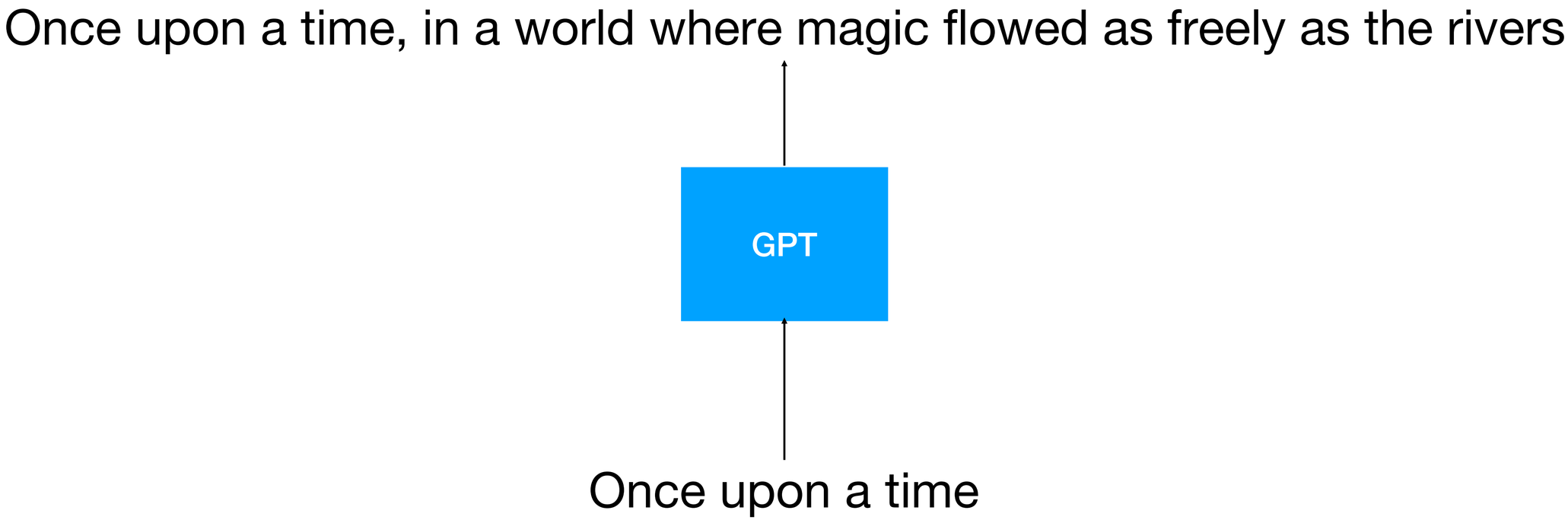
Understanding Transformers - 2

Vinay P. Namboodiri

Using Transformers – an example

First we will try to understand GPT

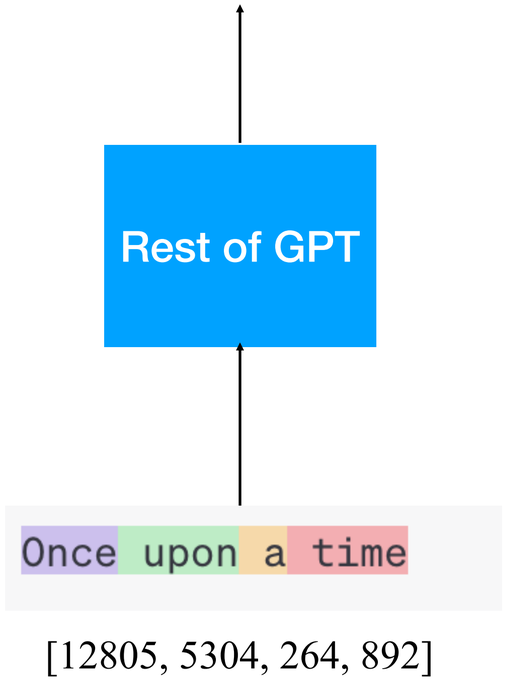
We want to understand what happens when we provide a prompt to GPT



First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT

Once upon a time, in a world where magic flowed as freely as the rivers



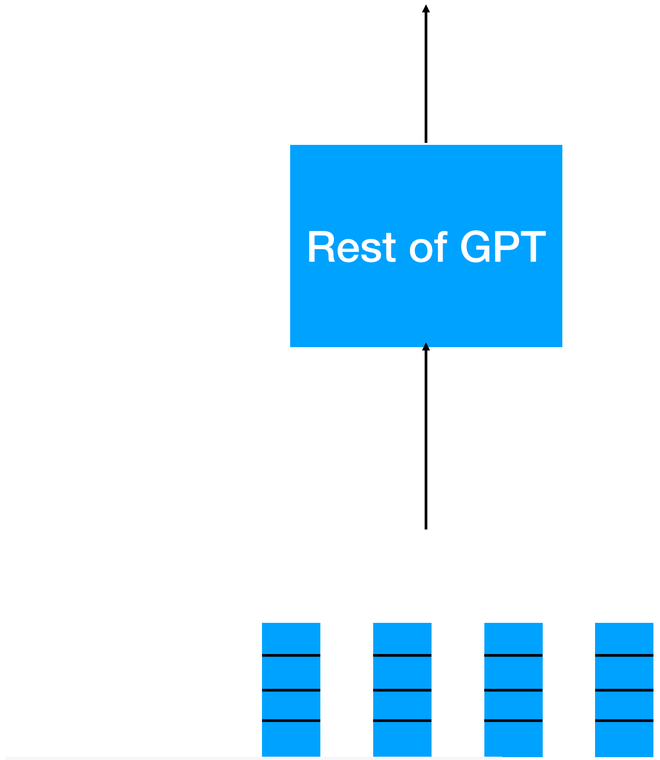
The text is converted into token vectors -

Once upon a time

First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT

Once upon a time, in a world where magic flowed as freely as the rivers



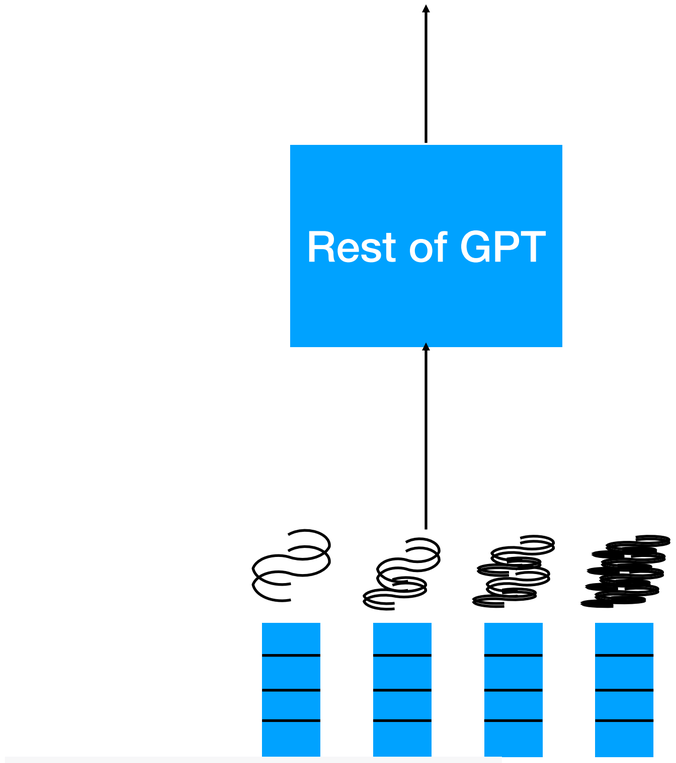
Sequence of integers is mapped to a set of embedding vectors by an embedding matrix W_E

Once upon a time [12805, 5304, 264, 892]

First we will try to understand GPT

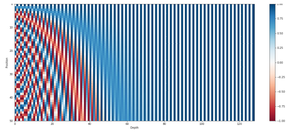
We want to understand what happens when we provide a prompt to GPT

Once upon a time, in a world where magic flowed as freely as the rivers



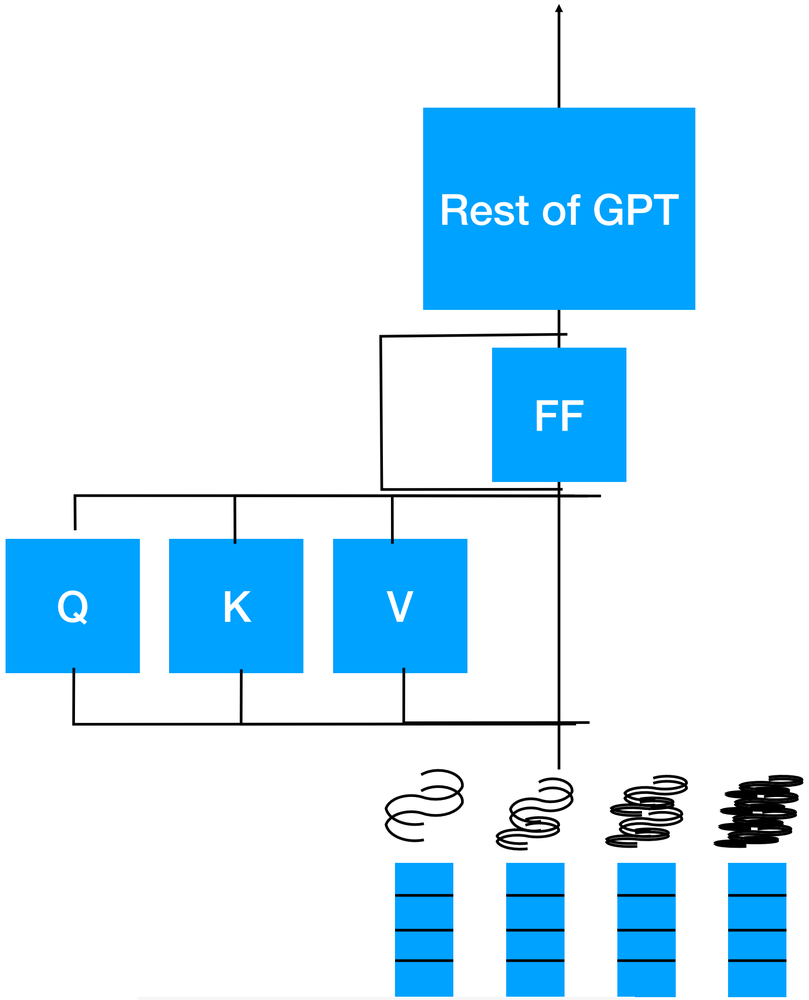
Positional encoding is added to the embedding -

$$\psi'(w_t) = \psi(w_t) + \overrightarrow{p_t}$$



First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT

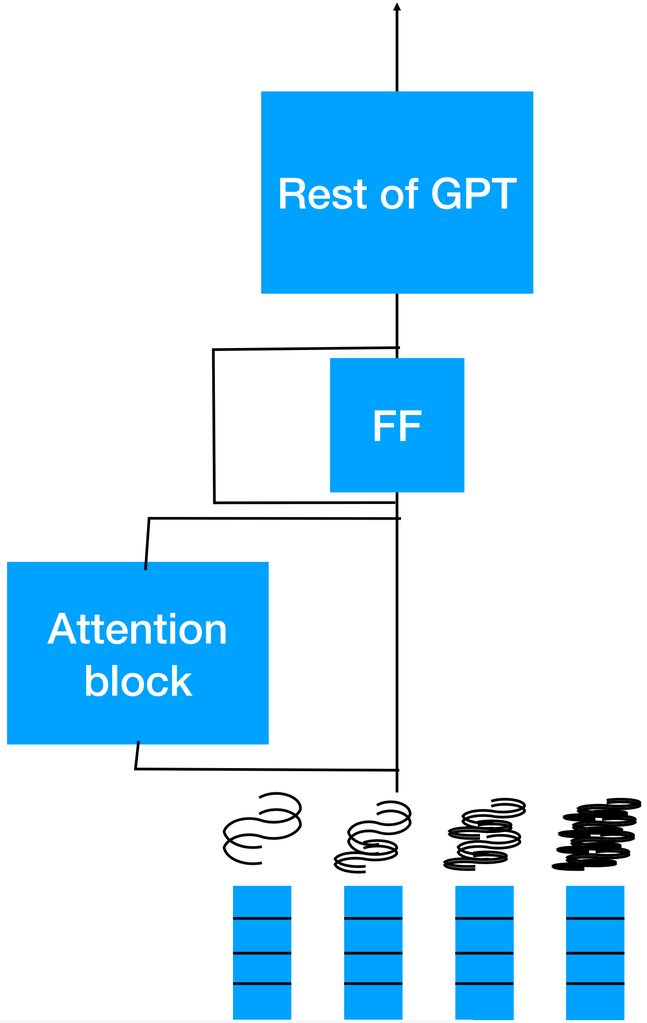


This residual stream is then passed through a ‘self attention + FF layer block

Once upon a time [12805, 5304, 264, 892]

First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT

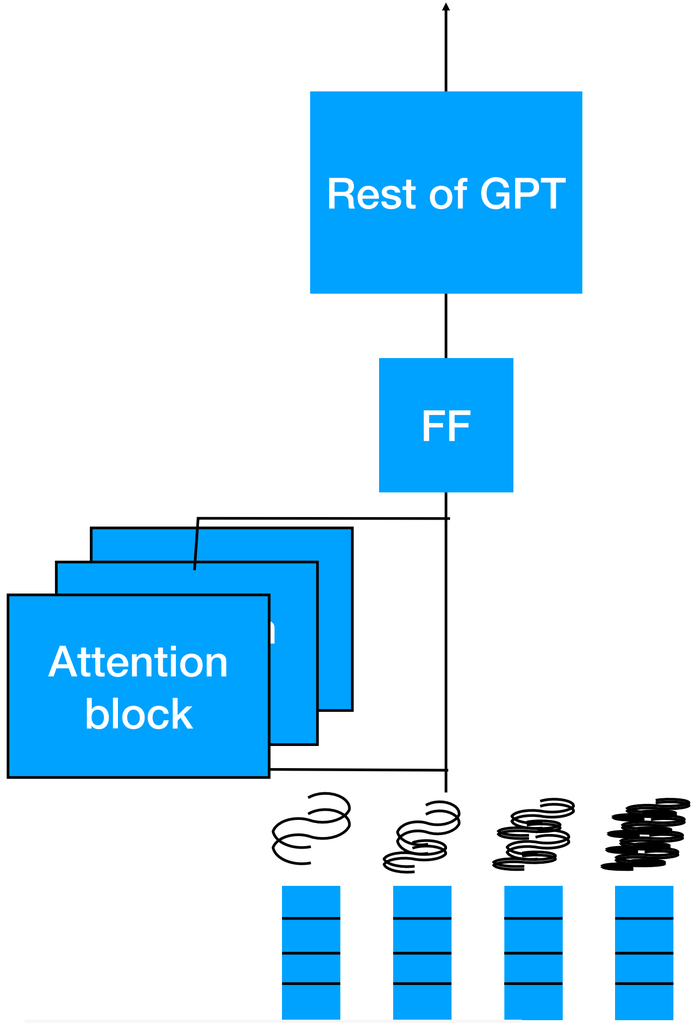


This residual stream is then passed through a ‘self attention + FF layer block

Once upon a time [12805, 5304, 264, 892]

First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT

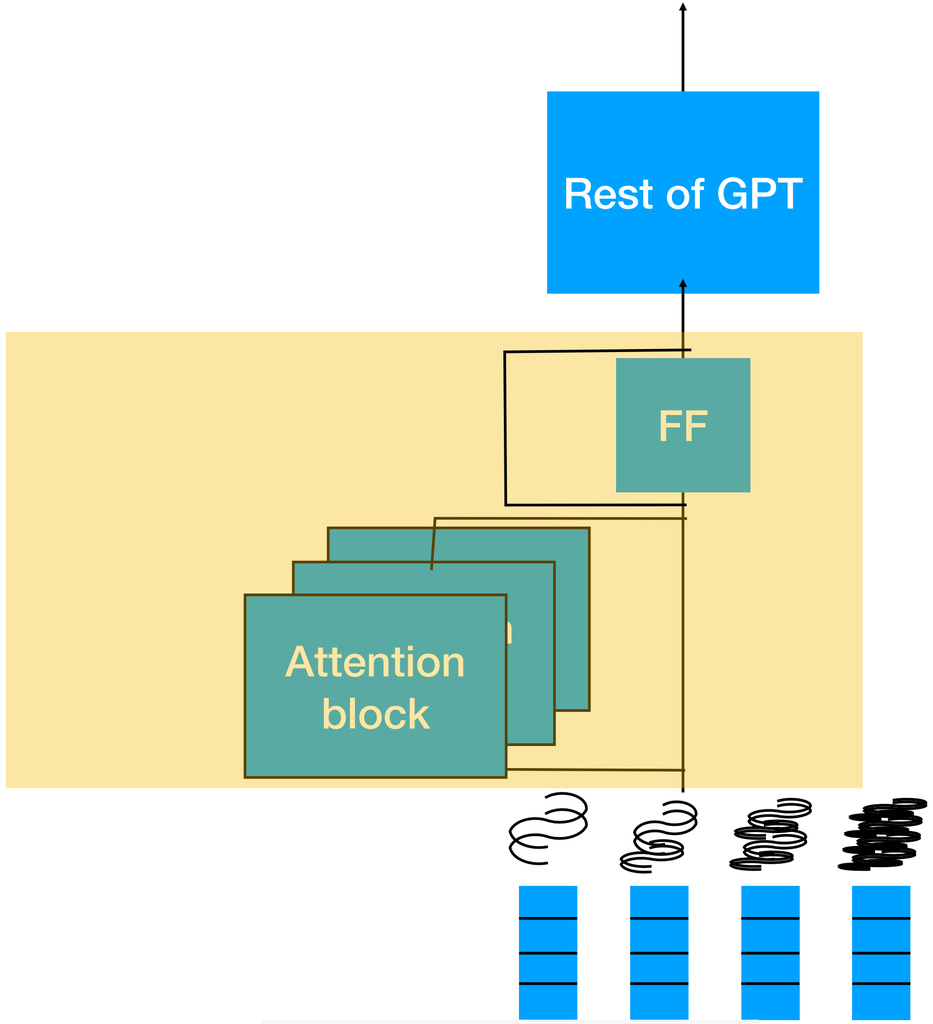


In practice, there are multiple heads running in parallel

Once upon a time [12805, 5304, 264, 892]

First we will try to understand GPT

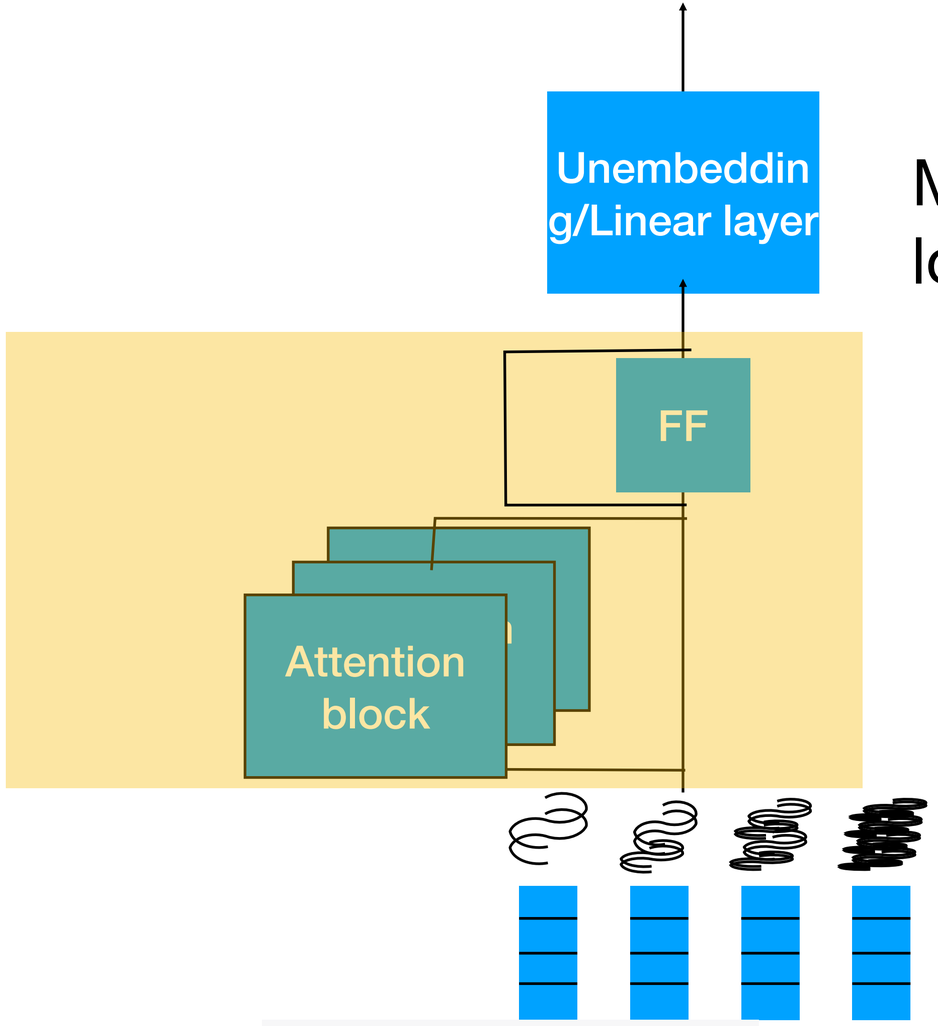
We want to understand what happens when we provide a prompt to GPT



This is one layer of the transformer - this structure is repeated many times

First we will try to understand GPT

We want to understand what happens when we provide a prompt to GPT



Maps from embedding space to logits for the token space

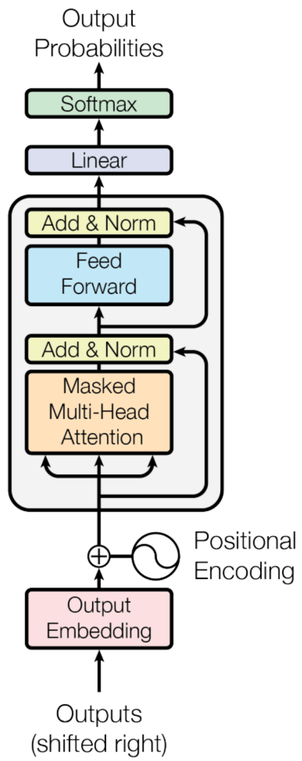
Many layers of this structure

Transformer architectures

NLP architectures

Decoder-only GPT

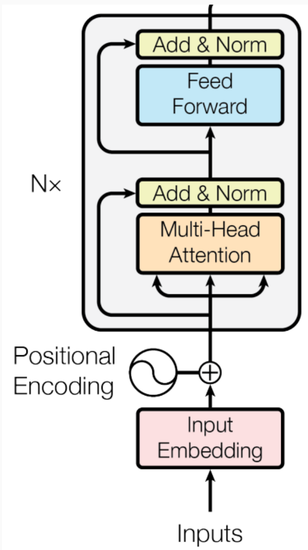
[sat_]



[START] [The_] [cat_]

Encoder-only BERT

[*] [*] **[sat_]** [*] **[the_]** [*]



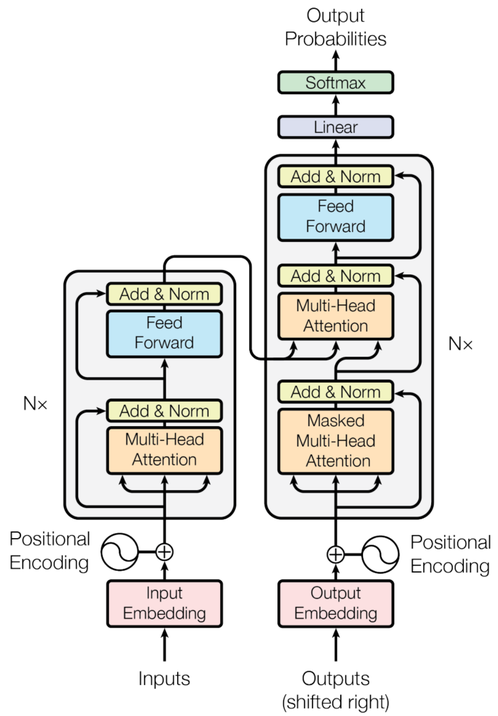
[The_] [cat_] **[MASK]** [on_] **[MASK]** [mat_]

Enc-Dec T5/M2M

Das ist gut.

A storm in Attala caused 6 victims.

This is not toxic.



Translate EN-DE: This is good.

Summarize: state authorities dispatched…

Is this toxic: You look beautiful today!

Transformer image source: "Attention Is All You Need" paper

BERT - Pretraining Task MLM

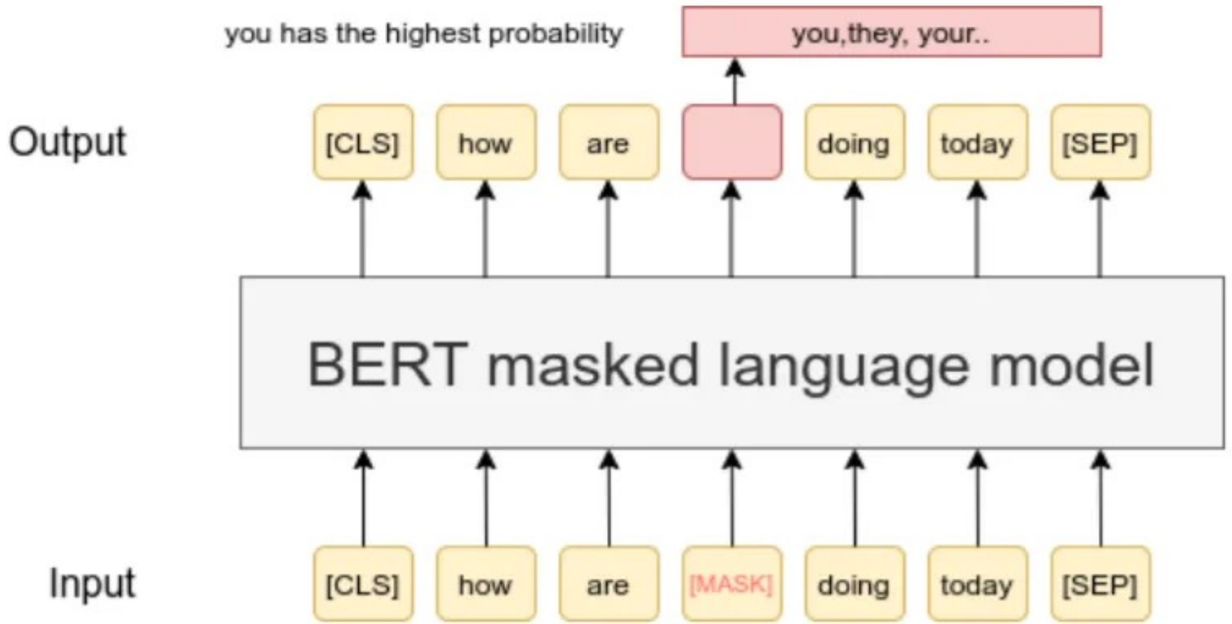


Fig from - https://medium.com/@shaikhrayyan123/a-comprehensive-guide-to-understanding-bert-from-beginners-to-advanced-2379699e2b51

BERT - Pretraining next sentence prediction

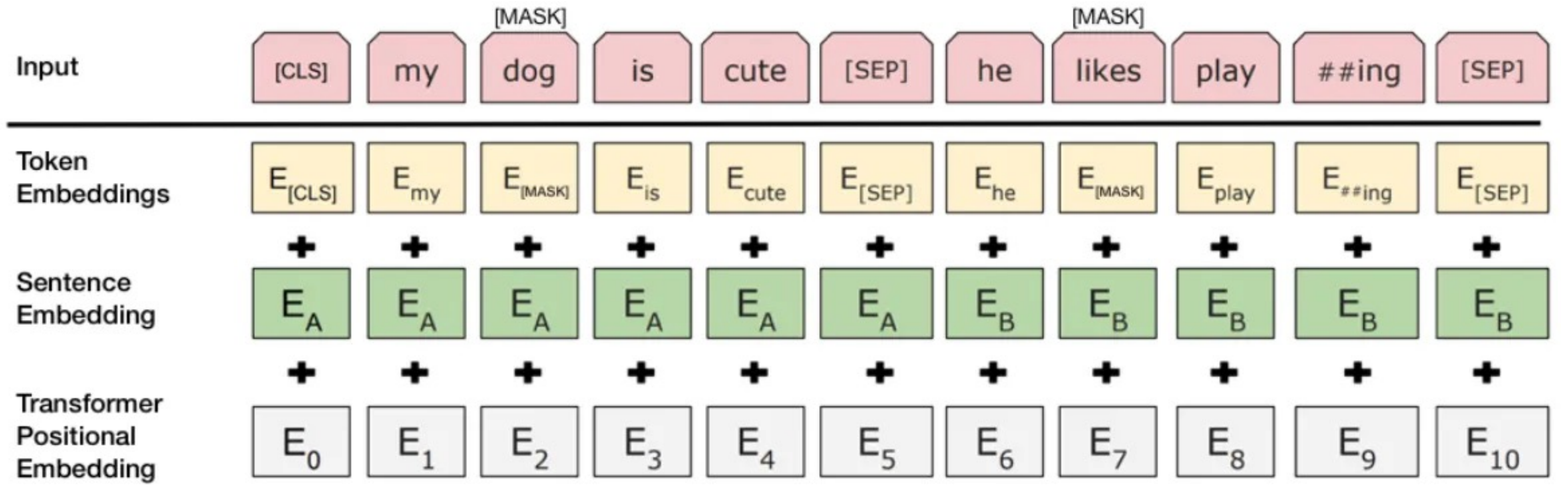


Fig from BERT paper - BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

BERT - usage for various tasks

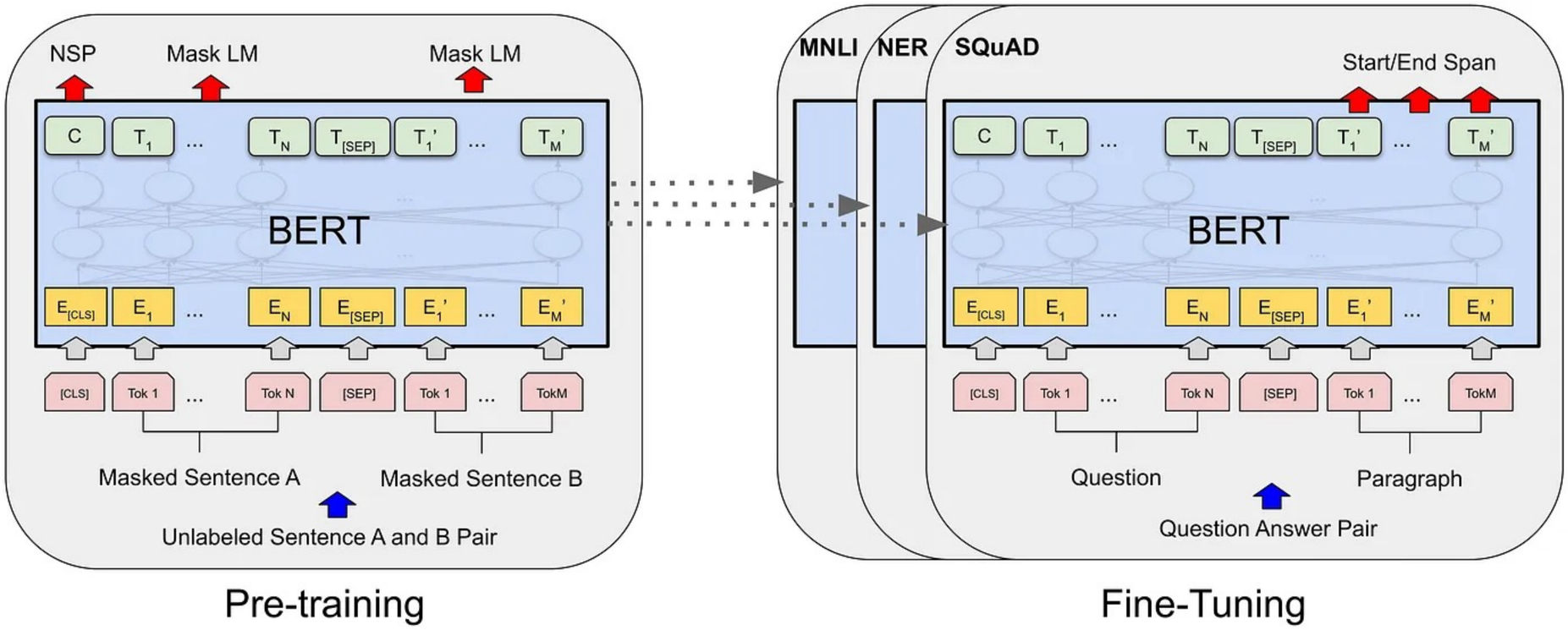


Fig from BERT paper - BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

GPT-3 Language Models are Few Shot Learners

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Fig from GPT3 paper - Language Models are Few Shot Learners

GPT-3 Language Models are Few Shot Learners

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

Fig from GPT3 paper - Language Models are Few Shot Learners

GPT-3 Language Models are Few Shot Learners

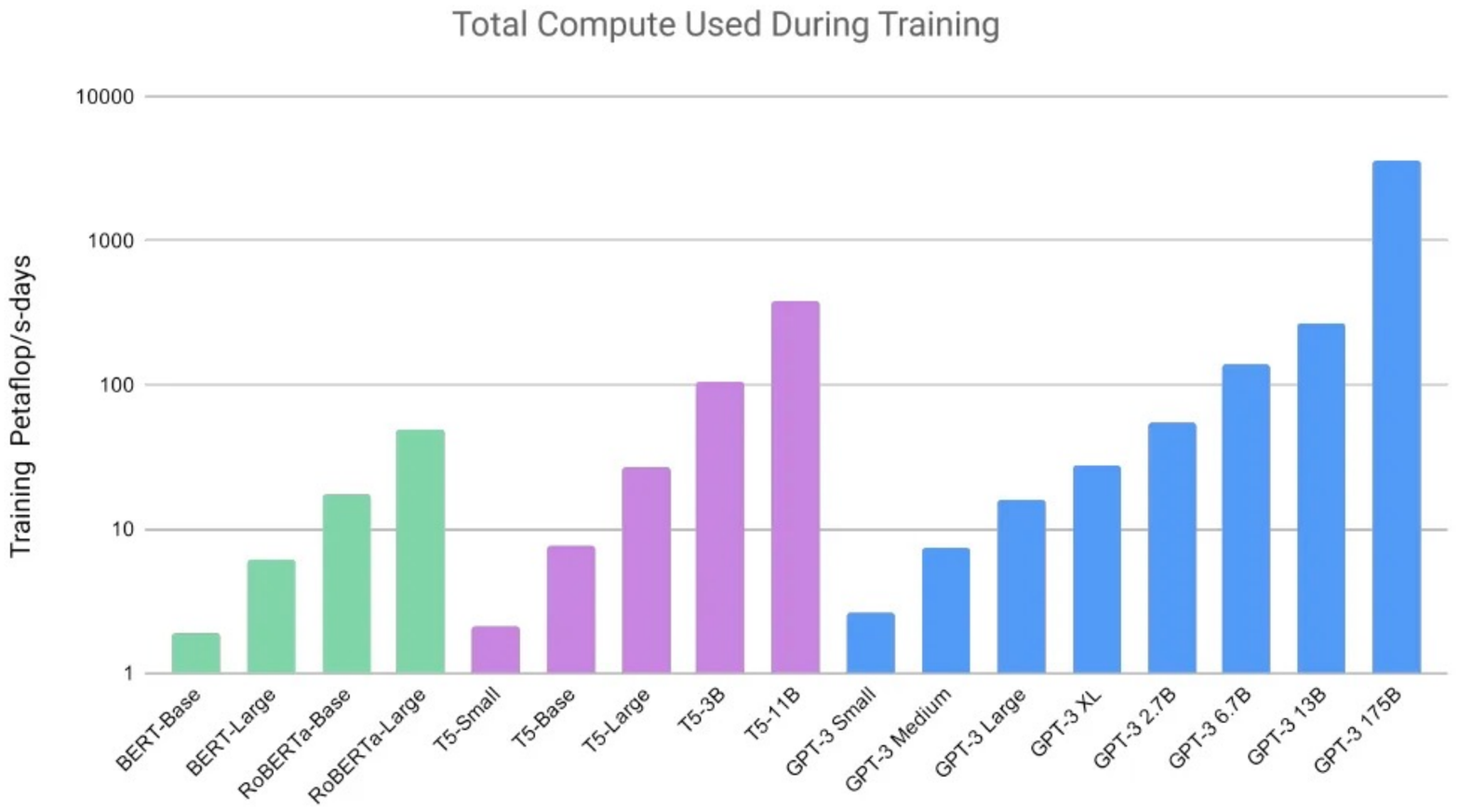


Fig from GPT3 paper - Language Models are Few Shot Learners

GPT-3 Language Models are Few Shot Learners

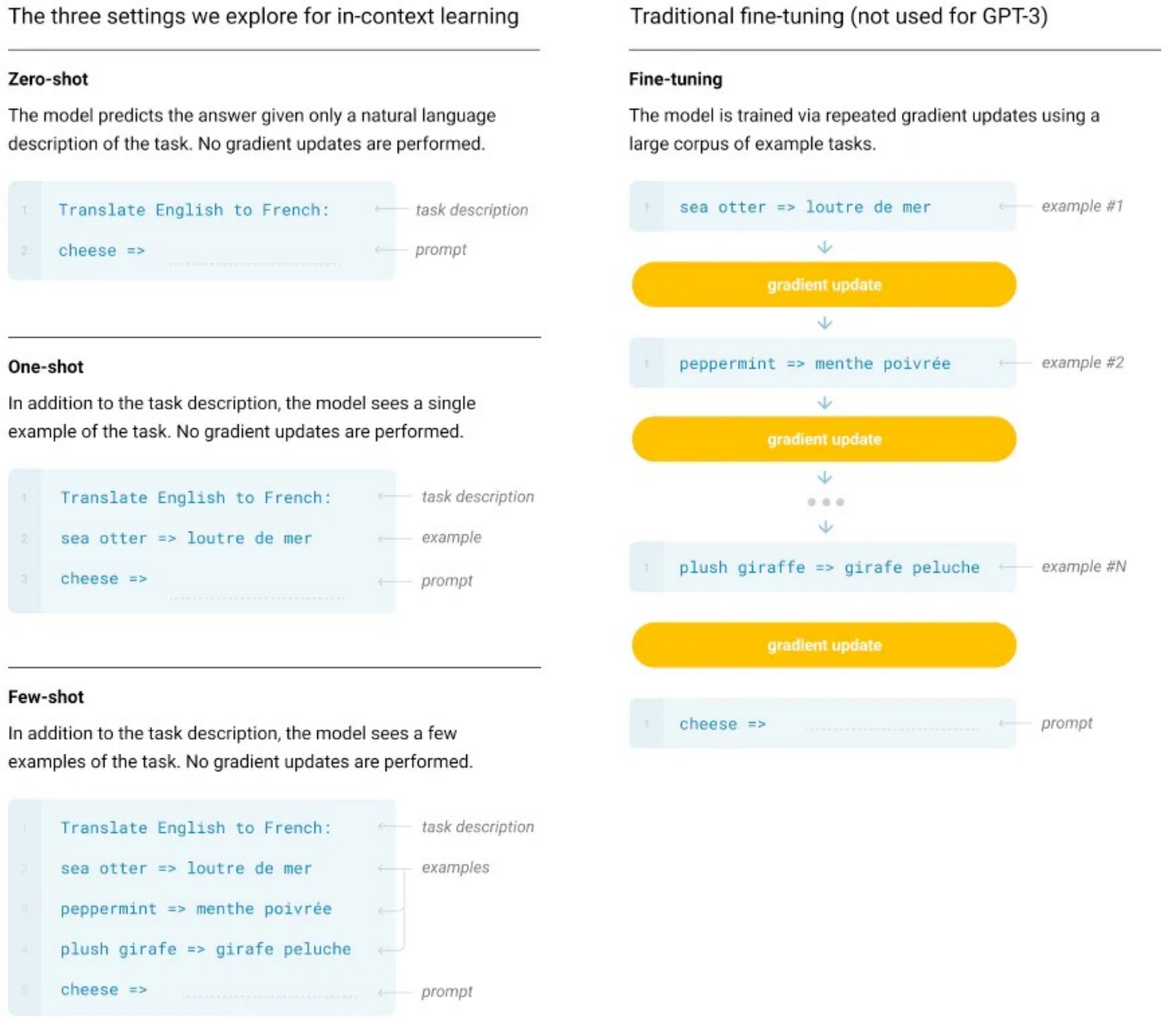


Fig from GPT3 paper - Language Models are Few Shot Learners

GPT-3 Language Models are Few Shot Learners

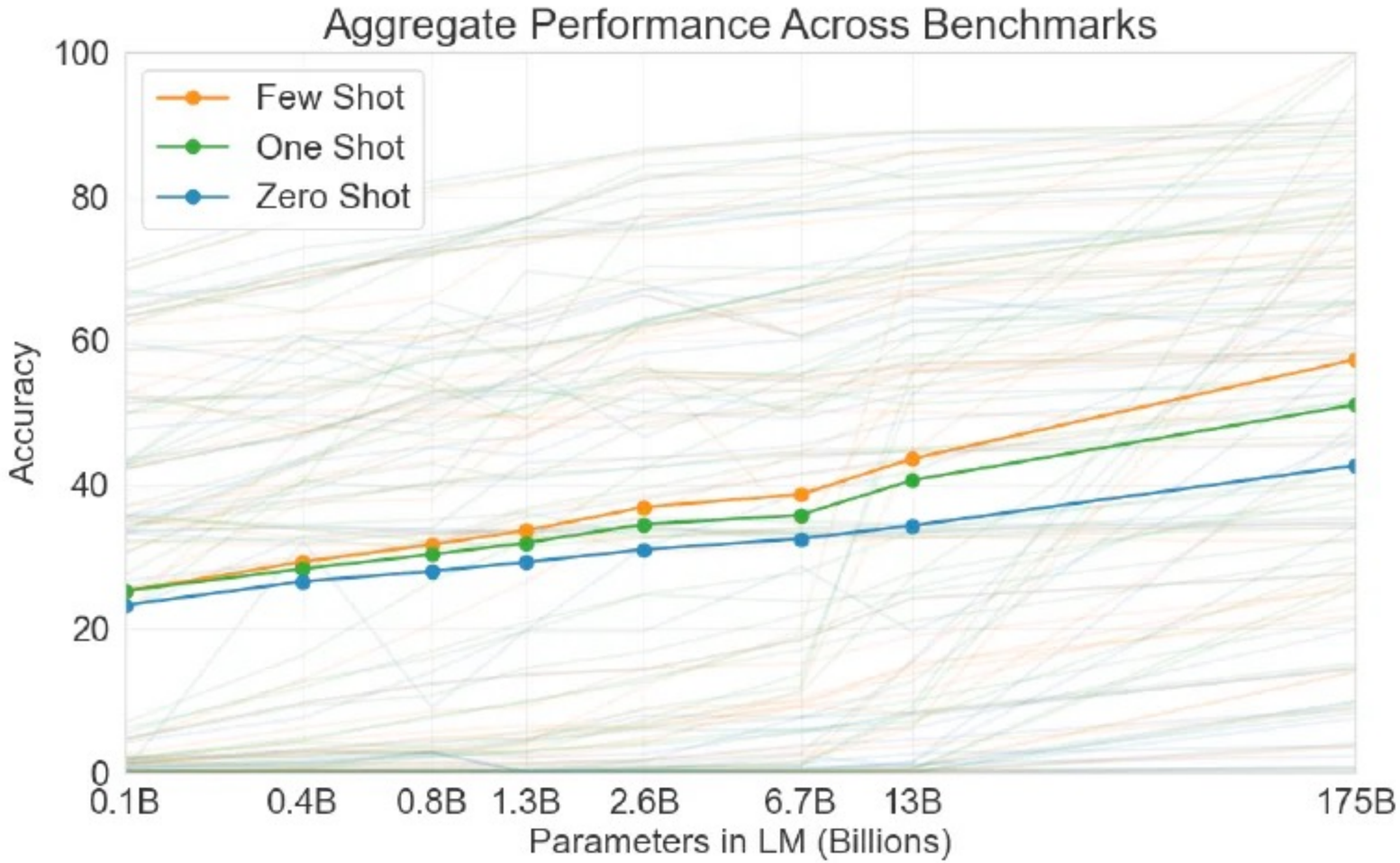
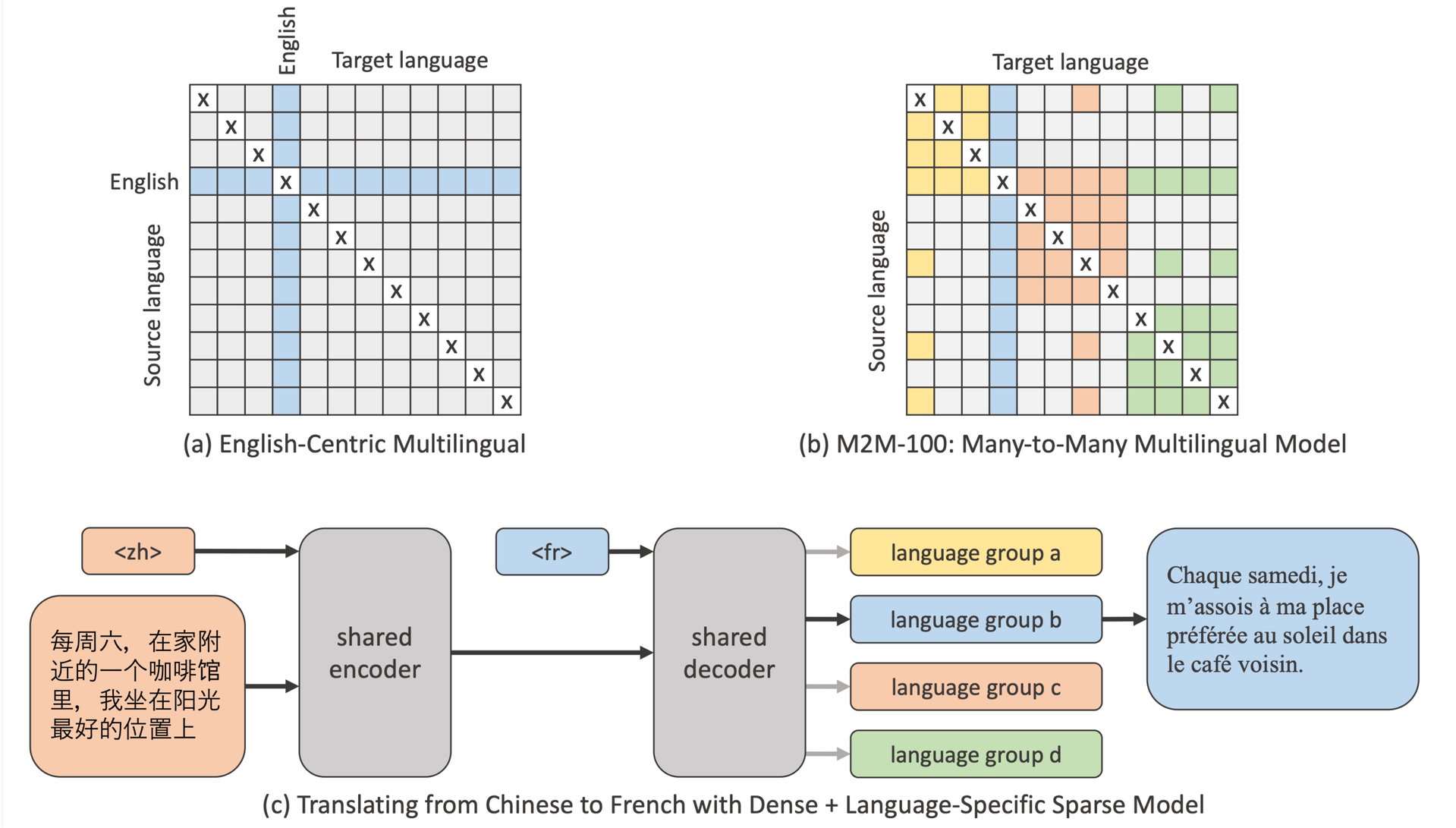


Fig from GPT3 paper - Language Models are Few Shot Learners

M2M100


English
Target language
English
Source language
(a) English-Centric Multilingual
Target language
Source language
(b) M2M-100: Many-to-Many Multilingual Model
<zh>
每周六，在家附近的一个咖啡馆里，我坐在阳光最好的位置上
shared encoder
<fr>
shared decoder
language group a
language group b
language group c
language group d
Chaque samedi, je m'assois à ma place préférée au soleil dans le café voisin.
(c) Translating from Chinese to French with Dense + Language-Specific Sparse Model

M2M100

ISO	Language	Family	Script	ISO	Language	Family	Script
af	Afrikaans	Germanic	Latin	ja	**Japanese**	Japonic	Kanji; Kana
da	Danish	Germanic	Latin	ko	**Korean**	Koreanic	Hangul
nl	**Dutch**	Germanic	Latin	vi	**Vietnamese**	Vietic	Latin
de	**German**	Germanic	Latin	zh	**Chinese Mandarin**	Chinese	Chinese
en	**English**	Germanic	Latin	bn	**Bengali**	Indo-Aryan	Eastern-Nagari
is	Icelandic	Germanic	Latin	gu	Gujarati	Indo-Aryan	Gujarati
lb	Luxembourgish	Germanic	Latin	hi	**Hindi**	Indo-Aryan	Devanagari
no	Norwegian	Germanic	Latin	kn	Kannada	Tamil	Kannada
sv	**Swedish**	Germanic	Latin	mr	Marathi	Indo-Aryan	Devanagari
fy	Western Frisian	Germanic	Latin	ne	Nepali	Indo-Aryan	Devanagari
yi	Yiddish	Germanic	Hebrew	or	Oriya	Indo-Aryan	Odia
ast	Asturian	Romance	Latin	pa	Panjabi	Indo-Aryan	Gurmukhi
ca	Catalan	Romance	Latin	sd	Sindhi	Indo-Aryan	Persian Devanagari
fr	**French**	Romance	Latin	si	Sinhala	Indo-Aryan	Sinhala
gl	Galician	Romance	Latin	ur	Urdu	Indo-Aryan	Arabic
it	Italian	Romance	Latin	ta	**Tamil**	Dravidian	Tamil
oc	Occitan	Romance	Latin	ceb	Cebuano	Malayo-Polyn.	Latin
pt	**Portuguese**	Romance	Latin	ilo	Iloko	Philippine	Latin
ro	Romanian	Romance	Latin	id	**Indonesian**	Malayo-Polyn.	Latin
es	**Spanish**	Romance	Latin	jv	Javanese	Malayo-Polyn.	Latin
be	Belarusian	Slavic	Cyrillic	mg	Malagasy	Malayo-Polyn.	Latin
bs	Bosnian	Slavic	Latin	ms	Malay	Malayo-Polyn.	Latin
bg	Bulgarian	Slavic	Cyrillic	ml	Malayalam	Dravidian	Malayalam
hr	Croatian	Slavic	Latin	su	Sundanese	Malayo-Polyn.	Latin
cs	**Czech**	Slavic	Latin	tl	Tagalog	Malayo-Polyn.	Latin
mk	Macedonian	Slavic	Cyrillic	my	Burmese	Sino-Tibetan	Burmese
pl	**Polish**	Slavic	Latin	km	Central Khmer	Khmer	Khmer
ru	**Russian**	Slavic	Cyrillic	lo	Lao	Kra-Dai	Thai; Lao
sr	Serbian	Slavic	Cyrillic; Latin	th	Thai	Kra-Dai	Thai
sk	Slovak	Slavic	Latin	mn	Mongolian	Mongolic	Cyrillic
sl	Slovenian	Slavic	Latin	ar	**Arabic**	Arabic	Arabic
uk	Ukrainian	Slavic	Cyrillic	he	**Hebrew**	Semitic	Hebrew
et	Estonian	Uralic	Latin	ps	Pashto	Iranian	Arabic
fi	**Finnish**	Uralic	Latin	fa	**Farsi**	Iranian	Arabic
hu	**Hungarian**	Uralic	Latin	am	Amharic	Ethopian	Ge'ez
lv	Latvian	Baltic	Latin	ff	Fulah	Niger-Congo	Latin
lt	**Lithuanian**	Baltic	Latin	ha	Hausa	Afro-Asiatic	Latin
sq	Albanian	Albanian	Latin	ig	Igbo	Niger-Congo	Latin
hy	Armenian	Armenian	Armenian	ln	Lingala	Niger-Congo	Latin
ka	Georgian	Kartvelian	Georgian	lg	Luganda	Niger-Congo	Latin
el	**Greek**	Hellenic	Greek	nso	Northern Sotho	Niger-Congo	Latin
br	Breton	Celtic	Latin	so	Somali	Cushitic	Latin
ga	Irish	Irish	Latin	sw	**Swahili**	Niger-Congo	Latin
gd	Scottish Gaelic	Celtic	Latin	ss	Swati	Niger-Congo	Latin
cy	Welsh	Celtic	Latin-Welsch	tn	Tswana	Niger-Congo	Latin
az	Azerbaijani	Turkic	Latin; Cyrillic Persian	wo	Wolof	Niger-Congo	Latin
ba	Bashkir	Turkic	Cyrillic	xh	Xhosa	Niger-Congo	Latin
kk	Kazakh	Turkic	Cyrillic	yo	Yoruba	Niger-Congo	Latin
tr	**Turkish**	Turkic	Latin	zu	Zulu	Niger-Congo	Latin
uz	Uzbek	Turkic	Latin; Cyrillic	ht	Haitian Creole	Creole	Latin

Table 1: **100 Languages grouped by family.** For each language, we display the ISO code, language name, language family, and script. Languages in bold are *bridge languages* (*Malayo-Polyn.* stands for *Malayo-Polynesian*).

M2M100

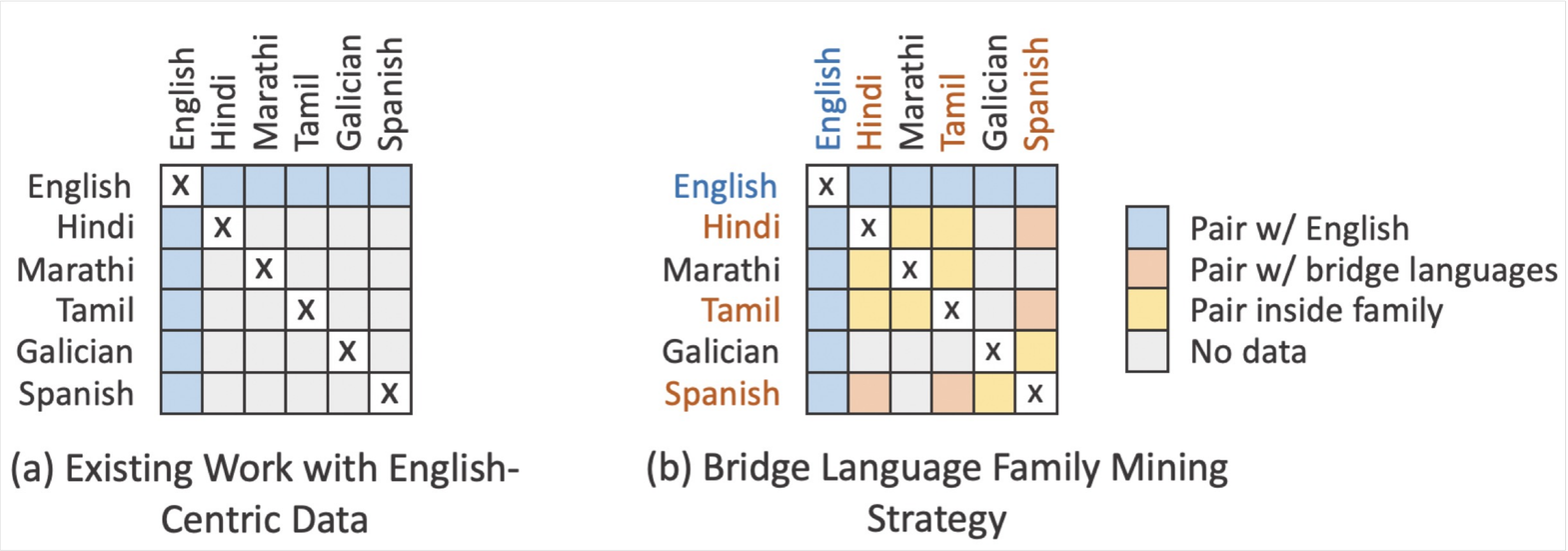


Figure 2: **Depiction of an English-Only data mining setting compared to the Bridge Language Mining Strategy**. We display a data matrix, where languages are shown on the X and Y axes. Data is mined in one direction (such as Hindi to Marathi) and used to train bidirectionally.

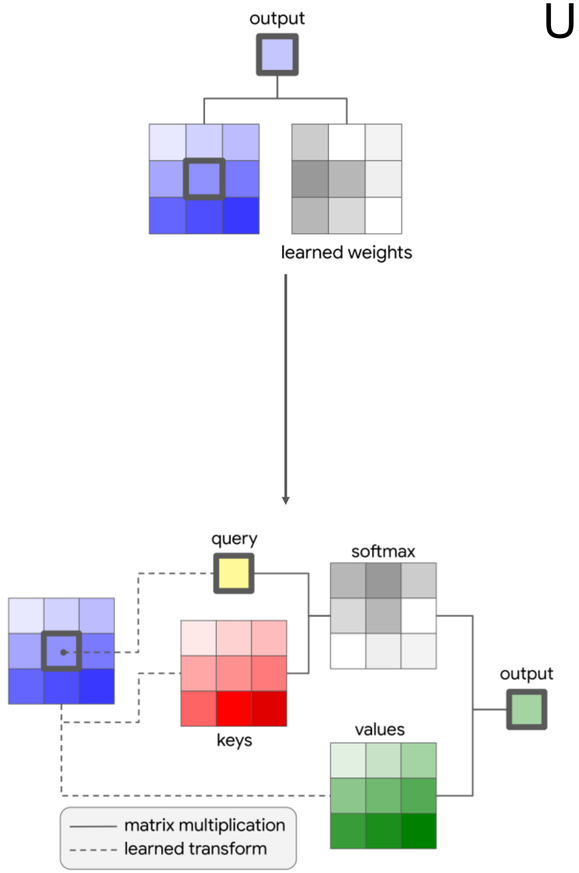
Computer Vision

Previous approaches

Many prior works attempted to introduce self-attention at the pixel level.

For 224px², that's 50k sequence length, too much!

1. On pixels, but locally or factorized



Usually replaces 3x3 conv in ResNet:

stage	output	ResNet-50	LR-Net-50 (7×7, m=8)
res1	112×112	7×7 conv, 64, stride 2	**1×1, 64** **7×7 LR, 64, stride 2**
res2	56×56	3×3 max pool, stride 2	3×3 max pool, stride 2
		[1×1, 64 3×3 conv, 64 1×1, 256] ×3	[1×1, 100 **7×7 LR, 100** 1×1, 256] ×3
res3	28×28	[1×1, 128 3×3 conv, 128 1×1, 512] ×4	[1×1, 200 **7×7 LR, 200** 1×1, 512] ×4
res4	14×14	[1×1, 256 3×3 conv, 256 1×1, 1024] ×6	[1×1, 400 **7×7 LR, 400** 1×1, 1024] ×6
res5	7×7	[1×1, 512 3×3 conv, 512 1×1, 2048] ×3	[1×1, 800 **7×7 LR, 800** 1×1, 2048] ×3
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params		**25.5**×10^6	**23.3**×10^6
FLOPs		**4.3**×10^9	**4.3**×10^9

Results:

Are not great

Do not justify increased complexity

Do not justify slowdown over convolutions

Examples:

Non-local NN (Wang et.al. 2017)

SASANet (Stand-Alone Self-Attention in Vision Models)

HaloNet (Scaling Local Self-Attn for Parameter Efficient...)

LR-Net (Local Relation Networks for Image Recognition)

SANet (Exploring Self-attention for Image Recognition)

...

Image credit: Stand-Alone Self-Attention in Vision Models by Ramachandran et.al.
Image credit: Local Relation Networks for Image Recognition by Hu et.al.

Slide credit Lucas Beyer

An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

2020, A Dosovitskiy, L Beyer, A Kolesnikov, D Weissenborn, X Zhai, T Unterthiner, M Dehghani, M Minderer, G Heigold, S Gelly, J Uszkoreit, N Houlsby

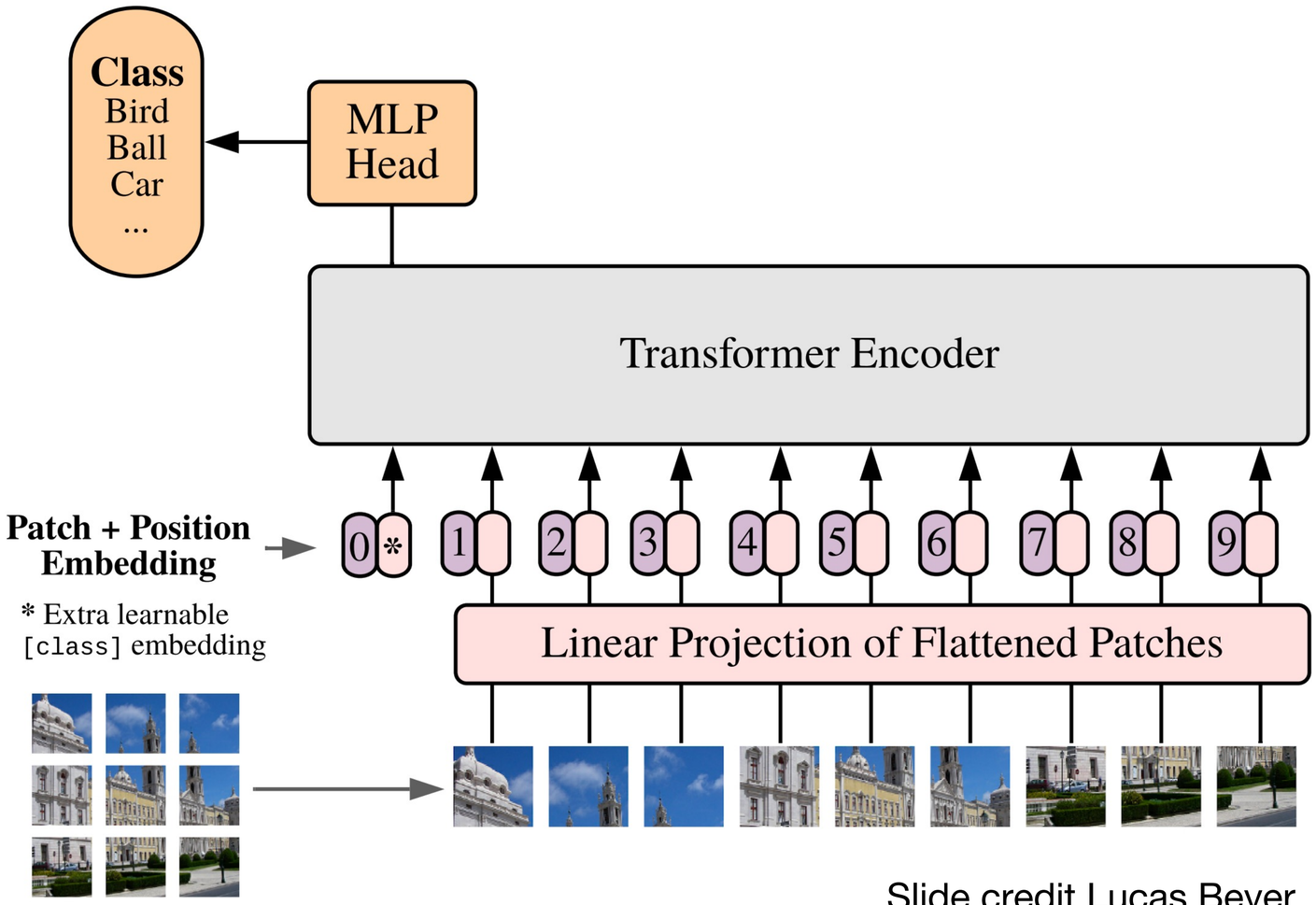
Many prior works attempted to introduce self-attention at the pixel level.

For 224px², that's 50k sequence length, too much!

Thus, most works restrict attention to local pixel neighborhoods, or as high-level mechanism on top of detections.

The **key breakthrough** in using the full Transformer architecture, standalone, was to **"tokenize" the image** by **cutting it into patches** of 16px², and treating each patch as a token, e.g. embedding it into input space.

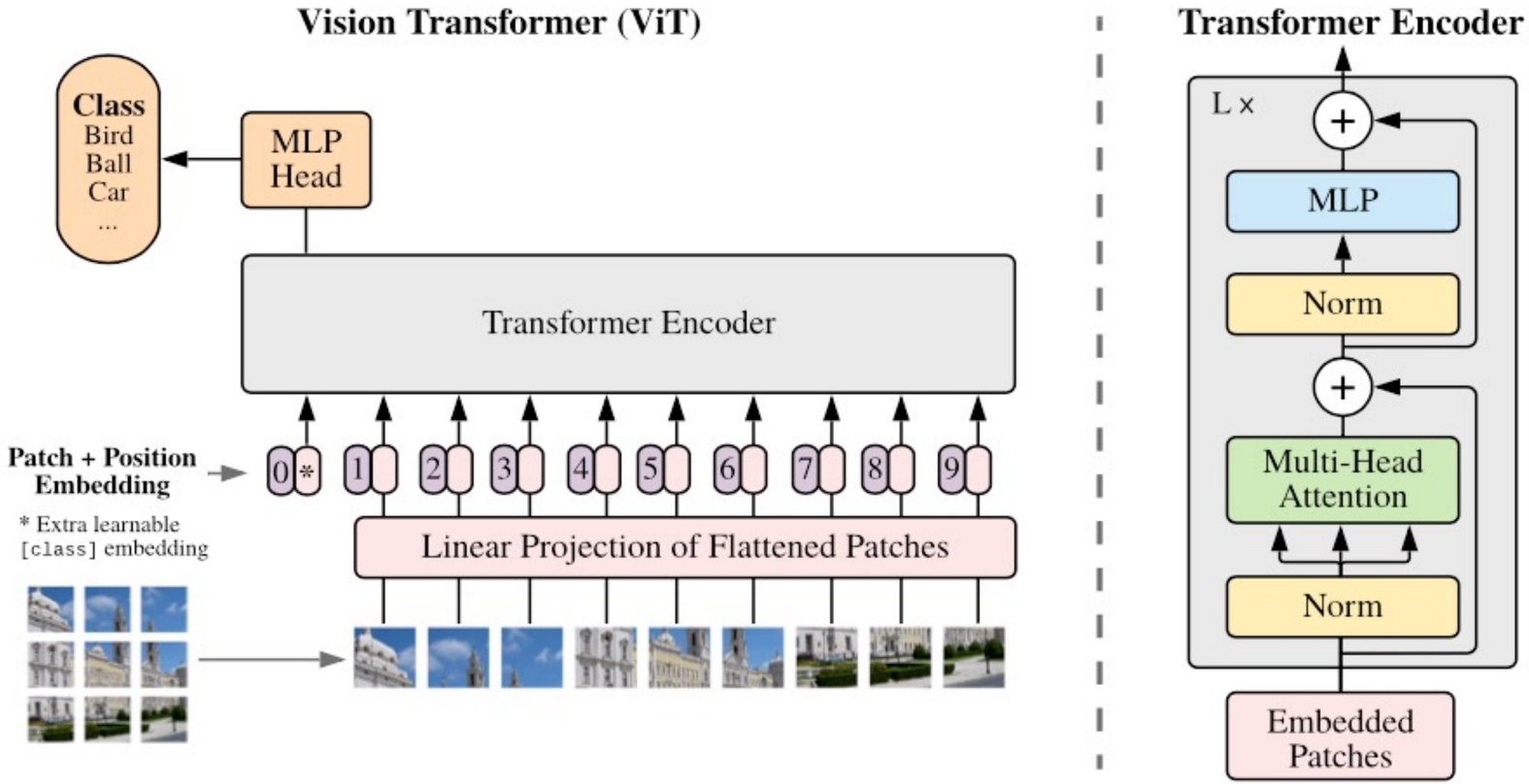
Vision Transformer (ViT)



Slide credit Lucas Beyer

An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

2020, A Dosovitskiy, L Beyer, A Kolesnikov, D Weissenborn, X Zhai, T Unterthiner, M Dehghani, M Minderer, G Heigold, S Gelly, J Uszkoreit, N Houlsby

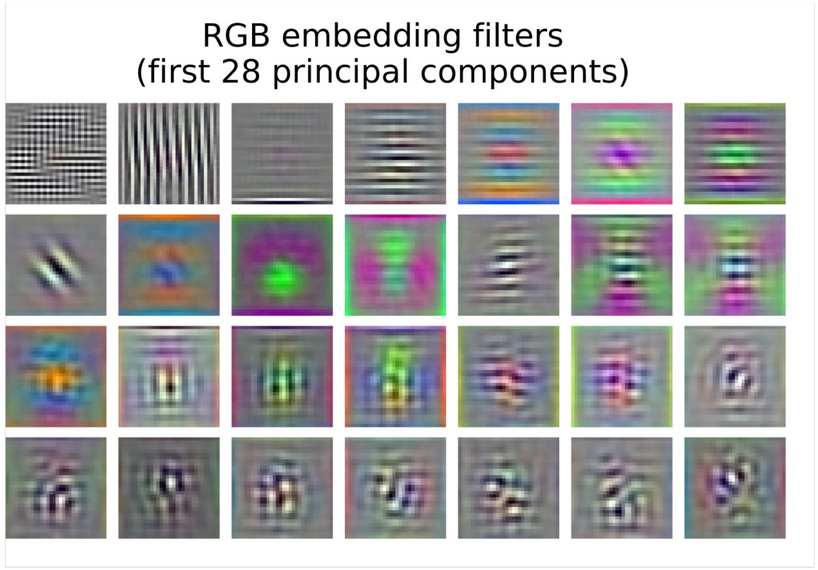


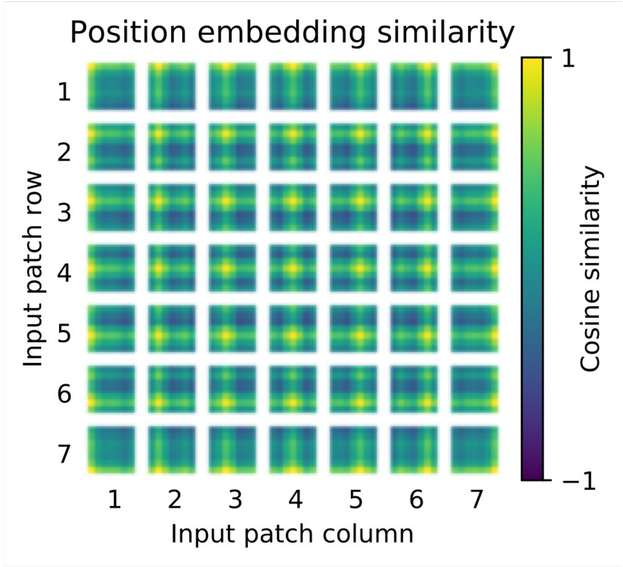
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

2020, A Dosovitskiy, L Beyer, A Kolesnikov, D Weissenborn, X Zhai, T Unterthiner, M Dehghani, M Minderer, G Heigold, S Gelly, J Uszkoreit, N Houlsby

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Details of Vision Transformer model variants.





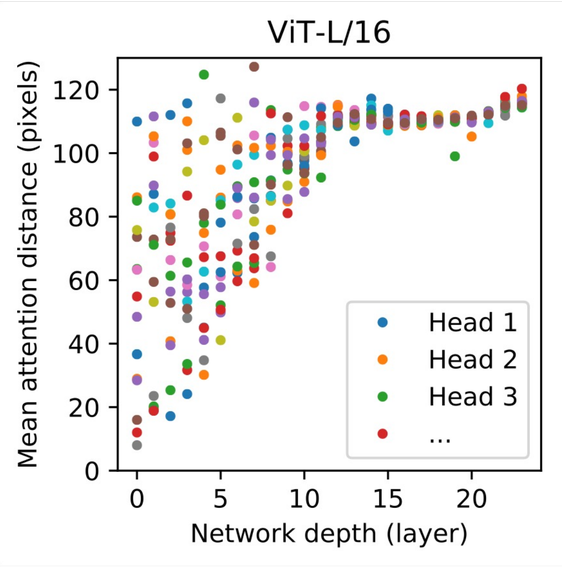


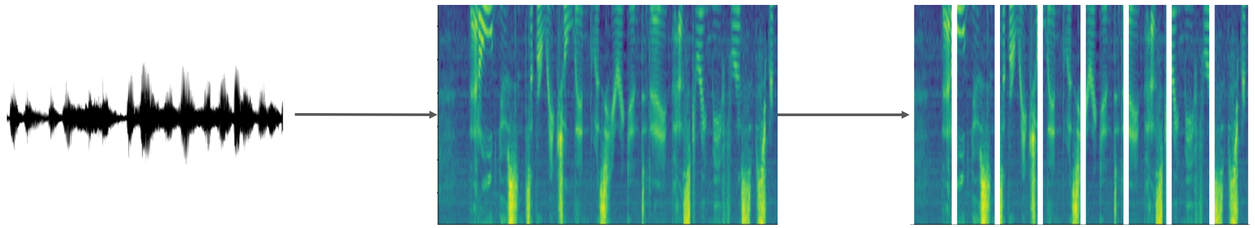
Figure 7: **Left:** Filters of the initial linear embedding of RGB values of ViT-L/32. **Center:** Similarity of position embeddings of ViT-L/32. Tiles show the cosine similarity between the position embedding of the patch with the indicated row and column and the position embeddings of all other patches. **Right:** Size of attended area by head and network depth. Each dot shows the mean attention distance across images for one of 16 heads at one layer. See Appendix D.7 for details.

Speech

Conformer: Convolution-augmented Transformer for Speech Recognition

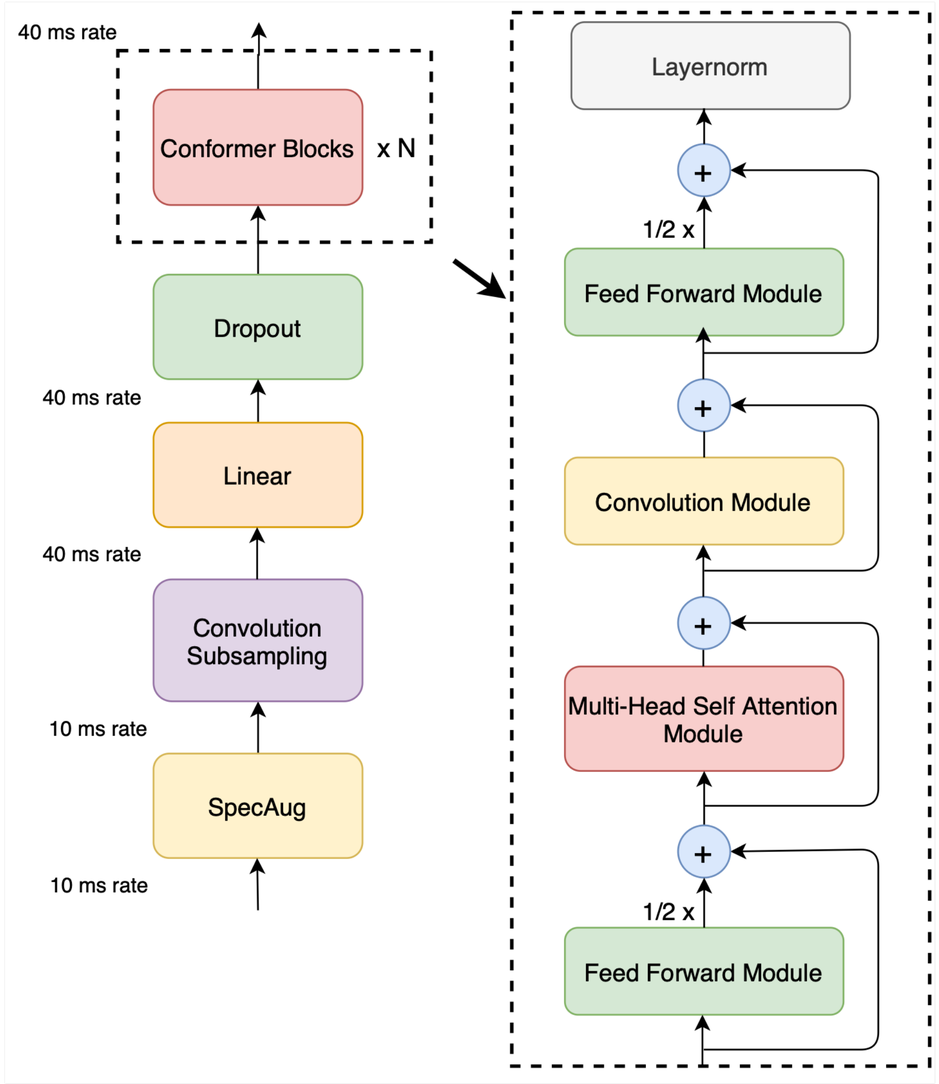
2020, A Gulati, J Qin, C-C Chiu, N Parmar, Y Zhang, J Yu, W Han, S Wang, Z Zhang, Y Wu, R Pang

Largely the same story as in computer vision.
But with spectrograms instead of images.



Conformer adds a third type of block using convolutions, and slightly reorder blocks, but overall very transformer-like.

Exists as encoder-decoder variant, or as encoder-only variant with CTC loss.



Transformer image source: "Attention Is All You Need" paper

Robust Speech Recognition via Large-Scale Weak Supervision - Whisper Model from OpenAI

Multitask training data (680k hours)

English transcription

"Ask not what your country can do for ···"

Ask not what your country can do for ···

Any-to-English speech translation

"El rápido zorro marrón salta sobre ···"

The quick brown fox jumps over ···

Non-English transcription

"언덕 위에 올라 내려다보면 너무나 넓고 넓은 ···"

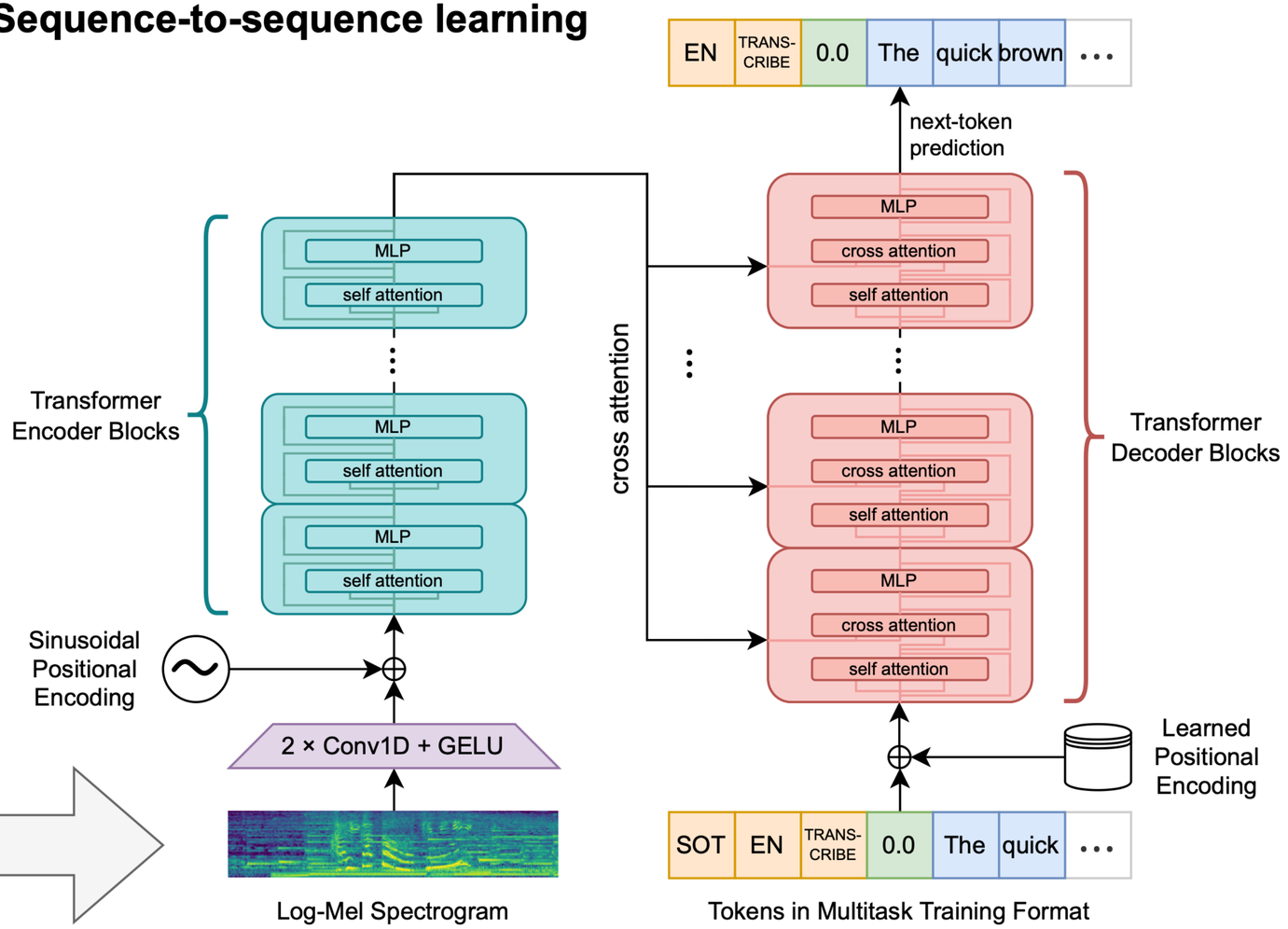
언덕 위에 올라 내려다보면 너무나 넓고 넓은 ···

No speech

(background music playing)

Ø

Sequence-to-sequence learning



- Whisper Model from OpenAI

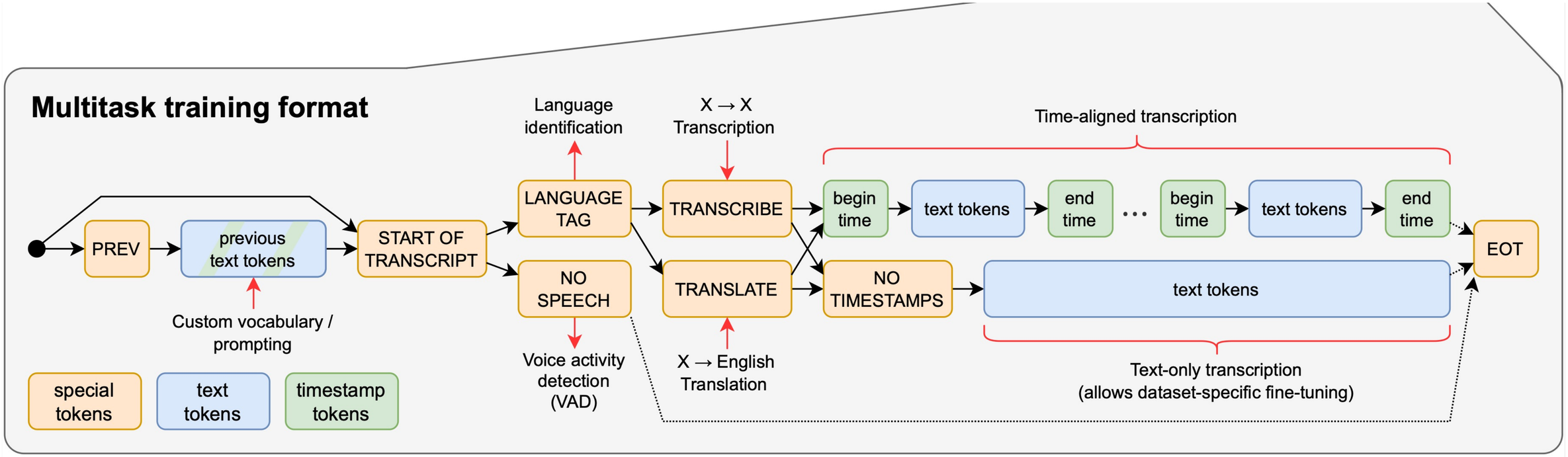
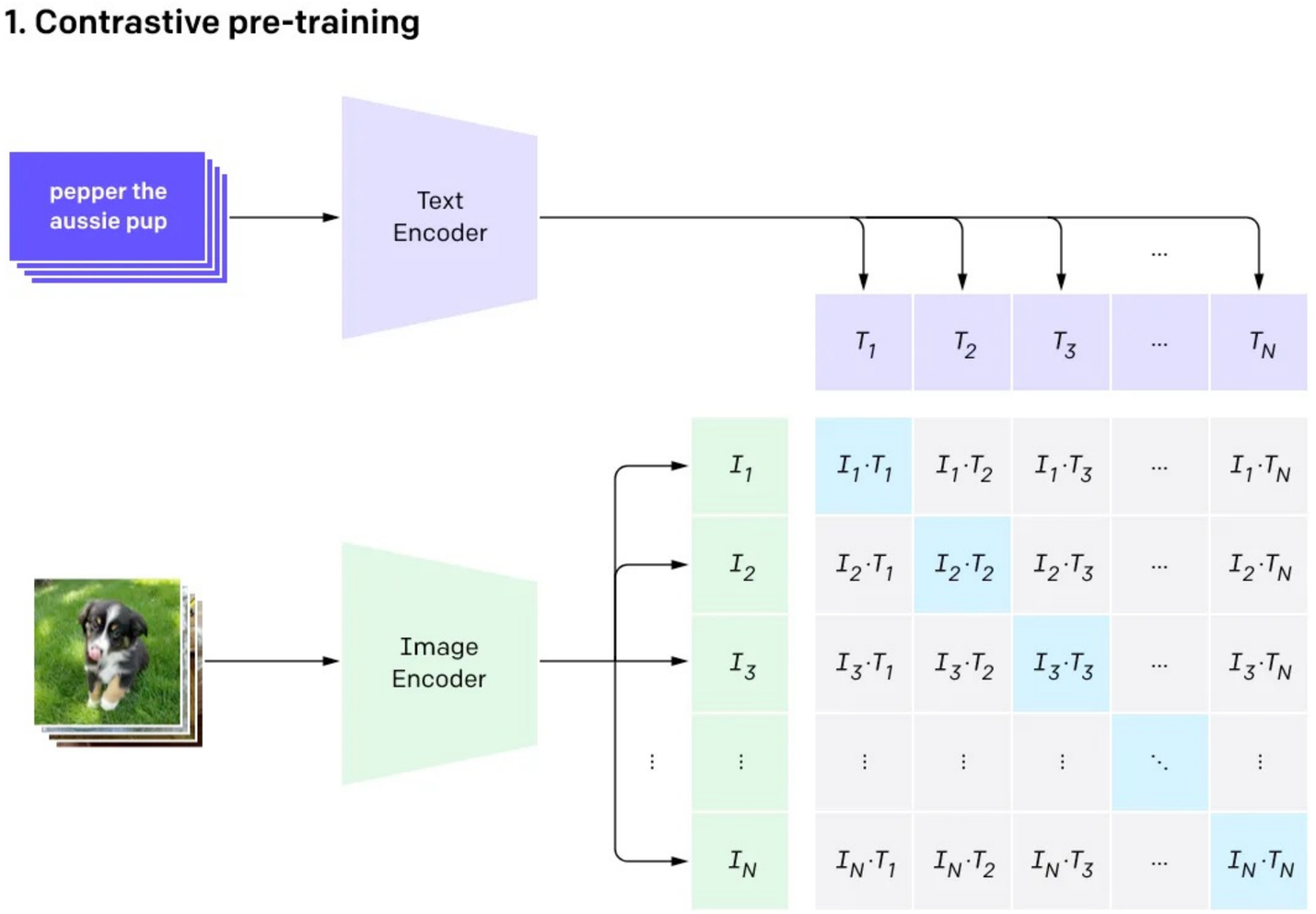


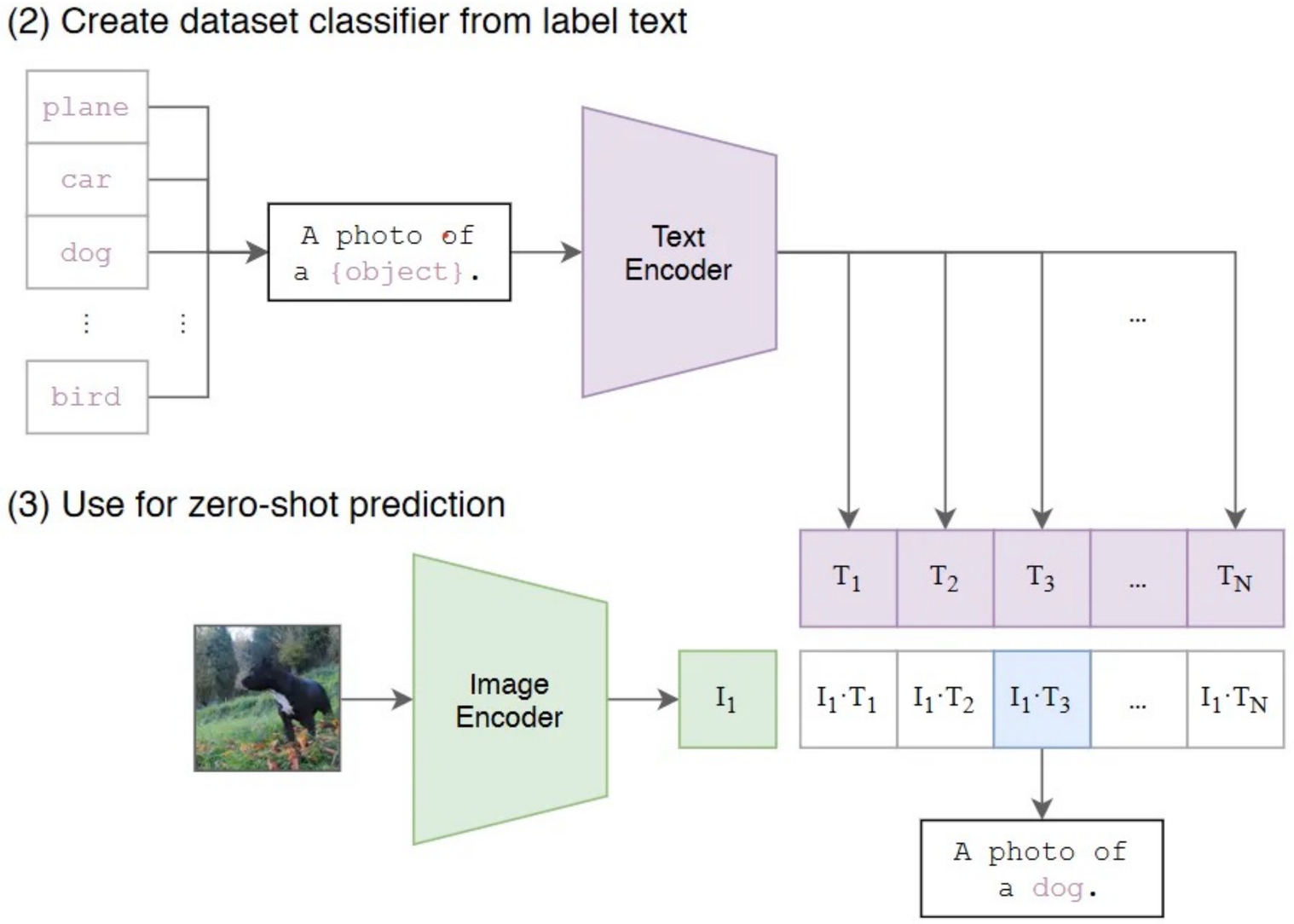
Figure 1. **Overview of our approach.** A sequence-to-sequence Transformer model is trained on many different speech processing tasks, including multilingual speech recognition, speech translation, spoken language identification, and voice activity detection. All of these tasks are jointly represented as a sequence of tokens to be predicted by the decoder, allowing for a single model to replace many different stages of a traditional speech processing pipeline. The multitask training format uses a set of special tokens that serve as task specifiers or classification targets, as further explained in Section 2.3.

Vision and Language Models

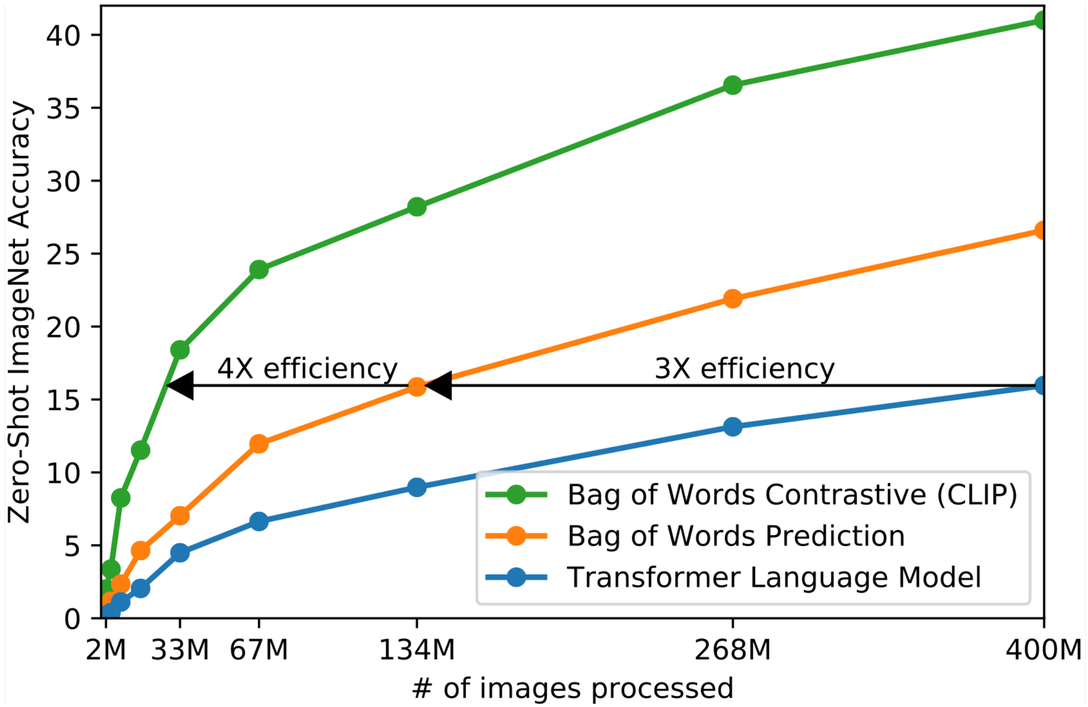
CLIP Model - OpenAI

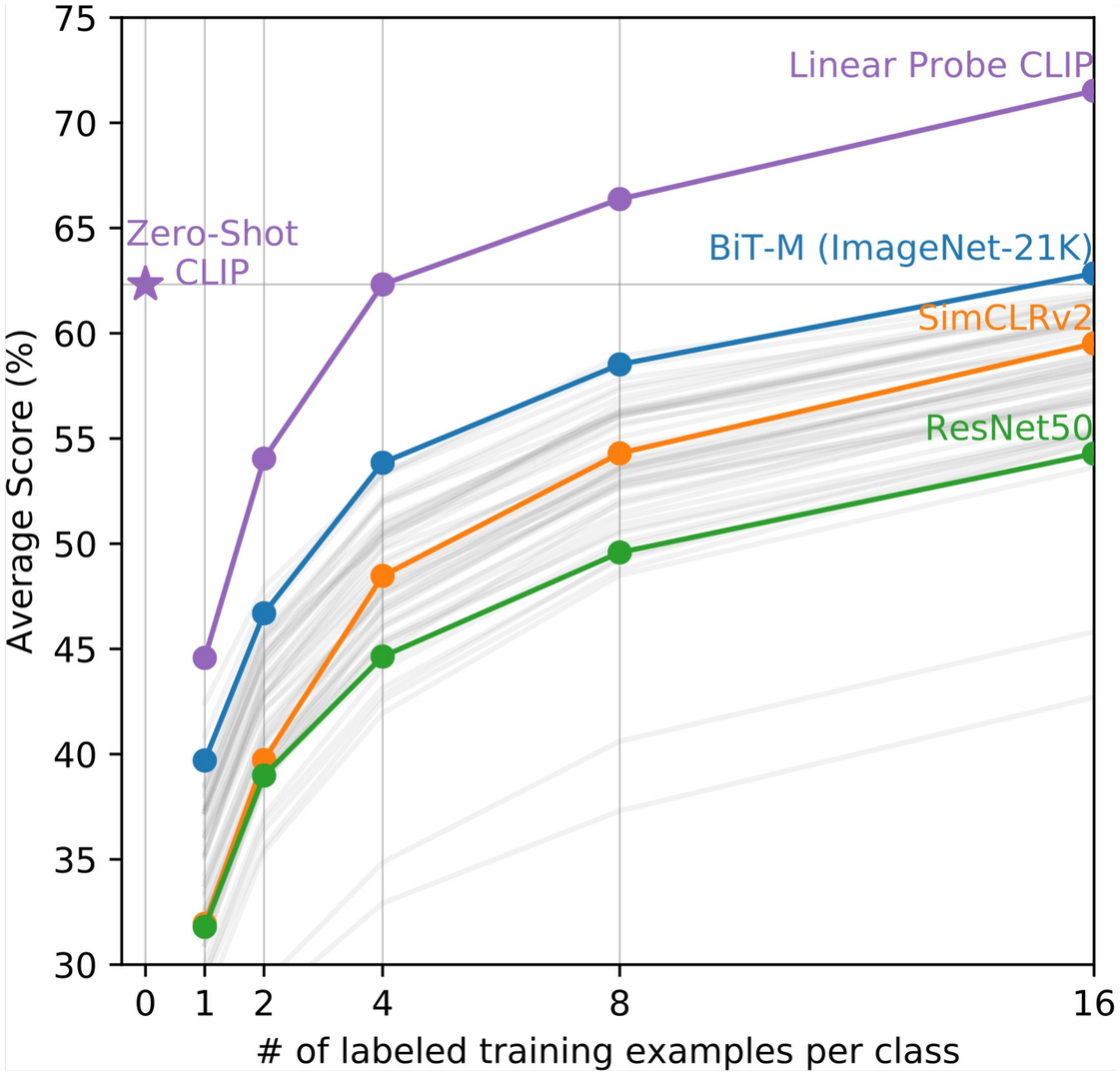


CLIP Model - OpenAI



CLIP Model - OpenAI



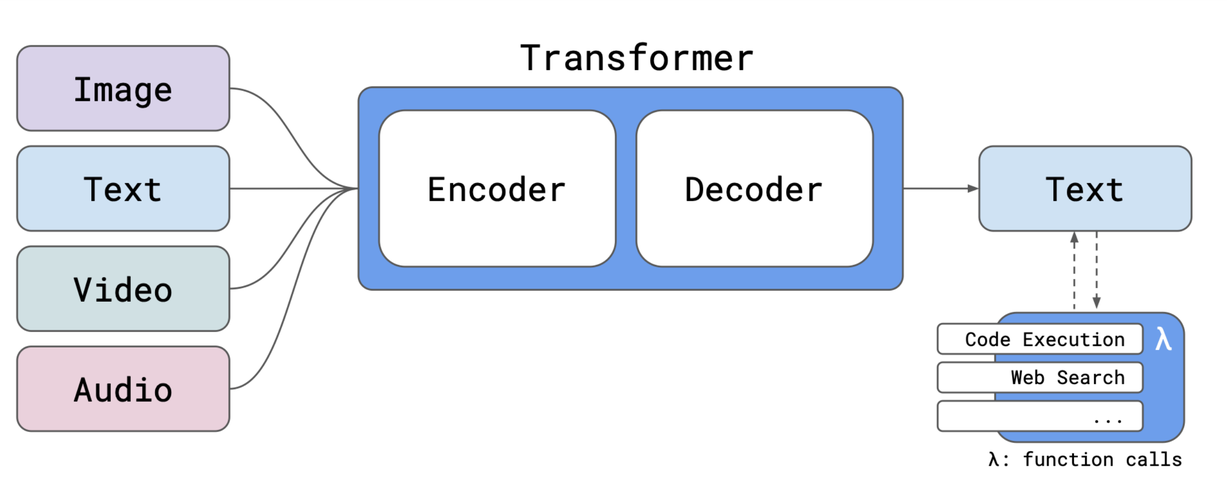


The Transformer's

Unification of communities

Reka Model - One of the recent GPT competitors

Figure 2: **Architectural overview for Reka Core, Flash & Edge models:** a modular encoder-decoder transformer supporting multimodal input (image, text, video & audio). The text output can invoke function calls, such as web search and code execution, then return the results.



Similar approach is followed by most large models

Reka Model - One of the recent GPT competitors

Table 5: Comparisons of our Reka Flash and Reka Core against other frontier models. Dashes (−) refer to either model not supporting modality or unavailable benchmark scores.

Model / Eval	Reka Core v0.5	Reka Flash v1.5	GPT-4	Claude 3 Opus	Claude 3 Sonnet	Gemini Ultra	Gemini Pro 1.5
MMLU (*Knowledge*)	83.2	75.9	86.4	86.8	79.0	83.7	81.9
GSM8K (*Reasoning*)	92.2	85.8	92.0	95.0	92.3	94.4	91.7
HumanEval (*Coding*)	76.8	72.0	76.5	84.9	73.0	74.4	71.9
GPQA (*main*) (*Hard QA*)	38.2	34.0	38.1	50.2	39.1	35.7	41.5
MMMU (*Image QA*)	56.3	53.3	56.8	59.1	53.1	59.4	58.5
VQAv2 (*Image QA*)	78.1	78.4	77.2	–	–	77.8	73.2
Perception-test (*Video QA*)	59.3	56.4	–	–	–	54.7	51.1[3]


Closed-source vs. Open-weight models (Arena ELO)
Closed-source models
Open-weight models
Arena ELO
Release Date
1250
1200
1150
1100
1050
Mar 23
Apr 23
May 23
Jun 23
Jul 23
Aug 23
Sep 23
Oct 23
Nov 23
Dec 23
Jan 24
Feb 24
Mar 24
Apr 24
GPT-4
GPT-4
GPT-4-Turbo
Claude 3 Opus
Gemini Pro
Claude 3 Sonnet
Command R+
Claude 3 Haiku
GPT-4
GPT-4
Mistral Large
Qwen1.5 72B
Mistral Medium
Command R
Claude 1
Qwen1.5-32B-Chat
Claude 2
GPT-3.5 Turbo
Starling-LM-7B-beta
Mixtral Instruct 8x7B
GPT-3.5 Turbo
WizardLM 70b
Yi-34B
DBRX Instruct
OpenChat-3.5
Vicuna 33B
Llama-2-70b-chat
Gemma-1.1-7B-it
DeepSeek-LLM-67B-Chat
OpenHermes-2.5-Mistral-7B
SOLAR-10.7B-Instruct-v1.0
Llama-2-13b-chat
MPT-30B-chat
Vicuna-13B-v1.5
Llama-2-7b-chat

A note on

Efficient Transformers

A note on Efficient Transformers

The self-attention operation complexity is $O(N^2)$ for sequence length N.

We'd like to use large N:

- Whole articles or books
- Full video movies
- High resolution images

Many O(N) approximations to the full self-attention have been proposed in the past two years.

Unfortunately, none provides a clear improvement.
They always trade-off between speed and quality.

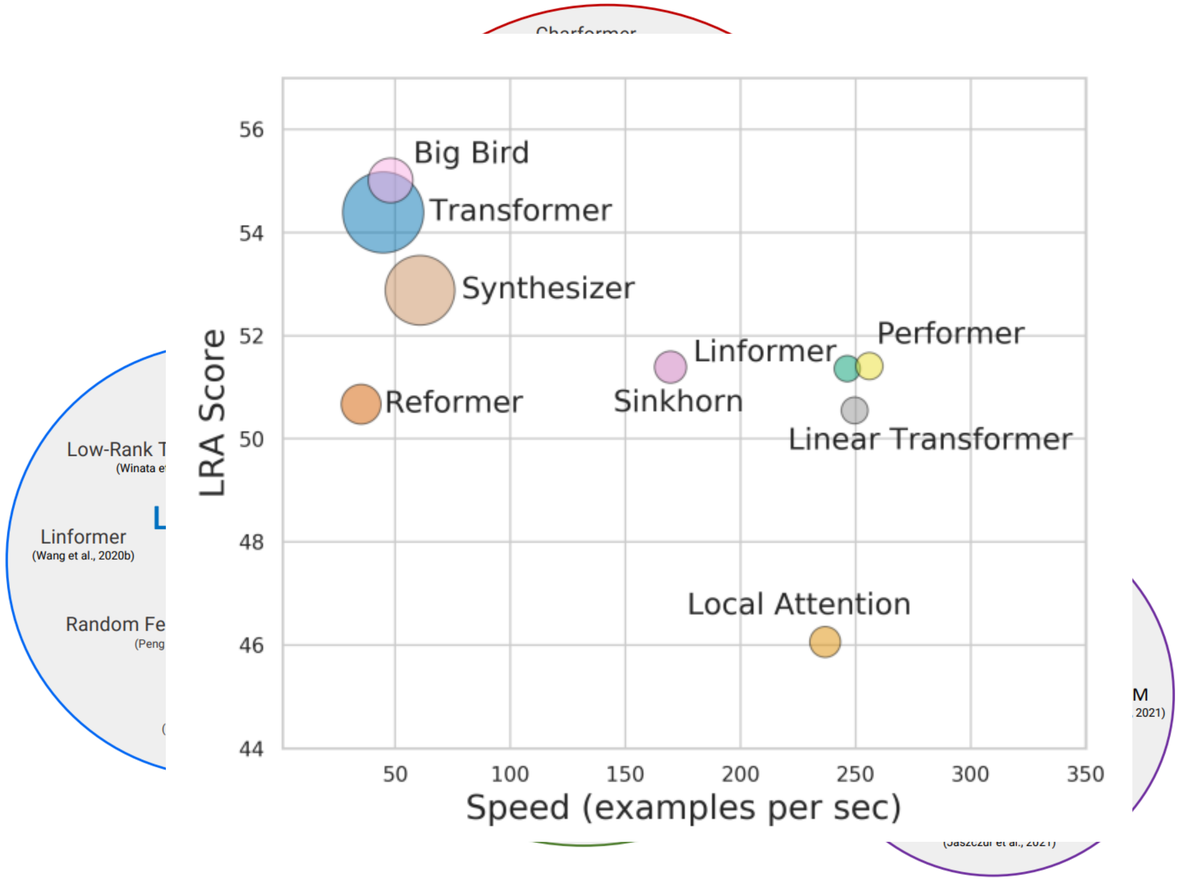


Figure 2: Taxonomy of Efficient Transformer Architectures.

Based on "Efficient Transformers: A Survey" by Yi Tay, Mostafa Dehghani, Dara Bahri, Donald Metzler
and "Long Range Arena: A Benchmark for Efficient Transformers" by Y Tay, M Dehghani, S Abnar, Y Shen, D Bahri, P Pham, J Rao, L Yang, S Ruder, D Metzler

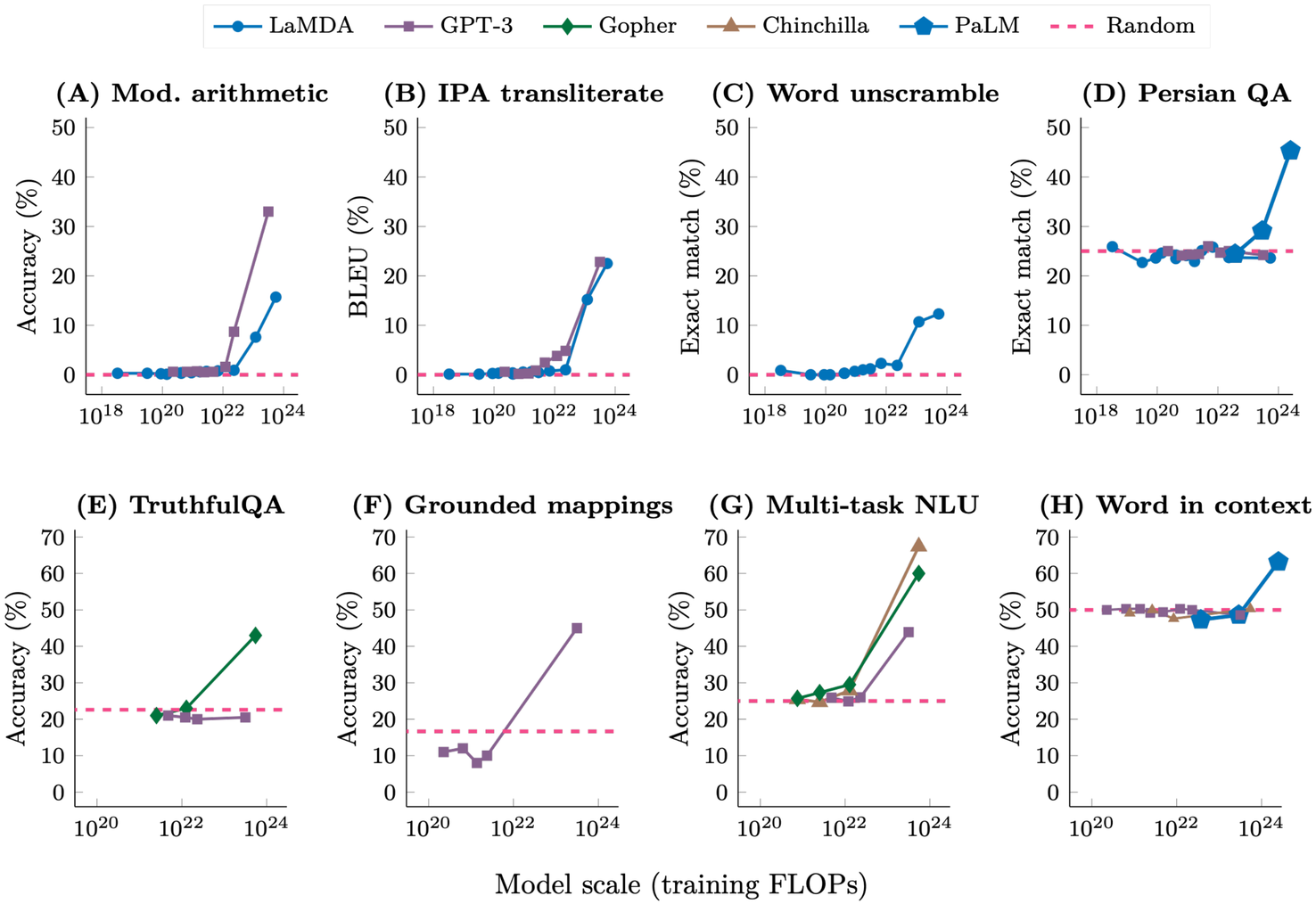
Slide credit Lucas Beyer

Emergence in LLMs

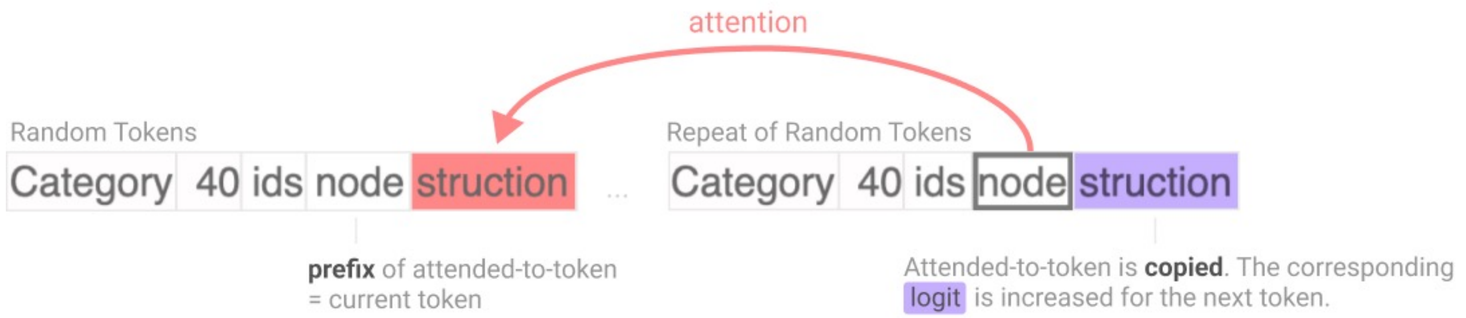
Emergent Abilities of Large Language Models

Emergence is when quantitative changes in a system result in qualitative changes in behavior.

An ability is emergent if it is not present in smaller models but is present in larger models.



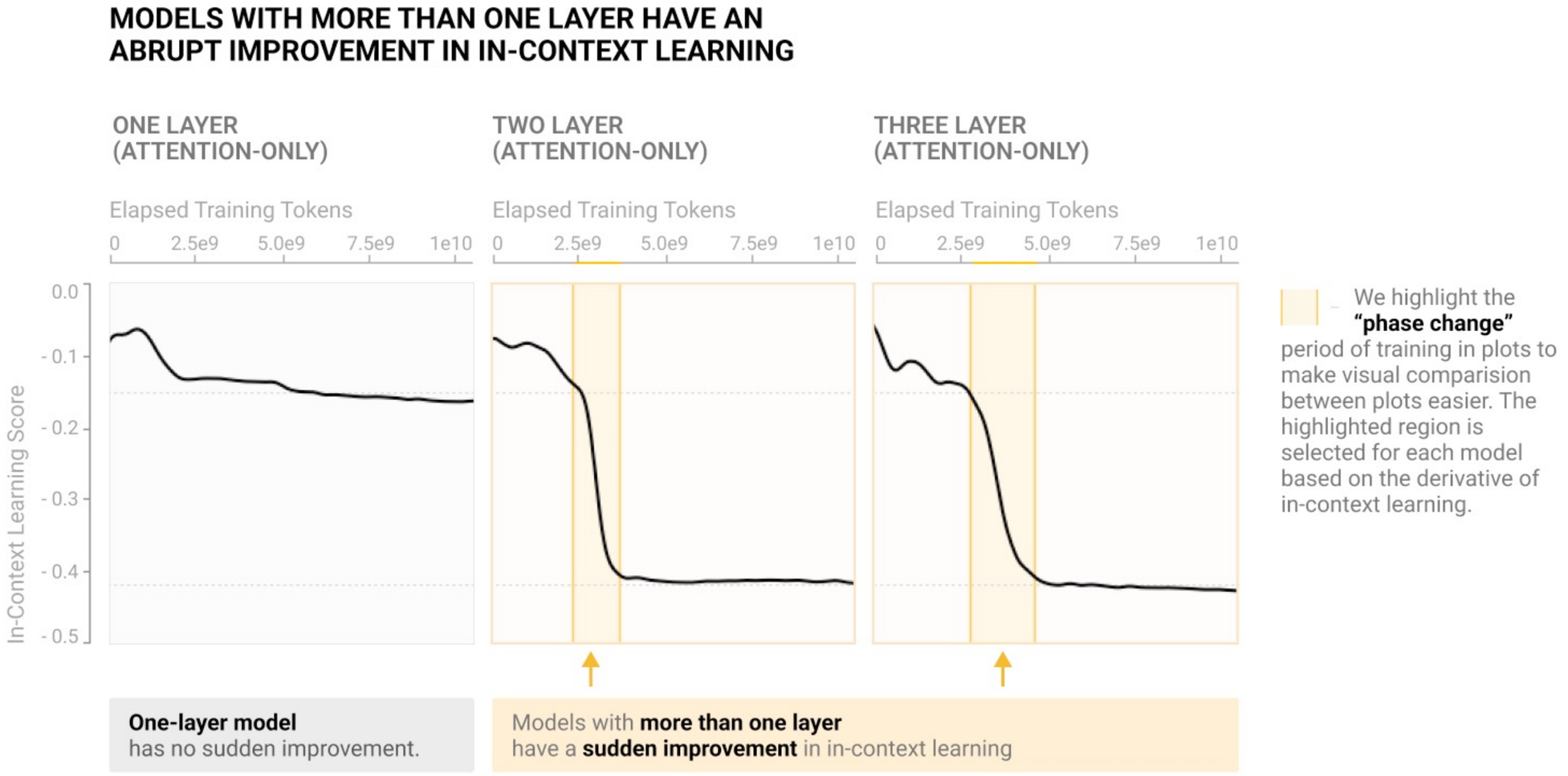
In-context Learning and Induction Heads



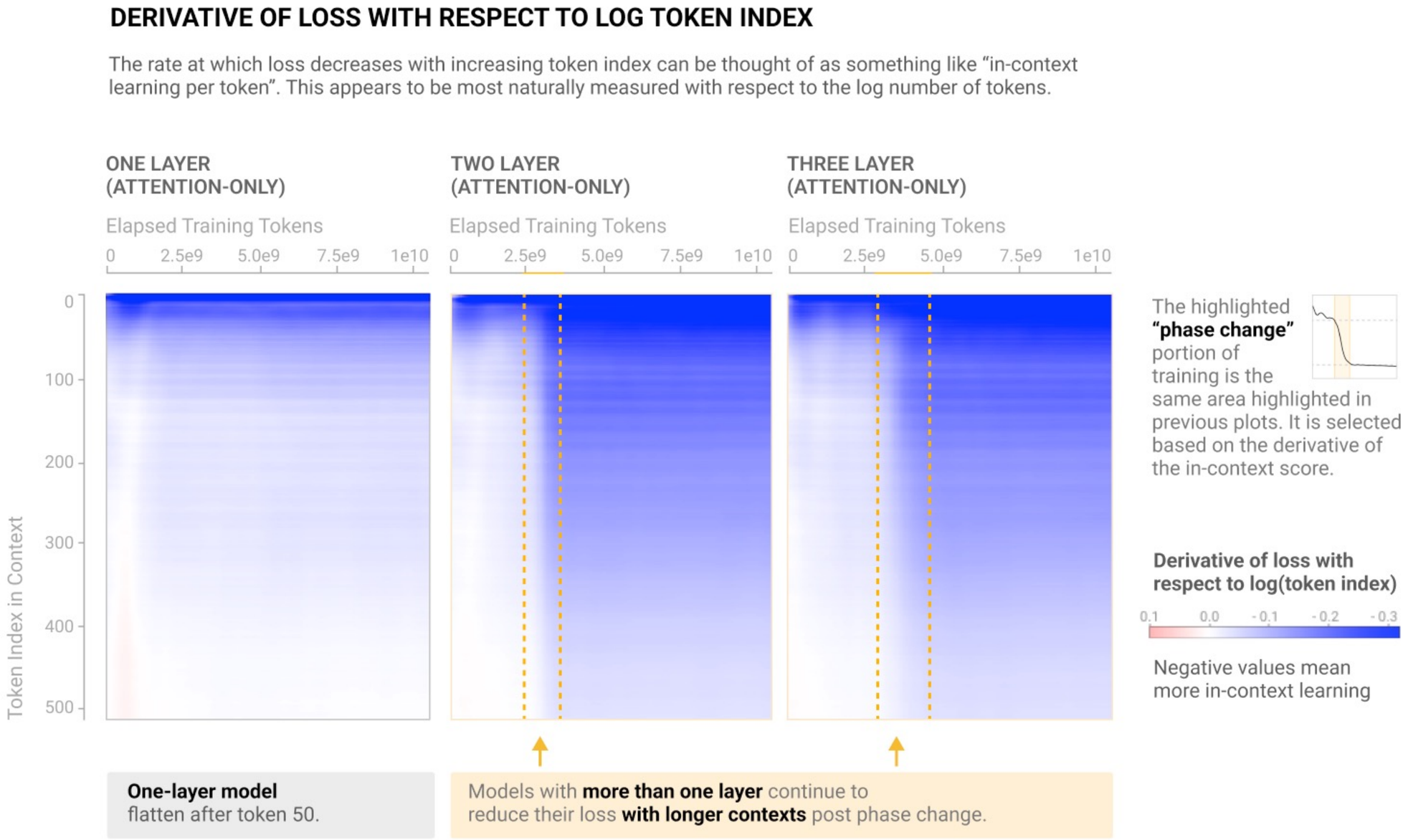
“We observed that induction heads could be seen as a kind of "in-context nearest neighbor" algorithm”

https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html

In-context Learning and Induction Heads



In-context Learning and Induction Heads



References

- 3Blue1Brown - Video 1, Video2
- Andrej Karpathy - NanoGPT implementation, Video Tutorial


?
That’s all folks
QUESTIONS?