모델 아키텍처

경쟁력있는 neural sequence transduction models는 (encoder-decoder 구조.
auto-regressive: consuming previously generated symbols as additional input.

encoder: $(x_1, \cdots, x_n)$ →(input, mapping) $(z_1, \cdots z_n)$

decoder: $(z_1, \cdots z_n)$ →(output) $(y_1, \cdots y_m)$

→ Transformer는 위의 구조를 이용 by using (stacked-self attention + pointwise fully connected layers.) ← 인코더, 디코더 구조.
(개별적)

1) encoder & Decoder stacks

encoder: 6개의 layers stacked.
└ each has 2 sub layers
- multi-head self attention: 8개 헤드로 병렬 연산 수행
- position wise fully connected.

각 sub layers에 residual connection 후, layer normalization.
(LayerNorm(x+Sublayer(x)))

임베딩 레이어 포함한 모든 레이어들의 output dim = 512 (d_{model}) ∵ residual connection 용이하게 하려고.

1개 문장 input
→ 6개의 인코딩 레이어 통과.
각 레이어 내에서 8개의 헤드 통해 각각 다른 방식으로 문맥 정보 학습 후,
최종 출력은 하나의 벡터로 압축 (디코더에서)

decoder: 6개 layers
└ has 3 sub layers
- multi-head self attention ← revised: 이후 포지션이 참조하지 않도록, 즉, 미래 데이터를 못보게 각 포지션이 (이전에서 생성된 단어들만 참조) masking.
- 인코더 스택의 아웃풋에 대한 multi head attention.

각 레이어 LayerNorm(x+Sublayer(x))

① 멀티헤드 어텐션 수행
→ 단어의 서로관계를 문맥 정보벡터
② 8개를 하나로 합침
→ concatenate + 가중치 변환
③ FFN 통과시켜서 최종 출력 벡터 생성
④ Softmax 적용해서 단어 분포로 변환
→ 가장 확률 높은 단어 선택

2) Attention

Q, (K, V) ⇒ output

mapping

output : Value의 가중합.

이때 가중치 : Q-K 간의 유사도 측정하는 호환성 함수로 결정

compatibility function

• Scaled Dot product Attention.

$Q, K \in dim = d_k$

$V \in dim = d_v$

$$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Q-K 중 연관성 가장 높은거 선택 → 합쳐서 표현

d_k 값이면 additional attention 성능이 dot보다 좋음 (d_k 스케일링 X) softmax 1에가까움

d_k 커지면 내적 크기 커져서 Softmax 함수의 기울기 매우 작게 된다. ↓ $\frac{\partial Softmax}{\partial x} = y(1-y)$ ↑ 0.000.....

↳ backprop 시에도 학습속도 ↓ or 학습불능

: gradient vanishing

$\Rightarrow \frac{1}{\sqrt{d_k}}$ 로 스케일링하여 해결!

• Multi-head Attention.

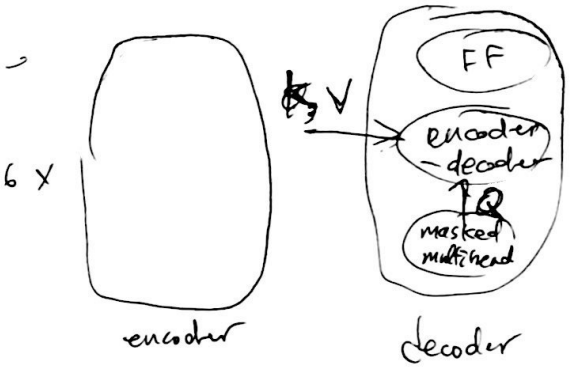
64×8개

$h = 8$ (parallel)

$$MultiHead(Q,K,V) = Concat(head_1, \dots, head_h)W^O$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$

$d_k, d_v = \frac{d_{model}}{h} = 64$



• Fully Connected $d_{ff} = 2048$ (inner layer)

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

ReLU

입력, output dim 역시 모두 512.

• positional encoding ← 입베딩의 위치정보

즉, dim=512

• 각 서브레이어 마다 dropout : residual dropout

(입베딩+위치인코딩 합한 값에도 드롭아웃

기본 확률값은 0.1 비율)

- 배치 : 여러 가능한 학습 시퀀스 중 학습에서 유망수보 선택

- 빔크기 : 한번에 고려할 수열시퀀스의 개수 선택

- 길이 패널티 : 생성 시퀀스 길이에 따라 패널티 부여
  가변적인 시퀀스 선택되게 해줌.

Result

(K, V) 데이터 h 의 variation 에 model variance 검증.

- $d_k \downarrow \rightarrow$ 모델성능 ↓ : dot product보다 작은 호환성함수 필요하다고?
  V에 곱해질 가중치 계산하는 방식

- 모델 클수록 성능 ↑

- dropout ← 과적합방지에 효과적

- Sinusoidal positional Encoding ≈ learned positional Encoding : base model과 별 차이 X

Conclusion

- recurrent와 제거한 attention 모델인 Transformer.
- Sota model 보다 효과적 (성능·비용·시간면에서)
- 제약 : 텍스트 이외 모달리티에서도 Transformer 활용.

Further Tasks → large input (이미지/비디오) 처리하려면 local, restricted 한 어텐션메커니즘 필요.