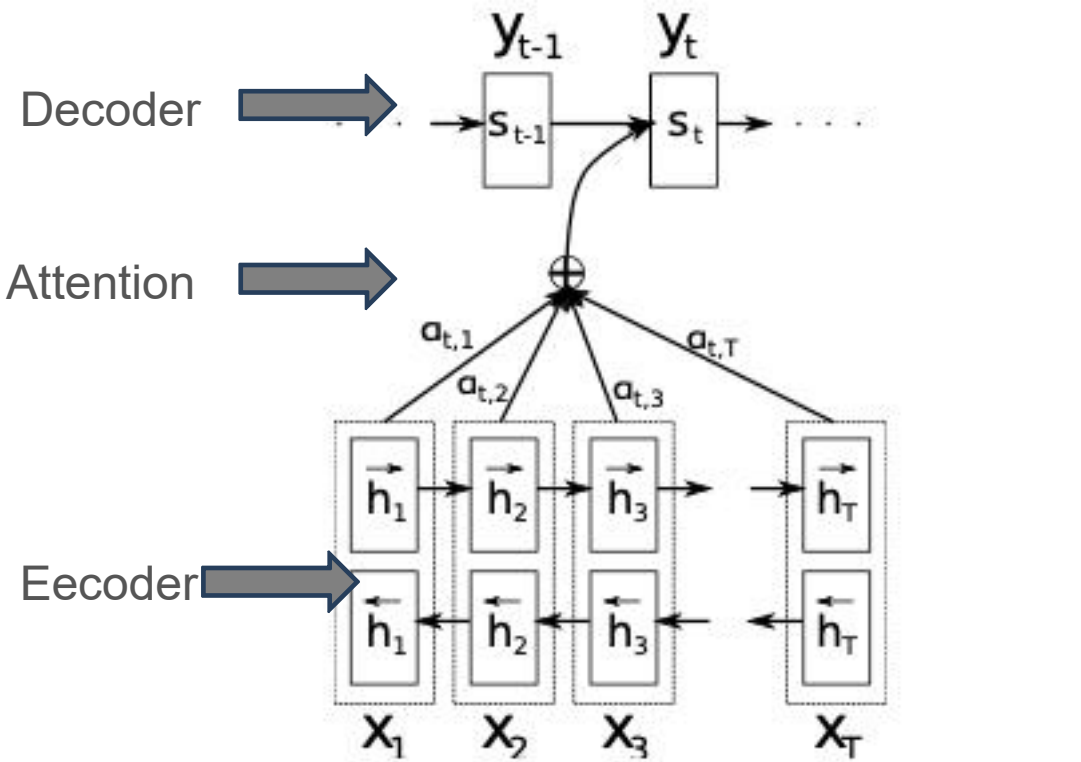
$$p(y_t \mid \left\{y_1, \cdots, y_{t-1}\right\}, c) = g(y_{t-1}, s_t, c),$$

[식- 3], [식-4]

Decoder는 context vector c가 encoder로부터 주어졌을 때,
c와 이전에 예측한 결과 y1,y2,...,yt′−1 을 기반으로 다음 단어 yt″ 를 예측하게 된다.
번역된 결과 y=(y1,...,yTy) 는 [식-3]과 같은 조건부 확률을 기반으로 생성
조건부 확률은 바로 직전 time인 (t−1) 에서 예측한 결과 yt−1 과,
RNN의 hidden state st,
그리고 non-linear function g() 를 이용해서 구할 수 있다.

(3) LEARNING TO ALIGN AND TRANSLATE

- **Attention 기법 (align and translate)**



$$h_j = \left[\overrightarrow{h}_j^\top ; \overleftarrow{h}_j^\top\right]$$

입력 문장에 대해서 이전에 나타는 내용과 이후에 나타나는 내용도 알아야지 좋은 번역이 가능하기 때문에,

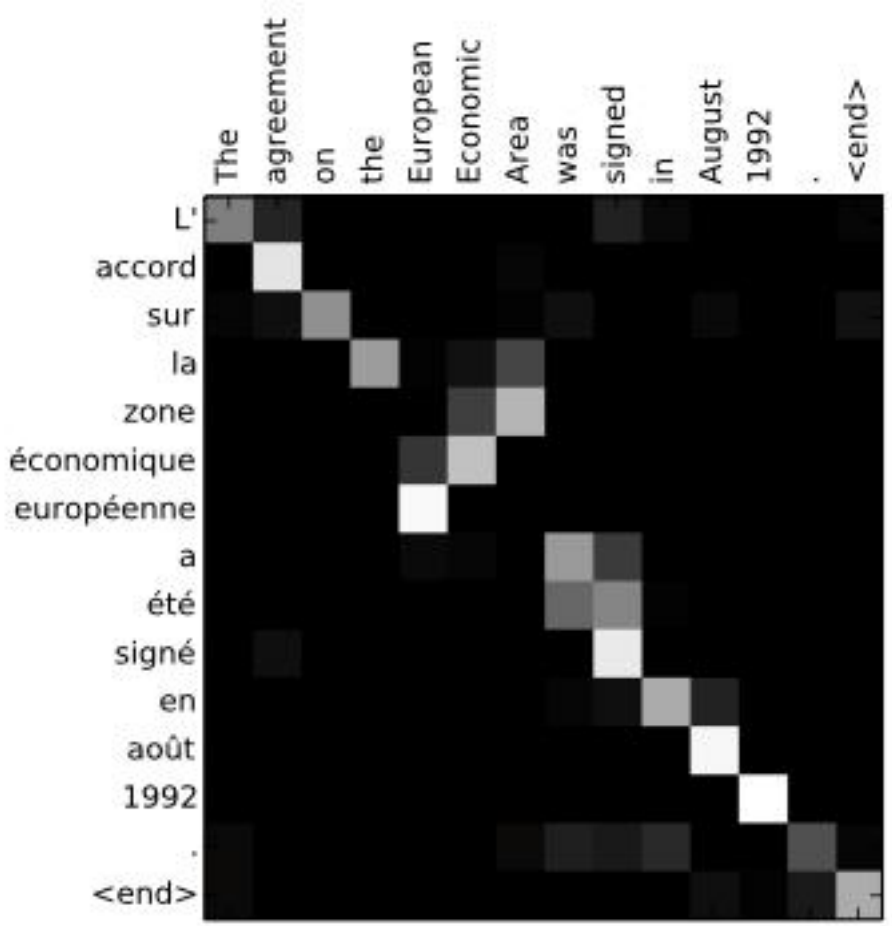
이번 모델에서는 **bidirectional RNN(BiRNN)**을 사용한다.
BiRNN은 두 개의 RNN**(forward RNN, backward RNN)**

BiRNN : 순방향과 역방향 RNN
순방향 RNN sequence를 처음부터 순서대로 읽고
forward hidden state 를 계산
역방향 RNN sequence를 역방향으로 마지막부터 처음까지 읽고
backward hidden state를 계산

각 단어 xj에 대해서 forward hidden sate와
backward hidden state를 concatenate
이 방법을 통해 annotation hj는 j번째 단어 앞뒤의 정보를 모두 포함

참고 링크 : https://misconstructed.tistory.com/49
https://youtu.be/l9pWT6BHpj0

(4,5) EXPERIMENT SETTINGS, RESULT



WMT'14 의 English-French parallel corpus
전체 corpus의 단어를 348M 개로 제한
monolingual data는 하나도 사용하지 않음
전체 단어 중 가장 많이 사용되는 30,000개

영어 - 프랑스어 번역결과

- Attention score 0 흰색 1검정
- 단순히 순서대로가 아님
- Alignment model을 통해 입력 시점 중 중요한 부분에 집중하고 있는 것을 확인 가능
- 한 단어가 여러 단어와의 관계도 나타낼 수 있어, 번역에 더 좋은 성능을 제공할 수 있음.

(5) RESULT

Attention 적용 X ->
Attention 적용 O ->

Model	All	No UNK°
RNNencdec-30	13.93	24.19
RNNsearch-30	21.50	31.44
RNNencdec-50	17.82	26.71
RNNsearch-50	26.75	34.16
RNNsearch-50*	28.45	36.15
Moses	33.30	35.63

* 성능 향상이 없을때까지 학습

- All : 모든 sentences
- No unk : unknown sentences가 없는 결과

- 전반적으로 더 좋은 성능을 보임

- 문장 길이가 계속 길어져도 성능저하가 없음.

- RNNencodec : 1000개의 hidden unit

- RNNsearch :
  - encoder - forward/backward 각각 1000개 hidden unit보유
  - decoder에서도 1000개의 hidden unit을 보유

(7) CONCLUSION

- **Neural machine translation**

기존 문제점
- 고정벡터 사용이 encoder-decoder bottleneck 문제
- 긴 문장 번역에 어려움

- 각 대상 단어를 생성할 때 입력한 단어 집합 또는 인코더가 검색하도록 하여, 고정 벡터로 인코딩할 필요가 없음.
- model focus only on information relevant to the generation of the next target word.
- Attention : 모델이 다음 target word를 생성하는 것과 관련 있는 정보에만 집중 하게 함.

- 영어-프랑스 번역 작업 테스트 결과 기존 encoder-decoder보다 성능이 좋음.
- 문장의 길이에 대해 관계 없이 더욱 ro-bust하다.