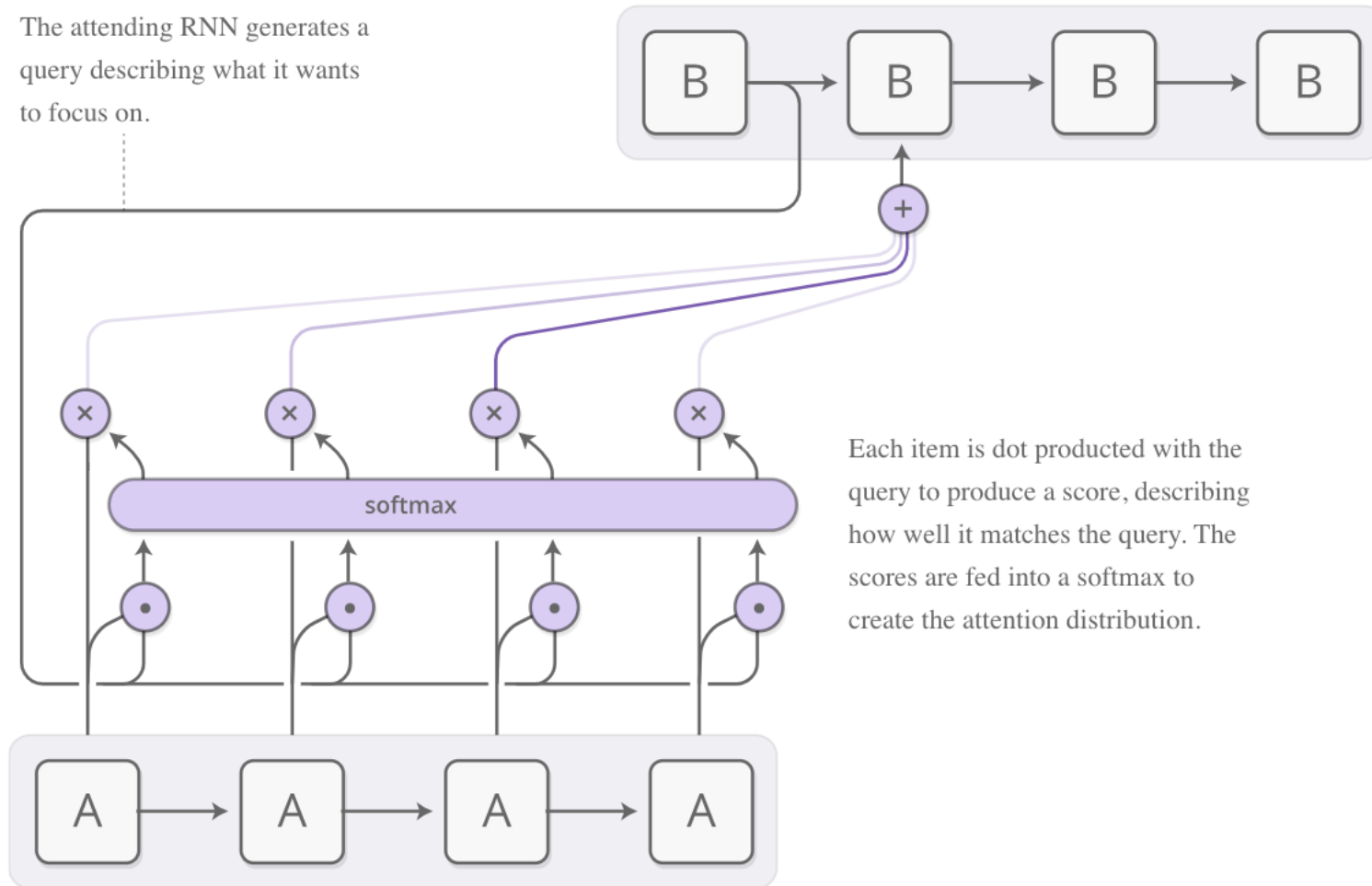
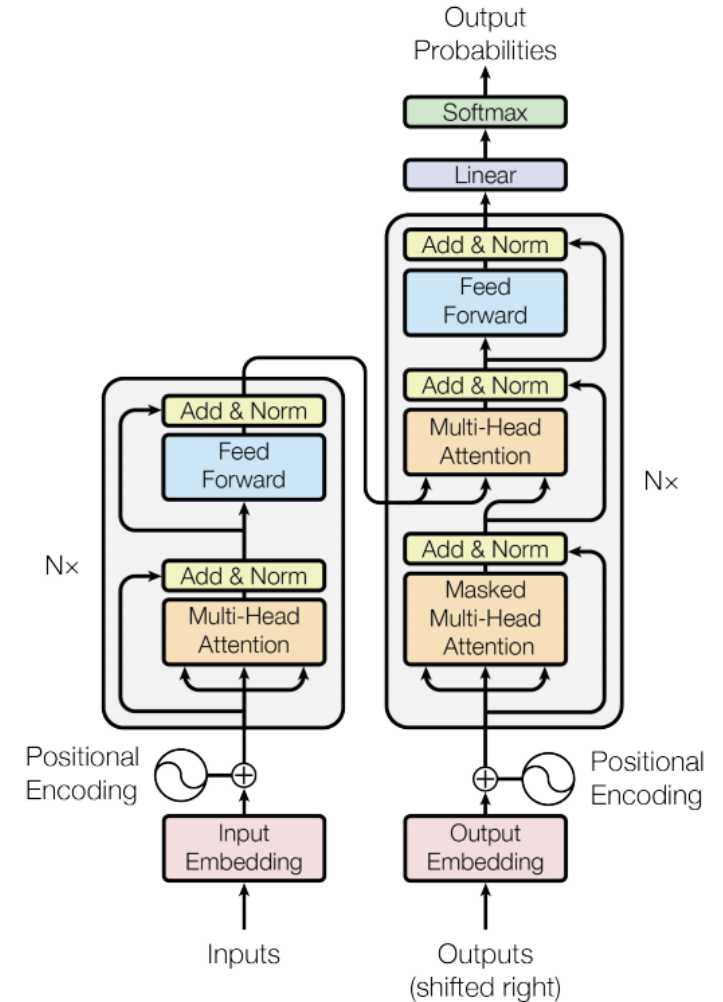


# Self-Attention and Transformer

The attending RNN generates a query describing what it wants to focus on.

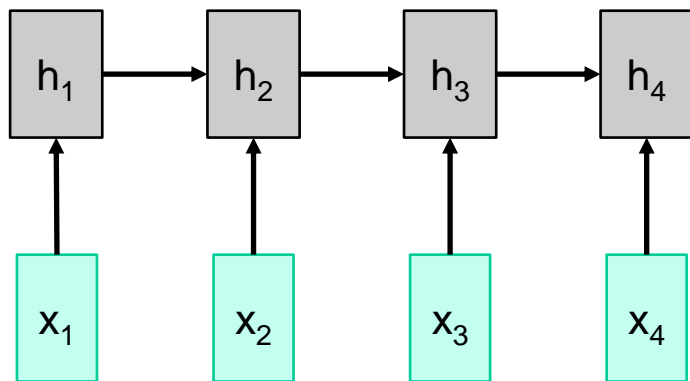


Each item is dot producted with the query to produce a score, describing how well it matches the query. The scores are fed into a softmax to create the attention distribution.



# Different ways of processing sequences

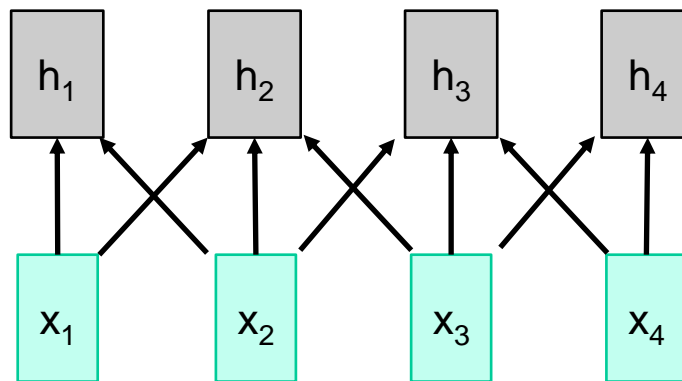
## RNN



Works on **ordered sequences**

- Pros: Good at long sequences: the last hidden vector encapsulates the whole sequence
- Cons: Not parallelizable: need to compute hidden states sequentially

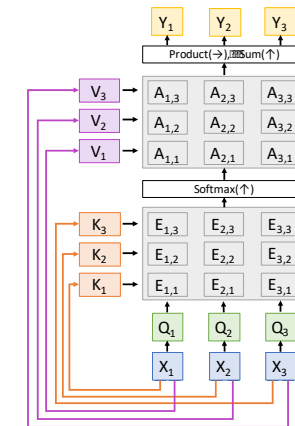
## 1D convolutional network



Works on **multidimensional grids**

- Con: Bad at long sequences: Need to stack many conv layers for outputs to “see” the whole sequence
- Pro: All outputs can be computed in parallel

## Self-Attention and Transformer



- Works on **sets of vectors**

# Outline

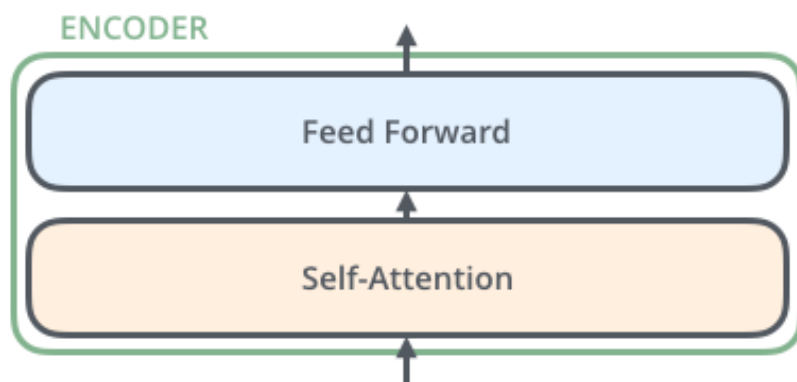
---

- Transformer architecture
  - Attention models
  - Implementation details
- Transformer-based language models
  - BERT
  - GPT and Other models
- Applications of transformers in vision

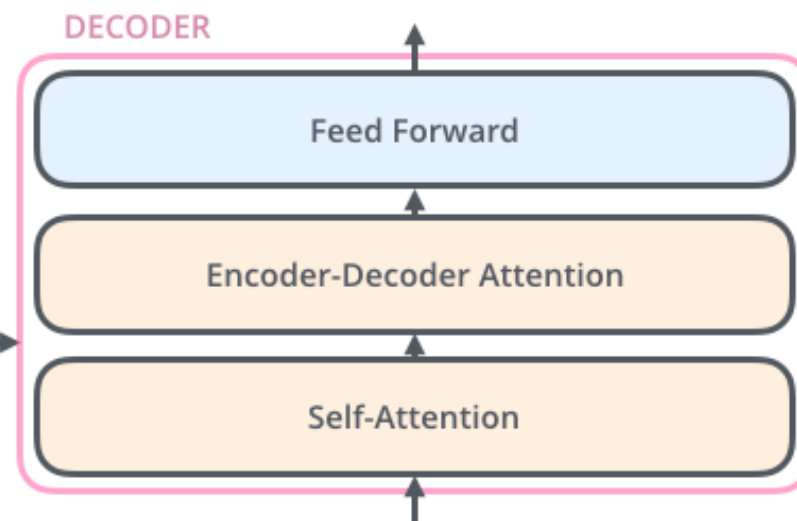
# Attention is all you need

- Neural Machine Translation (**NMT**) architecture using only point-wise processing and attention (no recurrent units or convolutions)

**Encoder:** receives entire input sequence and outputs encoded sequence of the same length



**Decoder:** predicts next token conditioned on encoder output and previously predicted tokens



A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin, [Attention is all you need](#), NeurIPS 2017

[Image source](#)

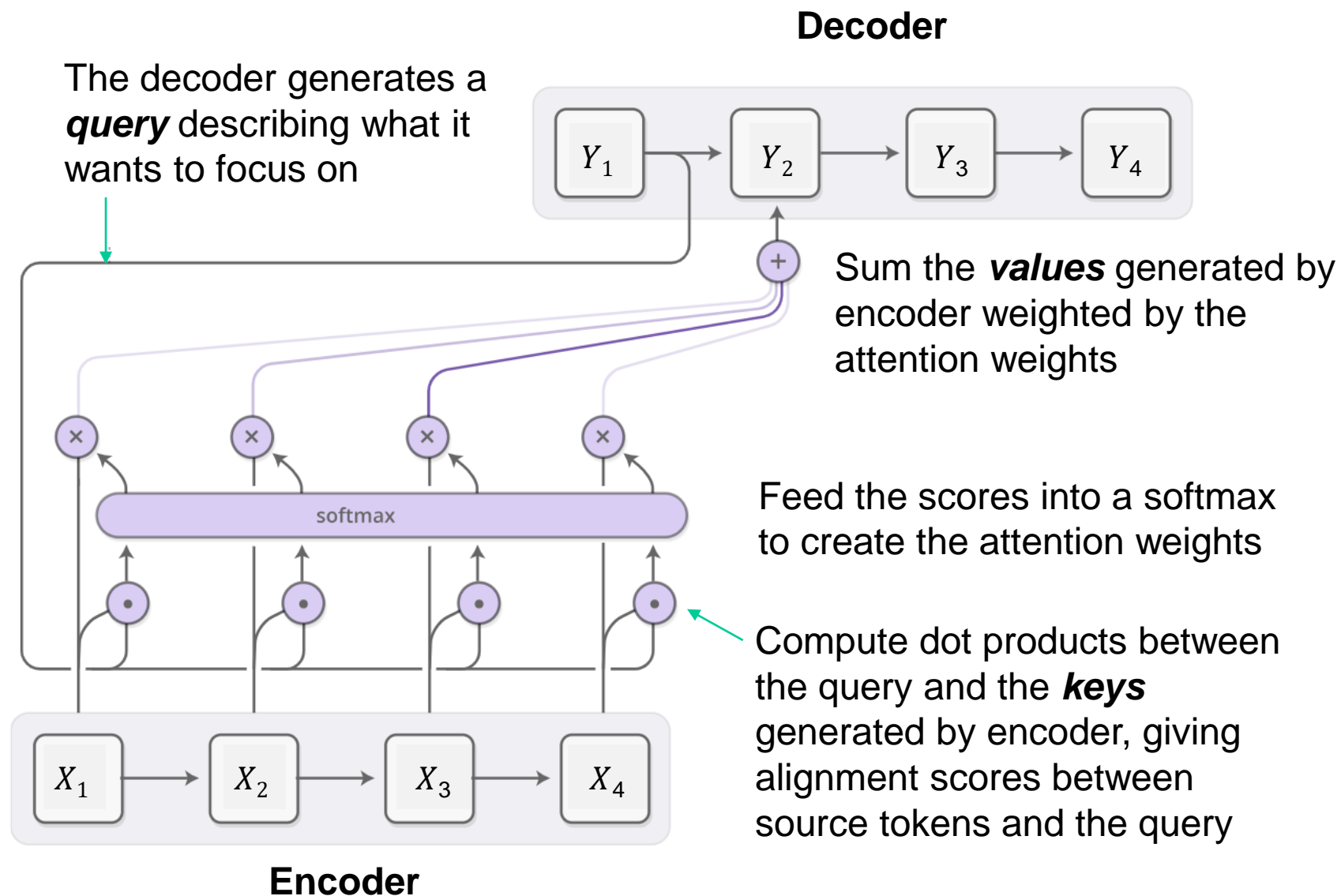
# Attention is all you need

---

- Neural Machine Translation (**NMT**) architecture using only point-wise processing and attention (no recurrent units or convolutions)
- More efficient and parallelizable than recurrent or convolutional architectures, faster to train, better accuracy

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser,  
I. Polosukhin, [Attention is all you need](#), NeurIPS 2017

# Key-Value-Query attention model



# Key-Value-Query attention model

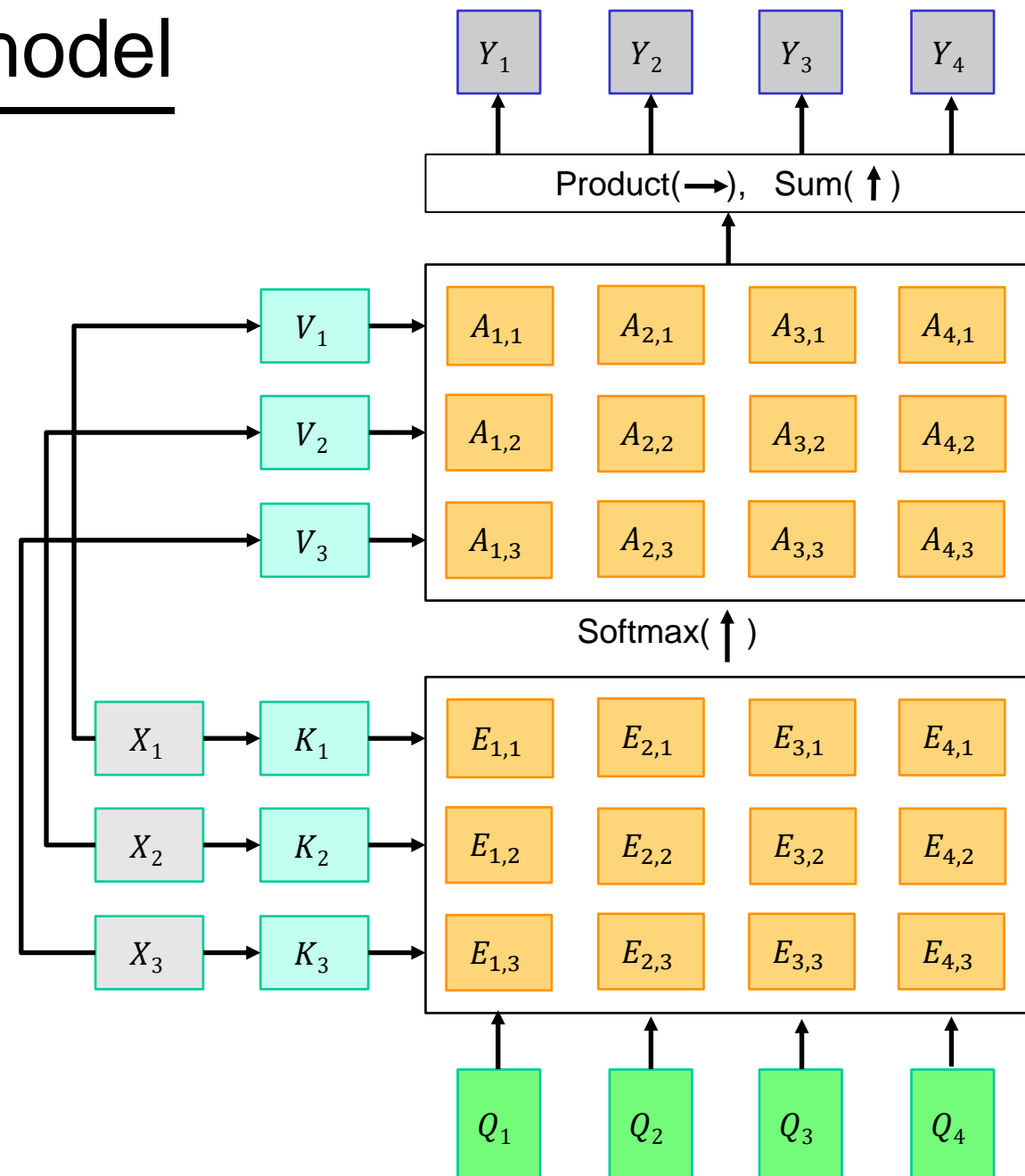
- Key vectors:  $K = XW_K$
- Value Vectors:  $V = XW_V$
- Query vectors
- Similarities: *scaled dot-product attention*

$$E_{i,j} = \frac{(Q_i \cdot K_j)}{\sqrt{D}}$$

( $D$  is the dimensionality of the keys)

- Attn. weights:  $A = \text{softmax}(E, \text{dim} = 1)$
- Output vectors:

$$Y_i = \sum_j A_{i,j} V_j \quad \text{or} \quad Y = AV$$



# Self-attention layer

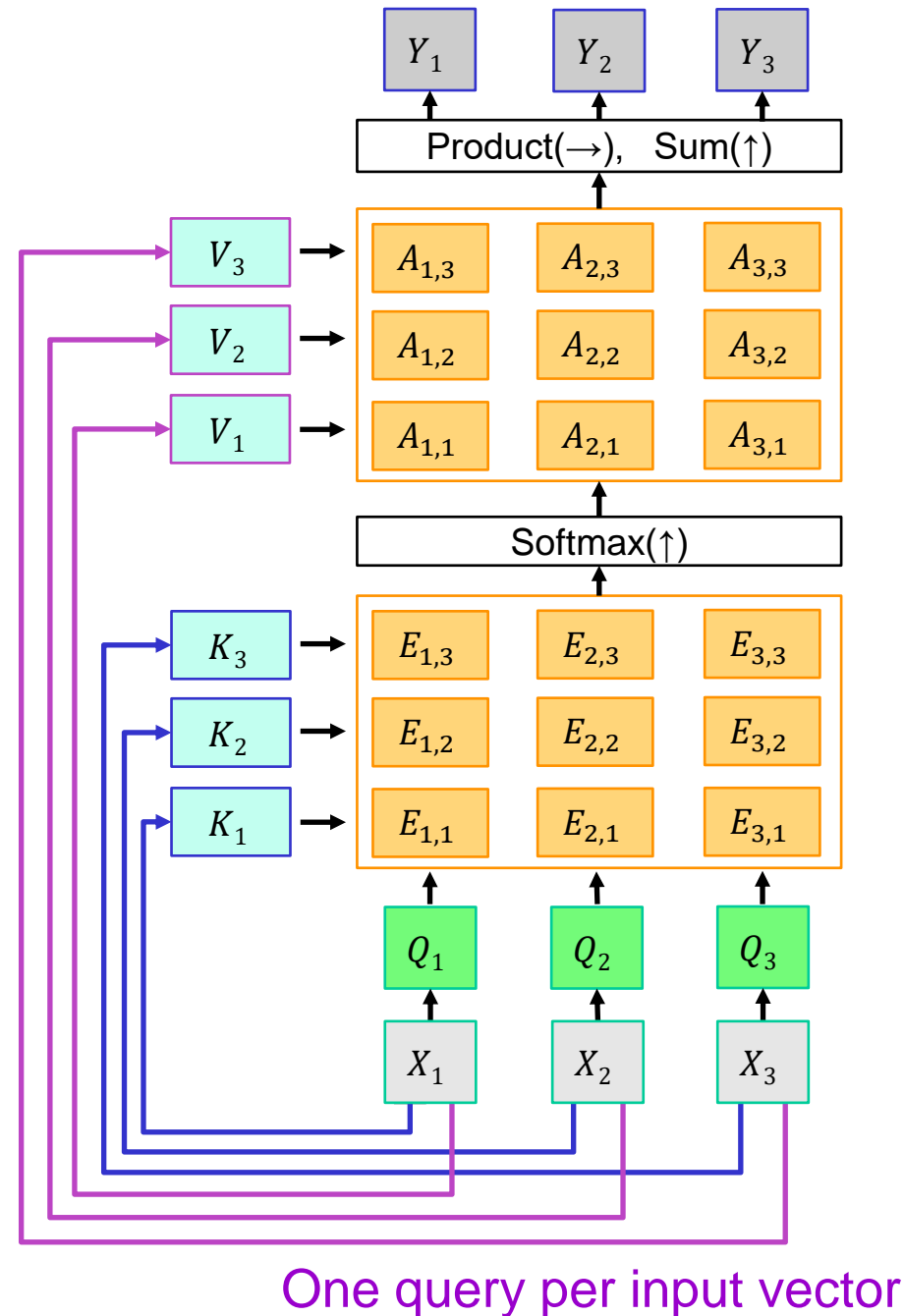
- Query vectors:  $Q = XW_Q$
- Key vectors:  $K = XW_K$
- Value vectors:  $V = XW_V$
- Similarities: *scaled dot-product attention*

$$E_{i,j} = \frac{(Q_i \cdot K_j)}{\sqrt{D}}$$

( $D$  is the dimensionality of the keys)

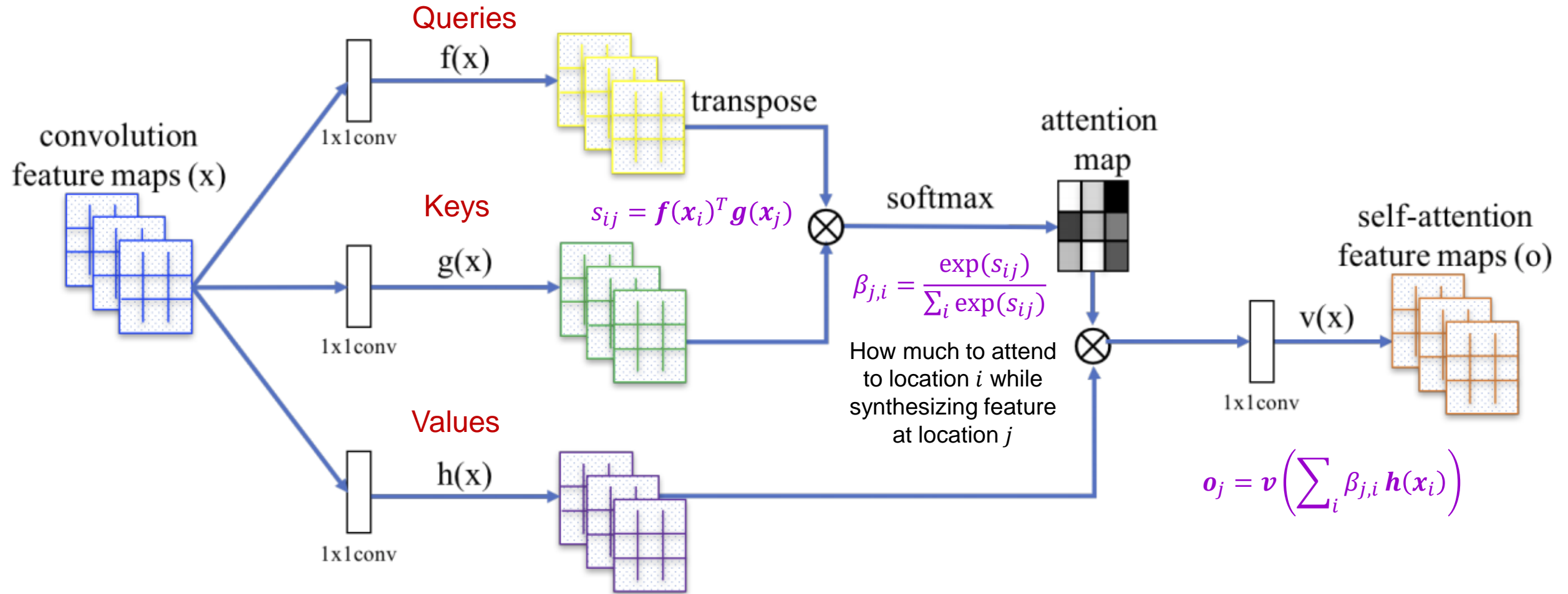
- Attn. weights:  $A = \text{softmax}(E, \text{dim} = 1)$
- Output vectors:

$$Y_i = \sum_j A_{i,j} V_j \quad \text{or} \quad Y = AV$$



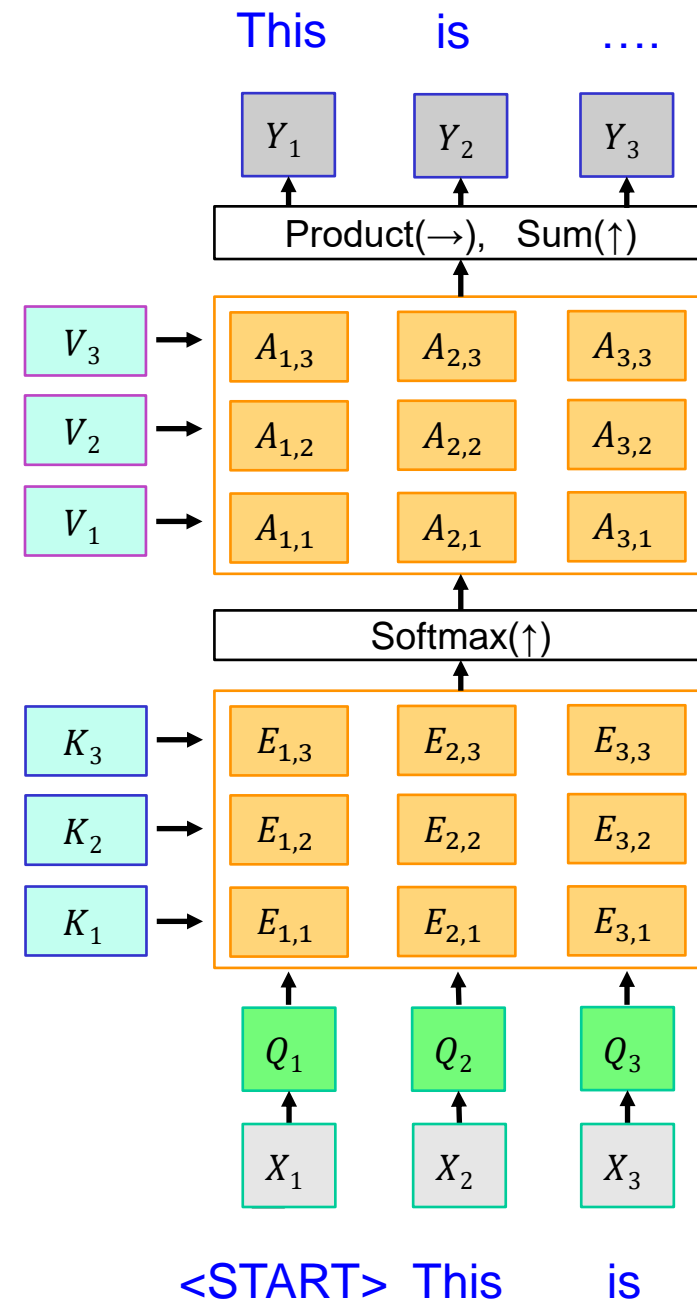


# Recall: Self-attention GAN



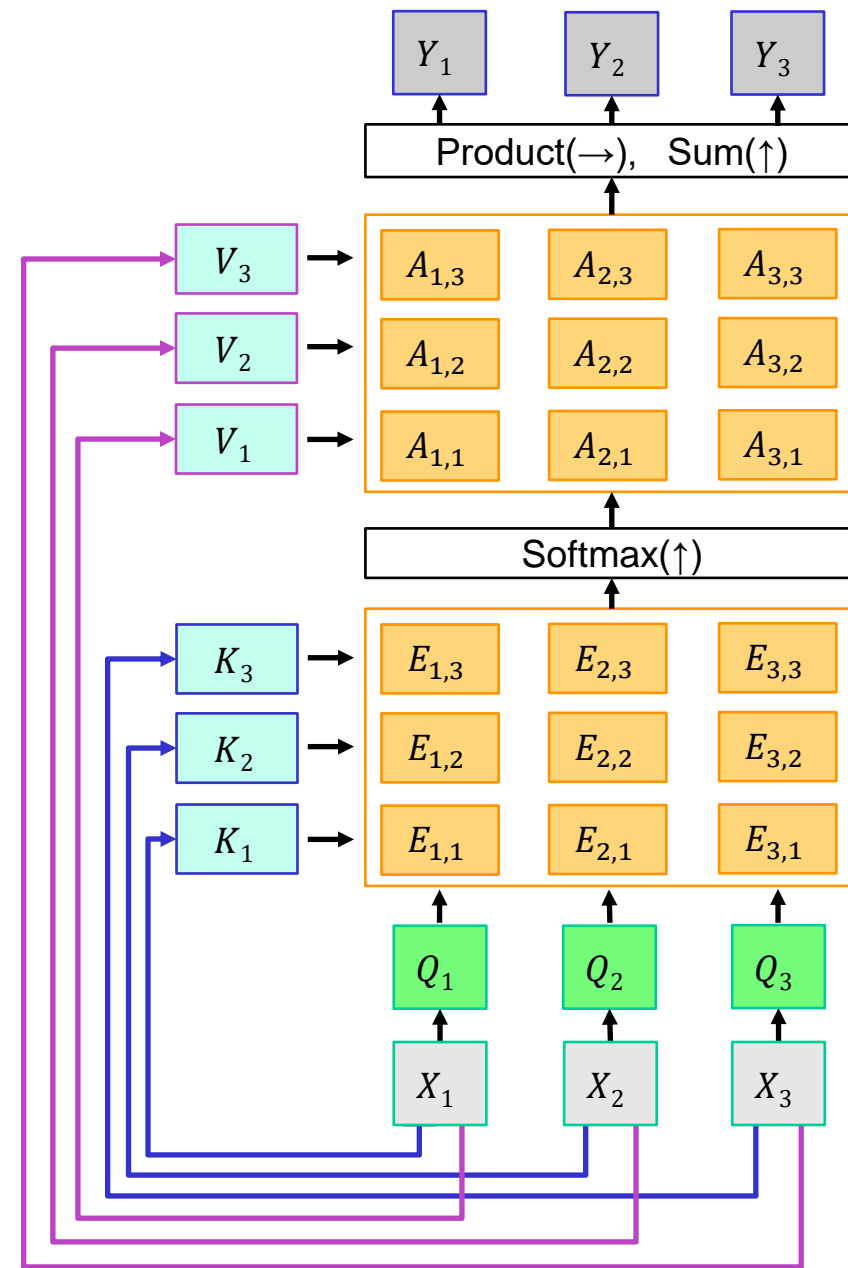
# Masked self-attention layer

- The decoder should not “look ahead” in the output sequence



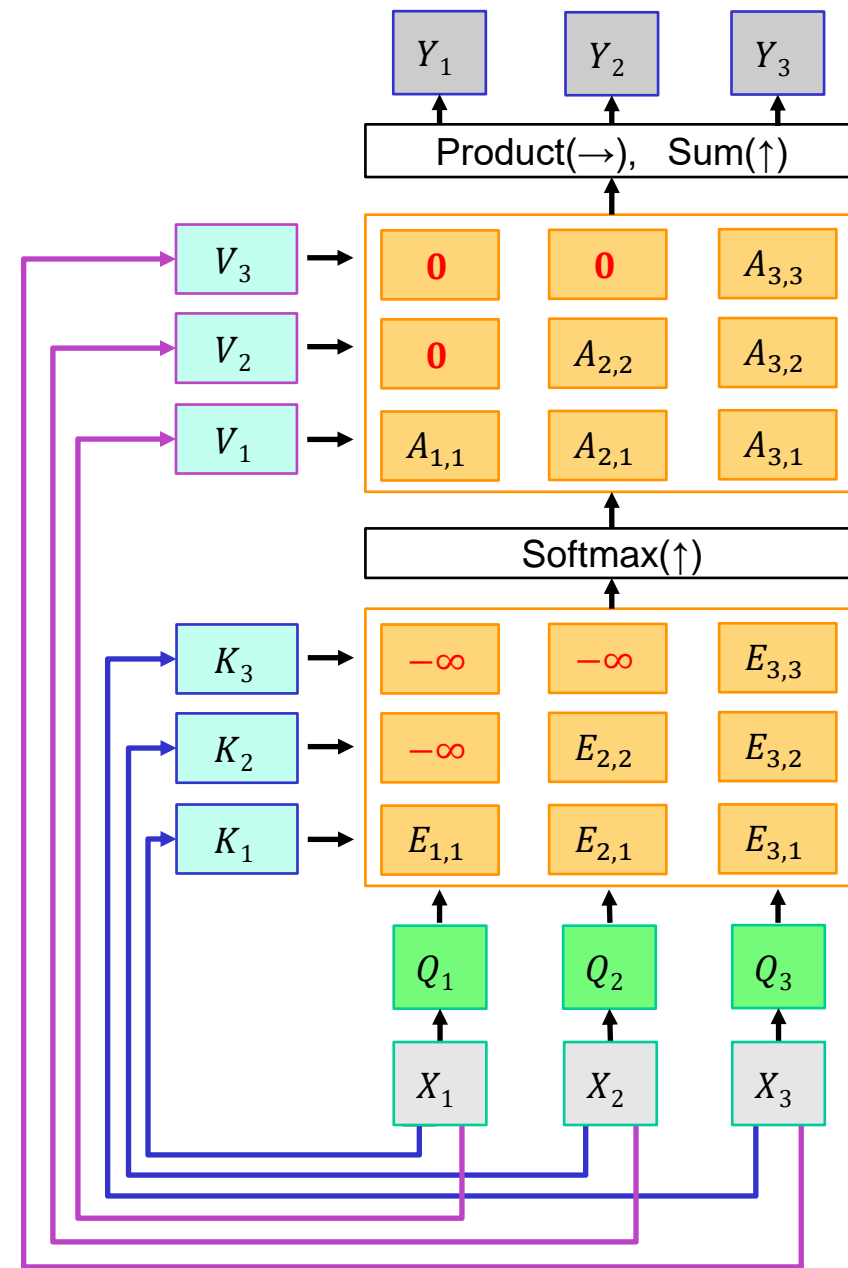
# Masked self-attention layer

- The decoder should not “look ahead” in the output sequence

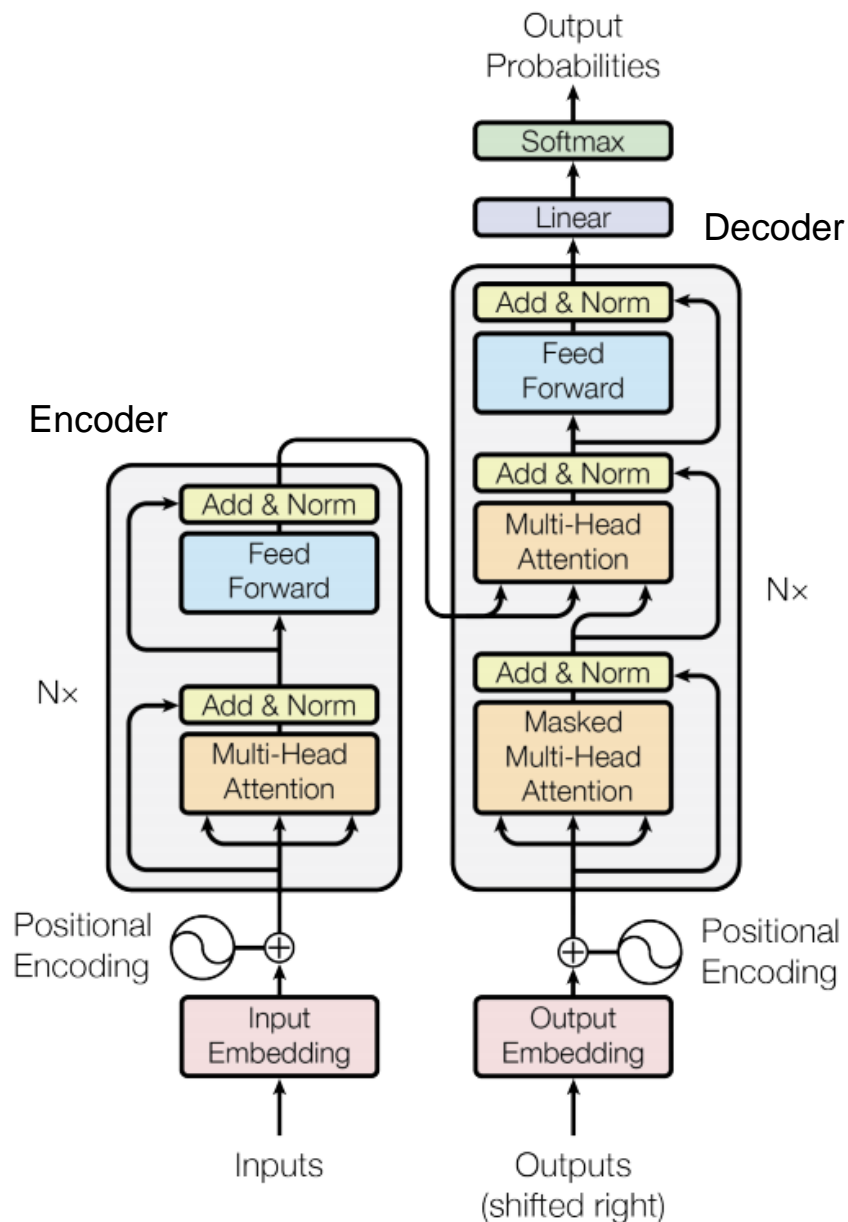


# Masked self-attention layer

- The decoder should not “look ahead” in the output sequence

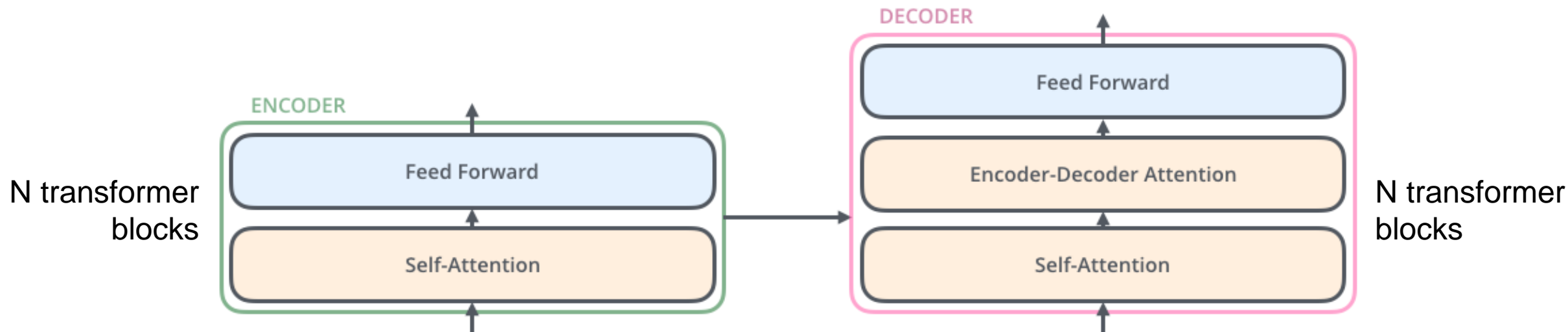


# Transformer architecture: Details



# Attention mechanisms

---

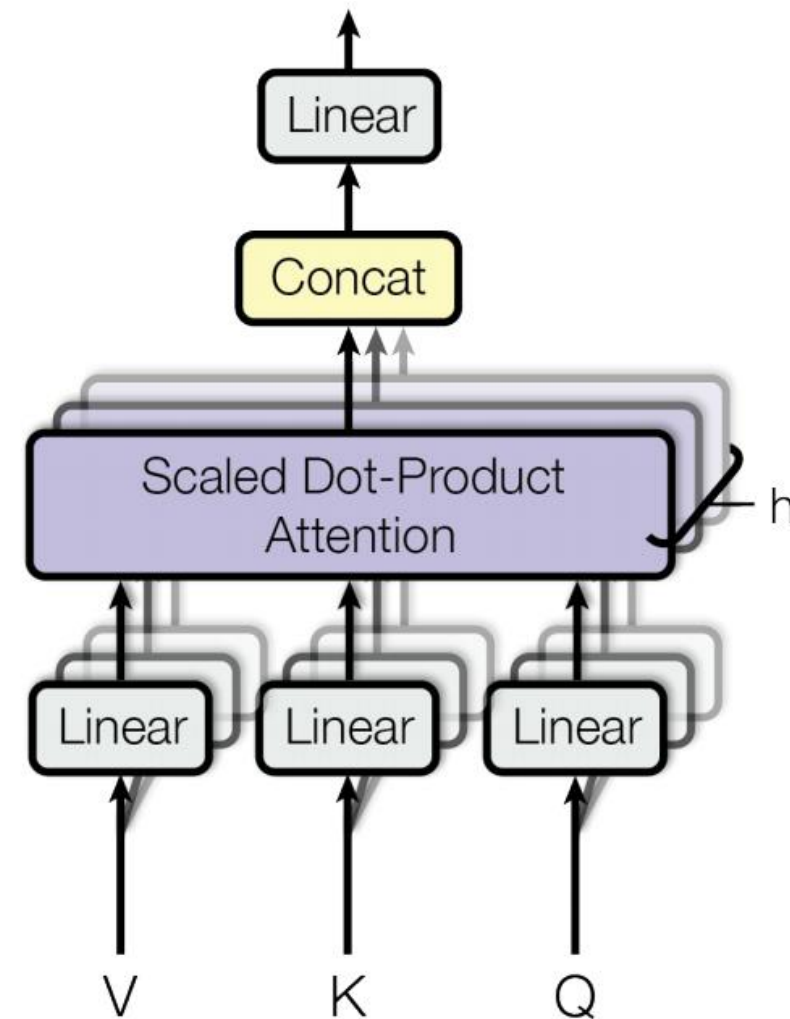


- **Encoder self-attention:** queries, keys, and values come from previous layer of encoder
- **Decoder self-attention:** values corresponding to future decoder outputs are masked out
- **Encoder-decoder attention:** queries come from previous decoder layer, keys and values come from output of encoder

# Multi-head attention

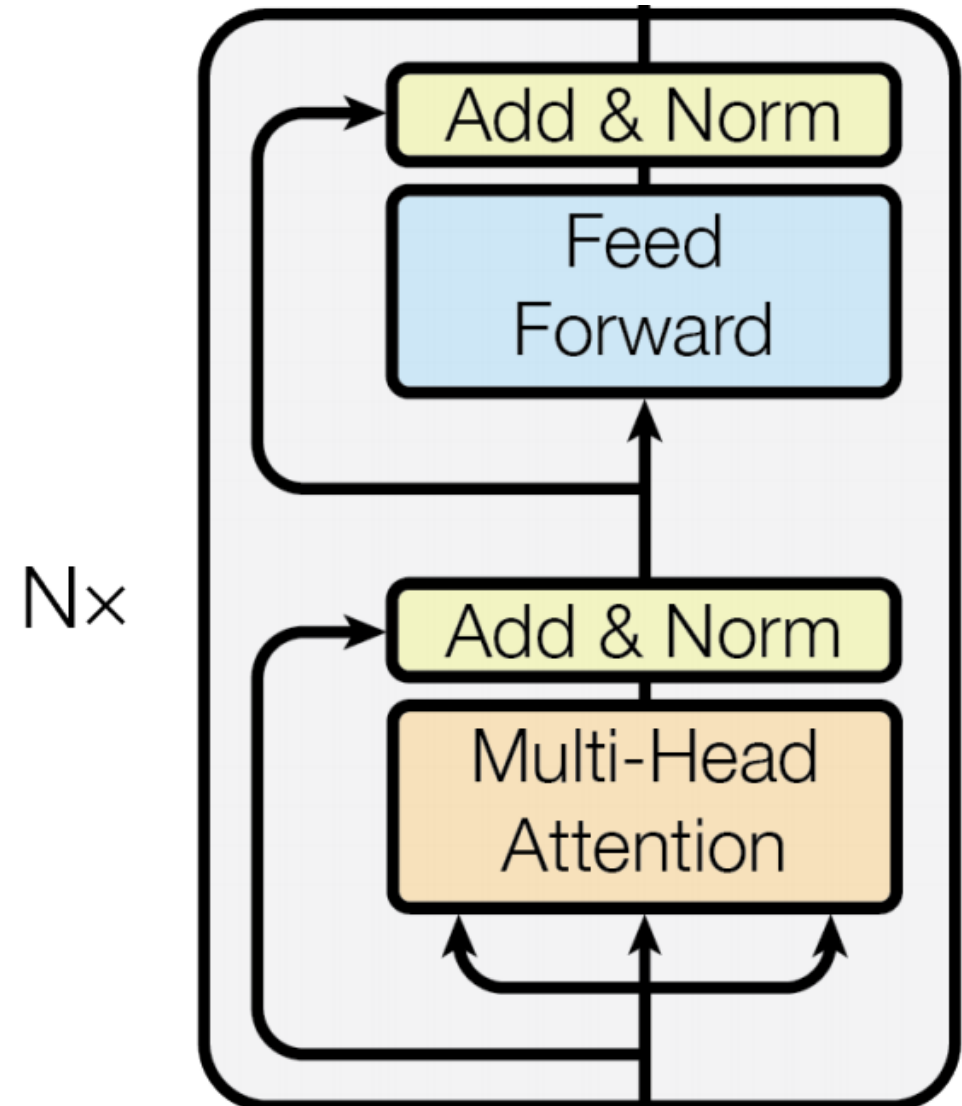
---

- Run  $h$  attention models in parallel on top of different linearly projected versions of  $Q, K, V$ ; concatenate and linearly project the results
- Intuition: enables model to attend to different kinds of information at different positions



# Transformer blocks

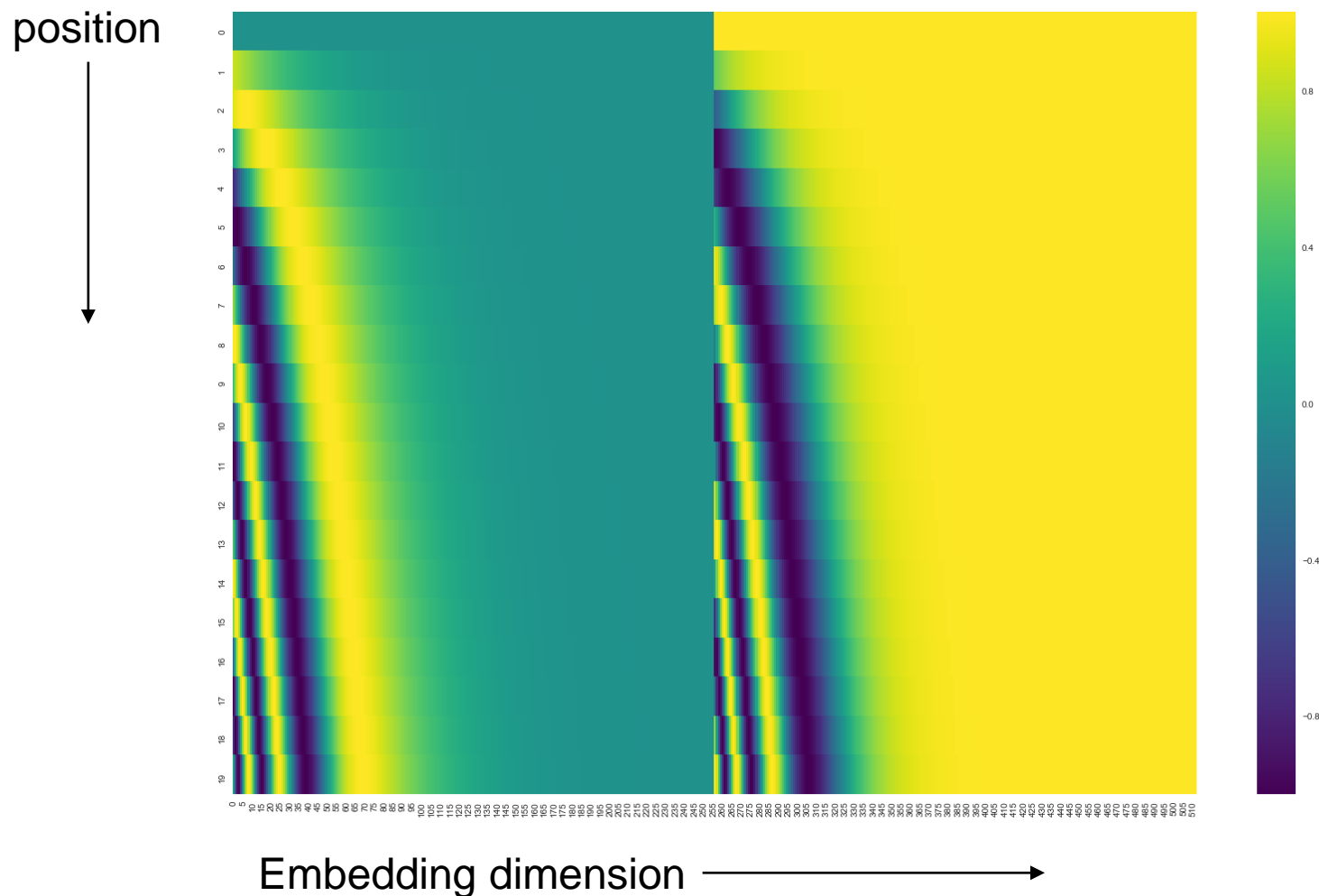
- A **Transformer** is a sequence of transformer blocks
  - Vaswani et al.: N=12 blocks, embedding dimension = 512, 6 attention heads
  - **Add & Norm:** residual connection followed by [layer normalization](#)
  - **Feedforward:** two linear layers with ReLUs in between, applied independently to each vector
- Attention is the only interaction between inputs!





# Positional encoding

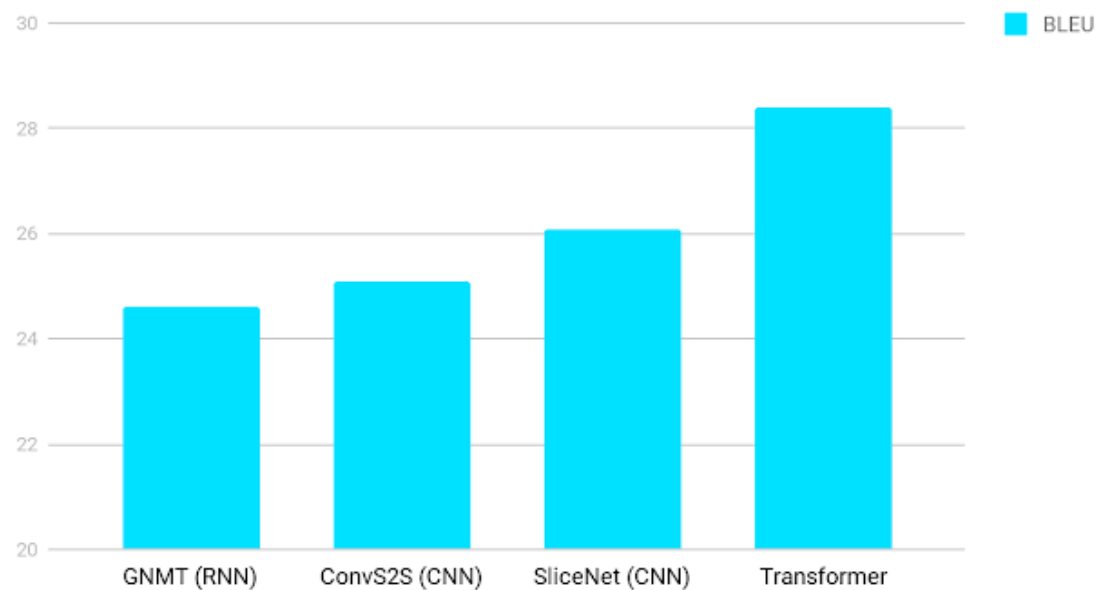
- To give transformer information about ordering of tokens, add function of position (based on sines and cosines) to every input



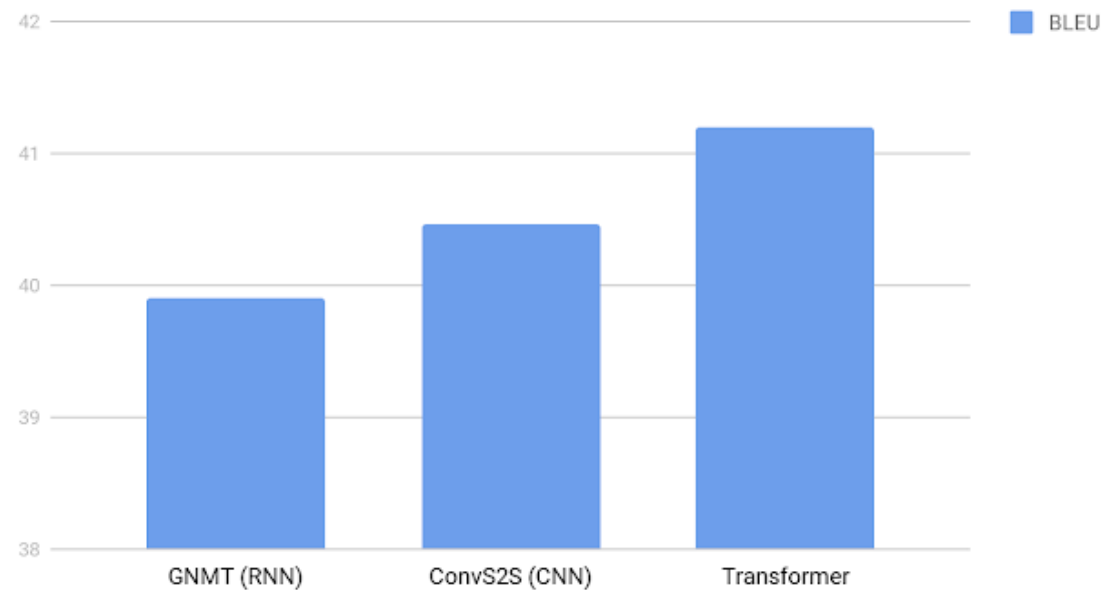
# Results

---

English German Translation quality



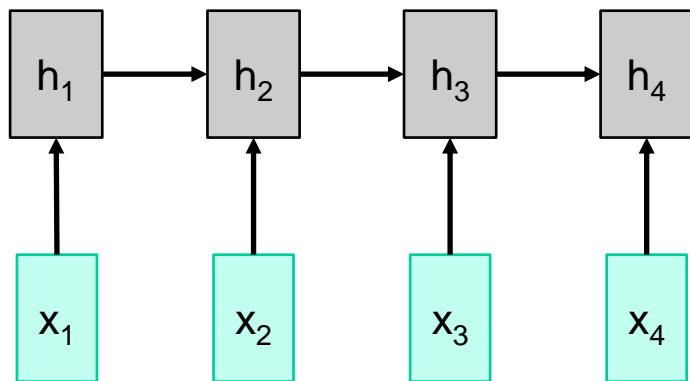
English French Translation Quality



<https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

# Different ways of processing sequences

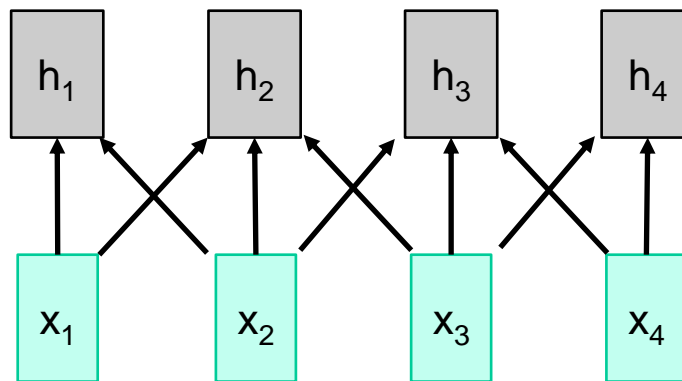
## RNN



Works on **ordered sequences**

- Pros: Good at long sequences: the last hidden vector encapsulates the whole sequence
- Cons: Not parallelizable: need to compute hidden states sequentially

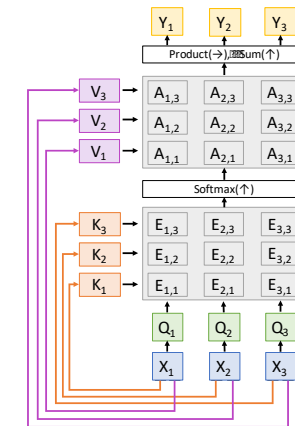
## 1D convolutional network



Works on **multidimensional grids**

- Con: Bad at long sequences: Need to stack many conv layers for outputs to “see” the whole sequence
- Pro: Highly parallel: Each output can be computed in parallel

## Self-Attention and Transformer



- Works on **sets of vectors**
- Pro: Good at long sequences: after one self-attention layer, each output “sees” all inputs!
- Pro: Highly parallel: Each output can be computed in parallel
- Con: Very memory-intensive

# Outline



---

- Transformer architecture
  - Attention models
  - Implementation details
- Transformer-based language models
  - BERT
  - GPT and Other models

# Self-supervised language modeling with transformers

---

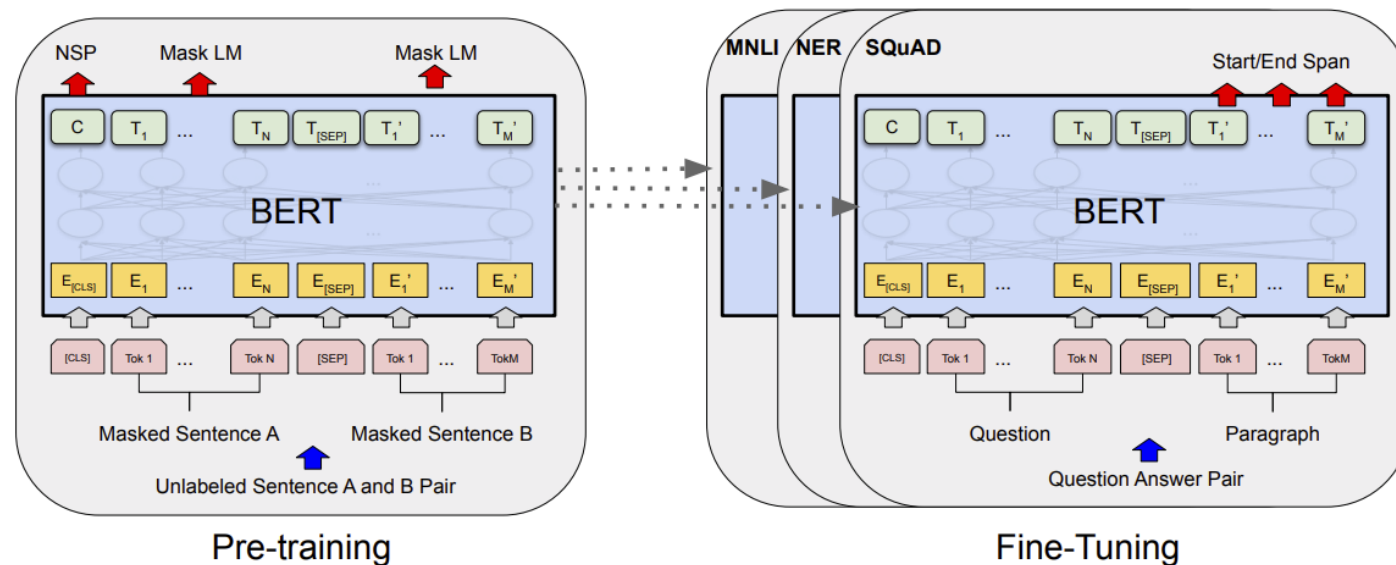
1. Download a lot of text from the internet
2. Train a transformer using a suitable pretext task
3. Fine-tune the transformer on desired NLP task

Model Alias	Org.	Article Reference
ULMfit	fast.ai	<i>Universal Language Model Fine-tuning for Text Classification</i> Howard and Ruder
 ELMo	AllenNLP	<i>Deep contextualized word representations</i> Peters et al.
OpenAI GPT	OpenAI	<i>Improving Language Understanding by Generative Pre-Training</i> Radford et al.
 BERT	Google	<i>BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding</i> Devlin et al.
XLNet	Facebook	<i>Cross-lingual Language Model Pretraining</i> Lample and Conneau

# Self-supervised language modeling with transformers

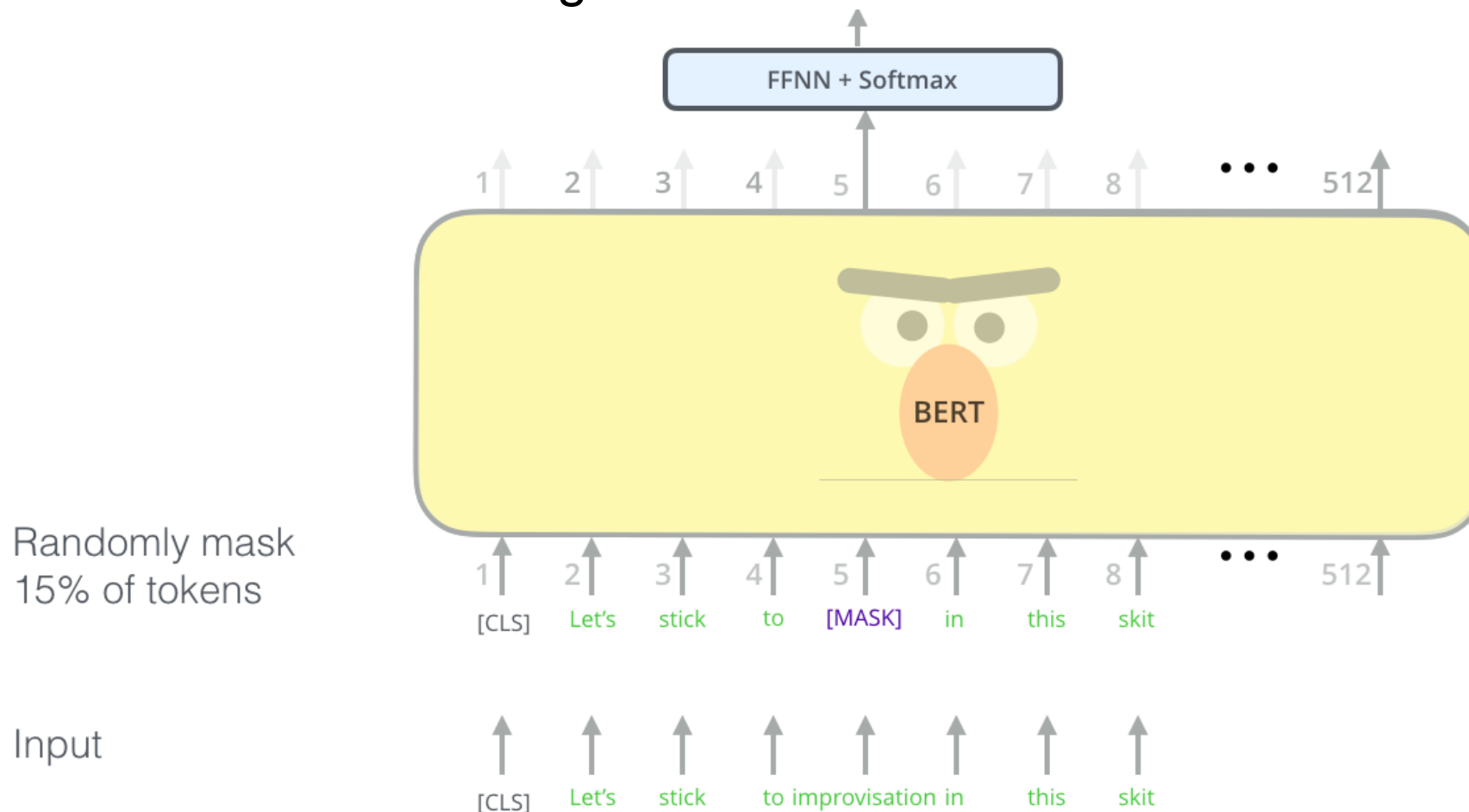
1. Download a lot of text from the internet
2. Train a transformer using a suitable pretext task
3. Fine-tune the transformer on desired NLP task

## Bidirectional Encoder Representations from Transformers (BERT)



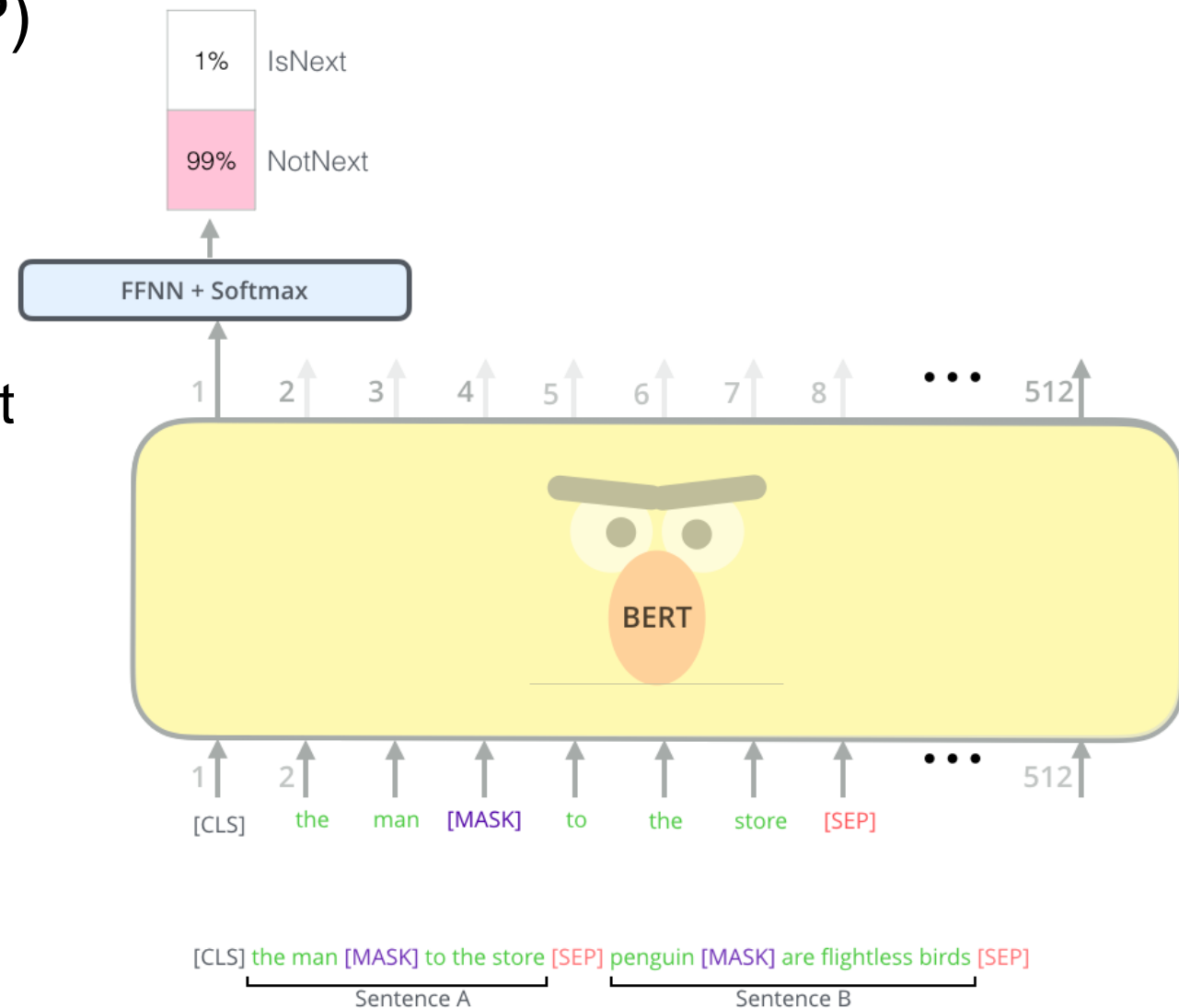
# BERT: Pretext tasks

- Masked language model (MLM)
  - Randomly mask 15% of tokens in input sentences, goal is to reconstruct them using bidirectional context



# BERT: Pretext tasks

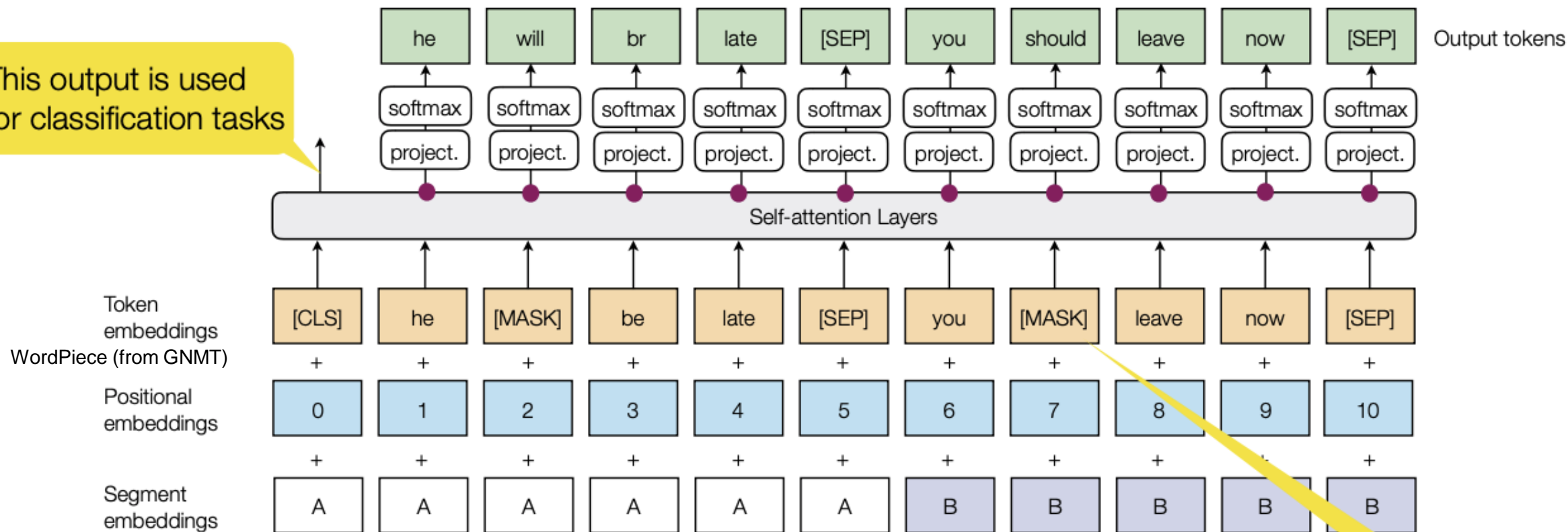
- Next sentence prediction (NSP)
  - Useful for Question Answering and Natural Language Inference tasks
  - In the training data, 50% of the time B is the actual sentence that follows A (labeled as IsNext), and 50% of the time it is a random sentence (labeled as NotNext).





# BERT: More detailed view

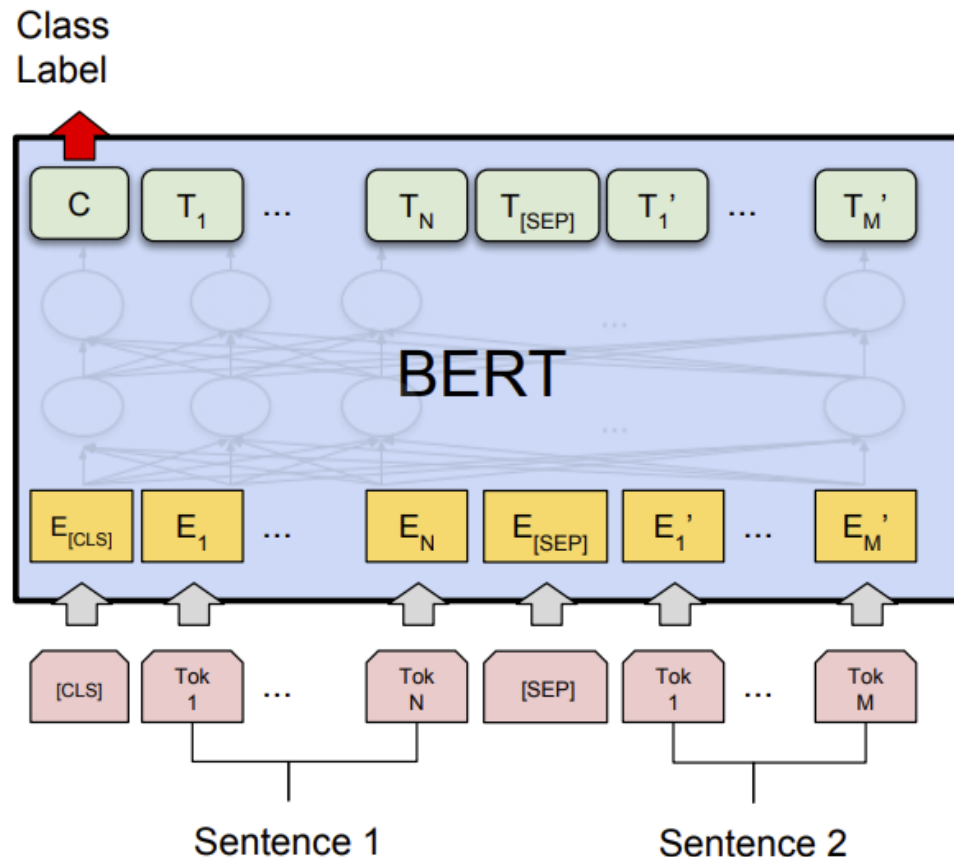
This output is used for classification tasks



15% of tokens get masked

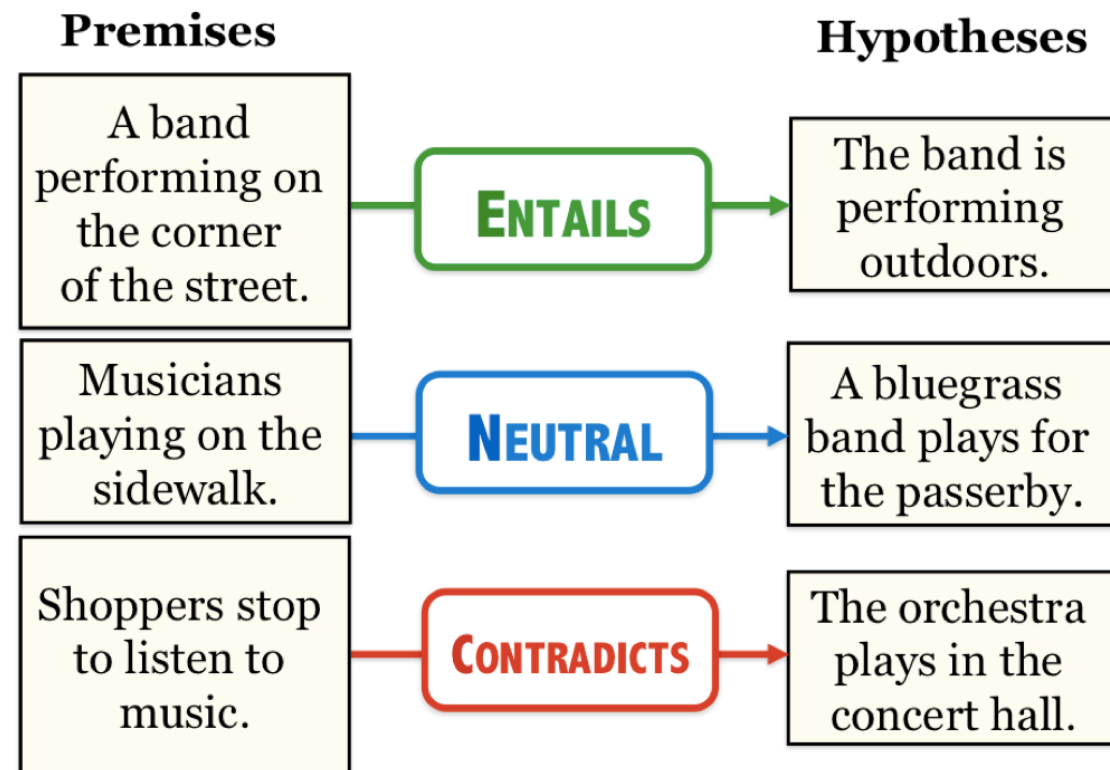
Trained on Wikipedia (2.5B words) + BookCorpus (800M words)

# BERT: Downstream tasks



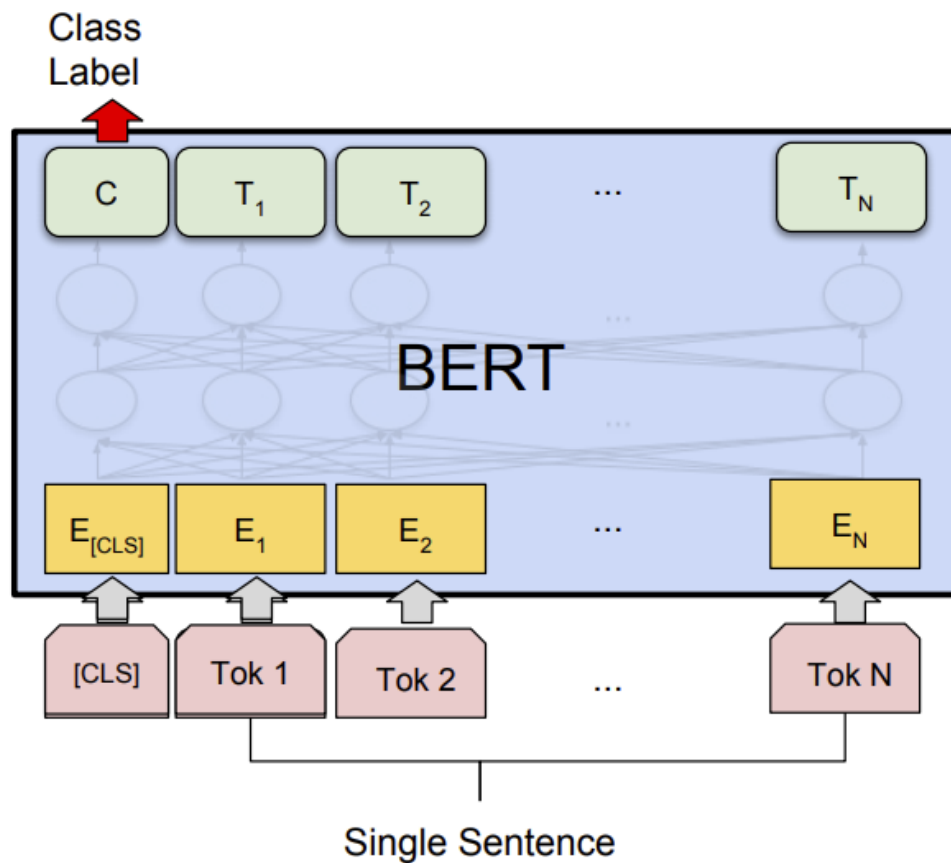
(a) Sentence Pair Classification Tasks:  
MNLI, QQP, QNLI, STS-B, MRPC,  
RTE, SWAG

## Textual entailment



Source: J. Hockenmaier

# BERT: Downstream tasks



(b) Single Sentence Classification Tasks:  
SST-2, CoLA

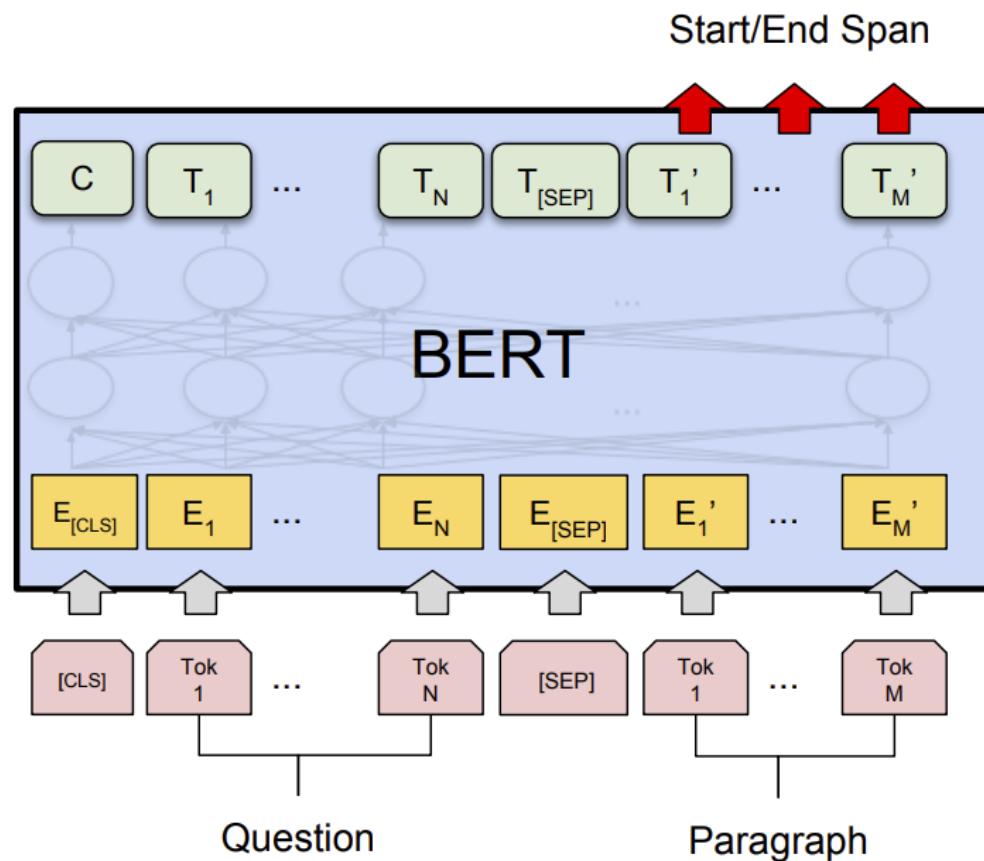
## CoLa

Sentence: The wagon rumbled down the road.  
Label: Acceptable

Sentence: The car honked down the road.  
Label: Unacceptable

Sentiment classification, linguistic acceptability

# BERT: Downstream tasks



(c) Question Answering Tasks:  
SQuAD v1.1

Find span in paragraph that contains the answer

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called “showers”.

What causes precipitation to fall?

**gravity**

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?

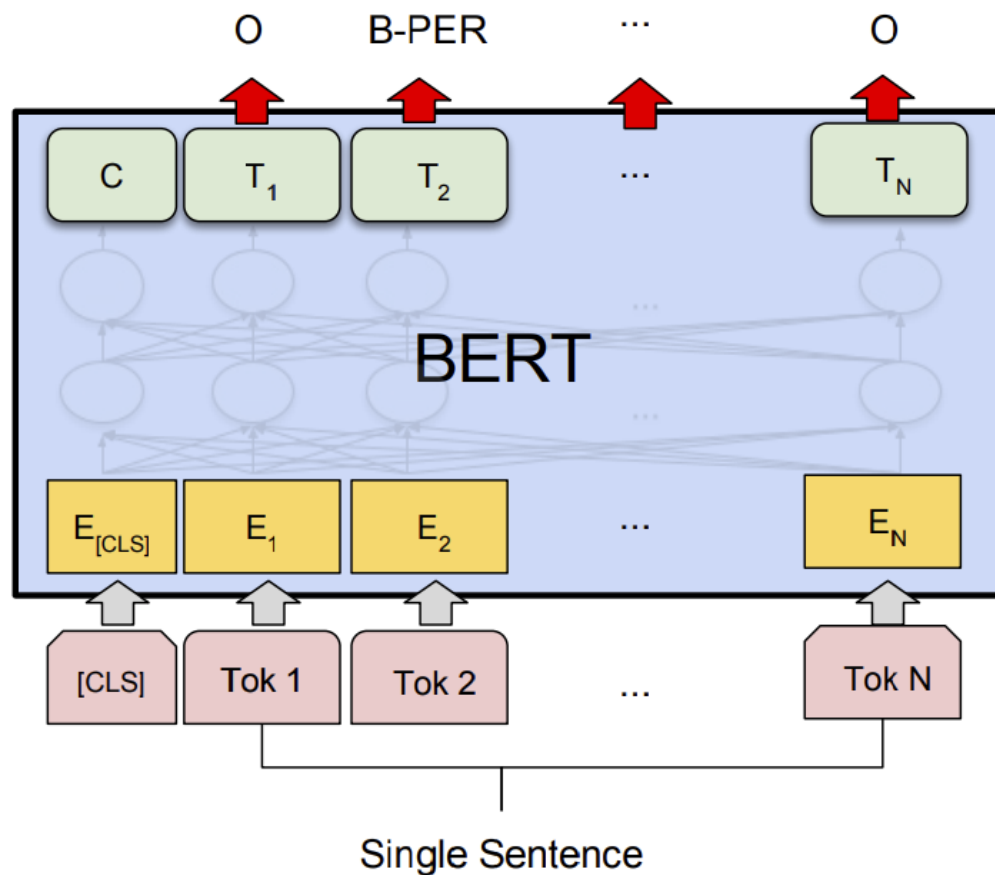
**graupel**

Where do water droplets collide with ice crystals to form precipitation?

**within a cloud**

Source: [SQuAD v1.1 paper](#)

# BERT: Downstream tasks



(d) Single Sentence Tagging Tasks:  
CoNLL-2003 NER

Named entity recognition

When Sebastian Thrun PERSON started at Google ORG in 2007 DATE , few people outside of the company took him seriously. "I can tell you very senior CEOs of major American NORP car companies would shake my hand and turn away because I wasn't worth talking to," said Thrun PERSON , now the co-founder and CEO of online higher education startup Udacity, in an interview with Recode ORG earlier this week DATE .

A little less than a decade later DATE , dozens of self-driving startups have cropped up while automakers around the world clamor, wallet in hand, to secure their place in the fast-moving world of fully automated transportation.



[Image source](#)

# Outline

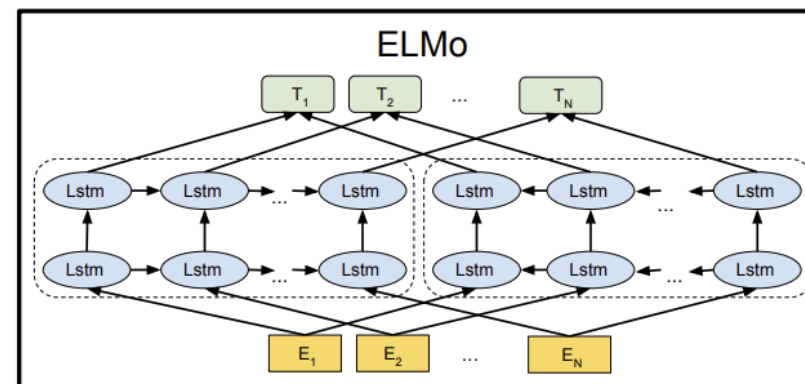
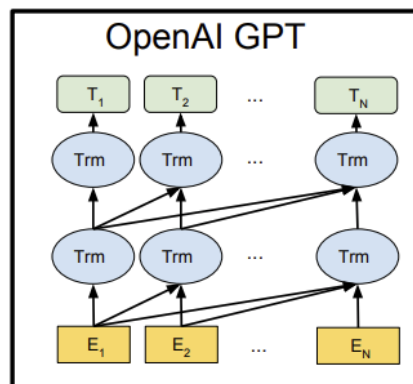
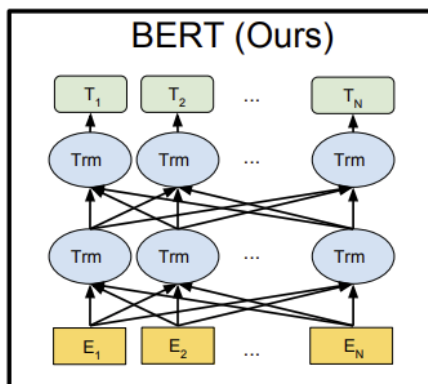
---

- Transformer architecture
  - Attention models
  - Implementation details
- Transformer-based language models
  - BERT
  - GPT and Other models

# Other language models

Model Alias	Org.	Article Reference
ULMfit	fast.ai	<i>Universal Language Model Fine-tuning for Text Classification</i> Howard and Ruder
 ELMo	AllenNLP	<i>Deep contextualized word representations</i> Peters et al.
OpenAI GPT	OpenAI	<i>Improving Language Understanding by Generative Pre-Training</i> Radford et al.
 BERT	Google	<i>BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding</i> Devlin et al.
XLNet	Facebook	<i>Cross-lingual Language Model Pretraining</i> Lample and Conneau

[Image source](#)



# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)



# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)

# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)
XLNet-Large	24	1024	16	~340M	126 GB	512x TPU-v3 (2.5 days)
RoBERTa	24	1024	16	355M	160 GB	1024x V100 GPU (1 day)

Yang et al., XLNet: Generalized Autoregressive Pretraining for Language Understanding, 2019  
Liu et al., RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019

# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)
XLNet-Large	24	1024	16	~340M	126 GB	512x TPU-v3 (2.5 days)
RoBERTa	24	1024	16	355M	160 GB	1024x V100 GPU (1 day)
GPT-2	48	1600	?	1.5B	40 GB	

# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)
XLNet-Large	24	1024	16	~340M	126 GB	512x TPU-v3 (2.5 days)
RoBERTa	24	1024	16	355M	160 GB	1024x V100 GPU (1 day)
GPT-2	48	1600	?	1.5B	40 GB	
Megatron-LM	72	3072	32	8.3B	174 GB	512x V100 GPU (9 days)

~\$430,000 on Amazon AWS!

# Scaling up transformers

---

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)
XLNet-Large	24	1024	16	~340M	126 GB	512x TPU-v3 (2.5 days)
RoBERTa	24	1024	16	355M	160 GB	1024x V100 GPU (1 day)
GPT-2	48	1600	?	1.5B	40 GB	
Megatron-LM	72	3072	32	8.3B	174 GB	512x V100 GPU (9 days)
Turing-NLG	78	4256	28	17B	?	256x V100 GPU

Microsoft, Turing-NLG: A 17-billion parameter language model by Microsoft, 2020

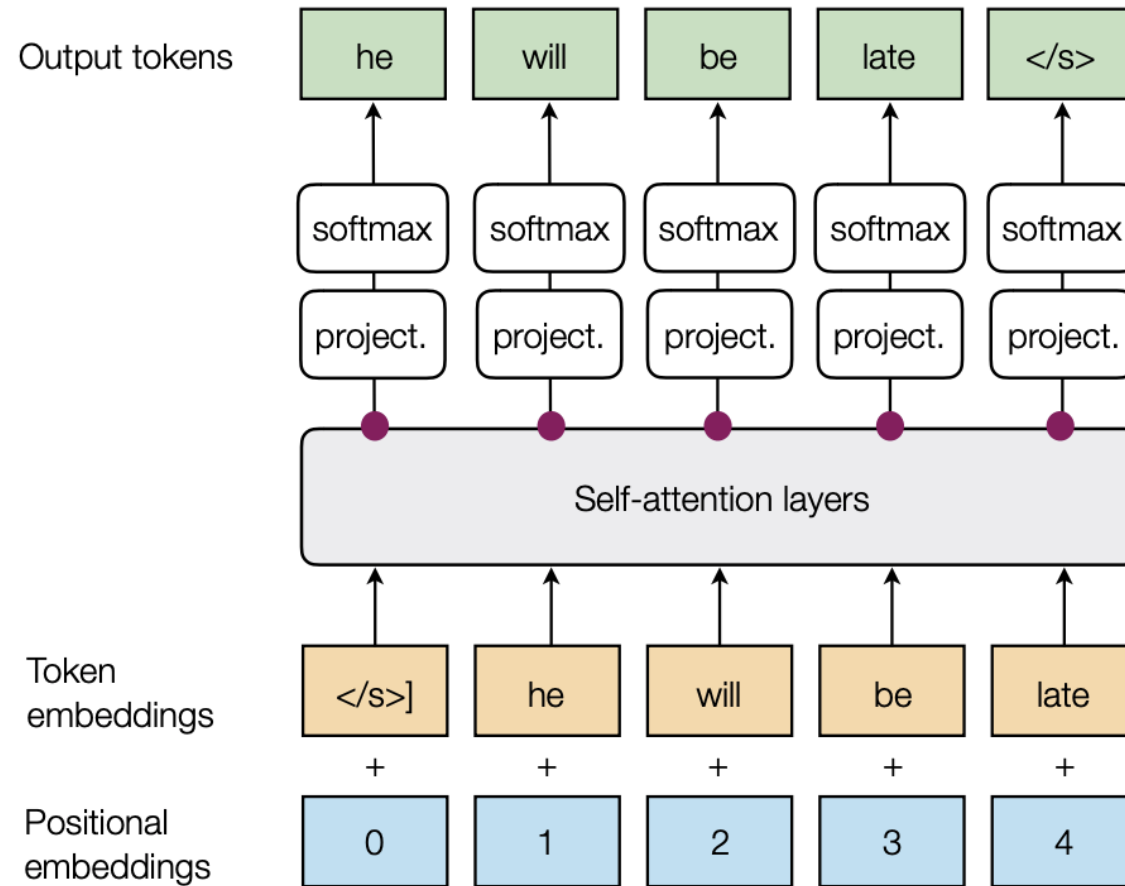
# Scaling up transformers

Model	Layers	Hidden dim.	Heads	Params	Data	Training
Transformer-Base	12	512	8	65M		8x P100 (12 hours)
Transformer-Large	12	1024	16	213M		8x P100 (3.5 days)
BERT-Base	12	768	12	110M	13 GB	4x TPU (4 days)
BERT-Large	24	1024	16	340M	13 GB	16x TPU (4 days)
XLNet-Large	24	1024	16	~340M	126 GB	512x TPU-v3 (2.5 days)
RoBERTa	24	1024	16	355M	160 GB	1024x V100 GPU (1 day)
GPT-2	48	1600	?	1.5B	40 GB	
Megatron-LM	72	3072	32	8.3B	174 GB	512x V100 GPU (9 days)
Turing-NLG	78	4256	28	17B	?	256x V100 GPU
GPT-3	96	12288	96	175B	694GB	?

~\$4.6M, 355 GPU-years  
([source](#))

# OpenAI GPT (Generative Pre-Training)

- Pre-training task: next token prediction



# GPT-2 and GPT-3

---

- Key idea: if the model and training datasets are big enough, model can adapt to new tasks without fine-tuning

GPT-2: A. Radford et al., [Language models are unsupervised multitask learners](#), 2019

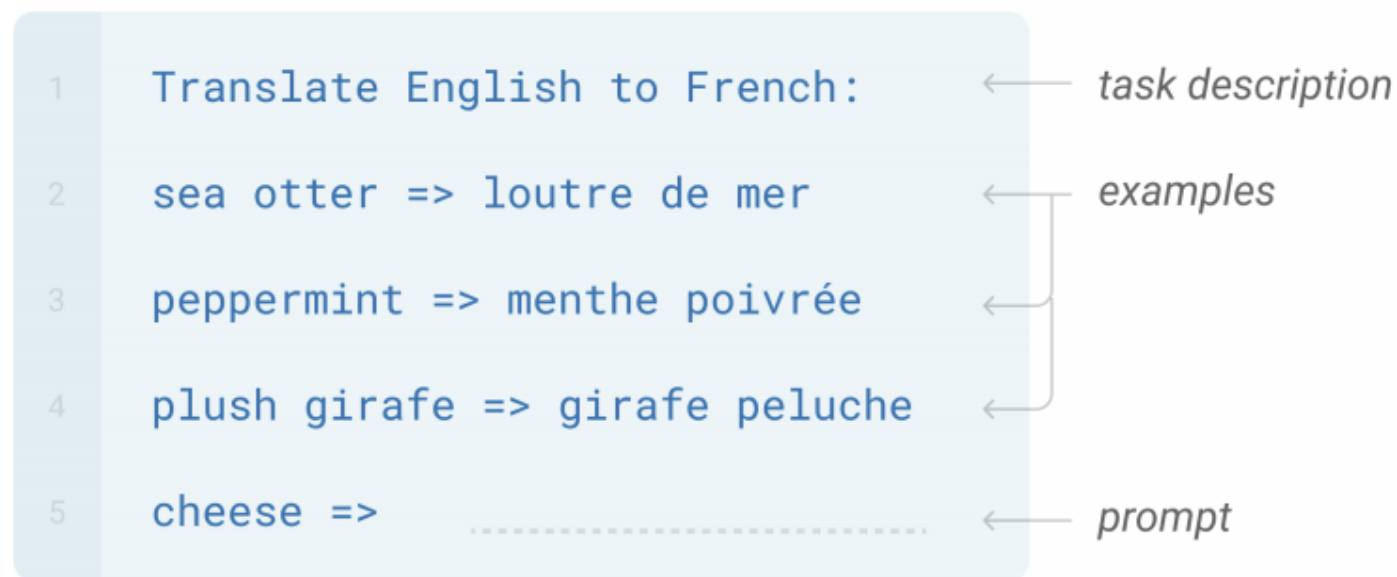
GPT-3: T. Brown et al., [Language models are few-shot learners](#), arXiv 2020



# GPT-3

---

- Key idea: if the model and training datasets are big enough, model can adapt to new tasks *without fine-tuning*
- **Few-shot learning:** In addition to the task description, the model sees a few examples of the task



# GPT-3

---

- Key idea: if the model and training datasets are big enough, model can adapt to new tasks *without fine-tuning*
- **One-shot learning:** In addition to the task description, the model sees a *single example* of the task

```
1  Translate English to French:
2  sea otter => loutre de mer
3  cheese =>
   .....
```

← task description

← example

← prompt

# GPT-3

---

- Key idea: if the model and training datasets are big enough, model can adapt to new tasks *without fine-tuning*
- **Zero-shot learning:** The model sees the task description and *no* training examples



# Task: Generate news article

---

Title: United Methodists Agree to Historic Split

Subtitle: Those who oppose gay marriage will form their own denomination

Article: After two days of intense debate, the United Methodist Church has agreed to a historic split - one that is expected to end in the creation of a new denomination, one that will be "theologically and socially conservative," according to The Washington Post. The majority of delegates attending the church's annual General Conference in May voted to strengthen a ban on the ordination of LGBTQ clergy and to write new rules that will "discipline" clergy who officiate at same-sex weddings. But those who opposed these measures have a new plan: They say they will form a separate denomination by 2020, calling their church the Christian Methodist denomination.

The Post notes that the denomination, which claims 12.5 million members, was in the early 20th century the "largest Protestant denomination in the U.S.," but that it has been shrinking in recent decades. The new split will be the second in the church's history. The first occurred in 1968, when roughly 10 percent of the denomination left to form the Evangelical United Brethren Church. The Post notes that the proposed split "comes at a critical time for the church, which has been losing members for years," which has been "pushed toward the brink of a schism over the role of LGBTQ people in the church." Gay marriage is not the only issue that has divided the church. In 2016, the denomination was split over ordination of transgender clergy, with the North Pacific regional conference voting to ban them from serving as clergy, and the South Pacific regional conference voting to allow them.

Gray: human prompts,  
boldface: GPT-3  
completions

(Three articles  
provided as training  
examples)

**Figure 3.14:** The GPT-3 generated news article that humans had the greatest difficulty distinguishing from a human written article (accuracy: 12%).

# Task: Use new word in sentence

---

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:

We were traveling in Africa and we saw these very cute whatpus.

---

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:

**One day when I was playing tag with my little sister, she got really excited and she started doing these crazy farduddles.**

A "yalubalu" is a type of vegetable that looks like a big pumpkin. An example of a sentence that uses the word yalubalu is:

**I was on a trip to Africa and I tried this yalubalu vegetable that was grown in a garden there. It was delicious.**

---

A "Burringo" is a car with very fast acceleration. An example of a sentence that uses the word Burringo is:

**In our garage we have a Burringo that my father drives to work every day.**

---

A "Gigamuru" is a type of Japanese musical instrument. An example of a sentence that uses the word Gigamuru is:

**I have a Gigamuru that my uncle gave me as a gift. I love to play it at home.**

---

To "screeg" something is to swing a sword at it. An example of a sentence that uses the word screeg is:

**We screeghed at each other for several minutes and then we went outside and ate ice cream.**

Gray: human prompts,  
boldface: GPT-3  
completions

# Task: Correct grammar

---

Poor English input: I eated the purple berries.

Good English output: I ate the purple berries.

Poor English input: Thank you for picking me as your designer. I'd appreciate it.

Good English output: Thank you for choosing me as your designer. I appreciate it.

Poor English input: The mentioned changes have done. or I did the alteration that you requested. or I changed things you wanted and did the modifications.

Good English output: The requested changes have been made. or I made the alteration that you requested. or I changed things you wanted and made the modifications.

Poor English input: I'd be more than happy to work with you in another project.

Good English output: I'd be more than happy to work with you on another project.

---

Poor English input: Please provide me with a short brief of the design you're looking for and that'd be nice if you could share some examples or project you did before.

Good English output: Please provide me with a brief description of the design you're looking for and that would be nice if you could share some examples or projects you have done before.

---

Poor English input: The patient was died.

Good English output: The patient died.

---

Poor English input: We think that Leslie likes ourselves.

Good English output: We think that Leslie likes us.

---

Poor English input: Janet broke Bill on the finger.

Good English output: Janet broke Bill's finger.

---

Poor English input: Mary arranged for, in St. Louis, John to rent a house cheap.

Good English output: Mary arranged for John to rent a house in St. Louis.

---

Gray: human prompts,  
boldface: GPT-3  
completions

# Task: Generate poems

---

## ----- Generated Poem 1 -----

I must have shadows on the way  
If I am to walk I must have  
Each step taken slowly and alone  
To have it ready made

And I must think in lines of grey  
To have dim thoughts to be my guide  
Must look on blue and green  
And never let my eye forget  
That color is my friend  
And purple must surround me too

The yellow of the sun is no more  
Intrusive than the bluish snow  
That falls on all of us. I must have  
Grey thoughts and blue thoughts walk with me  
If I am to go away at all.

## ----- Generated Poem 4 -----

Nobody will come to this place. It is a road that leads nowhere.  
The solitude is deep. The mountains are high.  
But they are desolate, and they turn the traveler's face  
Towards the North. All the sounds of the world are far away.  
When the wind rises above the trees,  
The boughs bow to the ground.  
Even the birds that inhabit the tangle of weeds  
That is the roadside cover, are silent. One listens,  
But hears no roar of the forest. One is alone.  
One will be taken.  
One will be taken.  
There is no utterance, there is no conversation,  
But one is uneasy all the same....  
There is a thin blue mist,  
A darkness rising like smoke,  
And within that darkness  
A possession of the heart.  
One will be taken.... It was here, and it will be here again-  
Here, under this sky empty and full of light.

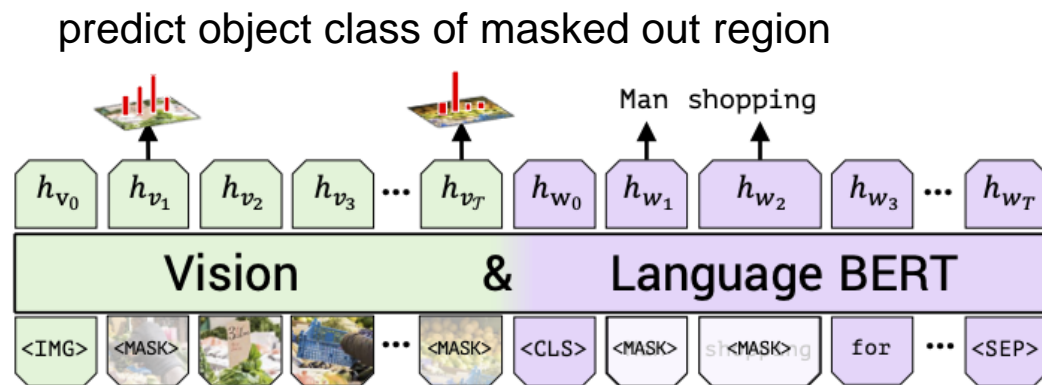
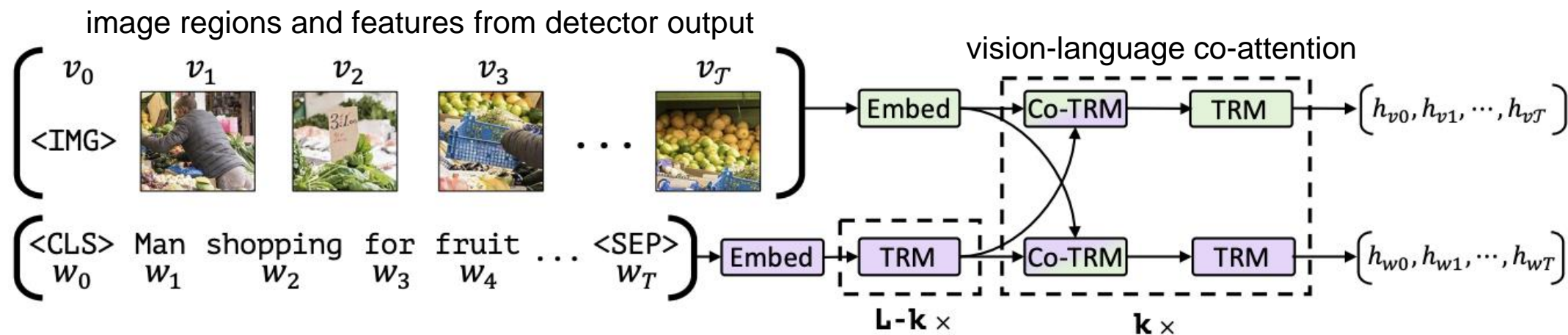
# Transformers: Outline

---

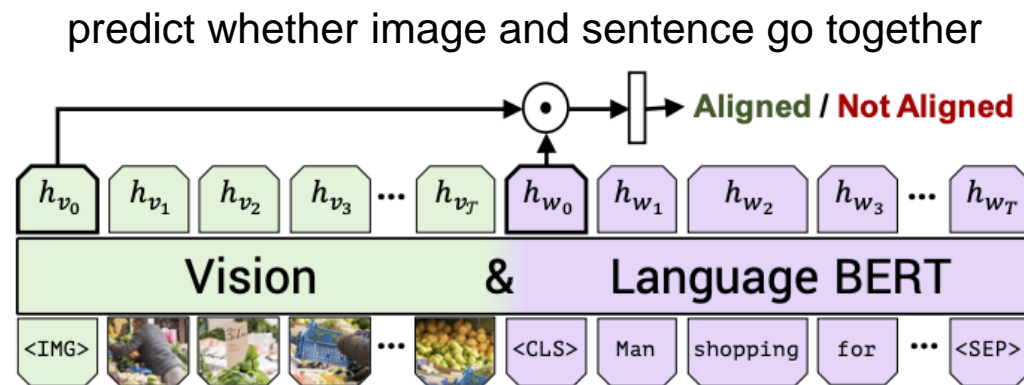
- Transformer architecture
  - Attention models
  - Implementation details
- Transformer-based language models
  - BERT
  - GPT and Other models
- Applications of transformers in vision



# Vision-and-language BERT

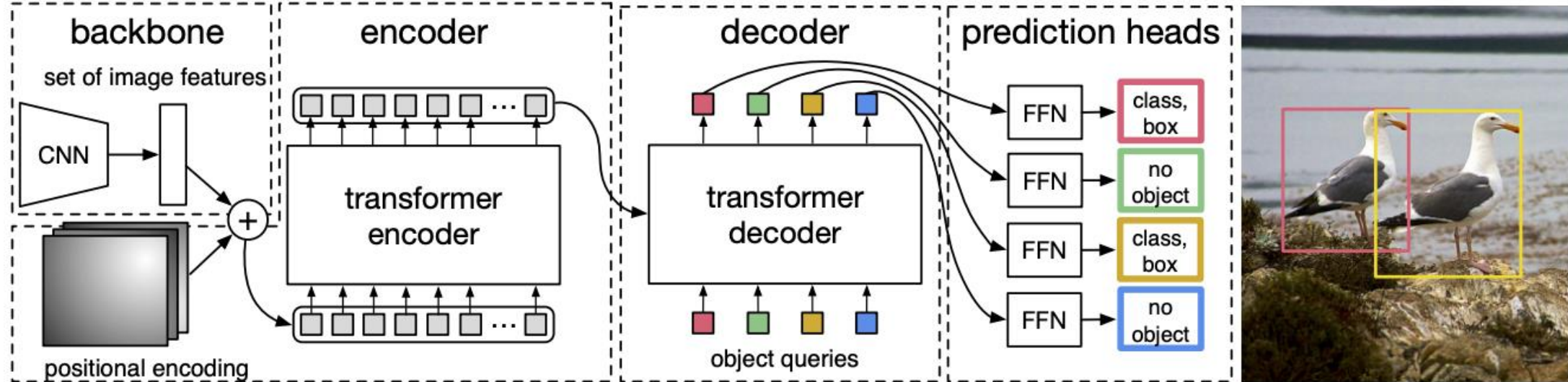
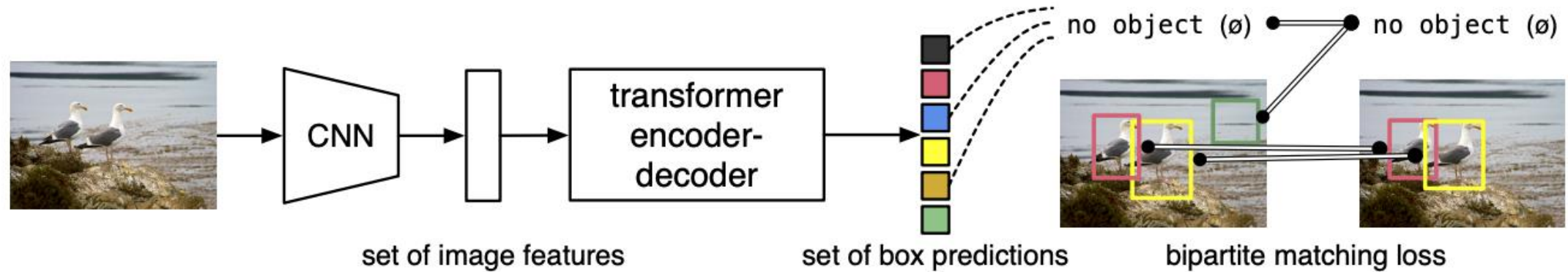


(a) Masked multi-modal learning



(b) Multi-modal alignment prediction

# Detection Transformer (DETR)



# Image transformer

---

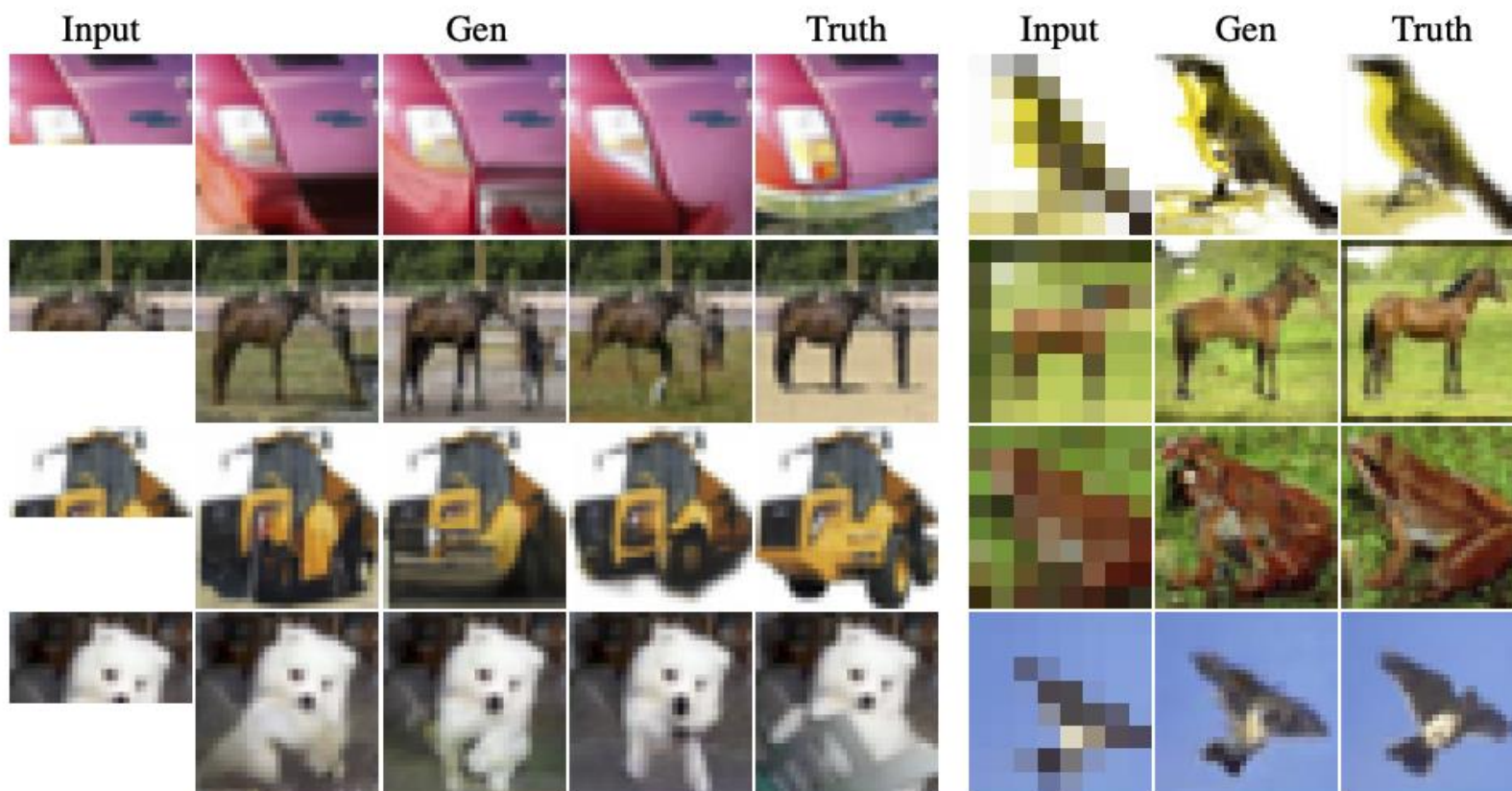
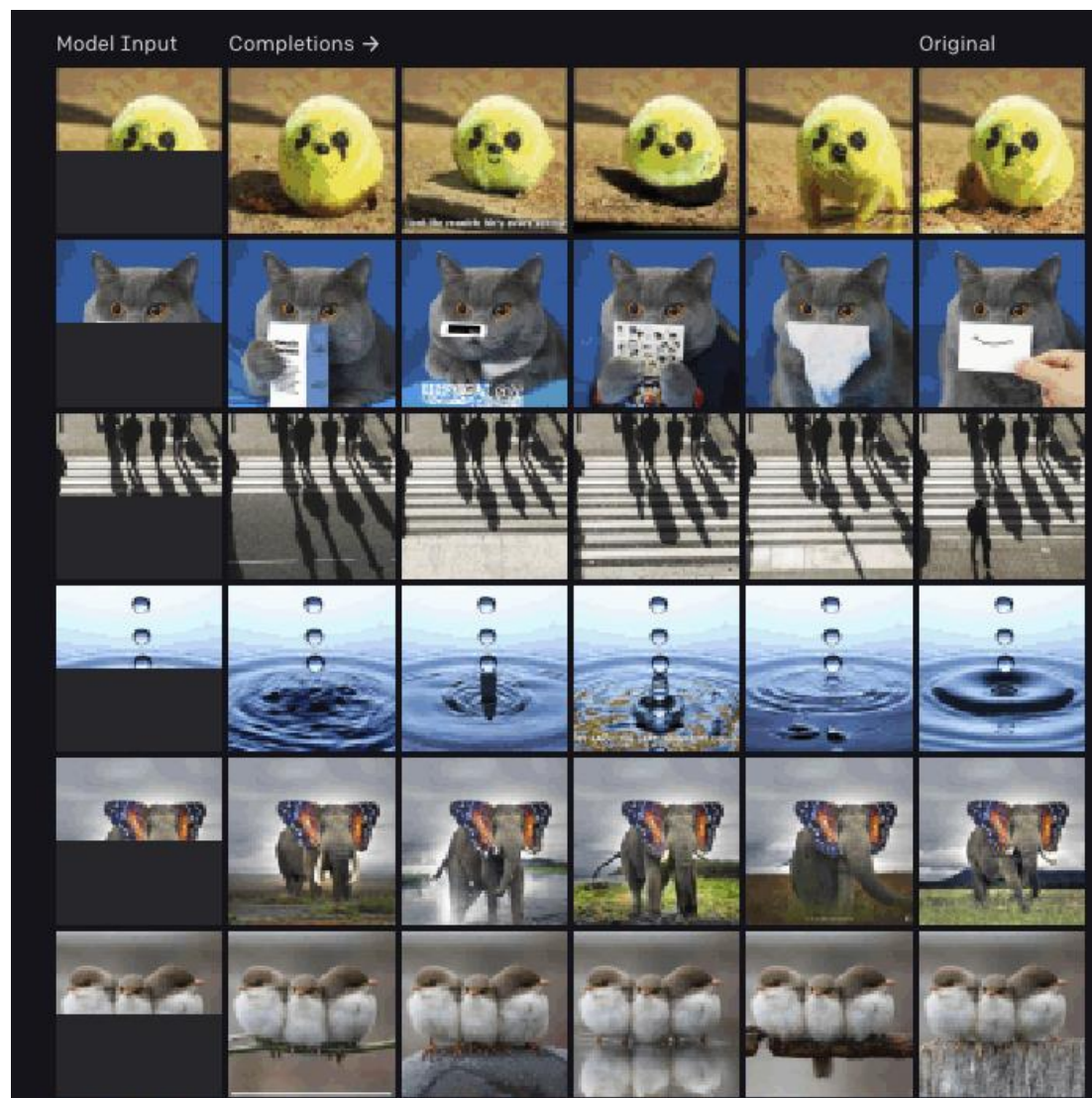


Table 2. On the left are image completions from our best conditional generation model, where we sample the second half. On the right are samples from our four-fold super-resolution model trained on CIFAR-10. Our images look realistic and plausible, show good diversity among the completion samples and observe the outputs carry surprising details for coarse inputs in super-resolution.

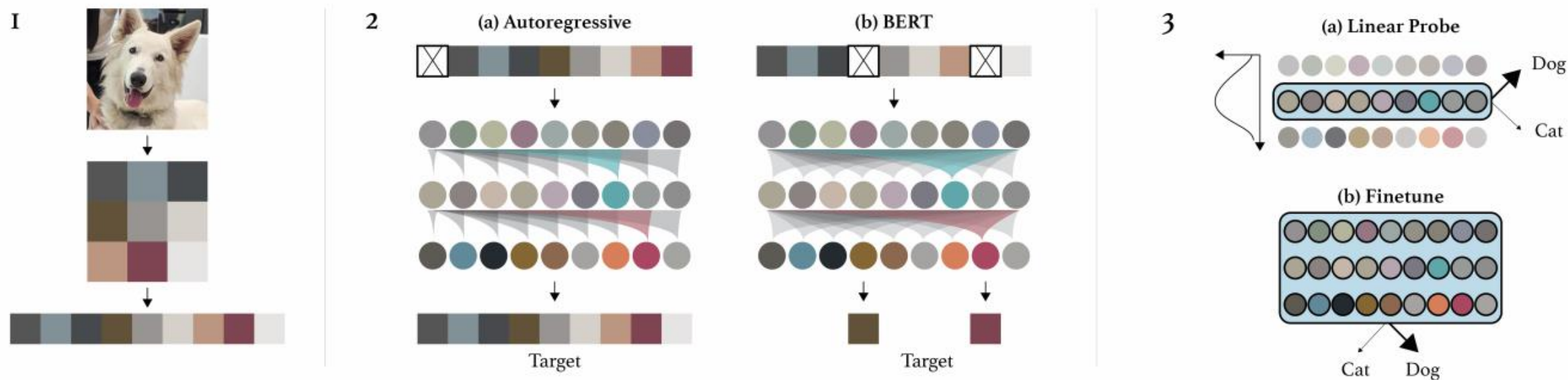


# Image GPT



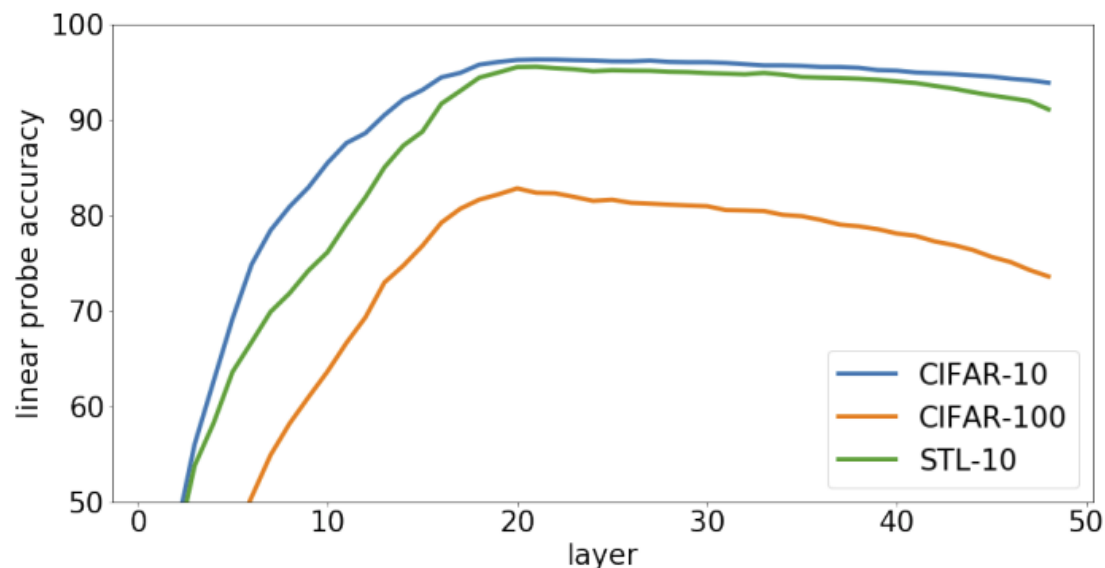
<https://openai.com/blog/image-gpt/>

# Image GPT



*Figure 1.* An overview of our approach. First, we pre-process raw images by resizing to a low resolution and reshaping into a 1D sequence. We then chose one of two pre-training objectives, auto-regressive next pixel prediction or masked pixel prediction. Finally, we evaluate the representations learned by these objectives with linear probes or fine-tuning.

# Image GPT



*Figure 2.* Representation quality depends on the layer from which we extract features. In contrast with supervised models, the best representations for these generative models lie in the middle of the network. We plot this unimodal dependence on depth by showing linear probes for iGPT-L on CIFAR-10, CIFAR-100, and STL-10.

*Table 2.* Comparing linear probe accuracies between our models and state-of-the-art self-supervised models. A blank input resolution (IR) corresponds to a model working at standard ImageNet resolution. We report the best performing configuration for each contrastive method, finding that our models achieve comparable performance.

Method	IR	Params (M)	Features	Acc
Rotation	orig.	86	8192	55.4
iGPT-L	$32^2 \cdot 3$	1362	1536	60.3
BigBiGAN	orig.	86	8192	61.3
iGPT-L	$48^2 \cdot 3$	1362	1536	65.2
AMDIM	orig.	626	8192	68.1
MoCo	orig.	375	8192	68.6
iGPT-XL	$64^2 \cdot 3$	6801	3072	68.7
SimCLR	orig.	24	2048	69.3
CPC v2	orig.	303	8192	71.5
iGPT-XL	$64^2 \cdot 3$	6801	15360	72.0
SimCLR	orig.	375	8192	76.5



# Acknowledgement

---

Thanks to the following courses and corresponding researchers for making their teaching/research material online

- Deep Learning, Stanford University
- Introduction to Deep Learning, University of Illinois at Urbana-Champaign
- Introduction to Deep Learning, Carnegie Mellon University
- Convolutional Neural Networks for Visual Recognition, Stanford University
- Natural Language Processing with Deep Learning, Stanford University
- And Many More .....