Deep Learning:

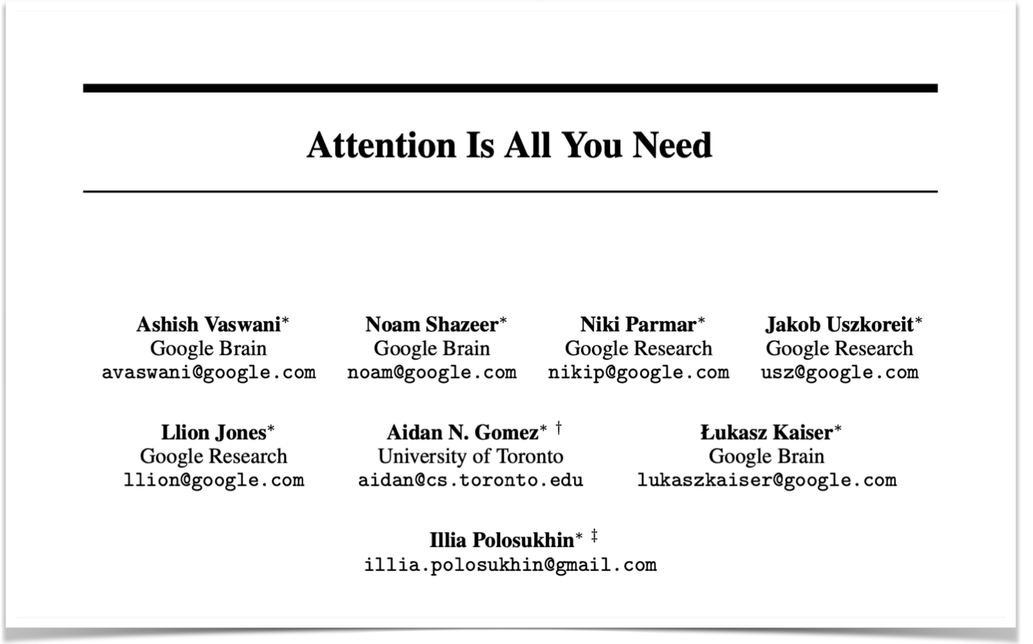
Attention & Transformers

Ozan Özdenizci
Institute of Theoretical Computer Science
ozan.ozdenizci@igi.tugraz.at

Deep Learning VO - WS 23/24
Lecture 12 - January 15th, 2024

Today

- Transformers
  - Attention
  - Transformer Architecture
  - Further Extensions
    - Generative Pre-Trained Transformer (GPT)
    - Vision Transformers (ViT)



Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Vaswani et al., “Attention is all you need”, NIPS 2017.

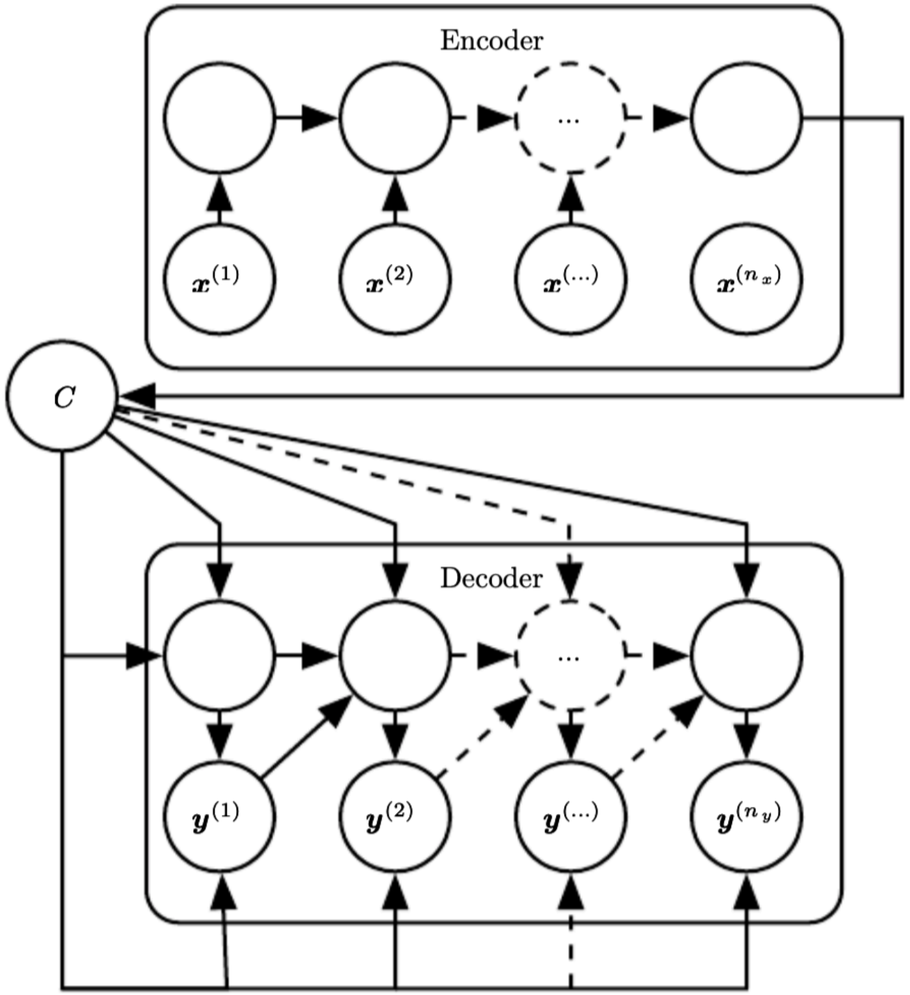
Recap: Text Translation with RNNs

Sequence to Sequence Learning with Neural Networks

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com



- If the output sequence does not have the same length as input sequence, e.g. in language translation.

Input: Sequence $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n_x)}$

Output: Sequence $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(n_y)}$

Encoder: Processes input and emits the context C, typically a simple function of its final hidden state.

Decoder: Generates the output sequence conditioned on this context C.

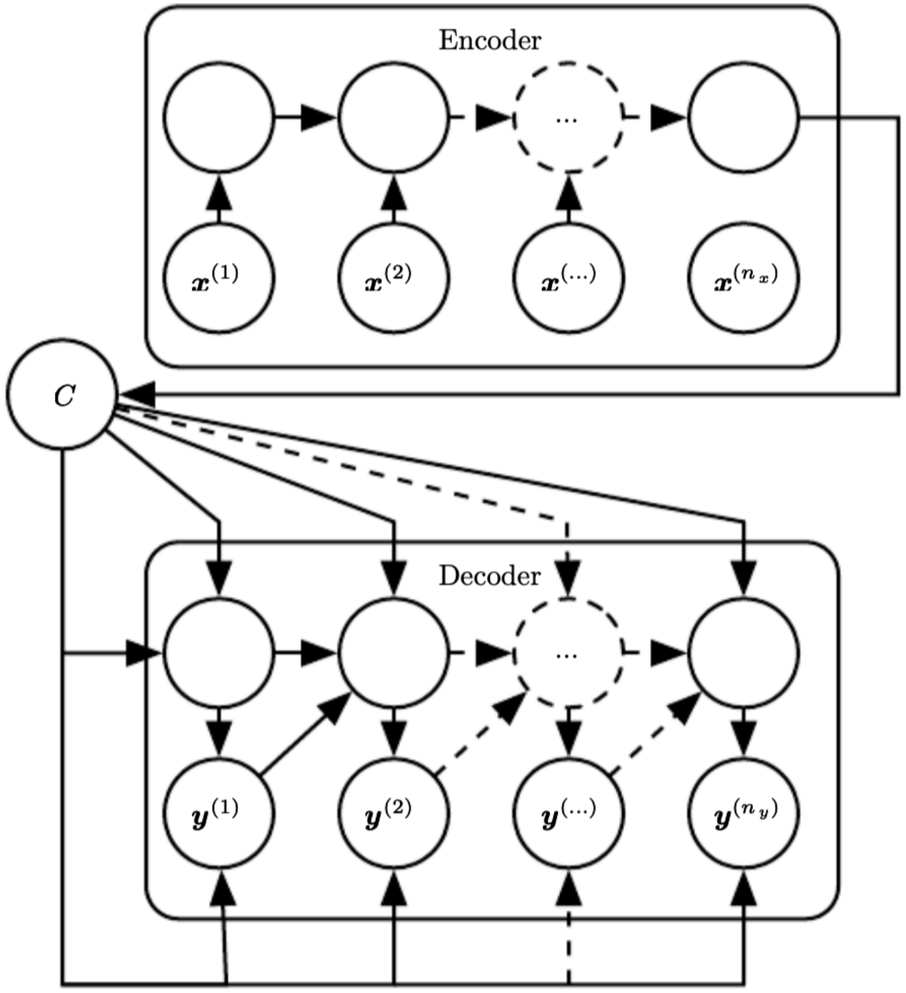
Motivation

Problem of RNNs for Sequence-to-Sequence learning:

- The context C has to capture all relevant information.
- Why can't we just take the hidden states during encoding at all time steps and use them in the decoder?

More problems:

- Long-range dependencies.
- Vanishing/exploding gradients.
- Bad for parallel computations.

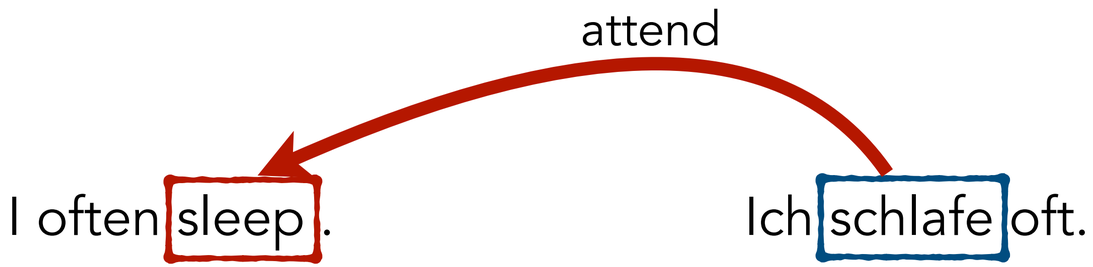


Motivation

We can do encoding-decoding without RNNs:

- The context C has to capture all relevant information.
- Why can't we just take the hidden states during encoding at all time steps and use them in the decoder?

Using all information is too much. Instead, attend to relevant information.



Attention is a mechanism that filters out irrelevant information.

Attention in Neural Networks

Similar to database queries.

Consider a database with L entries, where each entry i consist of a key $\mathbf{k}_i$ and a value $\mathbf{v}_i$

Query
$\mathbf{q}$

Keys	Values
$\mathbf{k}_1$	$\mathbf{v}_1$
$\mathbf{k}_2$	$\mathbf{v}_2$
.	.
.	.
$\mathbf{k}_L$	$\mathbf{v}_L$

Result: The value for which the key matches the query best.

$\mathbf{v}_i$ s.t.

$\text{sim}(\mathbf{q}, \mathbf{k}_i) \geq \text{sim}(\mathbf{q}, \mathbf{k}_j)\ \forall j$

In a neural network, all needs to be differentiable:

Vector $\mathbf{q}$ for query, $\mathbf{k}_i$ for i-th key, and $\mathbf{v}_i$ for i-th value, are all $\in \mathbb{R}^d$.

Attention in Neural Networks

In a neural network, all needs to be differentiable:

Vector $\mathbf{q}$ for query, $\mathbf{k}_i$ for i-th key, and $\mathbf{v}_i$ for i-th value, are all $\in \mathbb{R}^d$.

Keys	Values
$\mathbf{k}_1$	$\mathbf{v}_1$
$\mathbf{k}_2$	$\mathbf{v}_2$
.	.
.	.
$\mathbf{k}_L$	$\mathbf{v}_L$

Similarity measure: e.g., scaled dot product

$$\text{sim}(\mathbf{q}, \mathbf{k}_i) = \frac{\mathbf{q}^T \mathbf{k}_i}{\sqrt{d}}$$

Attention weights $\mathbf{a}$: Tells us how much to attend to each of the values.

$$\mathbf{a} = \text{softmax}\left(\text{sim}(\mathbf{q}, \mathbf{k}_1), \ldots, \text{sim}(\mathbf{q}, \mathbf{k}_L)\right)$$

Attention value (=output): Sum of values, weighted by attention weights:

$$\text{attention value} = \sum_{i=1}^{L} a_i \mathbf{v}_i$$

Attention in Neural Networks: Example

$$K = \begin{bmatrix} \mathbf{k}_1^T \\ \mathbf{k}_2^T \\ \mathbf{k}_3^T \end{bmatrix} = \begin{bmatrix} 0.9, & 0.4, -1.1 \\ 0.2, -1.0, & 0.5 \\ -1.0, & 0.0, & 1.0 \end{bmatrix} \qquad V = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \mathbf{v}_3^T \end{bmatrix} = \begin{bmatrix} 1,0,0 \\ 0,2,0 \\ 0,0,1 \end{bmatrix}$$

$$\mathrm{sim}(\mathbf{q}, \mathbf{k}_i) = \frac{\mathbf{q}^T \mathbf{k}_i}{\sqrt{d}}$$

$$\mathbf{a} = \mathrm{softmax}\left(\mathrm{sim}(\mathbf{q}, \mathbf{k}_1), \ldots, \mathrm{sim}(\mathbf{q}, \mathbf{k}_L)\right)$$

$$\text{attention value} = \sum_{i=1}^{L} a_i \mathbf{v}_i$$

$$\mathbf{q} = \begin{bmatrix} 1 \\ 0.5 \\ -1 \end{bmatrix} \longrightarrow \text{a single query}$$

$$\frac{K\mathbf{q}}{\sqrt{3}} = \left[1.27, -0.46, -1.15\right]^T$$

$$\mathbf{a} = [0.79, 0.14, 0.07]$$

$$\text{attention value: } (\mathbf{a}V)^T = \begin{bmatrix} 0.79 \\ 0.28 \\ 0.07 \end{bmatrix}$$

Keys	Values
$\mathbf{k}_1^T = (\ 0.9, \ 0.4, -1.1)$	$\mathbf{v}_1^T = (1,0,0)$
$\mathbf{k}_2^T = (\ 0.2, -1.0, \ 0.5)$	$\mathbf{v}_2^T = (0,2,0)$
$\mathbf{k}_3^T = (-1.0, \ 0.0, \ 1.0)$	$\mathbf{v}_3^T = (0,0,1)$

Attention in Neural Networks: Summary with Matrices

Matrix V (L x d) : each row one **value**

Matrix K (L x d) : each row corresponding **key**

Matrix Q (N x d) : each row one **query** (for batch processing, several queries at once)

$$V = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_L^T \end{bmatrix}$$

d : dimension of vectors

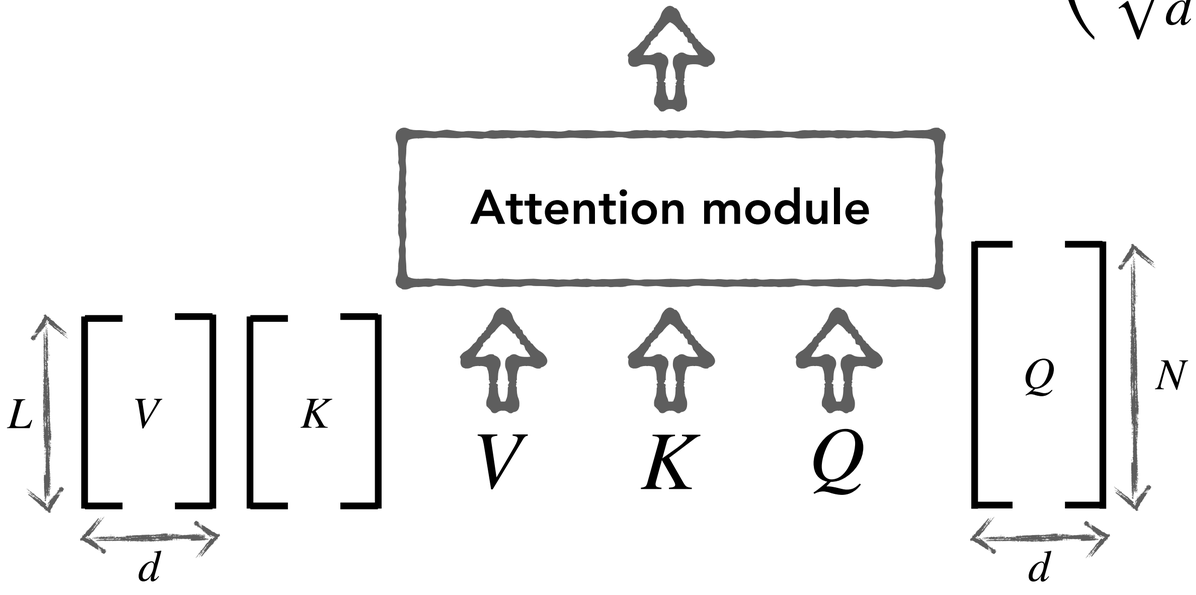
L : number of key-value pairs

N : number of queries

$$\text{Attention}(Q, K, V) \in \mathbb{R}^{N \times d}$$

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

$$K = \begin{bmatrix} \mathbf{k}_1^T \\ \mathbf{k}_2^T \\ \vdots \\ \mathbf{k}_L^T \end{bmatrix}$$



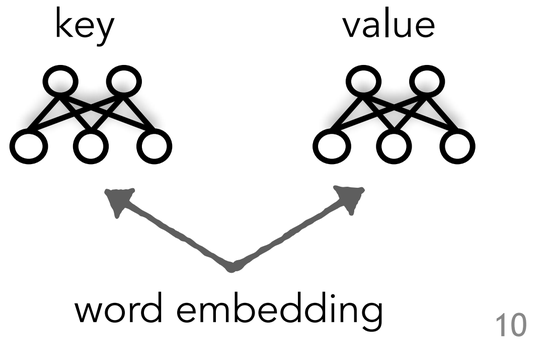
$$Q = \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_N^T \end{bmatrix}$$

Simple Encoder

Encoder:

- A sequence of word embeddings (more generally, a sequence of *tokens*) is taken as input.
- A neural network computes a key and a value vector for each word (more generally, for all tokens of the input sequence).
- Result: a key-matrix K and a value-matrix V

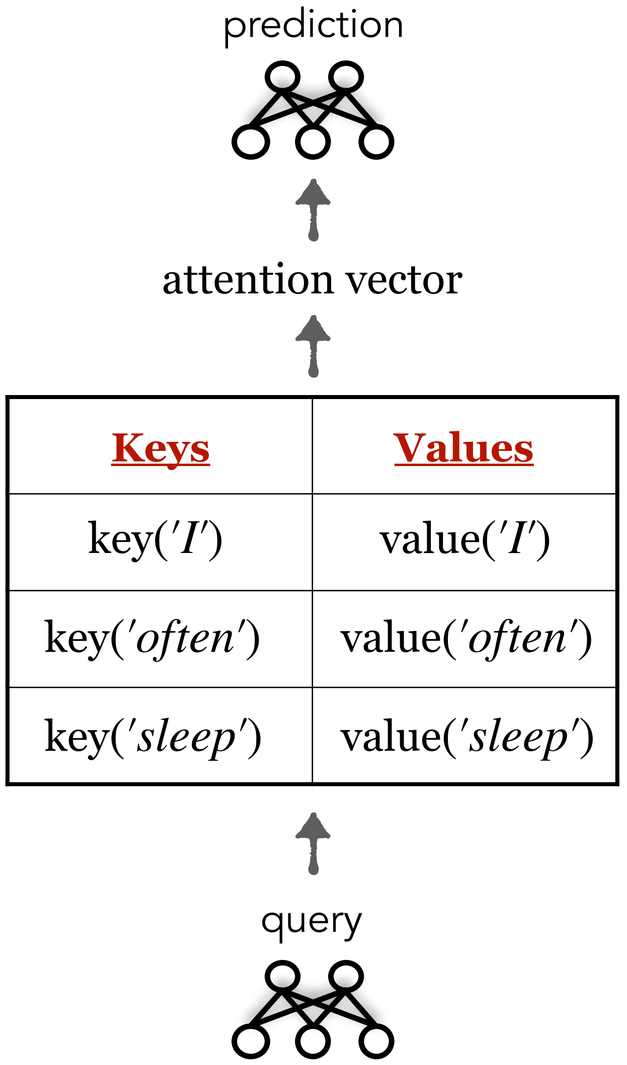
Keys	Values
key($'I'$)	value($'I'$)
key($'often'$)	value($'often'$)
key($'sleep'$)	value($'sleep'$)



Simple Encoder

Encoder:

- A sequence of word embeddings (more generally, a sequence of *tokens*) is taken as input.
- A neural network computes a key and a value vector for each word (more generally, for all tokens of the input sequence).
- Result: a key-matrix K and a value-matrix V
- To produce a prediction, another network computes a query vector which is applied.
- The attention vector is used by another network to compute the prediction.



Simple Encoder with Position Encoding

Encoder:

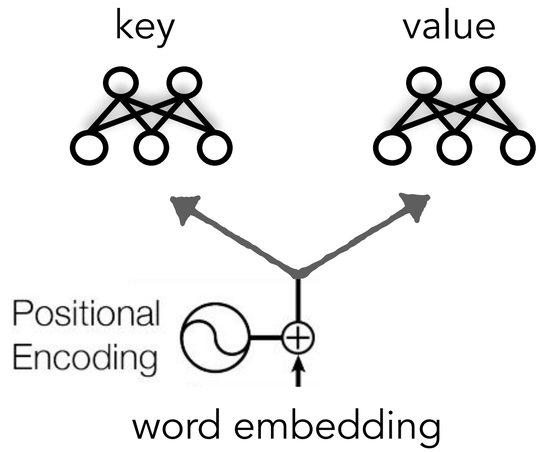
- Keys and values contain no information about position of words in the sentence.
- We can add a position encoding to include such information.

Positional encoding:

Add information about word position to the embedding of $\mathbf{x}_i$

$$\hat{\mathbf{x}}_i = \text{Emb}(\mathbf{x}_i) + \mathbf{p}_i$$

Keys	Values
$\text{key}('I, pos1')$	$\text{value}('I, pos1')$
$\text{key}('often, pos2')$	$\text{value}('often, pos2')$
$\text{key}('sleep, pos3')$	$\text{value}('sleep, pos3')$



Self-Attention

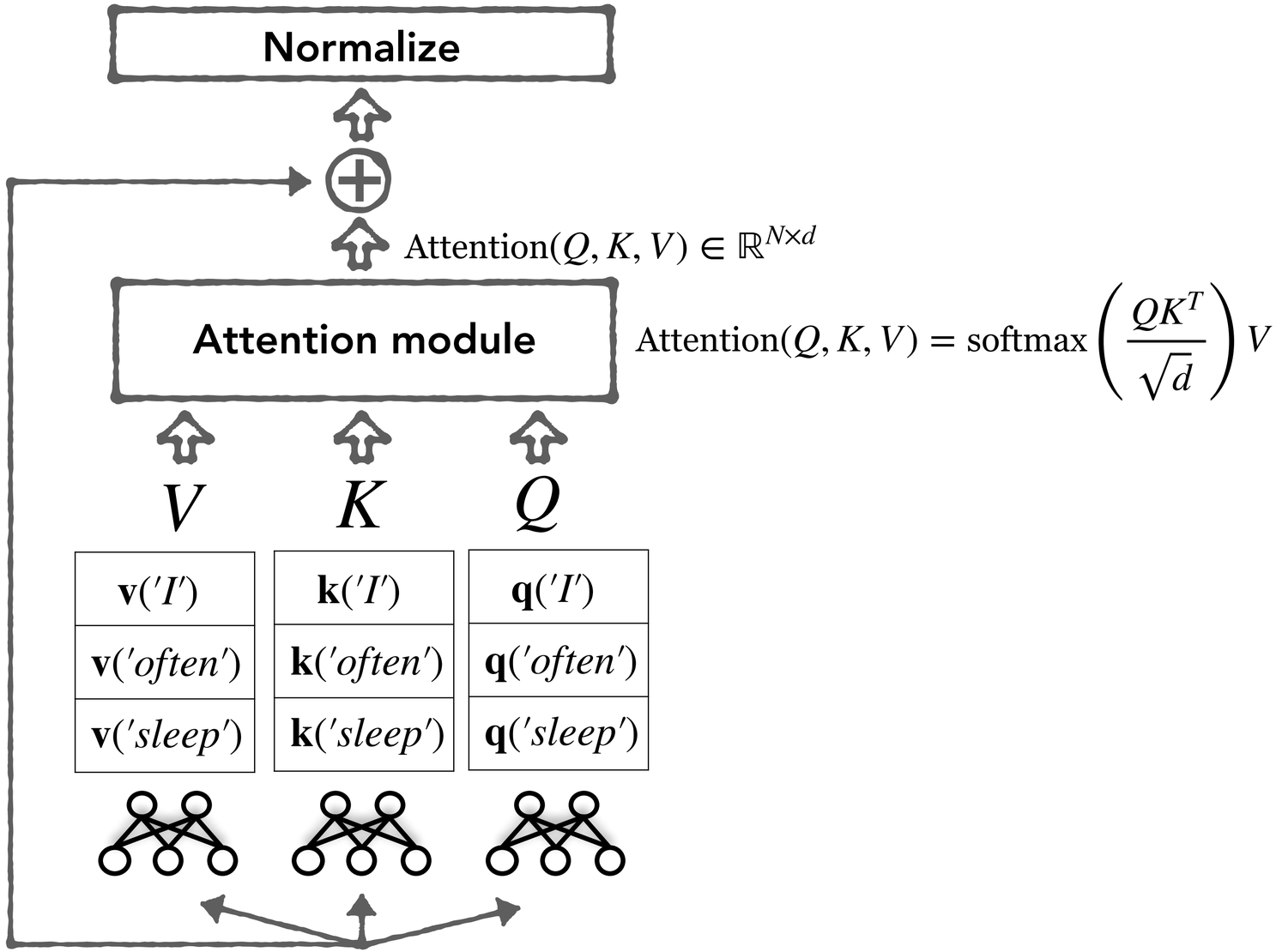
Goal: For each word in an input sentence (sequence of tokens), enrich its representation with context from other words.

- *"The child was scared, since it was alone"*, the word "*it*" could be enriched with "*child*".

Self-Attention: Example

Goal: For each word in an input sentence (sequence of tokens), enrich its representation with context from other words.

- *"The child was scared, since it was alone"*, the word "*it*" could be enriched with "*child*".



Self-Attention: Complexity

Layer Type	Self-Attention	RNN
Computational Complexity per layer	$O(L^2 d)$	$O(Ld^2)$
Sequential Operations	$O(1)$	$O(L)$
Maximum Path Length	$O(1)$	$O(L)$

Computational Complexity:

- **Attention:** For each token, the attention over all other tokens is computed: $O(L^2 d)$
  - This limits the length of sequences that can be tackled.
- **RNN:** (d being the number of neurons in recurrent layer). $O(d^2)$ recurrent weights, used L times: $O(Ld^2)$.

Sequential Operations: Indicates parallelizability of the computation.

- **Attention:** We can perform the computations on all tokens in parallel.
- **RNN:** Hidden for i-th token state depends on hidden state for previous tokens: $O(L)$

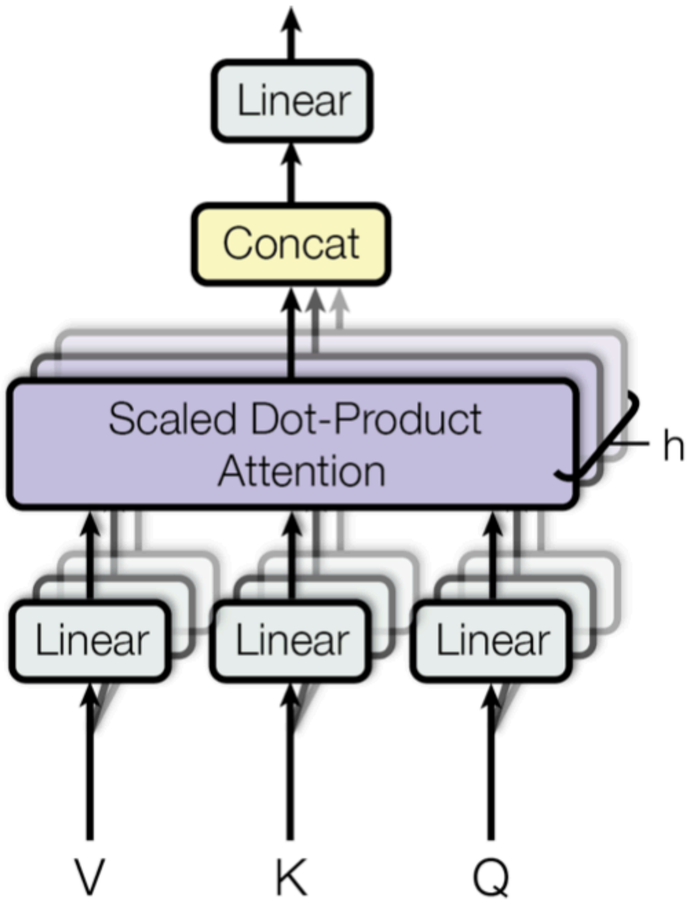
Maximum Path Length: Length between long-range dependencies in the network.

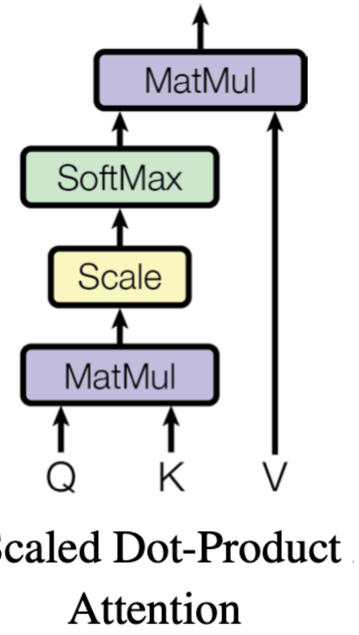
- **Attention:** We can attend directly to any position in the sequence: $O(1)$
- **RNN:** Signals need to travel through the temporally unfolded network.

Multi-Head Attention

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

- **Idea:** Several attention-modules in parallel with different weight matrices are used.
- Outputs of these are then linearly combined to obtain same output dimensionality as in single-headed attention.
- This allows the model to jointly attend to information from different representation subspaces at different positions.





Today

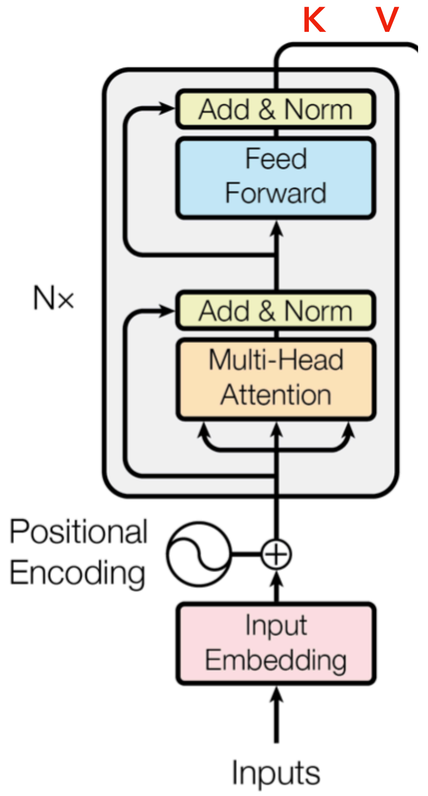
- [ ] Transformers
  - [x] Attention
  - [ ] Transformer Architecture
  - [ ] Further Extensions
    - [ ] Generative Pre-Trained Transformer (GPT)
    - [ ] Vision Transformers (ViT)

Encoder-Decoder Architecture

[Vaswani et al., "Attention is all you need", NIPS 2017.]

Encoder:

- A sequence of word embeddings (more generally, a sequence of *tokens*) is taken as input.
- Perform Multi-Head Self-Attention.
- Perform a feed-forward neural network transformation .
- Outputs a key-matrix K and a value-matrix V.



Encoder-Decoder Architecture

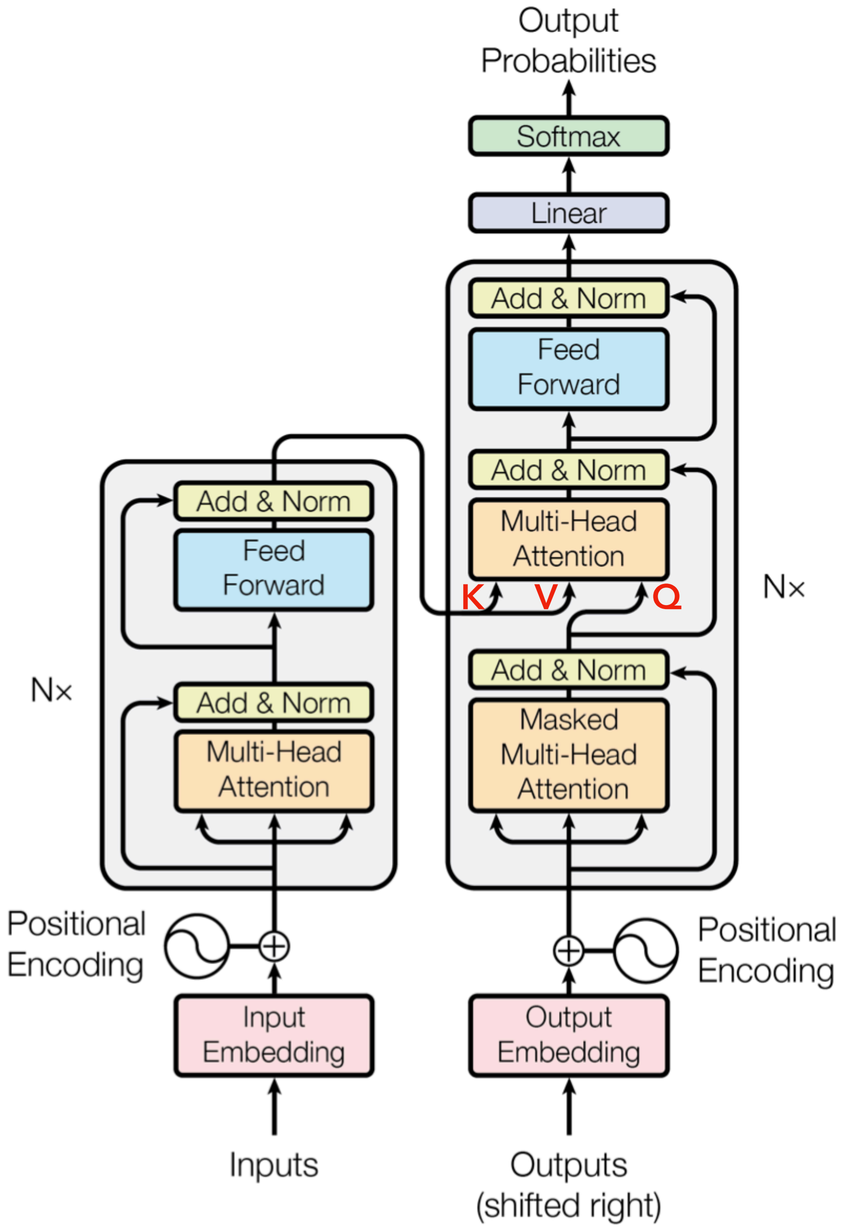
[Vaswani et al., "Attention is all you need", NIPS 2017.]

Encoder:

- A sequence of word embeddings (more generally, a sequence of *tokens*) is taken as input.
- Perform Multi-Head Self-Attention.
- Perform a feed-forward neural network transformation .
- Outputs a key-matrix K and a value-matrix V.

Decoder:

- Get as input all words that have been predicted so far.
  - For the first word, the input is <start>.
- Perform Multi-Head Self-Attention on it, which results in a query.
- Use this to query the encoder's output key-value matrix.
- One more network to transform the attention vector into a prediction.



Transformer Architecture

[Vaswani et al., "Attention is all you need", NIPS 2017.]

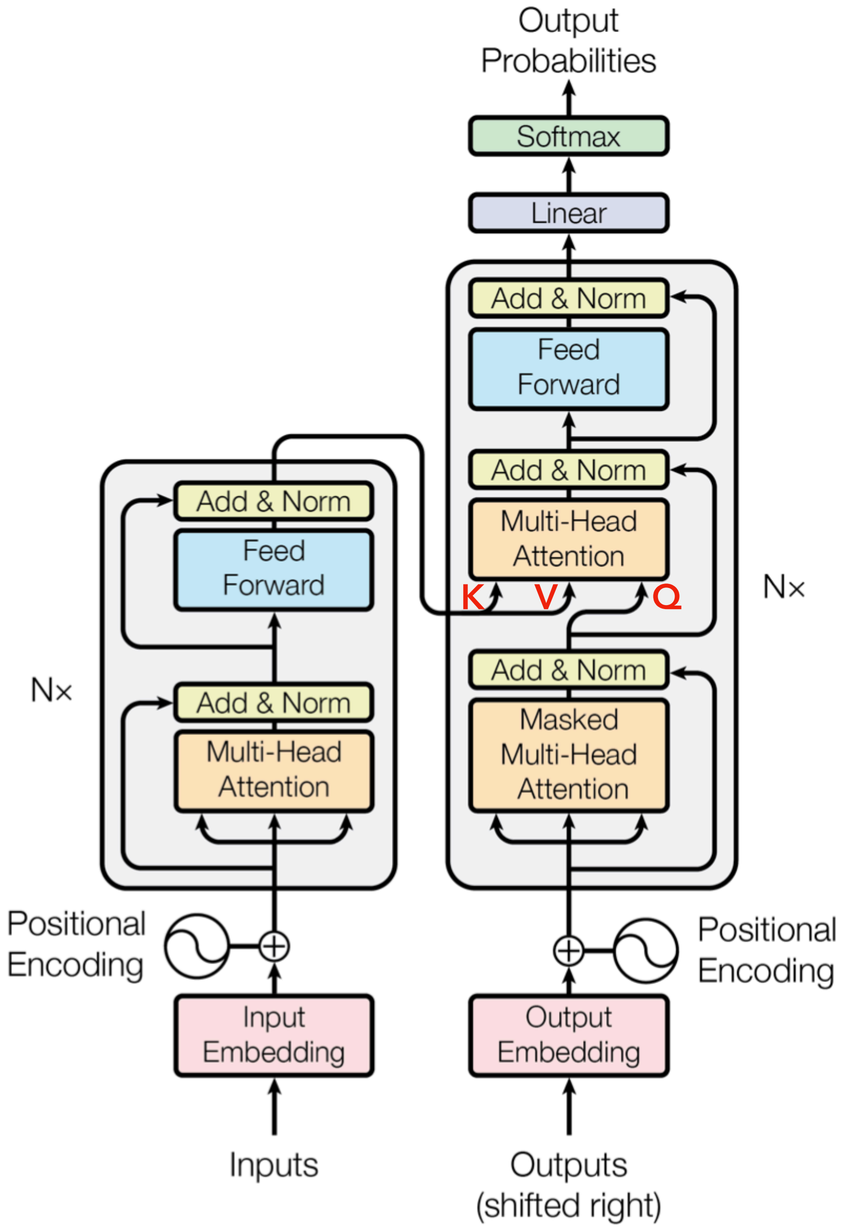
Add and Norm:

- Add residual connections
- Normalize layer output to zero-mean and unit variance (using layer norm).

Feed Forward:

- Standard neural network with one hidden layer and a ReLU activation in the hidden layer.

Nx: The block is replicated 6 times.



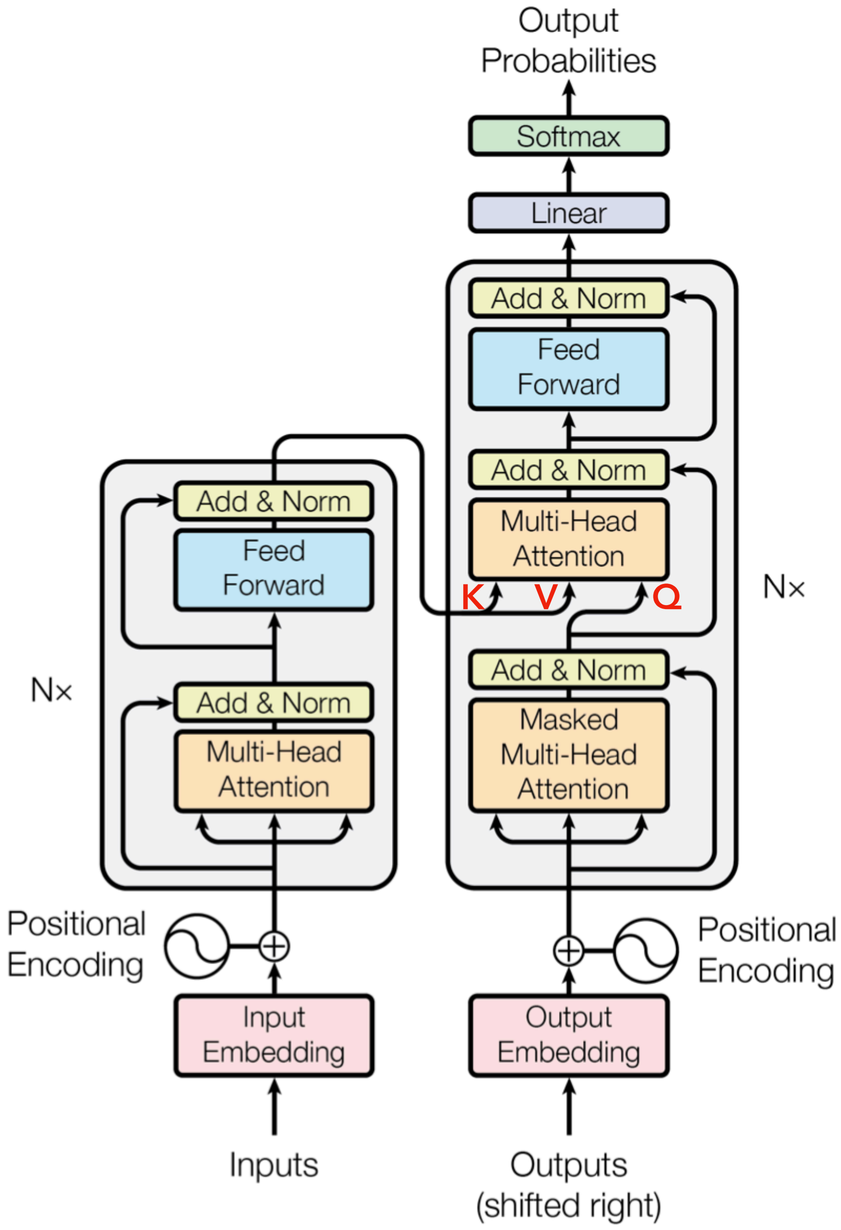
Transformer Architecture

[Vaswani et al., "Attention is all you need", NIPS 2017.]

Decoder - When predicting word i:

- Get predicted words 1 to i-1 as input (on first word, input is <start>)
- Perform Multi-Head Self-Attention on it.
- Then perform cross-attention:
  - Query from own MHA output, and use the key-value matrices from the encoder output.
- Feed forward neural network transformation.
- Finally, one linear layer and a Softmax predicts the word.

Masked MHA: Technicality for training, to not use future predicted words in self-attention.



Today

- [ ] Transformers
  - [x] Attention
  - [x] Transformer Architecture
  - [ ] Further Extensions
    - [ ] Generative Pre-Trained Transformer (GPT)
    - [ ] Vision Transformers (ViT)

Decoder-only Architecture

[Liu et al., "Generating Wikipedia by Summarizing Long Sequences", ICLR 2018.]

Recent architectures (e.g., GPT) skip the encoder

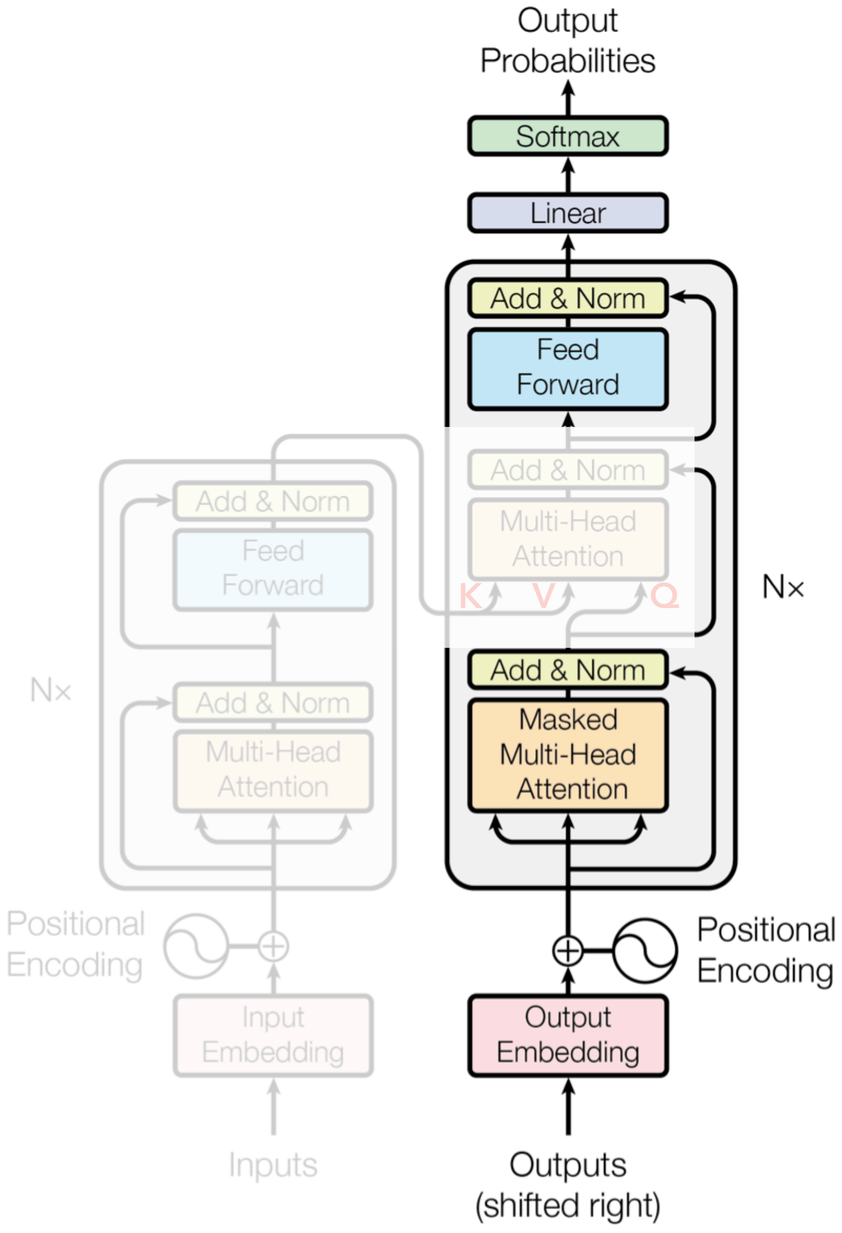
(= Transformer-Decoder architecture)

The transformer block then typically consists of:

- Multi-Head Self-Attention (with residual connections)
- Layer Norm
- Feed Forward Layer (with residual connections)
- Layer Norm

Advantages:

- Reduces number of parameters by almost half.
- Encoder and decoder typically learn redundant representations.



Decoder-only Architecture

[Liu et al., "Generating Wikipedia by Summarizing Long Sequences", ICLR 2018.]

Input:

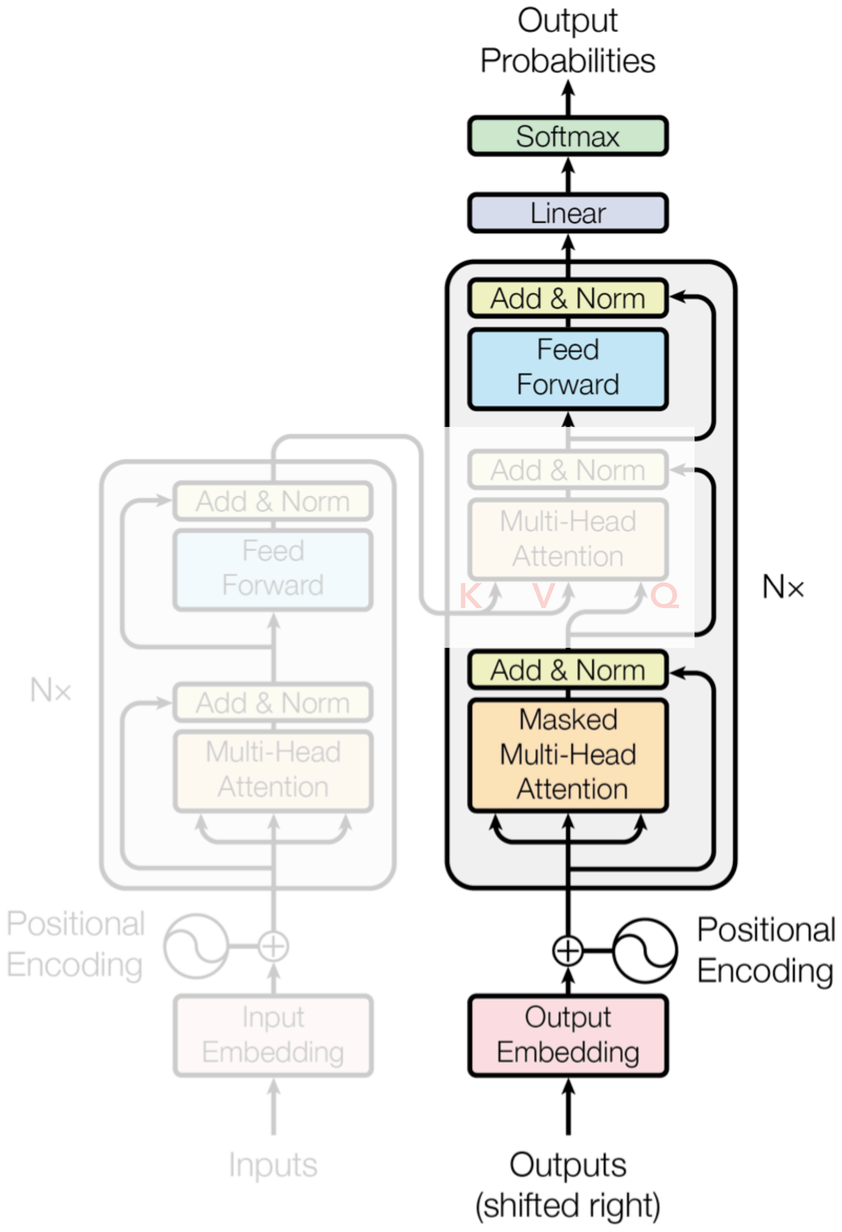
A sequence transduction example $(x^1, \ldots, x^n) \rightarrow (y^1, \ldots, y^m)$ is converted into a sequence $(x^1, \ldots, x^n, \delta, y^1, \ldots, y^m)$ with:

- x^i : i-th input token
- y^i : i-th output token
- δ : delimiter token

Model is trained to predict the next word given the previous ones:

$$p(w^1, \ldots, w^{n+m}) = \prod_j p(w^j \mid w^1, \ldots, w^{j-1})$$

- During training, the model is also trained to predict the input.
- **Inference:** Start with input sequence and delimiter, predict output.



Decoder-only Architecture

[Liu et al., “Generating Wikipedia by Summarizing Long Sequences”, ICLR 2018.]

Application: Abstractive Summarization

Generating a Wikipedia article by multi-document summarization (e.g., first paragraph of the Wikipedia text).

Input:

- Text from references of the Wikipedia article + documents from web-search on the topic.
- Paragraphs in all documents are pre-ranked. These paragraphs are then given as a single sequence as input to the transformer with a length of up to 11.000 words.

Decoder-only Architecture

Application: Abstractive Summarization

Generating a Wikipedia article by multi-document summari

Input:

- Text from references of the Wikipedia article + document
- Paragraphs in all documents are pre-ranked. These parag input to the transformer with a length of up to 11.000 wo

Dewey & LeBoeuf<EOT>on an april morning in manhattan last year , steven davis , the former chairman of the law firm of dewey & leboeuf , reached for his ringing cell phone . he was sitting in the back seat of a taxi , on the way downtown to renew his passport . dewey & leboeuf , which was often referred to in the press as a global `` super firm , '' was largely his creation . in 2007 , he had engineered the merger of a profitable but staid midsized specialty firm -- leboeuf , lamb , greene & macrae -- with a less profitable but much better - known firm , dewey ballantine . (thomas e. dewey , the former republican presidential nominee , was for many years the guiding partner .) `` dewey married money , leboeuf married up '' was how some characterized the union . it was the largest merger of new york law firms in history , and the new firm had more than thirteen hundred lawyers . dewey & leboeuf handled high - profile transactions for an enviable roster of corporate clients : lloyd 's and a.i.g. in insurance ; duke and bp in energy ; jpmorgan chase and barclays in banking ; disney in media and entertainment ; dell and ebay in technology ; and alcoa in manufacturing . under davis 's leadership , a number of the firm 's partners had joined the ranks of the highest - paid corporate lawyers in the country . dewey & leboeuf llp (dewey) , an international law firm headquartered in new york city , was formed in october 2007 through the combination of dewey ballantine llp and leboeuf , lamb , greene , & macrae llp . at its height , approximately 1,300 partners and employees worked in dewey 's manhattan office , and nearly 3,000 partners and employees worked for the firm worldwide . in may 2012 , dewey collapsed , resulting in the largest law firm bankruptcy in history . jacobs v. altorelli (in re dewey & leboeuf llp) involves the bankruptcy of dewey & leboeuf (dewey) . at its peak , dewey was one of the largest and most prestigious law firms in america . following a wave of partner departures during the first half of 2012 , dewey filed for bankruptcy protection on may 29 , 2012 . when dewey ballantine and leboeuf , lamb , greene & macrae decided in 2007 to join forces to become dewey & leboeuf , mortgage backed securities were still the rage , business was booming and few appreciated the intensity of the storm on the horizon . a mere one year later however , dewey & leboeuf as well as every other major law firm had seen virtually all of its structured finance work disappear and some of those firms were soon to be history . on march 15 , 2012 , the new york times summarized dewey & leboeuf 's predicament as follows : `` tens of millions of dollars in deferred compensation are owed to dewey 's partners . some have been told they are being paid a fraction of what they were promised . the firm is cutting 5 percent of its lawyers and 6 percent of its staff . nineteen of its 300 partners have left dewey since january , including heads of major practice areas . about a dozen more departures are expected ... after the merger , the firm went on a hiring binge , poaching big producers away from rivals with multiyear , multimillion - dollar guarantees . in 2011 alone , it brought on 37 so - called lateral partners . on top of those obligations , the firm , in order to retain essential talent at the time of the merger , gave contracts to dozens of its partners . yet dewey , like many law firms , has failed to see a meaningful recovery from the lean post-financial crisis years . the firm posted sluggish results last year , showing no increase in earnings over 2010 . dewey had budgeted for a double - digit percentage rise in profits . the firm 's enormous compensation commitments , combined with disappointing financial performance , have created a significant shortfall , forcing the firm to slash or defer pay for numerous partners . ' to say that this has caused a morale problem here is something of an understatement , ' said a lawyer at dewey on the condition of anonymity . '' leboeuf lamb greene and macrae llp is an internationally

Decoder-only Architecture

[Liu et al., "Generating Wikipedia by Summarizing Long Sequences", ICLR 2018.]

Transformer-DMAC:
A memory-efficient variant that allows for longer input sequences.

Application: Abstractive Summarization

Generating a Wikipedi

Input:

- Text from references
- Paragraphs in all doc
  input to the transforr

Transformer-encoder-decoder, *L*=100 (log-perplexity: 2.63)
dewey & leboeuf llp (dewey & leboeuf llp) is an american law firm headquartered in new york city . dewey & leboeuf is one of the largest law firms in the united states . dewey & leboeuf has offices in new york city , los angeles , washington , d.c. , washington , d.c. , and washington , d.c.

Transformer decoder, *L*=500 (log-perplexity: 2.60)
dewey & leboeuf llp is an international law firm headquartered in new york city . dewey was formed in october 2007 through the combination of dewey ballantine llp and leboeuf , lamb , greene , & macrae llp .

Transformer-DMAC, L=7000, 256 experts (log-perplexity: 1.90)
dewey & leboeuf llp is an international law firm headquartered in new york city . it was formed in october 2007 through the combination of dewey ballantine llp and leboeuf , lamb , greene & macrae llp . at its height , approximately 1,300 partners and employees worked in dewey 's manhattan office , and nearly 3,000 partners and employees worked for the firm worldwide . in may 2012 , dewey collapsed , resulting in the largest law firm bankruptcy

Wikipedia (ground truth)
dewey & leboeuf llp was a global law firm , headquartered in new york city , that is now in bankruptcy . the firm 's leaders have been indicted for fraud for their role in allegedly cooking the company 's books to obtain loans while hiding the firm 's financial plight . the firm was formed in 2007 through the merger of dewey ballantine and leboeuf , lamb , greene & macrae . dewey & leboeuf was known for its corporate , insurance , litigation , tax and restructuring practices . at the time of the bankruptcy filing , it employed over 1,000 lawyers in 26 offices around the world . in 2012 , the firm 's financial difficulties and indebtedness became public . in the same period , many partners departed , and the manhattan district attorney 's office began to investigate alleged false statements by firm chairman steven davis . as a result of these difficulties , dewey & leboeuf 's offices began to enter administration in may 2012 . the firm filed for bankruptcy in new york on may 28 , 2012 . on march 6 , 2014 , the former chairman , chief financial officer and the executive director of dewey & leboeuf were indicted on charges of grand larceny by the manhattan district attorney .

Generative Pre-Trained Transformer (GPT)

Idea: One can pre-train a Transformer-Decoder architecture as in [Liu et al., 2018] to predict the next word in a large text corpus.

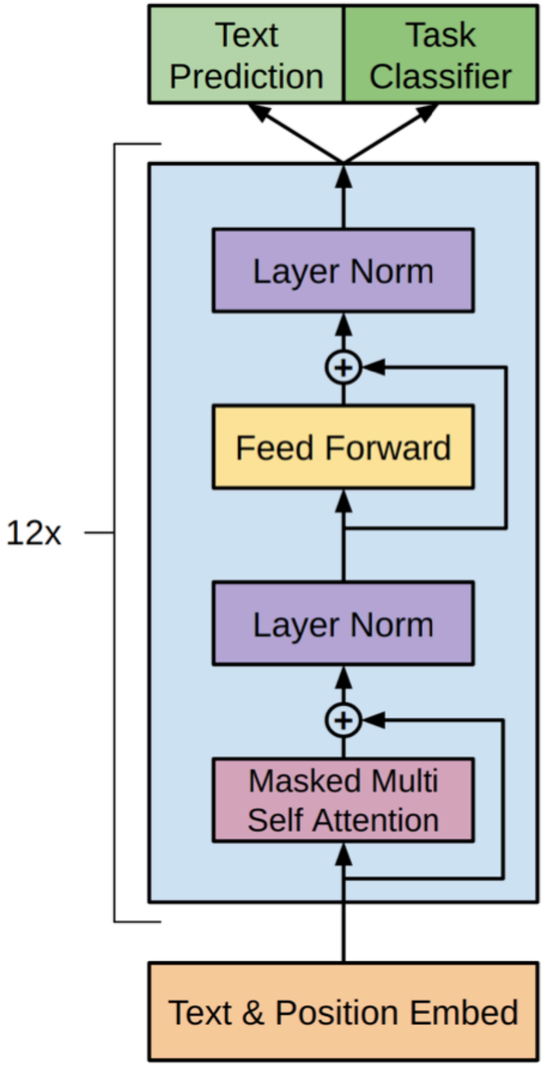
Unsupervised pre-training:

Given an unsupervised corpus of tokens $\mathcal{U} = \langle u^1, \ldots, u^n \rangle$, optimize the likelihood:

$$L(\mathcal{U}) = \sum_i \log P(u^i \mid u^{i-k}, \ldots, u^{i-1}; \theta)$$

k : size of context window

θ : parameters of Transformer that models P.



[Radford et al., "Improving Language Understanding by Generative Pre-Training", 2018.]

Generative Pre-Trained Transformer (GPT)

Idea: One can pre-train a Transformer-Decoder architecture as in [Liu et al., 2018] to predict the next word in a large text corpus.

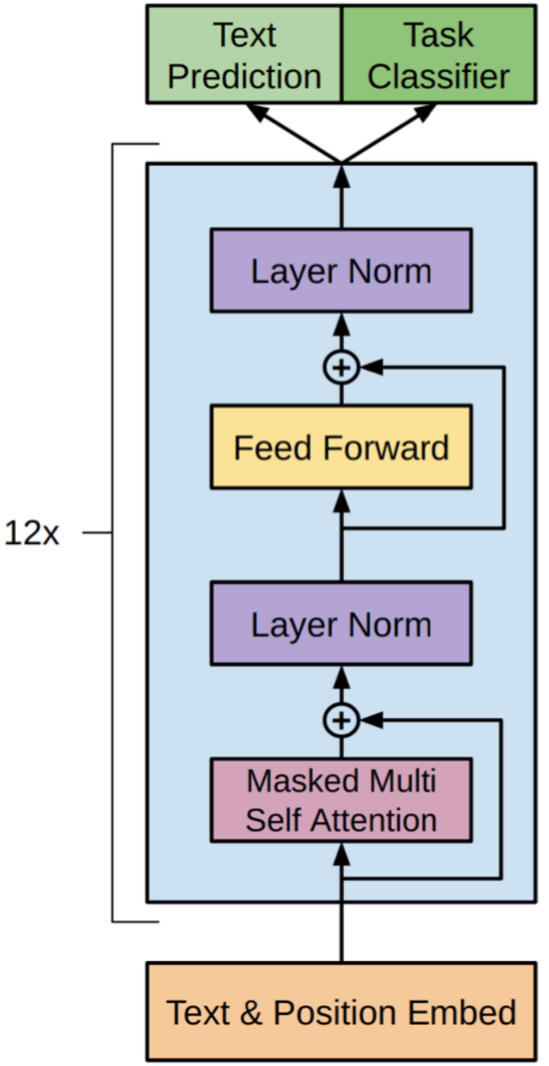
Unsupervised pre-training:

Given an unsupervised corpus of tokens $\mathcal{U} = \langle u^1, \ldots, u^n \rangle$, optimize the likelihood:

$$L(\mathcal{U}) = \sum_i \log P(u^i \mid u^{i-k}, \ldots, u^{i-1}; \theta)$$

After pre-training: A supervised task can be considered, where each instance is given by a sequence $x^1, \ldots, x^m$ and a label y.

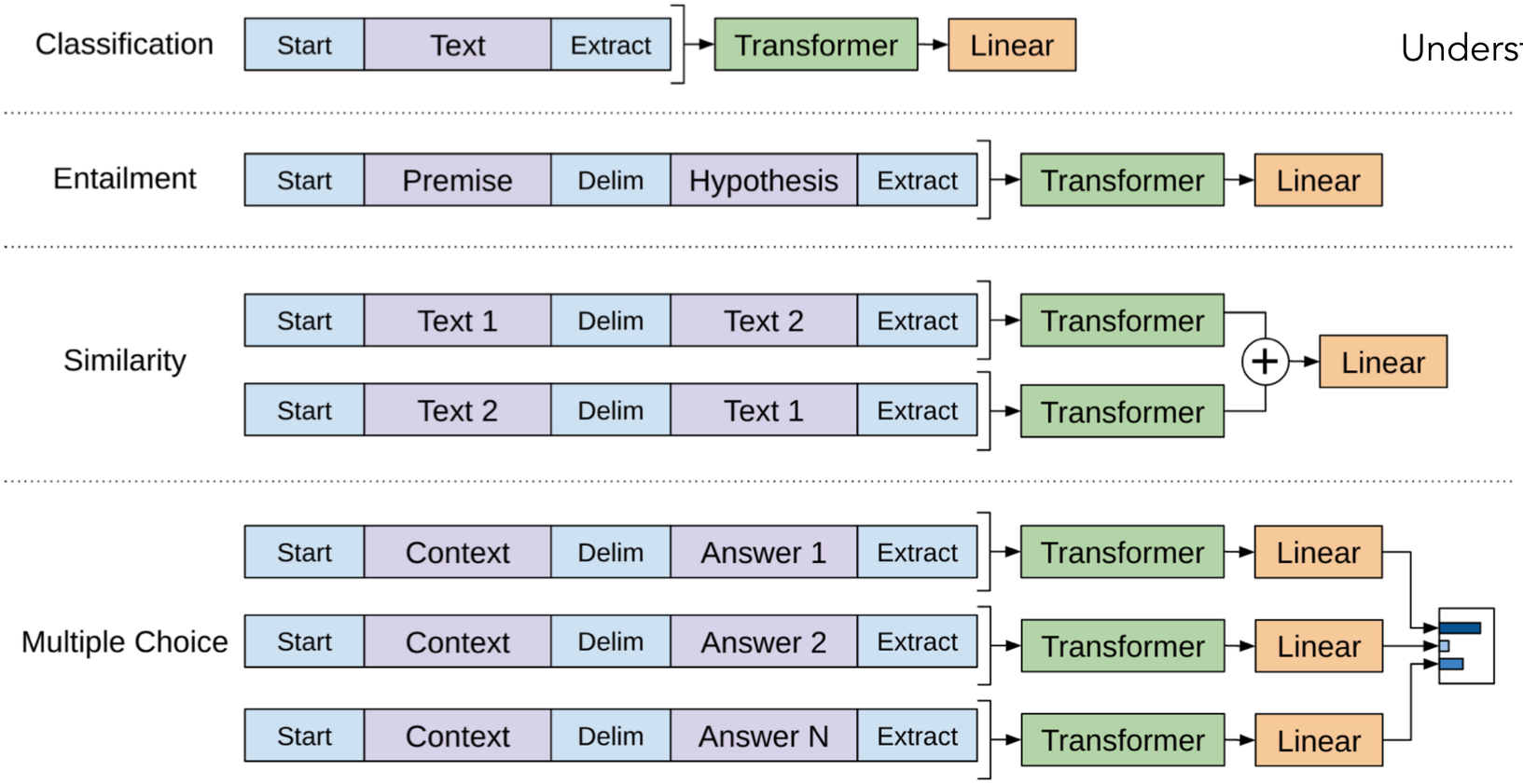
- Inputs are passed through the pre-trained transformer. Its final transformer-block activation is fed into an added linear output layer with parameters W to predict y.

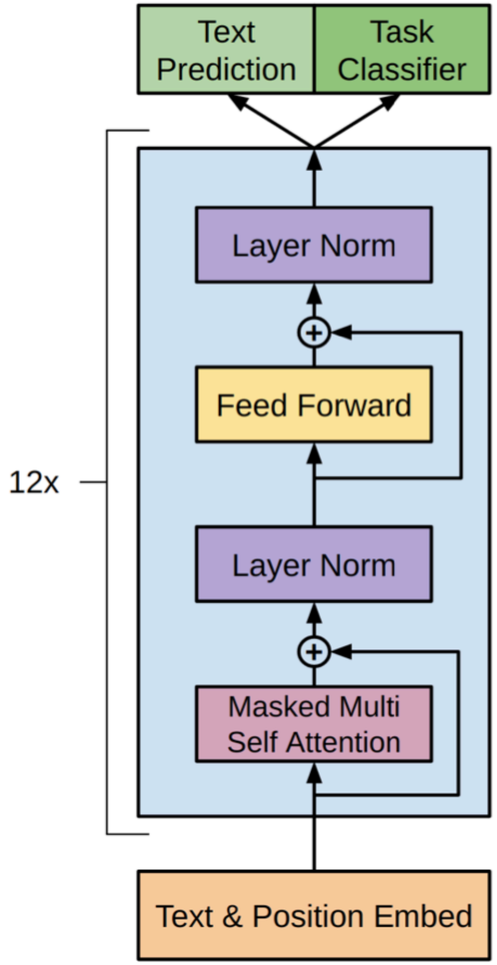


[Radford et al., "Improving Language Understanding by Generative Pre-Training", 2018.]

Generative Pre-Trained Transformer (GPT)

[Radford et al., "Improving Language Understanding by Generative Pre-Training", 2018.]





- **Classification:** Classify text into one of several categories.
- **Entailment:** Classify if the hypothesis follows from the premise.
- **Similarity:** Classify if Text 1 is semantically equal to Text 2.
- **Multiple Choice:** Which answer about the Context text is correct?

Example: RACE Dataset for Reading Comprehension

Passage:
In a small village in England about 150 years ago, a mail coach was standing on the street. It didn't come to that village often. People had to pay a lot to get a letter. The person who sent the letter didn't have to pay the postage, while the receiver had to.
"Here's a letter for Miss Alice Brown," said the mailman.
" I'm Alice Brown," a girl of about 18 said in a low voice.
Alice looked at the envelope for a minute, and then handed it back to the mailman.
"I'm sorry I can't take it, I don't have enough money to pay it", she said.
A gentleman standing around were very sorry for her. Then he came up and paid the postage for her.
When the gentleman gave the letter to her, she said with a smile, " Thank you very much, This letter is from Tom. I'm going to marry him. He went to London to look for work. I've waited a long time for this letter, but now I don't need it, there is nothing in it."
"Really? How do you know that?" the gentleman said in surprise.
"He told me that he would put some signs on the envelope. Look, sir, this cross in the corner means that he is well and this circle means he has found work. That's good news."
The gentleman was Sir Rowland Hill. He didn't forgot Alice and her letter.
"The postage to be paid by the receiver has to be changed," he said to himself and had a good plan.
"The postage has to be much lower, what about a penny? And the person who sends the letter pays the postage. He has to buy a stamp and put it on the envelope." he said . The government accepted his plan. Then the first stamp was put out in 1840. It was called the "Penny Black". It had a picture of the Queen on it.

Questions:

1): The first postage stamp was made _.
A. in England B. in America C. by Alice D. in 1910

2): The girl handed the letter back to the mailman because _ .
A. she didn't know whose letter it was
B. she had no money to pay the postage
C. she received the letter but she didn't want to open it
D. she had already known what was written in the letter

3): We can know from Alice's words that _ .
A. Tom had told her what the signs meant before leaving
B. Alice was clever and could guess the meaning of the signs
C. Alice had put the signs on the envelope herself
D. Tom had put the signs as Alice had told him to

4): The idea of using stamps was thought of by _ .
A. the government
B. Sir Rowland Hill
C. Alice Brown
D. Tom

5): From the passage we know the high postage made _ .
A. people never send each other letters
B. lovers almost lose every touch with each other
C. people try their best to avoid paying it
D. receivers refuse to pay the coming letters

Answer: ADABC

GPT-3: Few-Shot Learning

[Brown et al., "Language Models are Few-Shot Learners", NeurIPS 2020]

Pre-trained models can be used for various tasks *without further fine-tuning*, using a larger GPT architecture.

Few-shot learning after pre-training:

- Randomly draw *K* examples (*<Input> <Delimiter> <Answer>*) and show to Transformer for conditioning.
- Then show an Input for test example, the model predicts the answer.
- For some tasks only a natural language prompt is provided (*K=0*, zero-shot learning).

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  cheese =>                           ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← example
3  cheese =>                           ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```

GPT-3: Few-Shot Learning

[Brown et al., "Language Models are Few-Shot Learners", NeurIPS 2020]

Pre-trained models can be used for various tasks *without further fine-tuning*, using a larger GPT architecture.

Few-shot learning after pre-training:

- Randomly draw *K* examples (*<Input> <Delimiter> <Answer>*) and show to Transformer for conditioning.
- Then show an Input for test example, the model predicts the answer.
- For some tasks only a natural language prompt is provided (*K=0*, zero-shot learning).

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3.

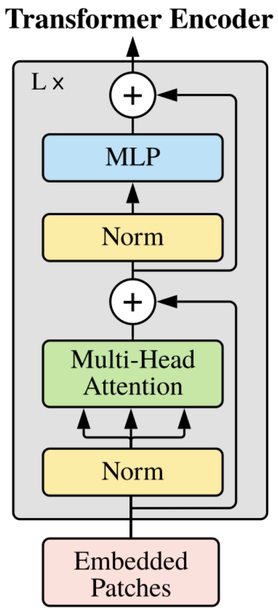
Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}
GPT-3 Small	125M	12	768	12	64
GPT-3 Medium	350M	24	1024	16	64
GPT-3 Large	760M	24	1536	16	96
GPT-3 XL	1.3B	24	2048	24	128
GPT-3 2.7B	2.7B	32	2560	32	80
GPT-3 6.7B	6.7B	32	4096	32	128
GPT-3 13B	13.0B	40	5140	40	128
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128

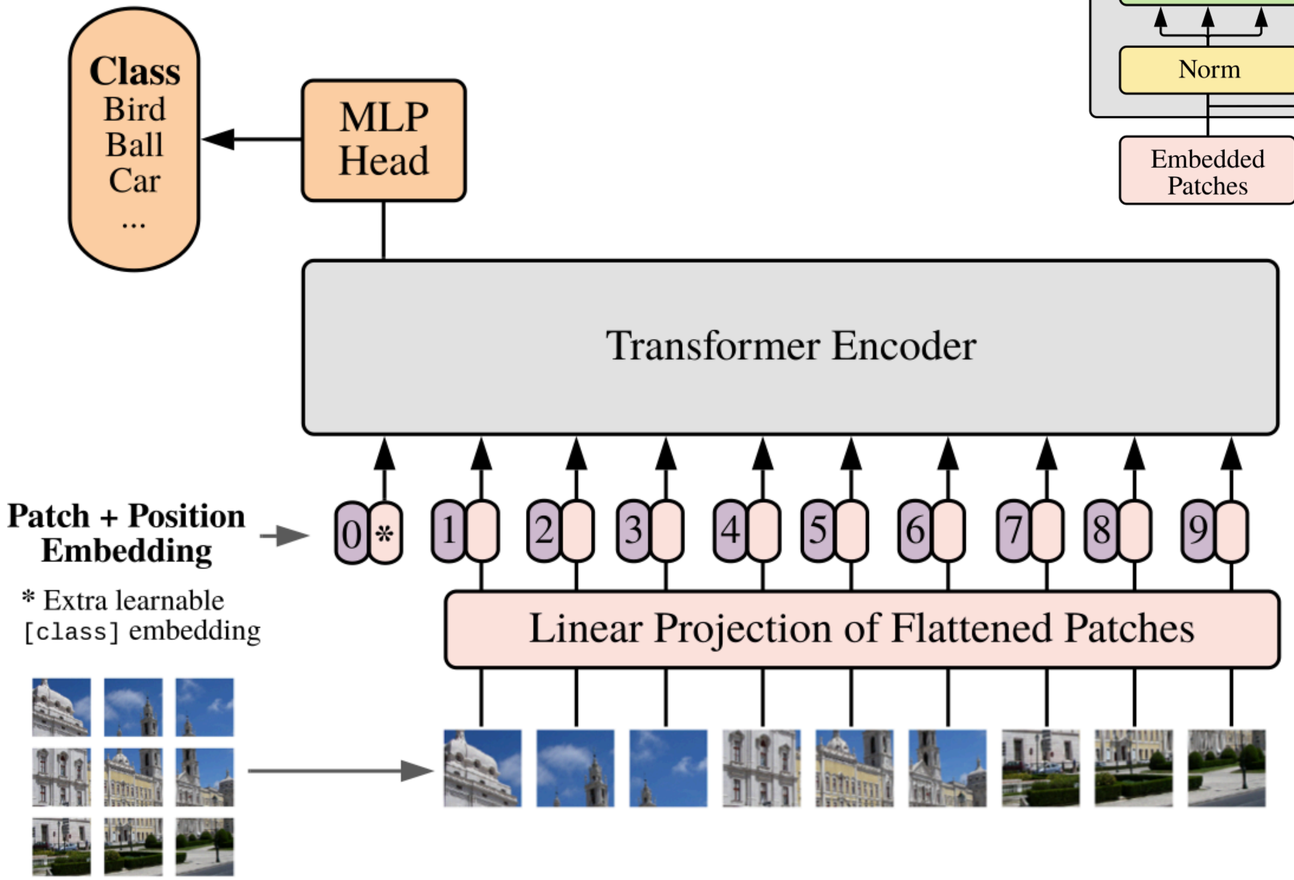
- Training is extremely expensive and energy demanding.

Vision Transformer (ViT)

Application of the Transformer architecture to images.

- Split the image into fixed-size patches.
- Linearly embed each of them and add position embeddings.
- Feed the resulting sequence of vectors to a Transformer encoder.
- For classification:
  - Prepend a learnable class embedding to the input sequence,
  - Train a classification head using this at the output.

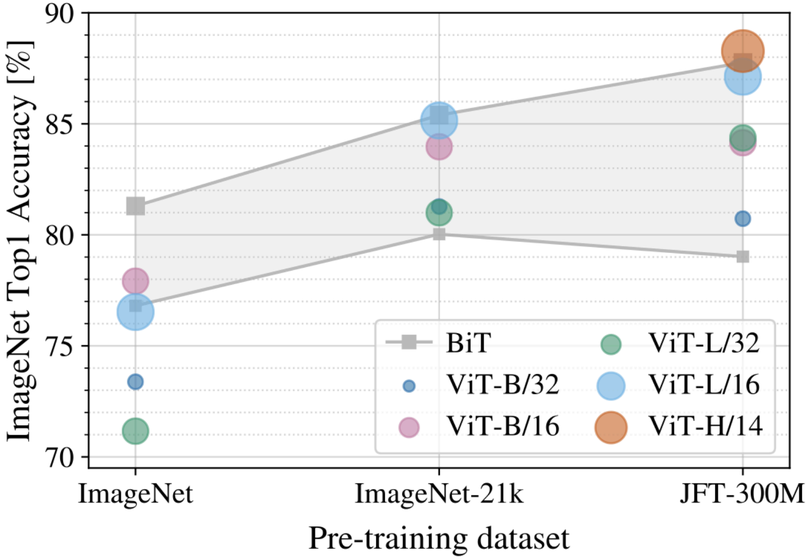




[Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale", ICLR 2021.]

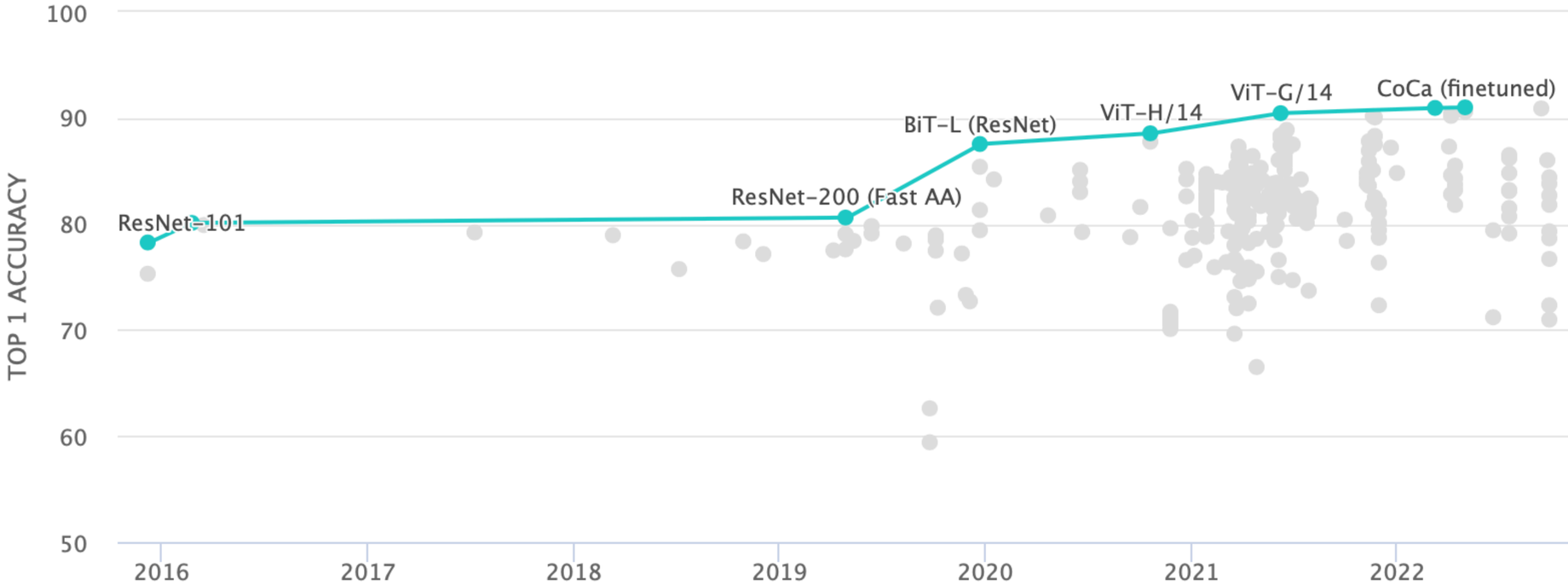
Vision Transformer (ViT)

[Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale", ICLR 2021.]



State-of-the-art models in image classification primarily rely on attention mechanisms & transformers.

Source: https://paperswithcode.com/sota/image-classification-on-imagenet/



Summary

- Transformers are state-of-the-art natural language processing models.
- They have also been applied to images with great success (e.g., ViT and extensions).
- Big advantage of Transformers: More parallelization possible than in RNNs.
- Can be trained on huge datasets.
- Simple predictive pre-training is very effective in generative modeling.
  - Larger models & scaling are claimed to reveal *emergent abilities*.
    [J. Wei et al. "Emergent Abilities of Large Language Models." TMLR 2022.]
- Their capabilities can be surprising (e.g., ChatGPT), but they are also limited.
  - Can make strange errors, have security vulnerabilities, but produces reasonable text.

Today

- ☑ Transformers
  - ☑ Attention
  - ☑ Transformer Architecture
  - ☑ Further Extensions
    - ☑ Generative Pre-Trained Transformer (GPT)
    - ☑ Vision Transformers (ViT)

Questions?