2. Transformer

나오게 된 이유

- seq2seq 는 recurrency를 이용해 문장의 순차성을 유지하면서 인코더와 디코더를 구현할 수 있었음
- 하지만 long-term dependency 에 취약
- 아울러 recurrent 한 특성 상 병렬 처리가 불가

=> long-term dependenct를 해결하는 방법 중 하나로 attention 을 써 봤지만 여전히 long-term dependency 해소가 잘 안되며 recurrent 한 특성은 해소 불가

=> recurrency를 최소화하면서 이 문제를 해결할 수 있는 방법은?

- 문장의 순차성을 어떻게 확보할 것인가?

Transformer

- 여전히 Encoder-decoder 구조를 가짐
- CNN 이나 RNN 셀 없이 Multi-head (self) attention을 이용해 문제를 해결

구조

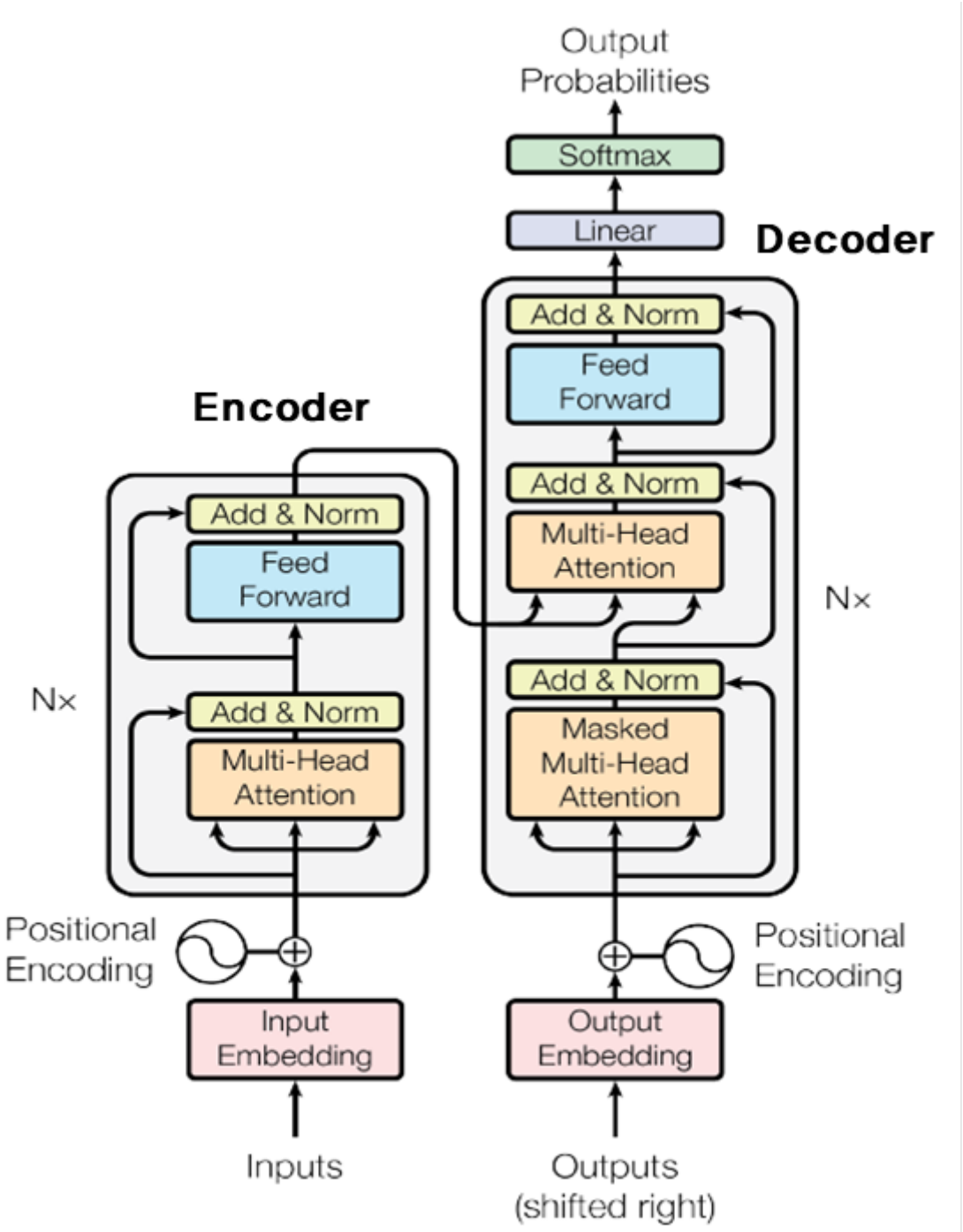
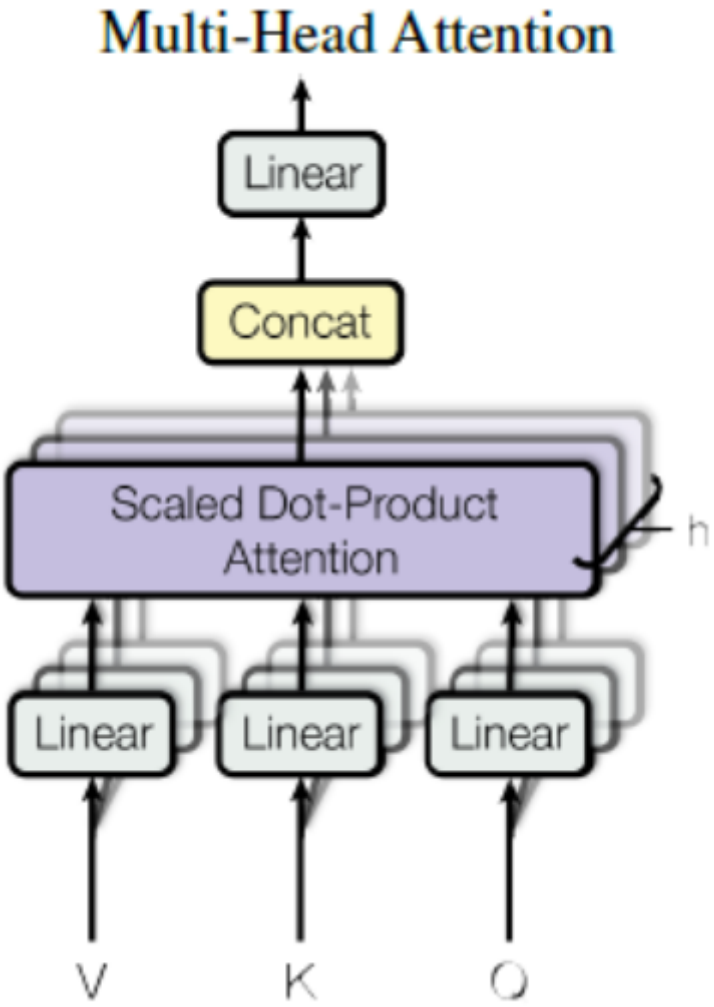


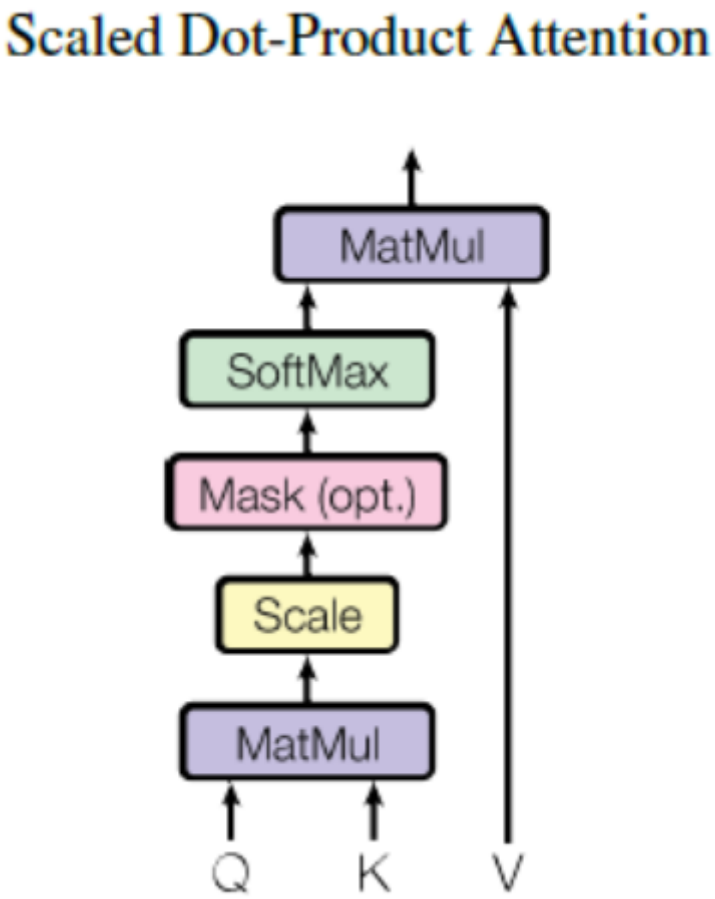
Figure 1: The Transformer - model architecture.

1. Embedding
    a. sinusoidal function 을 이용해 인풋/아웃풋 시퀀스가 각각 차지하는 위치를 인코딩
2. Encoder
    a. N(논문에서는 6) 개의 동일한 레이어로 구성
    b. 매 레이어의 아웃풋이 다음 레이어의 인풋이 됨
    c. 각 레이어는 서브 레이어로 multi-head self-attention 과 위치(position) 단위로 fc feed forward 레이어를 가짐

i. multi-head attention



ii. scaled dot-product attention



d. 각 서브 레이어는 residual(skip) connection 을 거쳐서 normalize 됨

3. Decoder
   a. 역시 N(6) 개의 동일한 레이어로 구성
   b. 인코더의 결과에 multi-head attention 을 수행하는 서브 레이어가 추가됨
   c. 디코더의 경우 결과를 순차적으로 생성해야 하기 때문에 self-attention 에 masking 절차를 거침(현재 position 앞 position에만 attention을 줌)
4. linear transform and softmax

Training of transformer

- Optimizer: adam