헷갈리는 개념 #2 . Attention, Transformer. GPT BERT

Attention — Decoder에서 예측시, Encoder에서의 전체 key 유사도를 고려하는 기법

Transformer — Attention을 이용해 RNN의 병렬처리 모델 + Multi-Head Attention Layer과 Subsequent Masking.

BERT — Transformer 기반의 MLM. NSP. Unsupervised task를 RNN에 적용한 Bidirectional Model. < MLM (Masked Language Model) NSP (Next Sentence Prediction)

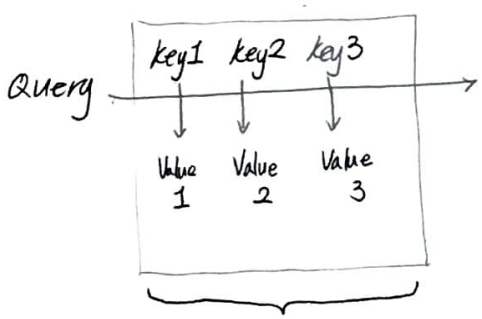
GPT — Transformer Decoder를 활용한 Pretrained LM.

1. Attention → seq2seq with Attention 이용용 종류 dot / scaled dot / general / concat location-base

seq2seq에 존재하는 문제점을 해결하기 위한 방안.

basic idea : 출력시 예측하는 매시점마다, 전체문장입력을 다시 한번 참고 연관있는 단어부분에 집중.

Attention (Q, K, V) = Attention Value.

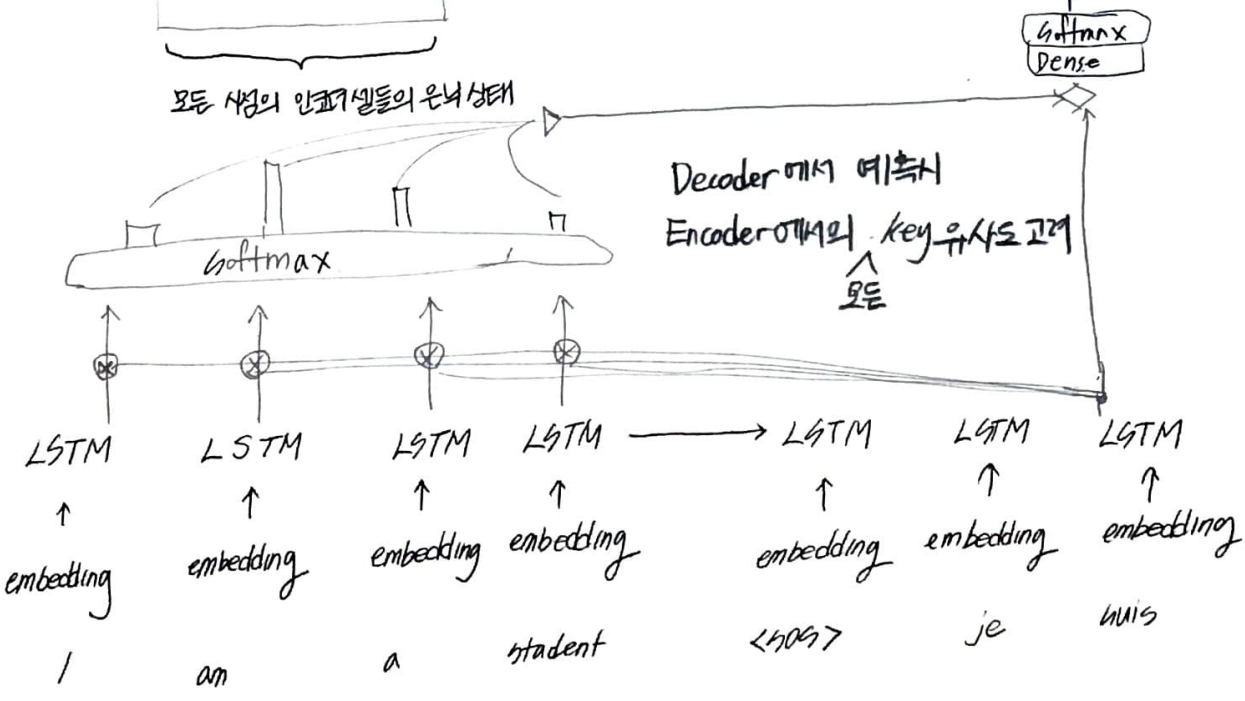


현재 key

Query에 따라 key의 유사도를 value에 저장.

(Query-key) 관계유사도

value 값을 모두 더한 attention value.

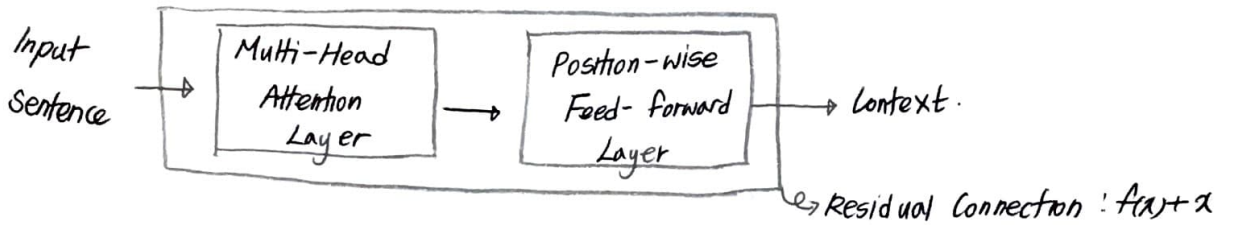


✓

2. Transformer

이전 RNN의 불가능했던 병렬처리를 극복. sentence to sentence 구조.

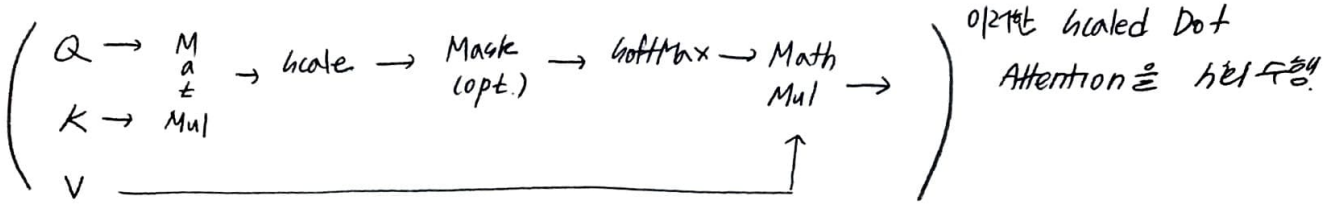
1) Encoder Layer



- Multi-Head Attention Layer

  : self Attention을 병렬적으로 여러개를 수행하는 Layer, Token 간의 유사도를 측정.

  같은 문장 내에서 다른 token에 대한 attention



- Position-wise Feed-Forward Layer.

  : 단순히 2개의 FC 갖는 Layer

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

2) Decoder

(Teacher forcing with subsequent masking)

- Input. context + sentence.

  - context. 의 Encoder의 output.

  - sentence.

    ↳ Teacher Forcing : 실제 labeled data (Ground Truth)를 RNN cell의 input으로 처리.

    + subsequent masking 처리 (수행하는 단은 masking)