Transformer
(Attention Is All You Need)

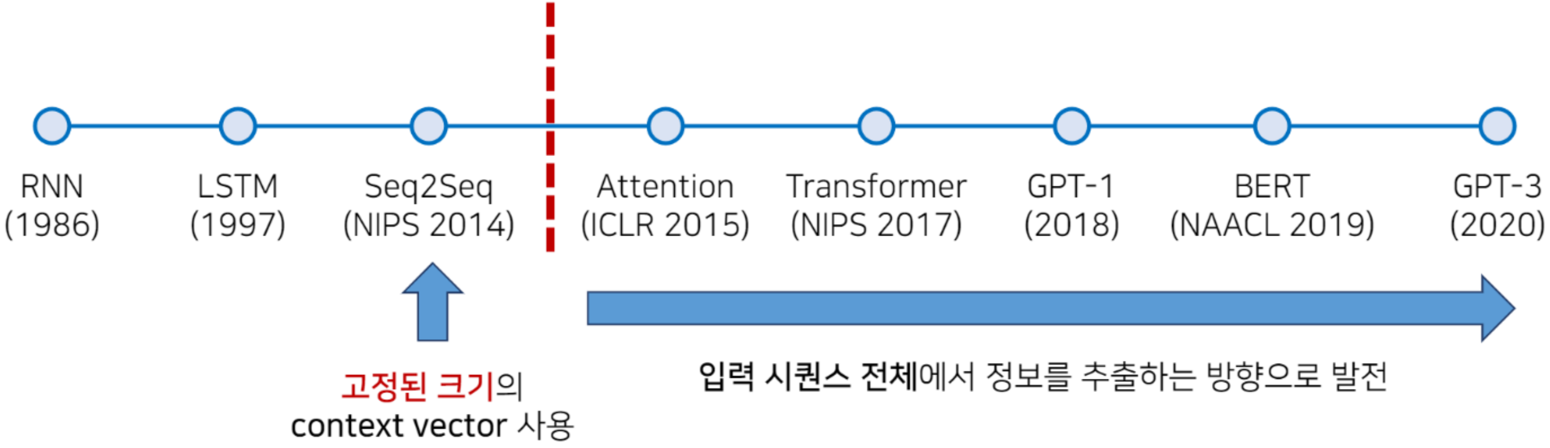
IBK시스템 플랫폼사업팀 곽효빈대리

2021-08-10

Background

딥러닝 기반의 기계 번역 발전 과정

- 2021년 기준으로 최신 고성능 모델들은 Transformer를 기반
  - GPT : Transformer의 Decoder 아키텍처를 활용
  - BERT : Transformer의 Encoder 아키텍처를 활용



Attention Mechanism

1. Attention Mechanism 정의

인간의 시각적 집중 (Visual Attention) 현상을 구현하기 위한 신경망적 기법

2. 가중치와 어텐션의 공통점과 차이점

가중치와 어텐션 모두 해당 값을 얼마나 가중시킬 것인가 나타내는 역할이지만,

어텐션은 가중치와 달리 전체 또는 특정 영역의 입력값을 반영하여, 그 중에 어떤 부분에 집중해야 하는지를 나타내는 것을 목표

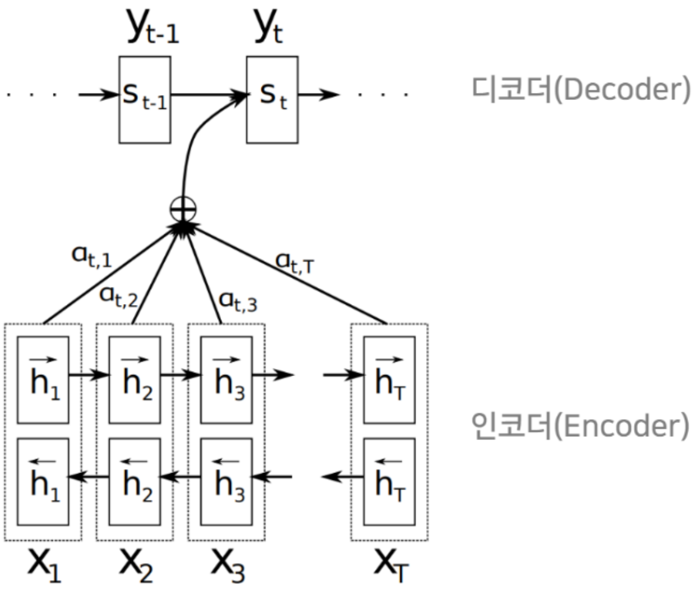
Seq2Seq with Attention

- Seq2Seq 모델에 Attention Mechanism을 사용
    - 디코더는 인코더의 모든 출력(outputs)을 참고

- 에너지(Energy) $e_{ij} = a(s_{i-1}, h_j)$
- 가중치(Weight) $\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$

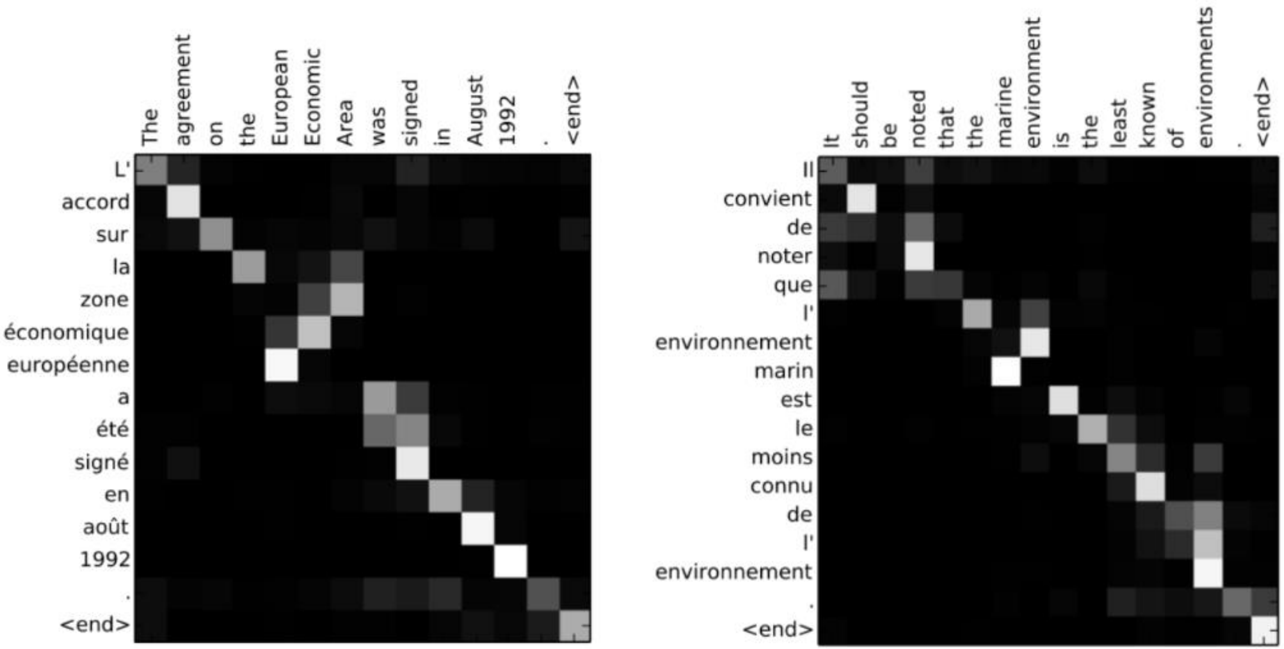
Weighted sum 이용

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$



Seq2Seq with Attention

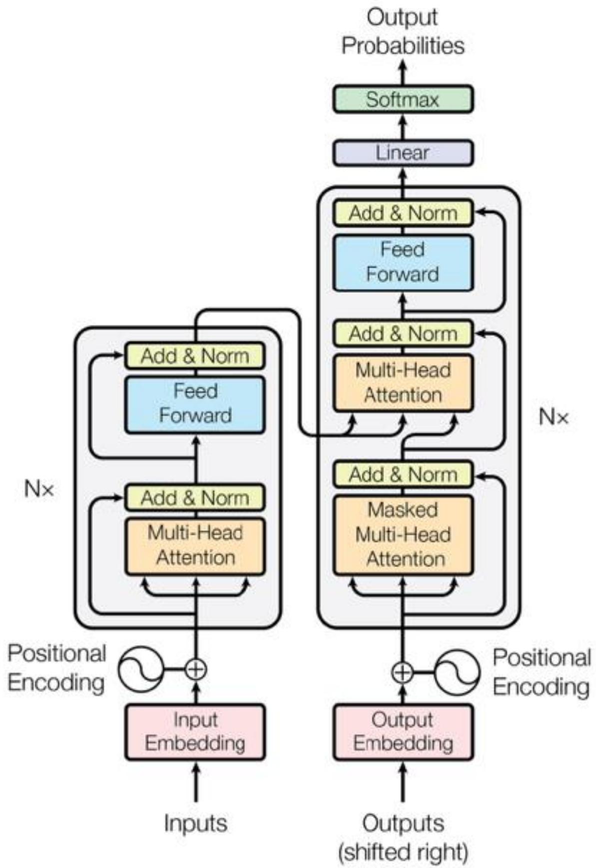
- Attention weight을 사용해 각 출력이 어떤 입력 정보를 참고했는지 시각화 가능



Attention is All You Need

Transformer

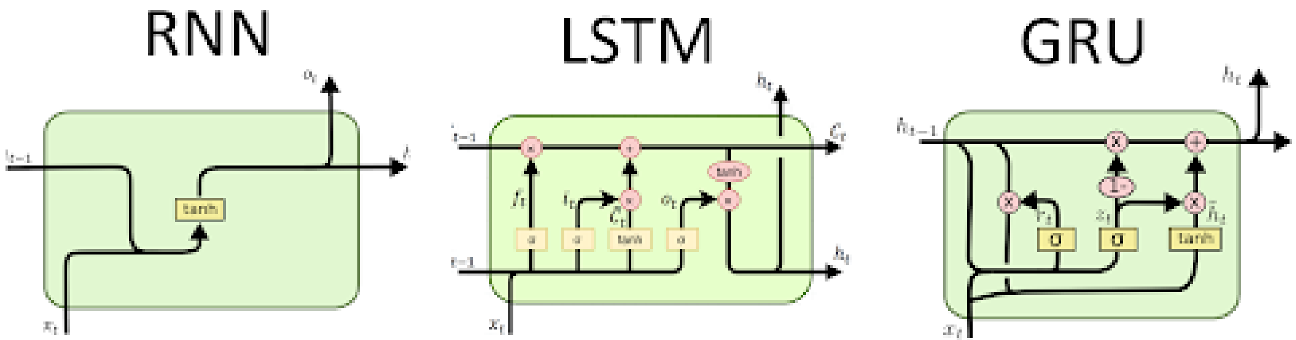
- RNN이나 CNN을 전혀 사용하지 않음
  - 대신 **Positional Encoding**을 사용
- 인코더와 디코더로 구성
  - Attention 과정을 여러 layer에서 반복
- BERT와 같은 향상된 네트워크에서도 채택되고 있음



1. Introduction & Background

1. Recurrent Neural Network

- 순차적인 특성이 유지되나, 정보간 거리에 따른 제약(i.e., long-term-dependency problem)을 지님
- 순차적인 특성 때문에 병렬처리를 할 수 없고, 계산속도가 느림

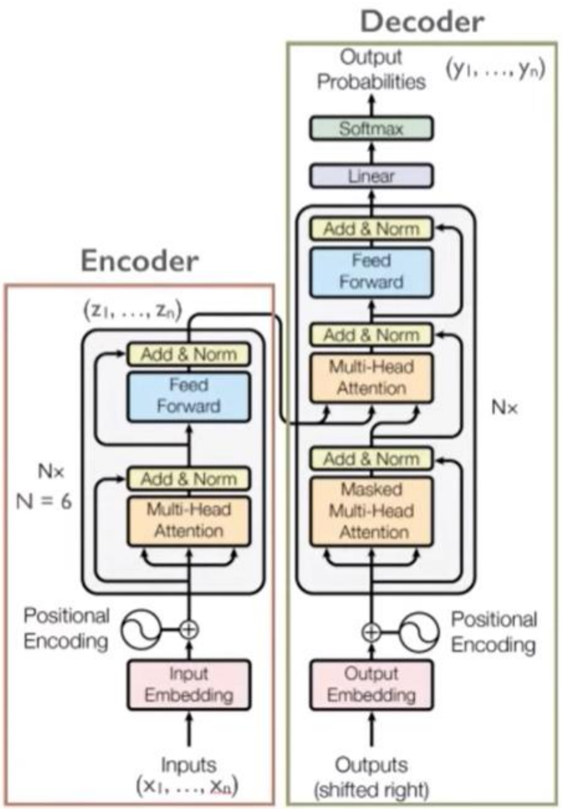


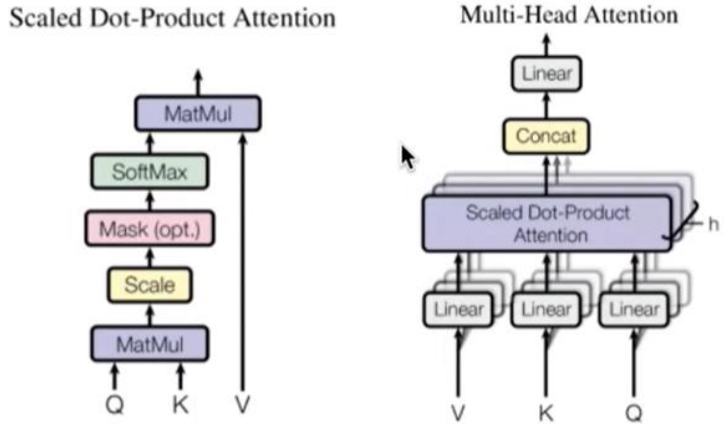
1. Introduction & Background

2. Self-Attention

- Intra-attention 이라고도 불린다
- 한 포지션을 모든 문장하고 비교
- 한 단어가 한 문장 안에서 어디에 집중하는지 계산한다.

2. Model Architecture



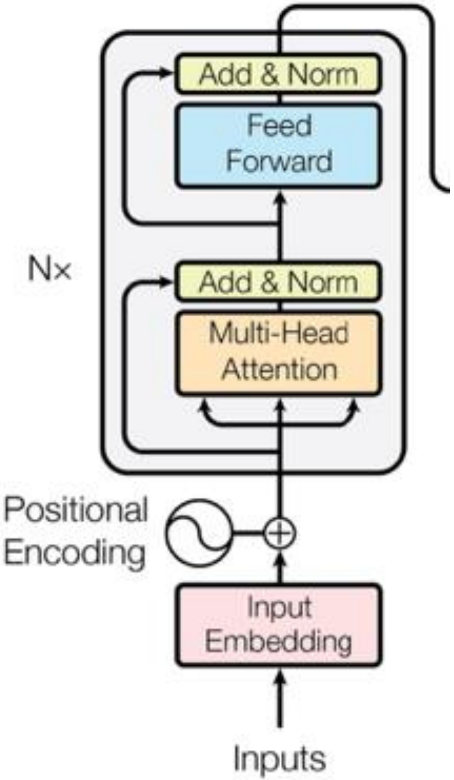


$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head_1}, ..., \mathrm{head_h})W^O$$
$$\text{where } \mathrm{head_i} = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$
$$W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}, W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}, W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}, W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$$

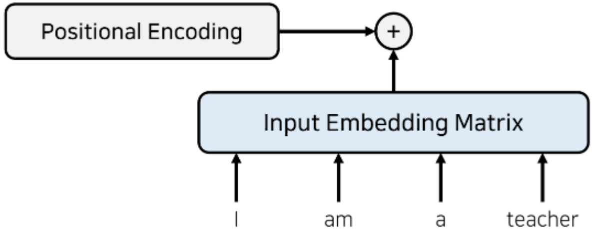
2. Model Architecture

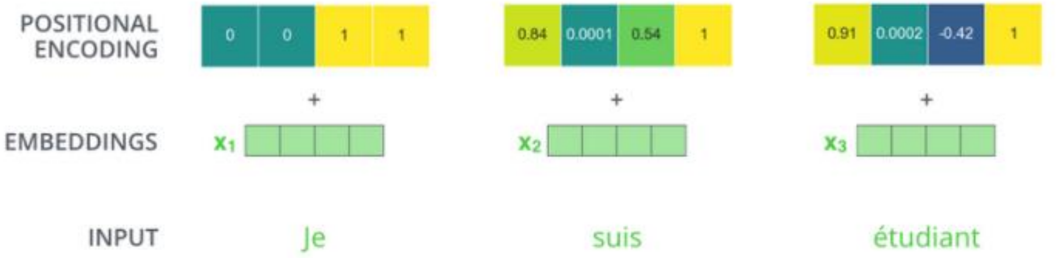
- Encoder
  - Each Layer has two sub layers
    - N=6 Layers
  - Multi-Head Attention
  - FC Feed Forward network
  - all sub layers produce ouputs of dimension d
    - d_model = 512



2. Model Architecture

- Positional Encoding
  - 위치 정보를 포함하고 있는 임베딩
  - 주기 함수를 활용한 공식 사용



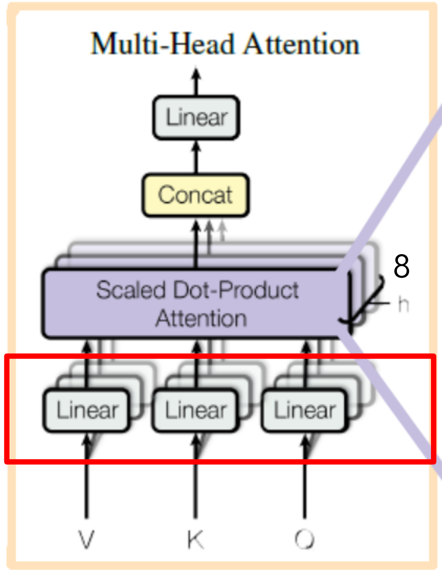


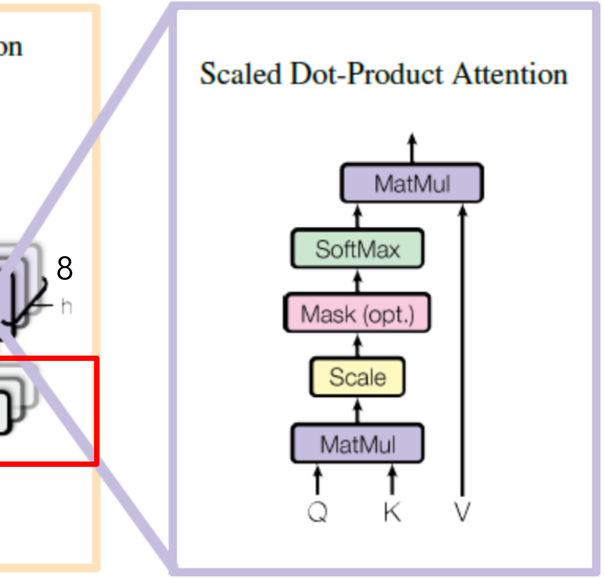
$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

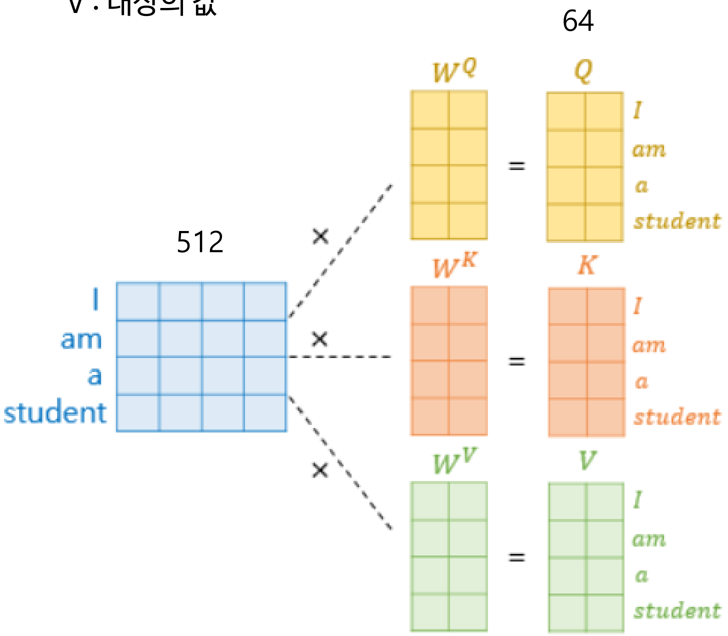
2. Model Architecture

- Multi-Head Attention

Q : 물어보는 주체
K : 물어보는 대상
V : 대상의 값



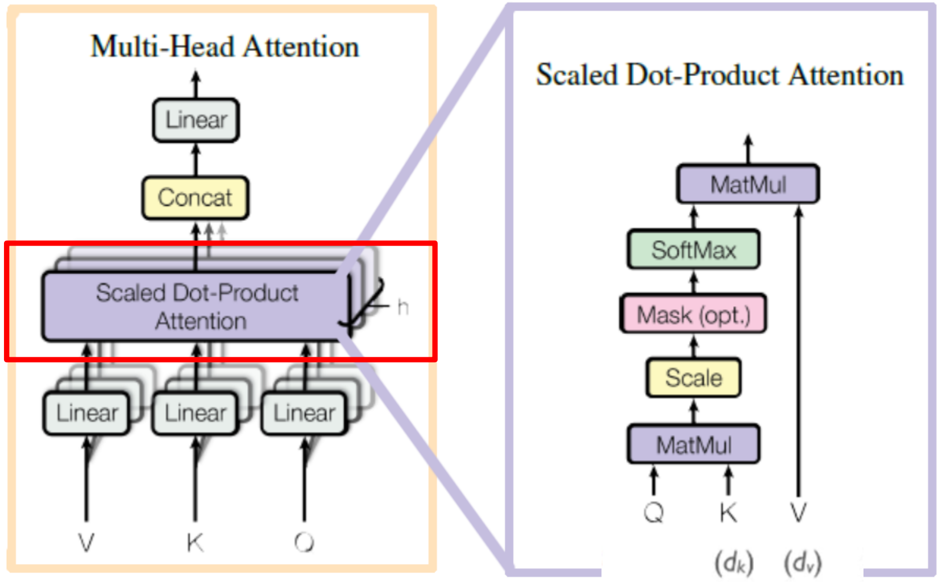




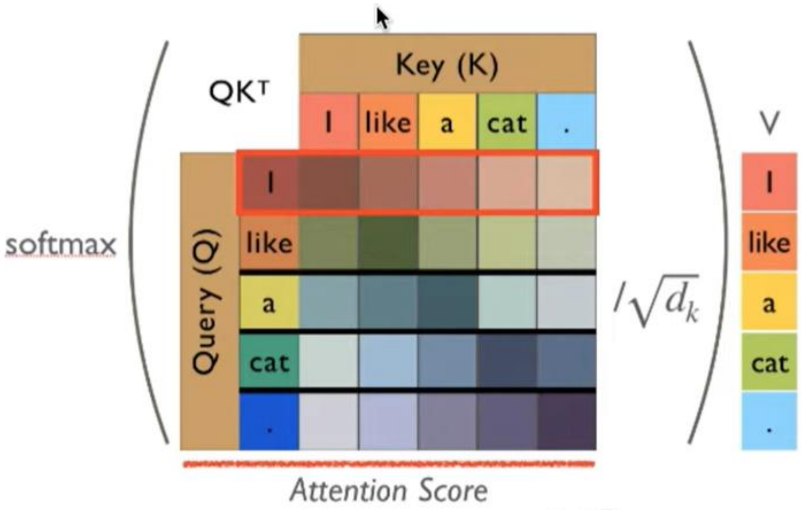
2. Model Architecture

- Multi-Head Attention

Q : 물어보는 주체, 검색어
K : 물어보는 대상
V : 대상의 값



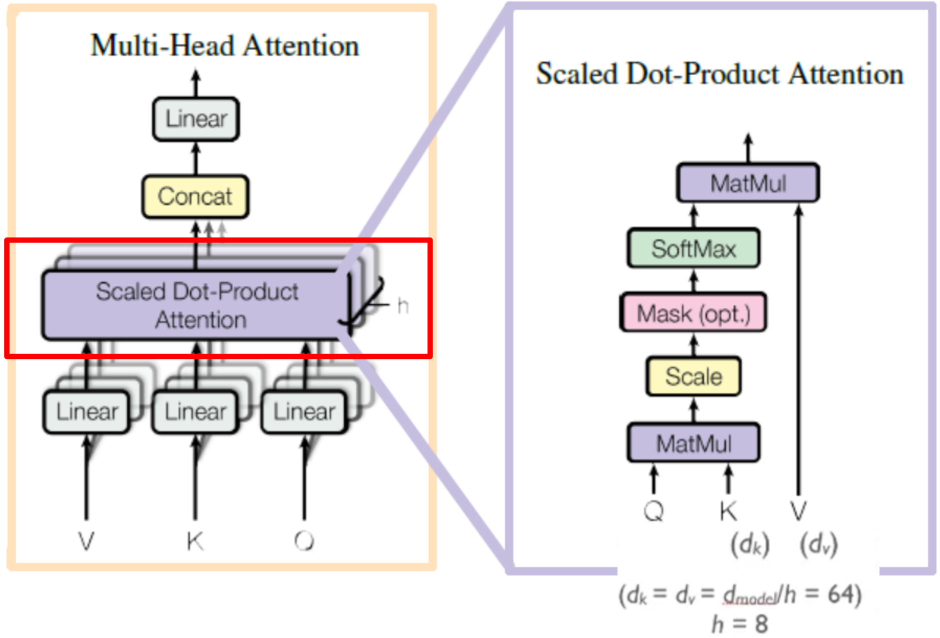
($d_k = d_v = d_{model}/h = 64$)
$h = 8$

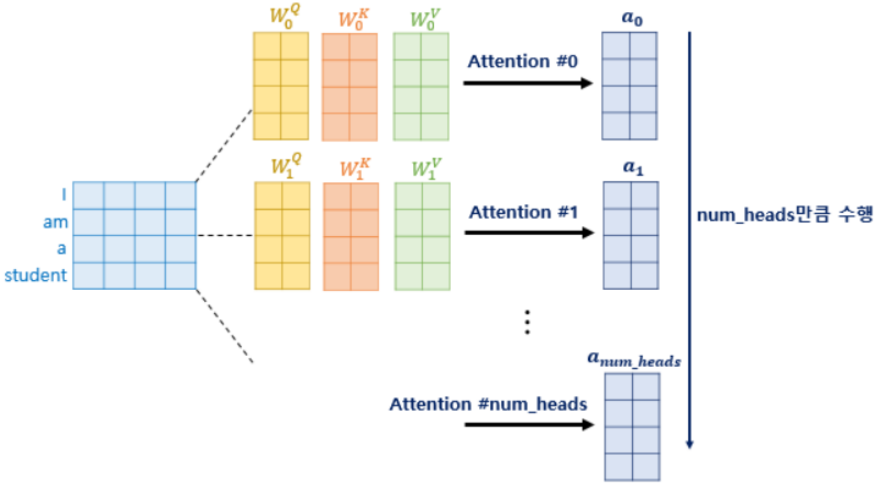


$$\mathrm{Attention}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

2. Model Architecture

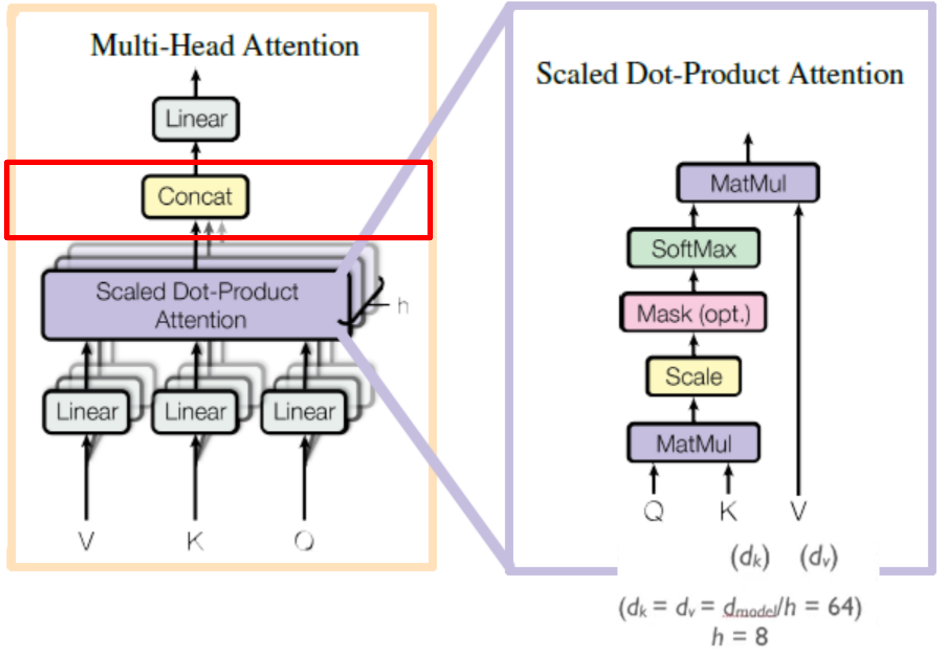
- Multi-Head Attention

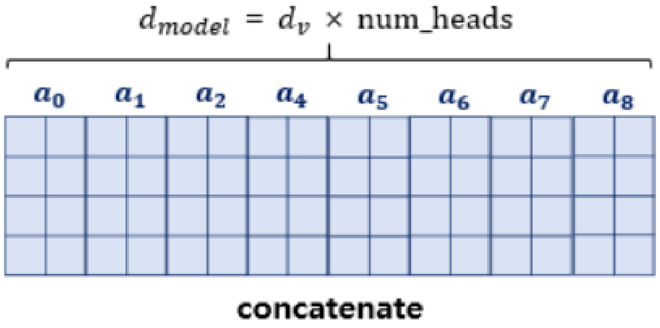




2. Model Architecture

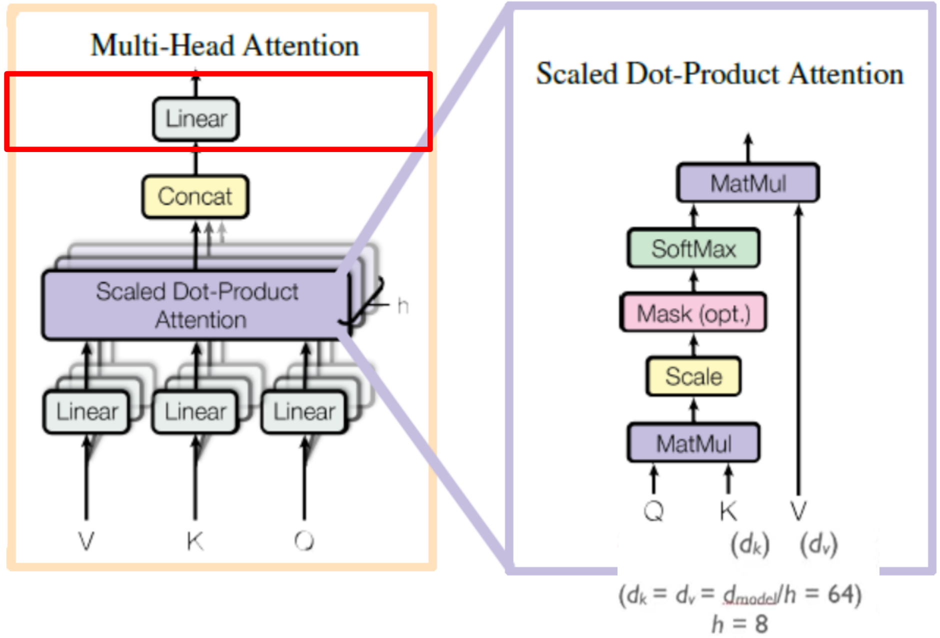
- Multi-Head Attention

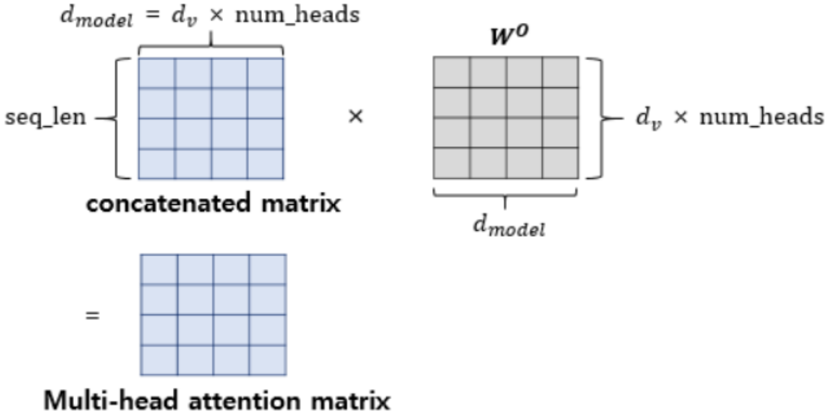




2. Model Architecture

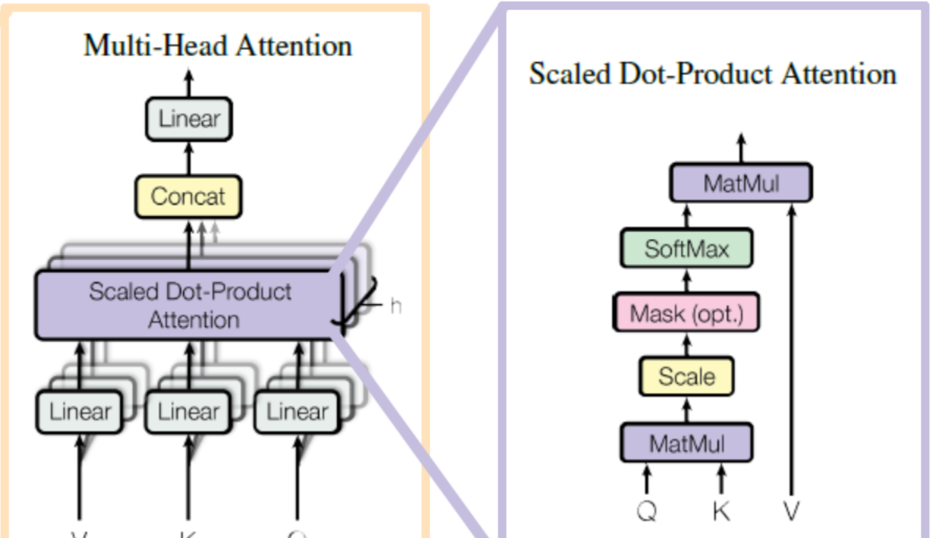
- Multi-Head Attention





2. Model Architecture

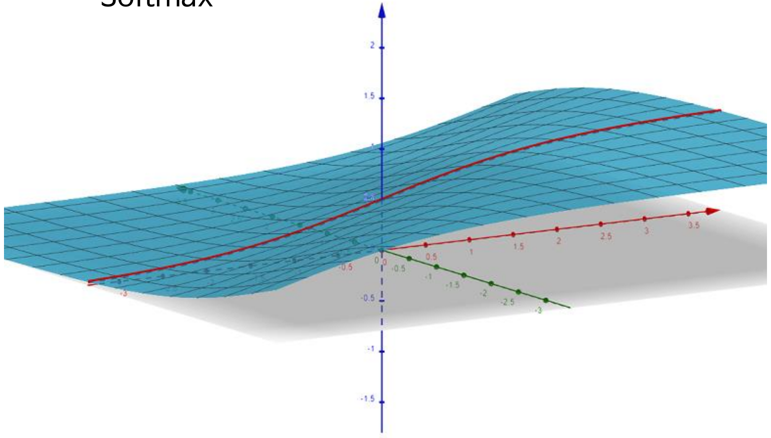
- Multi-Head Attention



(d_k) (d_v)

$(d_k = d_v = d_{model}/h = 64)$
$h = 8$

Softmax

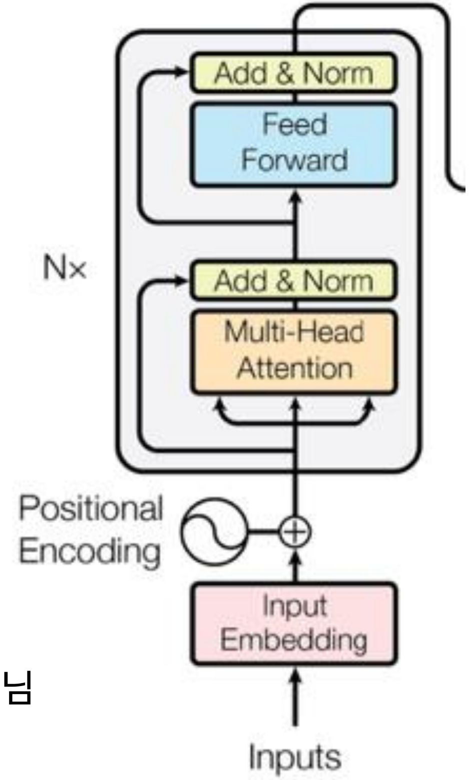


$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{ for } i = 1, \dots, K \text{ and } \mathbf{z} = (z_1, \dots, z_K) \in \mathbb{R}^K$$

2. Model Architecture

- Encoder
  - Multi-Head Attention
    - 임베딩이 끝난 후에는 Attention 진행
  - Add + Norm
    - Add: Adding residual connection
    - Norm : LayerNorm(x+Sublayer(x))
  - Feed Forward
    - Position-wise fully connected Networks
    - FFN의 파라미터 W, b는 같은 encoder내에서는 동일한 값을 지님

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



Attention Visualizations

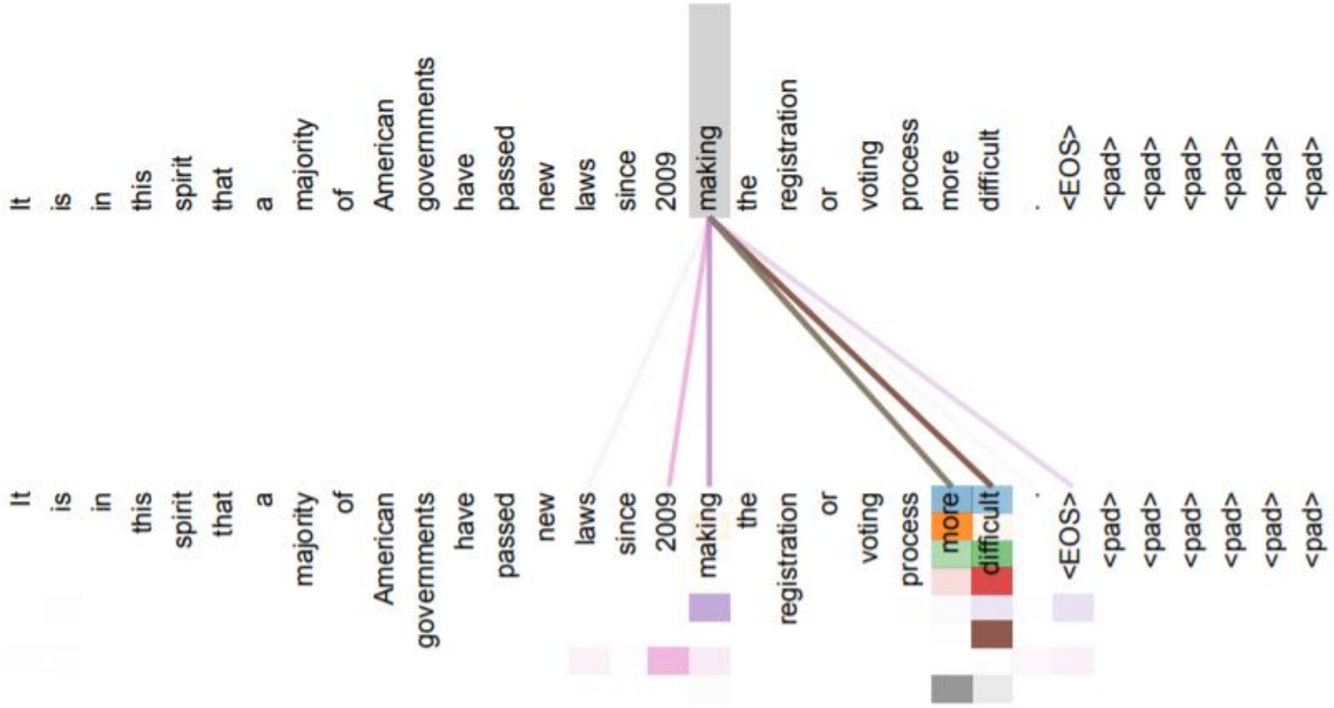


Figure 3: An example of the attention mechanism following long-distance dependencies in the encoder self-attention in layer 5 of 6. Many of the attention heads attend to a distant dependency of the verb 'making', completing the phrase 'making...more difficult'. Attentions here shown only for the word 'making'. Different colors represent different heads. Best viewed in color.

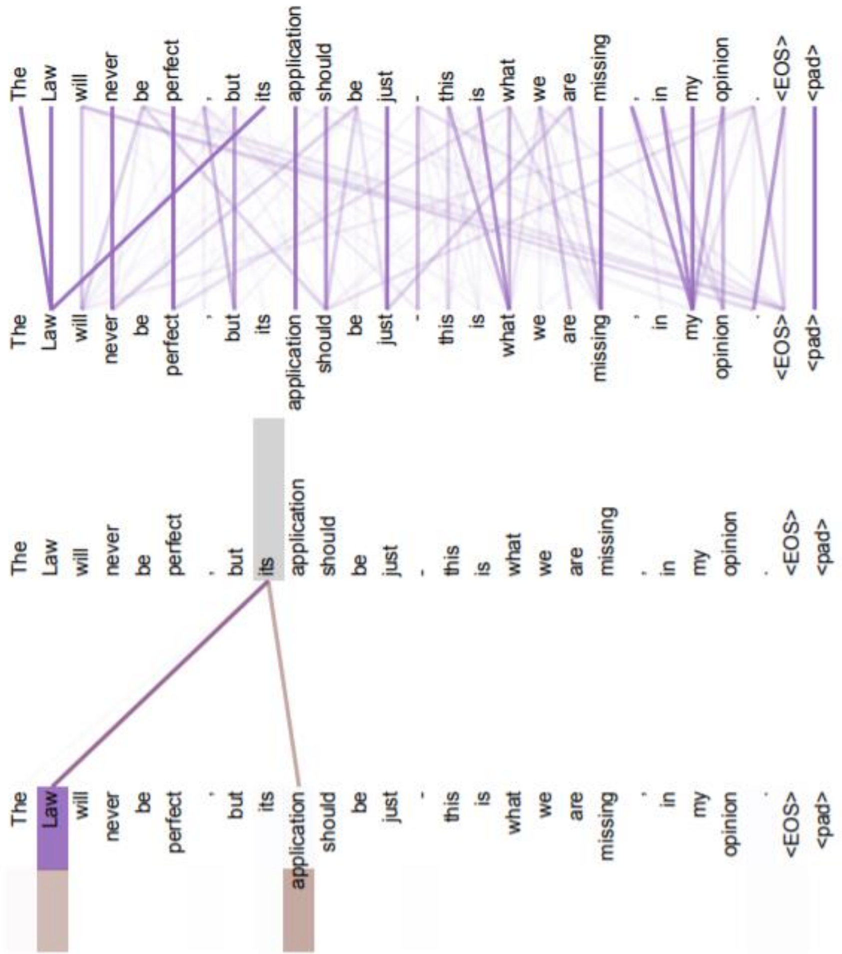


Figure 4: Two attention heads, also in layer 5 of 6, apparently involved in anaphora resolution. Top: Full attentions for head 5. Bottom: Isolated attentions from just the word 'its' for attention heads 5 and 6. Note that the attentions are very sharp for this word.

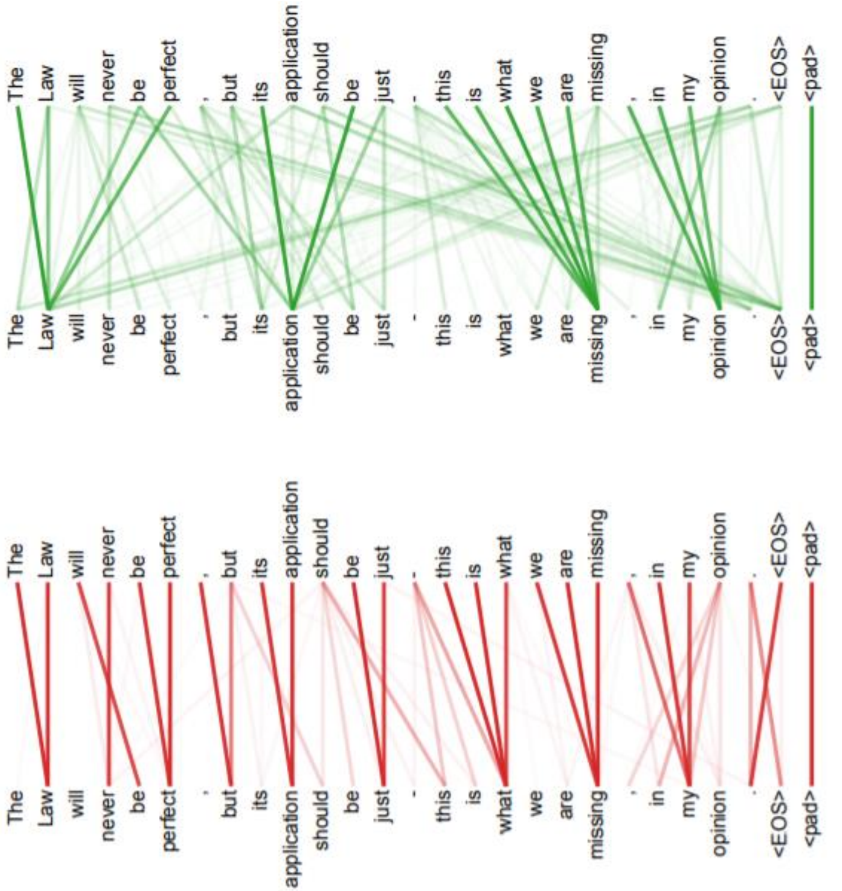
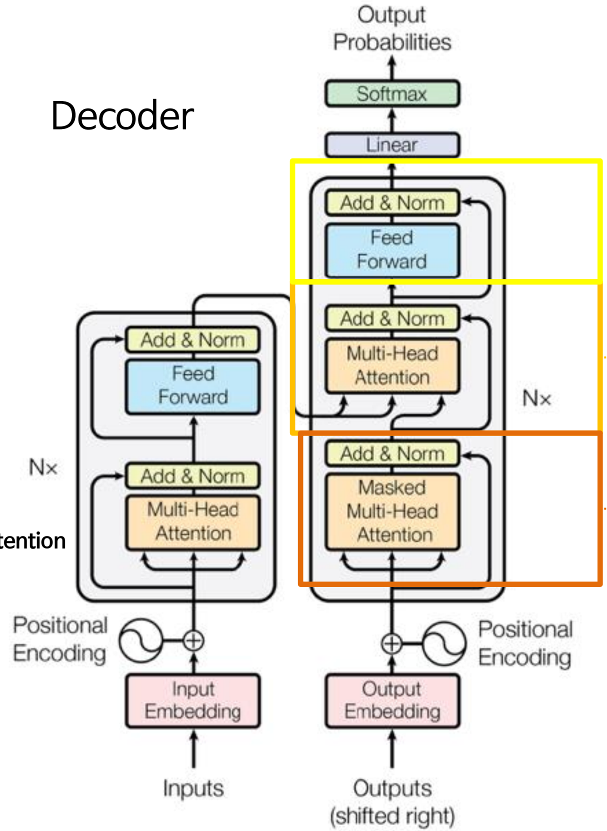


Figure 5: Many of the attention heads exhibit behaviour that seems related to the structure of the sentence. We give two such examples above, from two different heads from the encoder self-attention at layer 5 of 6. The heads clearly learned to perform different tasks.

2. Model Architecture

Decoder

Encoder Self-Attention

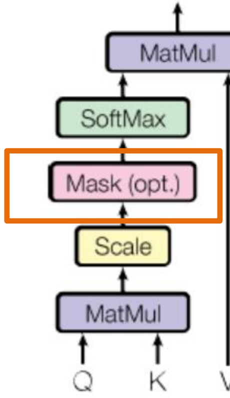


3rd sub layer
Position-wise fully connected
Feed-forward network

2nd sub layer
Enconder-Decoder Attention

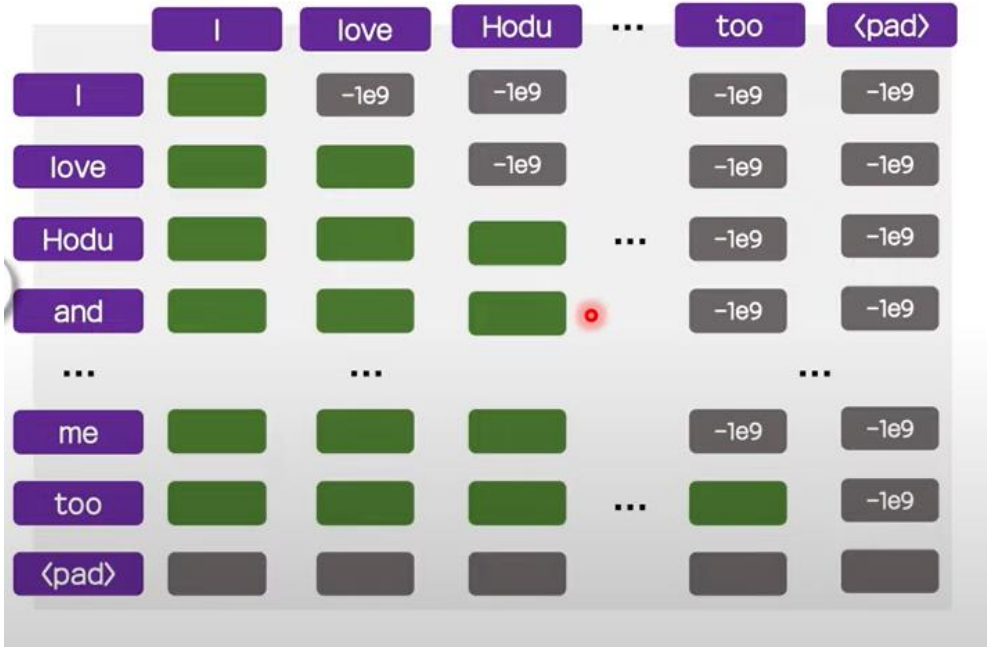
1st sub layer
Masked Decoder Self-Attention

2nd sub-layer Input :
- Queries : from previous decoder layer
- Keys,Values : from output of encoder



Masking out (setting to $-\infty$) all values in the input of the softmax which correspond to illegal connections

2. Model Architecture



3. Why Self-Attention

1. 레이어당 전체 연산량이 줄어든다
2. 병렬화가 가능한 연산이 늘어난다
3. Long-range term dependency도 잘 학습할 수 있게 된다
4. 모델 자체 동작을 해석하기 쉬워진다

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

4. Training

4.1 Traing Data and Batching

- WMT 2014 English-French dataset
: 4.5 million sentence pairs (37000 tokens)

```
iron cement is a ready for use paste which is laid as a fillet by putty knife or finger in the mould edges
iron cement protects the ingot against the hot , abrasive steel casting process .
a fire restant repair cement for fire places , ovens , open fireplaces etc .
Construction and repair of highways and ...
An announcement must be commercial character .
Goods and services advancement through the P.O.Box system is NOT ALLOWED .
Deliveries ( spam ) and other improper information deleted .
Translator Internet is a Toolbar for MS Internet Explorer .
It allows you to translate in real time any web pasge from one language to another .
You only have to select languages and TI does all the work for you ! Automatic dictionary updates ....
This software is written in order to increase your English keyboard typing speed , through teaching the bas
keyboard and give some training examples .
Each lesson teaches some extra keys , and there is also a practice , if it is chosen , one can practice the
previous lessons . The words chosen in the practice are mostly meaningful and relates to the tough keys ...
Are you one of millions out there who are trying to learn foreign language , but never have enough time ?
```

https://nlp.stanford.edu/projects/nmt/

- WMT 2014 English-G dataset
: 36 mililion sentences (32000 word-piece vocabulary)

- Training batch : 25000 source tokens and 25000 target tokens

4. Training

4.2 Hardware and Schedule

- Hardware : 8 NVIDIA P100 GPUs
- Schedule : [Each training step] 0.4sec
  [Base models;a total of 100,000 steps] 12hrs, [Big models;300,000 steps] 3.5 days

4.3 Optimizer

- Adam Optimizer

4.4 Regularization

- Residual Dropout
  1) Applied to the ouput of each sub-layer; before it is added to the sub-layer input and normalized
  2) Applied to the sums of the embeddings and the positional encodings in both the encoder and decoder
  (P drop = 0.1)

- Label Smoothing
  : This hurts perplexity, as the model learns to be more unsure, but improves accuracy and BLEU score.

5. Results

5.1 Machine Translation

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	$\mathbf{3.3 \cdot 10^{18}}$	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

5. Results

5.2 Model Variations

Table 3: Variations on the Transformer architecture. Unlisted values are identical to those of the base model. All metrics are on the English-to-German translation development set, newstest2013. Listed perplexities are per-wordpiece, according to our byte-pair encoding, and should not be compared to per-word perplexities.

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	ϵ_{ls}	train steps	PPL (dev)	BLEU (dev)	params $\times 10^6$
base	6	512	2048	8	64	64	0.1	0.1	100K	4.92	25.8	65
(A)				1	512	512				5.29	24.9	
				4	128	128				5.00	25.5	
				16	32	32				4.91	25.8	
				32	16	16				5.01	25.4	
(B)					16					5.16	25.1	58
					32					5.01	25.4	60
(C)	2									6.11	23.7	36
	4									5.19	25.3	50
	8									4.88	25.5	80
		256			32	32				5.75	24.5	28
		1024			128	128				4.66	26.0	168
			1024							5.12	25.4	53
			4096							4.75	26.2	90
(D)							0.0			5.77	24.6	
							0.2			4.95	25.5	
								0.0		4.67	25.3	
								0.2		5.47	25.7	
(E)	positional embedding instead of sinusoids									4.92	25.7	
big	6	1024	4096	16			0.3		300K	**4.33**	**26.4**	213

6. Conclusion

- Transformer : Recurent, Convolution을 사용하지 않고, attention만 사용한 모델
- 다른 모델들보다 훨씬 빠른 학습속도 그리고 좋은 성능을 지님
- 번역 뿐만 아니라 이미지 등 큰 입력을 갖는 분야에도 적용될 것이 기대됨

감사합니다.