IS883: Synthesizing Digital Efforts

Mohannad Elhamod

BOSTON UNIVERSITY

Boston University Questrom School of Business

The Transformer

How to solve the forgetting problem…

BOSTON UNIVERSITY

Boston University Questrom School of Business

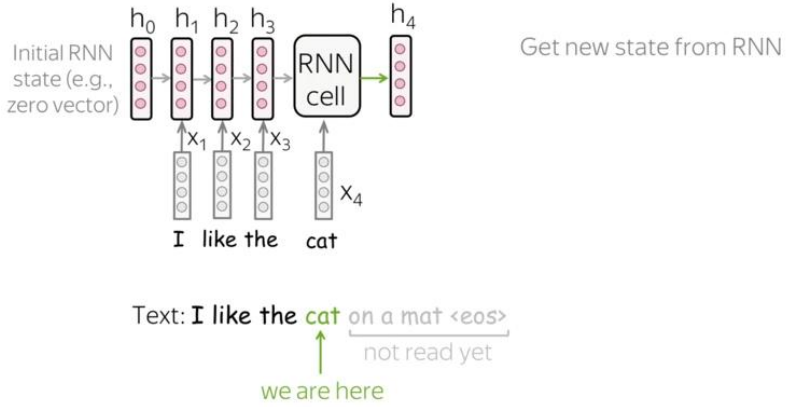
Last time on IS883…

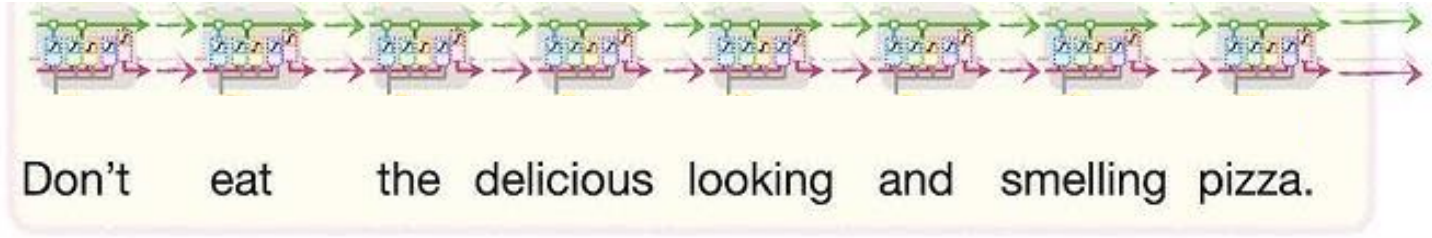
Catastrophic forgetting… !

How do we solve it?

- We started with having one memorization path…
- Then we added a second path (long memory and short memory).
- Next…?

Lena-voita





StatQuest

Attention!

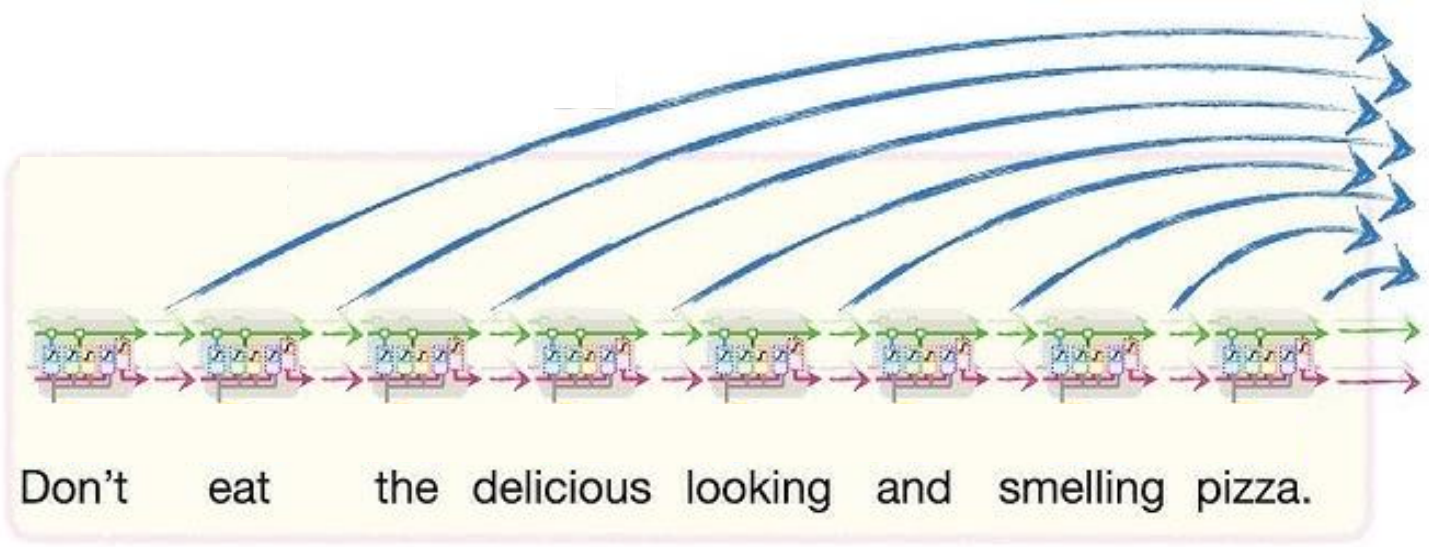
Instead, let's get information from all past cells!

Published as a conference paper at ICLR 2015

NEURAL MACHINE TRANSLATION
BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

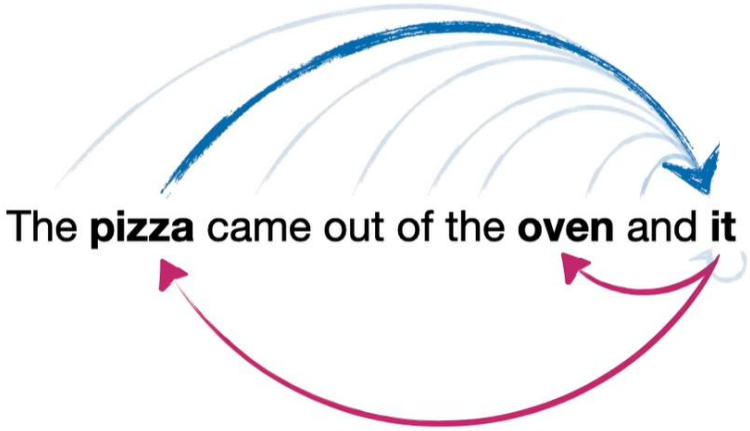


StatQuest

Attention!

A bit more detail:

- What are we doing with this information?
  - We are trying to find the similarity between a previous state and the current state.
  - We want to pay attention to the right part of the sentence.

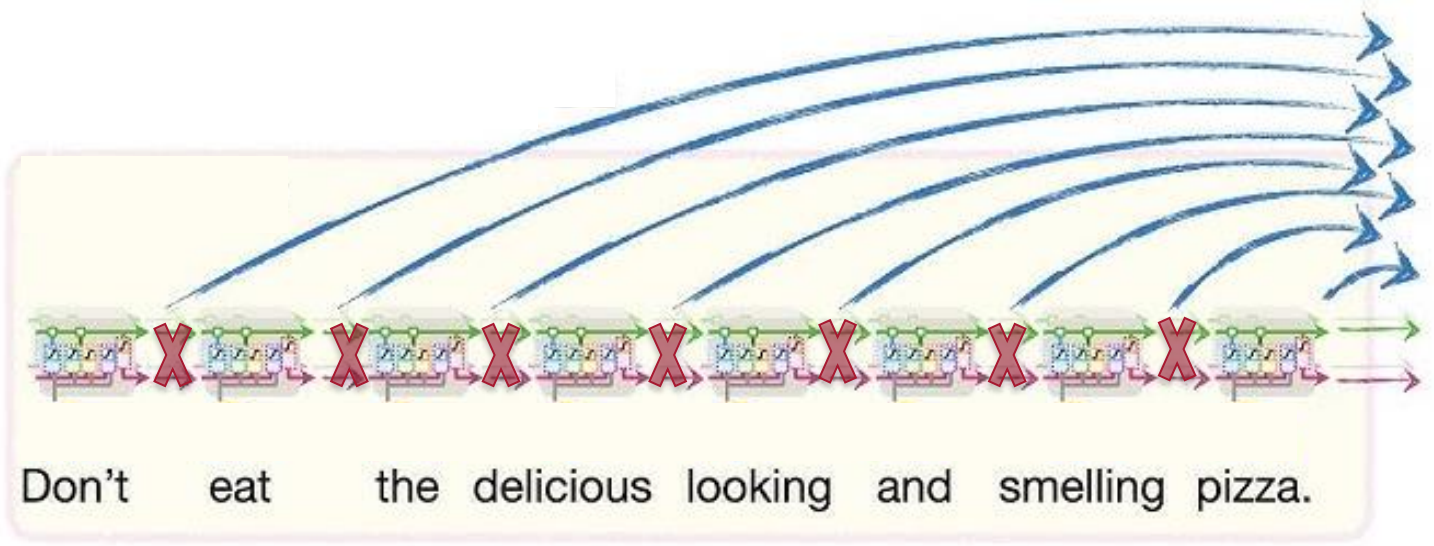


StatQuest

Attention!

What do you think the next step is?

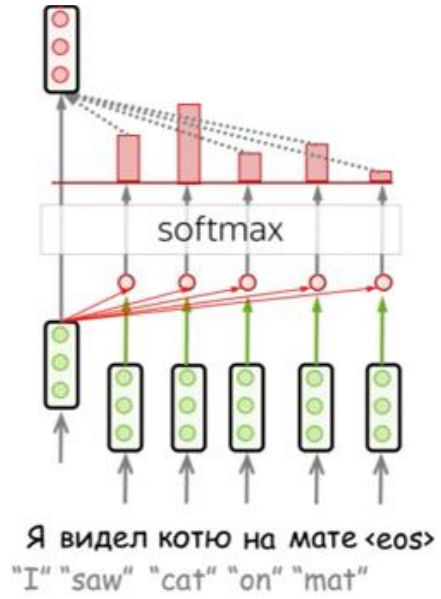
- We don’t really need the long short-term part…
- We only need to calculate the **attention** to each token.



StatQuest

Attention!

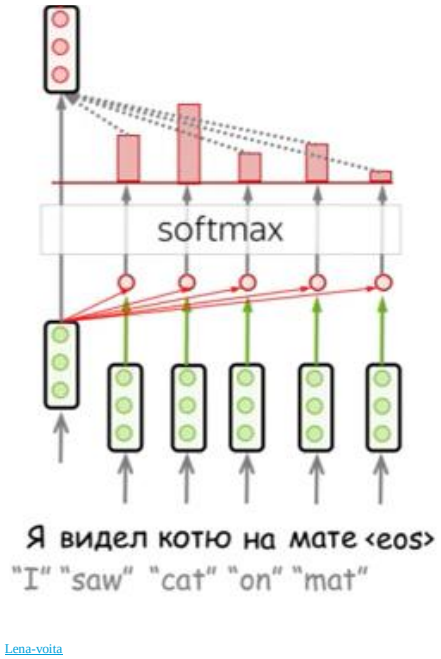
- If each state (i.e., embedding) is a vector, then similarity is calculated by?...
  - Dot product!



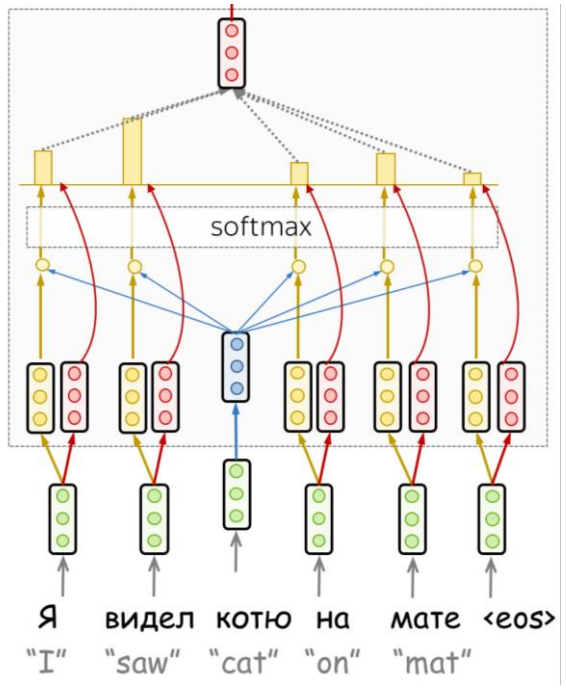
Lena-voita

Attention!

- But this only captures embedding similarity. What if I want a more nuanced notion of similarity?



Lena-voita

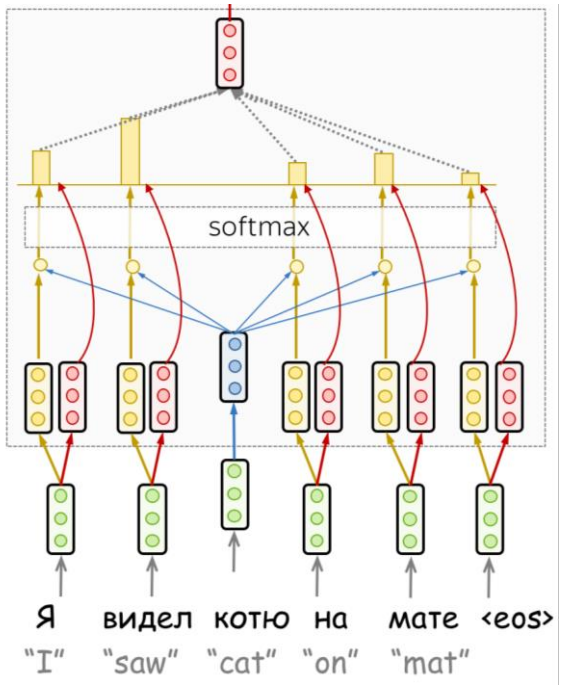


Attention!

- But this only captures embedding similarity. What if I want a more nuanced notion of similarity?

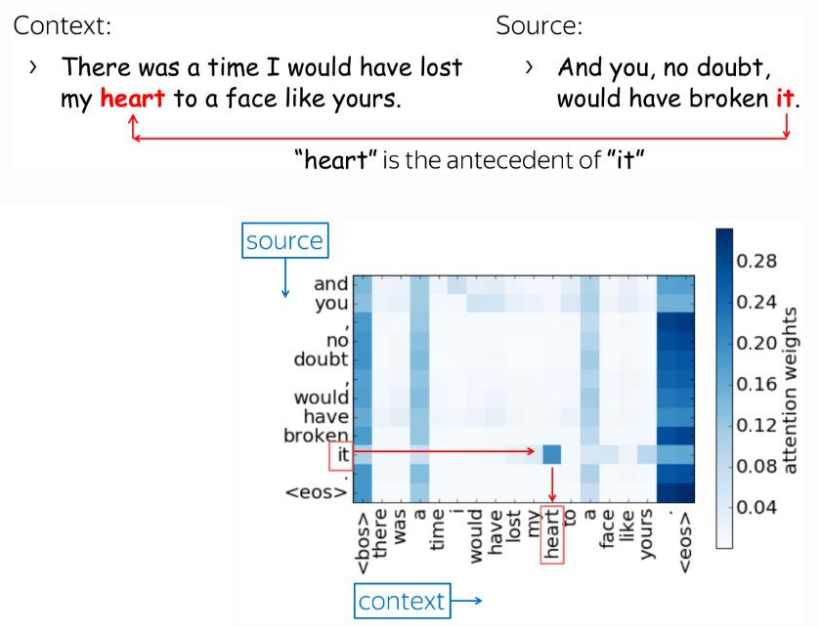
  - **Query: Attention from…**
  - **Key: Attention to…**
    - Their multiplication gives me the **attention scores**.
  - **Values : used as embeddings weighted by these scores.**

Lena-voita

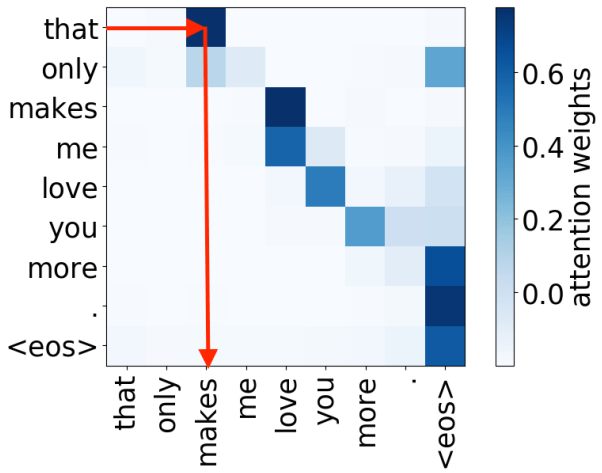


Attention!

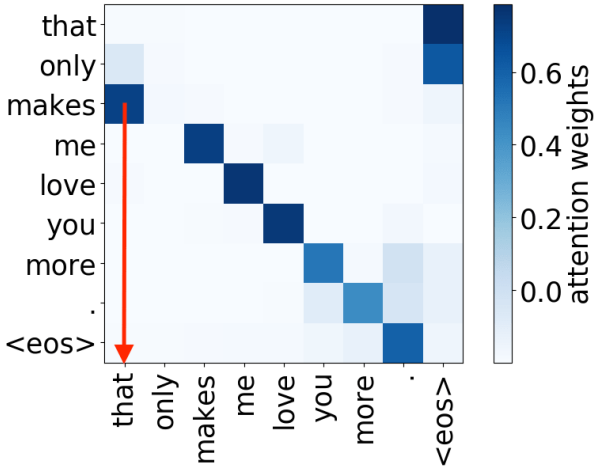
- But what if I want to capture more than one notion of “a nuance”?



Lena-voita

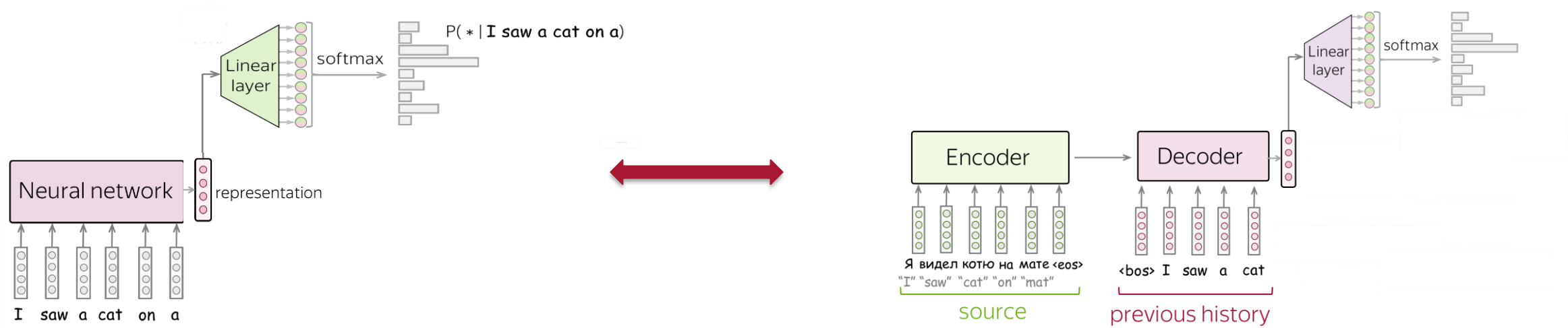


Subject -> verb



Verb -> subject

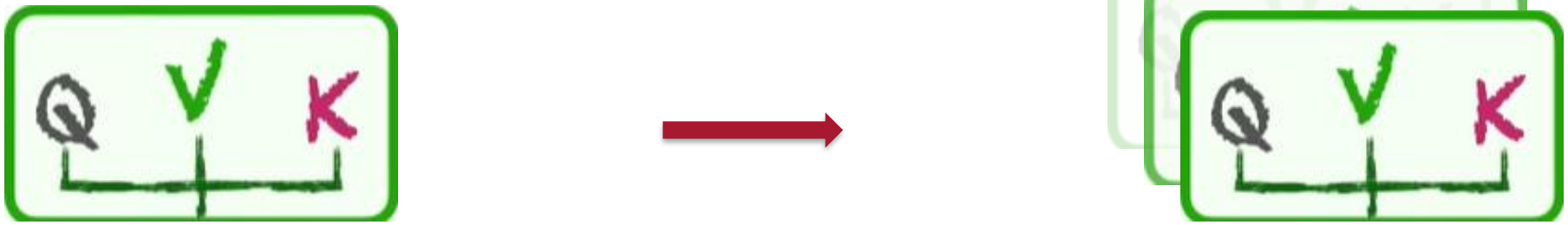
An encoder-decoder approach



Lena-voita

Attention!

- But what if I want to capture more than one notion of “a nuance”…



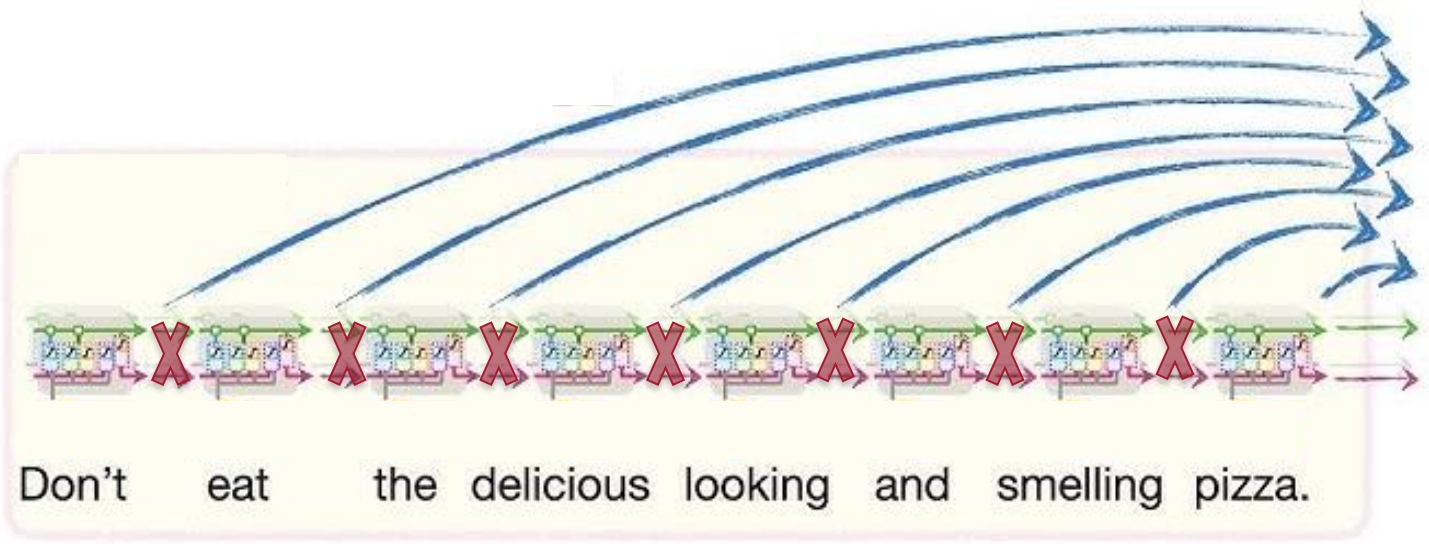
StatQuest

Multi-head attention

Issues? (1)

What issues did we cause by removing the sequential information?

Word order!

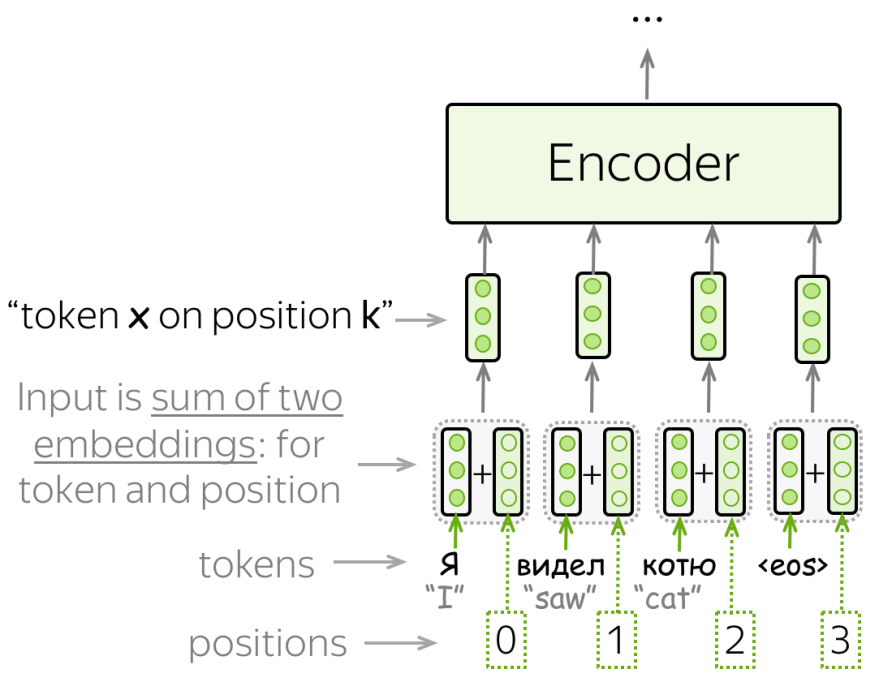


StatQuest

Issues? (1)

We need to keep track of word order in the sentenc

- **Positional encoding.**

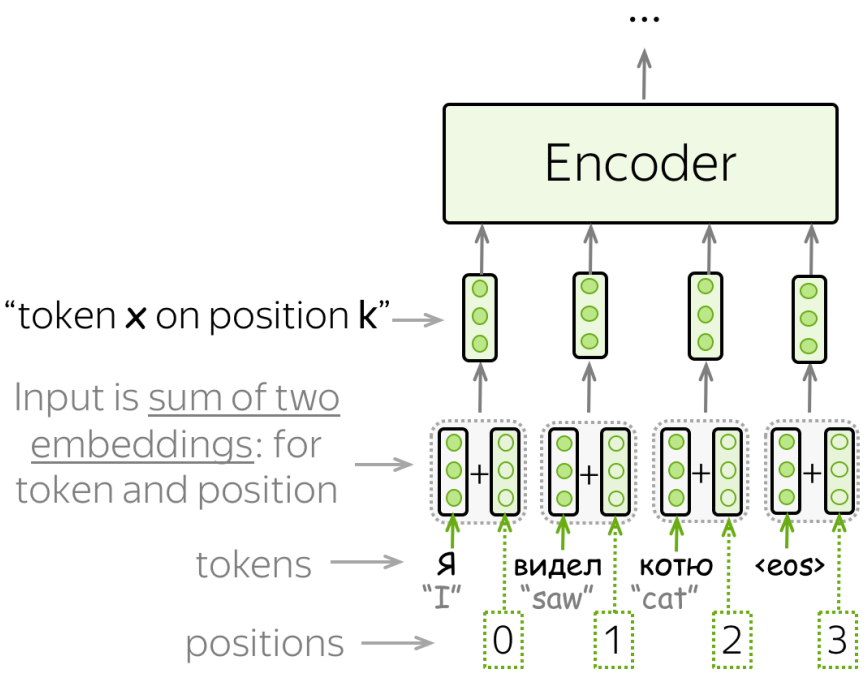


Lena-voita

Issues? (2)

- Word2vec was trained on a its own corpus (Google News) in advance. Is this a problem? Can we do better?

Learn the word embeddings!

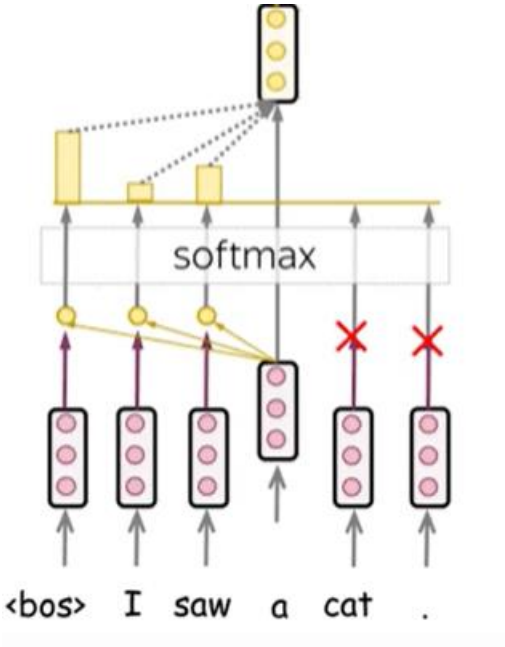


Lena-voita

Issues? (3)

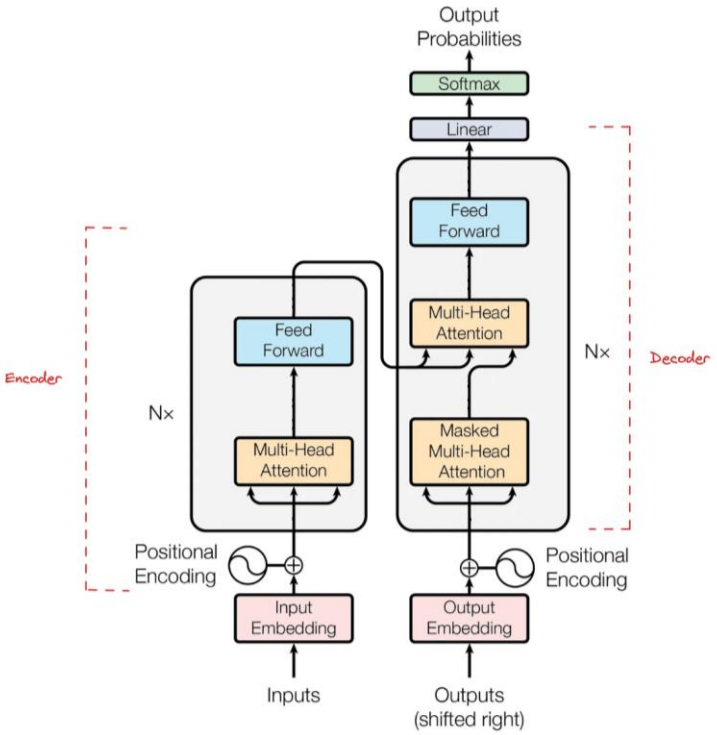
- When we are training, the decoder should only look at the past

Use “masked” attention!



Lena-voita

The Transformer is born!



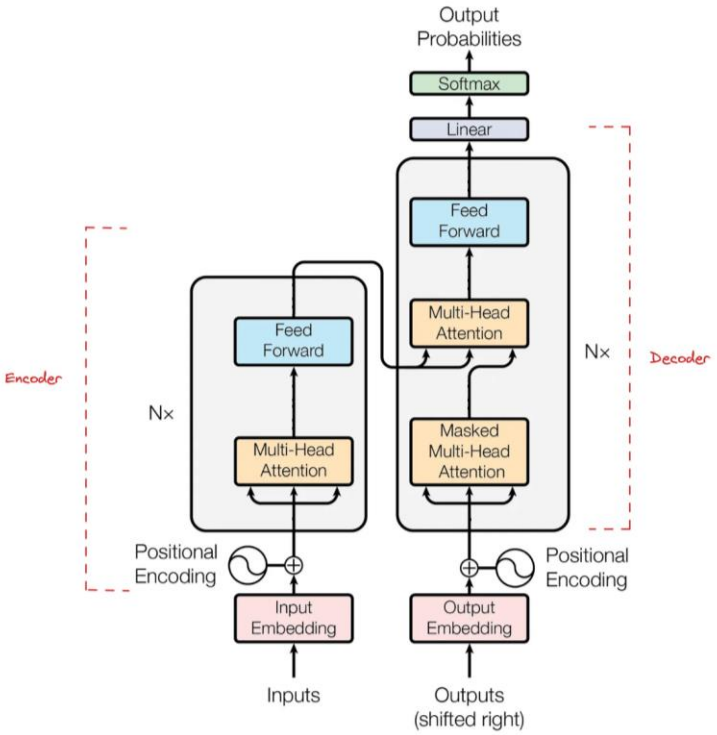
Javinkarla

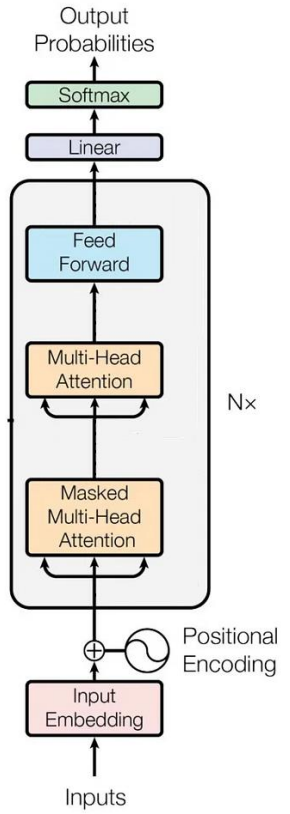
12 Jun 2017

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

GPT is born!


Output Probabilities
Softmax
Linear
Feed Forward
Multi-Head Attention
Masked Multi-Head Attention
Nx
Encoder
Decoder
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

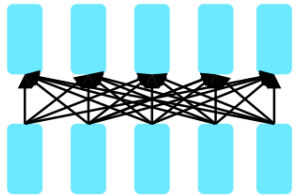

Output Probabilities
Softmax
Linear
Feed Forward
Multi-Head Attention
Masked Multi-Head Attention
Nx
Positional Encoding
Input Embedding
Inputs

Model type summary

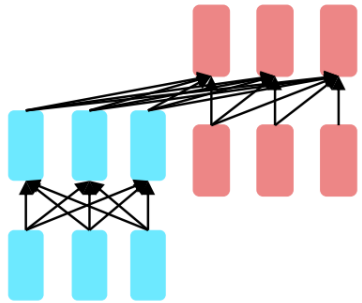
What is each one good for…

Model	Examples	Tasks
Encoder	ALBERT, BERT, DistilBERT, ELECTRA, RoBERTa	Sentence classification, named entity recognition, extractive question answering
Decoder	CTRL, GPT, GPT-2, Transformer XL	Text generation
Encoder-decoder	BART, T5, Marian, mBART	Summarization, translation, generative question answering

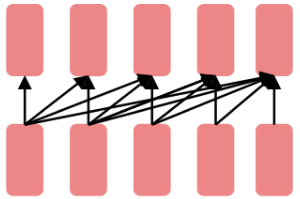
Javinkarla



Encoders



Encoder-Decoders



Decoders

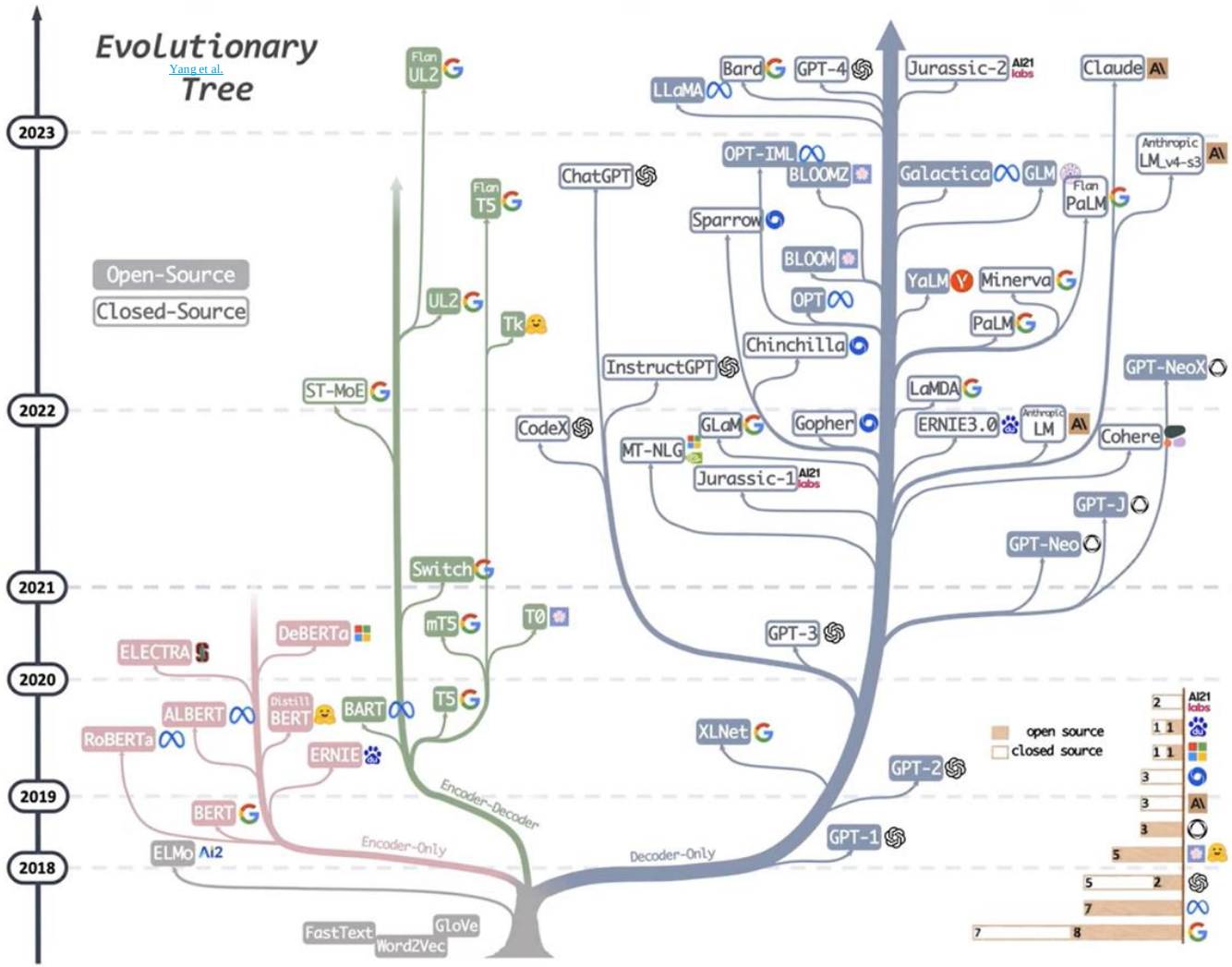
Models in the wild

BOSTON UNIVERSITY

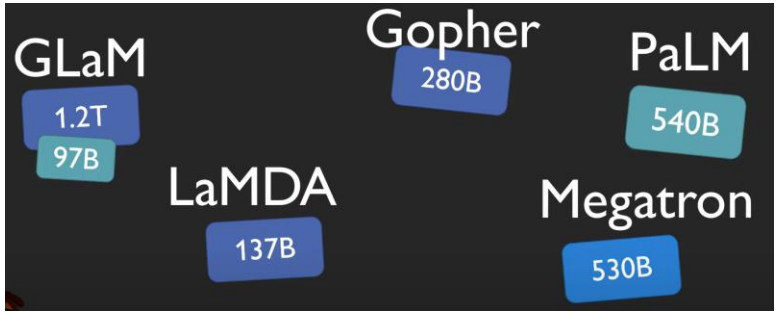
Why so many?

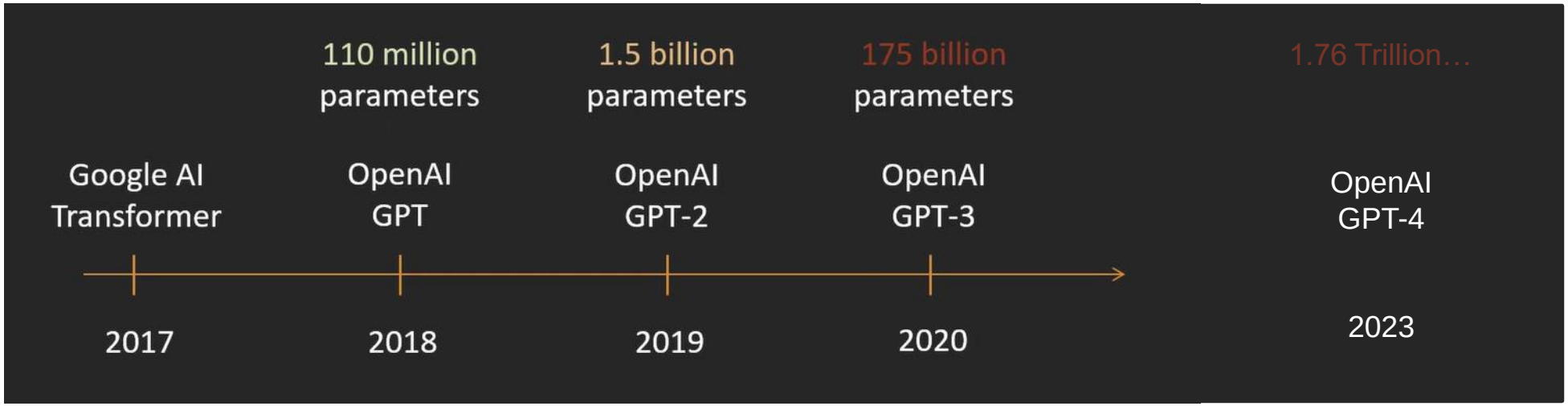
Where do the differences come from?

- Data.
- Model type and size.
- Hyperparameters (context size, embedding size,…).
- Training process (the cost function, fine-tuning, human feedback, etc.).



The GPT evolution...





AI Coffee Beak with Letitia

Book Corpus
WebText

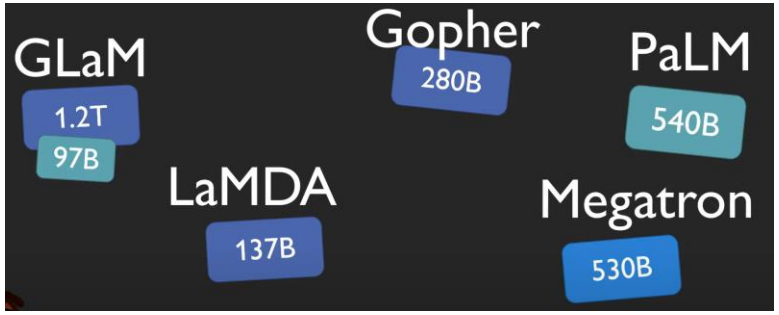
1,038 books (around 74M sentences and 1G words) of 16 different sub-genres (e.g., Romance, Historical, Adventure, etc.)

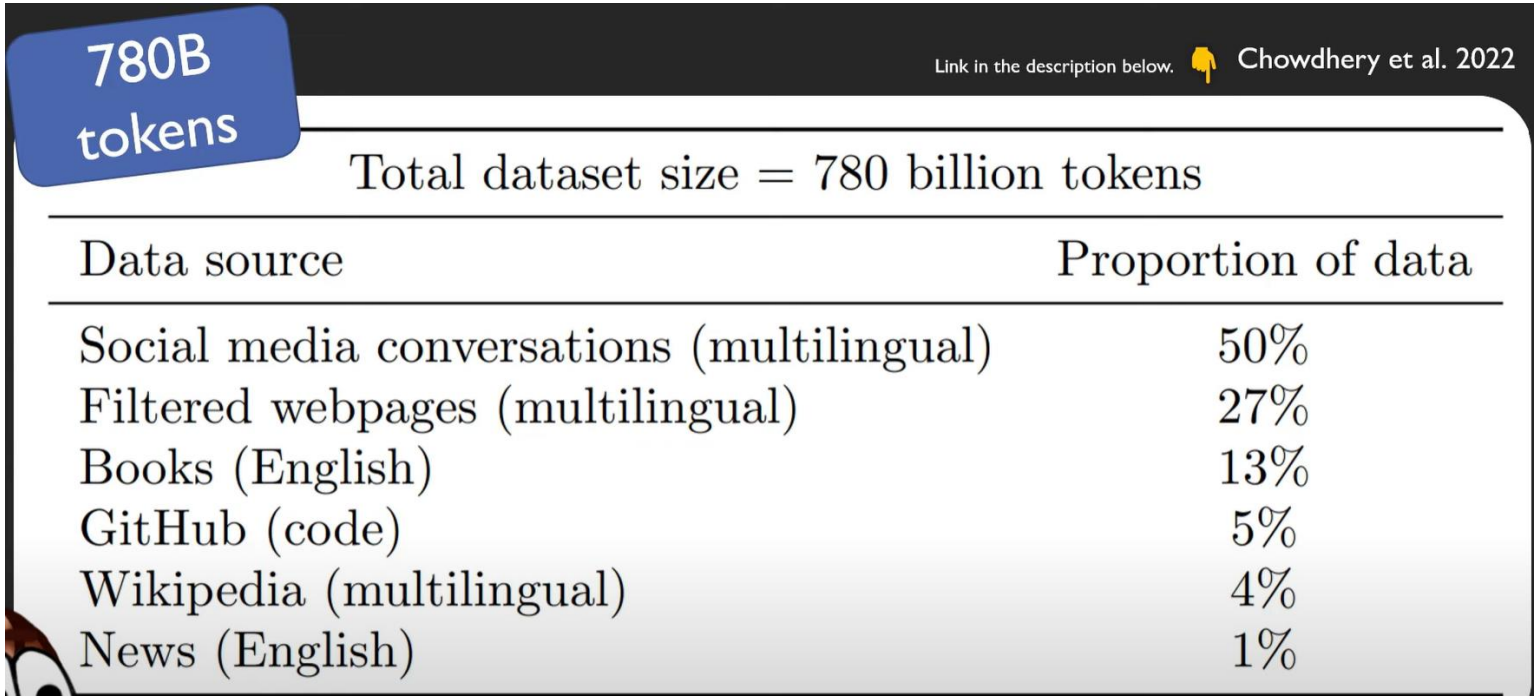
Common Crawl + ...

Over 240 billion pages. Petabytes of data.

????

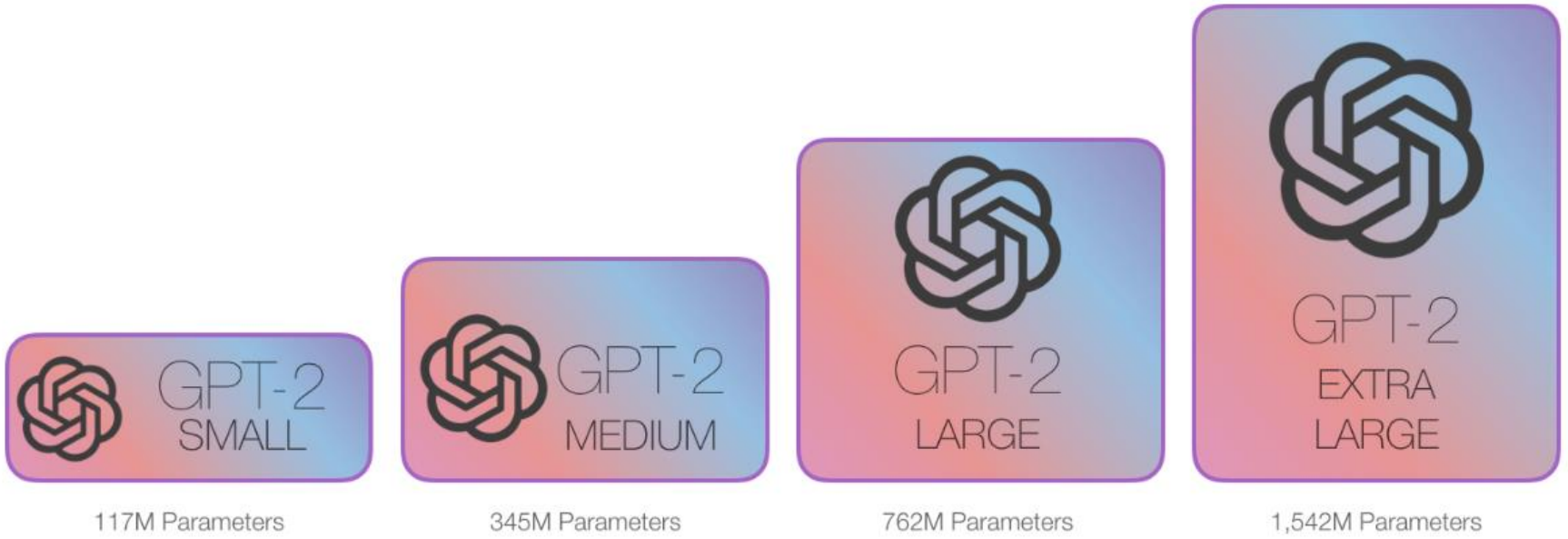
The GPT evolution...





AI Coffee Beak with Letitia

Different model sizes



Jay Alammar

Different model sizes

[OpenAI model reference]

[HuggingFace tasks]
[HuggingFace models]

PAPER	PUBLISHED	MODEL NAME IN PAPER	MODEL NAME IN API	PARAMETERS[2]
[2005.14165] Language Models are Few-Shot Learners	22 Jul 2020	GPT-3 175B	davinci	175B
		GPT-3 6.7B	curie	6.7B
		GPT-3 1B	babbage	1B
[2107.03374] Evaluating Large Language Models Trained on Code	14 Jul 2021	Codex 12B	code-cushman-001[3]	12B
[2201.10005] Text and Code Embeddings by Contrastive Pre-Training	14 Jan 2022	GPT-3 unsupervised cpt-text 175B	text-similarity-davinci-001	175B
		GPT-3 unsupervised cpt-text 6B	text-similarity-curie-001	6B
		GPT-3 unsupervised cpt-text 1.2B	No close matching model on API	1.2B
[2009.01325] Learning to summarize from human feedback	15 Feb 2022	GPT-3 6.7B pretrain	No close matching model on API	6.7B
		GPT-3 2.7B pretrain	No close matching model on API	2.7B
		GPT-3 1.3B pretrain	No close matching model on API	1.3B
[2203.02155] Training language models to follow instructions with human feedback	4 Mar 2022	InstructGPT-3 175B SFT	davinci-instruct-beta	175B
		InstructGPT-3 175B	No close matching model on API	175B
		InstructGPT-3 6B	No close matching model on API	6B
		InstructGPT-3 1.3B	No close matching model on API	1.3B

How much training does it take?

2 example models

GPT-3 (2020)

50,257 vocabulary size
2048 context length
175B parameters
Trained on 300B tokens

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}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

Training: (rough order of magnitude to have in mind)

- O(1,000 - 10,000) V100 GPUs
- O(1) month of training
- O(1-10) $M

LLaMA (2023)

32,000 vocabulary size
2048 context length
65B parameters
Trained on 1-1.4T tokens

params	dimension	n heads	n layers	learning rate	batch size	n tokens
6.7B	4096	32	32	$3.0e^{-4}$	4M	1.0T
13.0B	5120	40	40	$3.0e^{-4}$	4M	1.0T
32.5B	6656	52	60	$1.5e^{-4}$	4M	1.4T
65.2B	8192	64	80	$1.5e^{-4}$	4M	1.4T

Table 2: **Model sizes, architectures, and optimization hyper-parameters.**

Training for 65B model:

- 2,048 A100 GPUs
- 21 days of training
- $5M

[Language Models are Few-Shot Learners, OpenAI 2020]
[LLaMA: Open and Efficient Foundation Language Models, Meta AI 2023]

Let's play!

https://tinyurl.com/2p9e9kke

BOSTON UNIVERSITY