Different Transformer Architectures

CS 5624: Natural Language Processing

Fall 2026

https://tuvllms.github.io/nlp-fall-2026/

Tu Vu

VIRGINIA TECH

Logistics

- Final projects
  - Group information due today
  - Search for teammates on Piazza (@5)/Discord or reach out to us at cs5624instructors@gmail.com

- HW0 / Quiz0 released

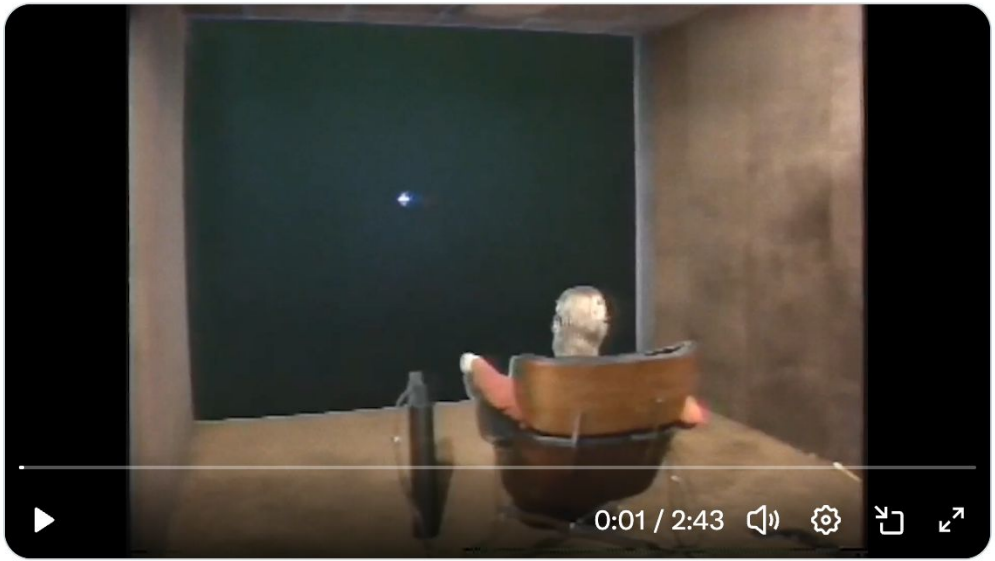
AI news

“Put that there”

OpenAI
@OpenAI

This is GPT-6 Astra.

Anything you can do on a computer, Astra can do for you. Fast.



3:32 PM · Sep 3, 2026 · **132.3M** Views

https://x.com/OpenAI/status/2095595741528125780

AGI (Artificial General Intelligence) is coming?

Jensen Huang
@JensenHuang

GPT-6 Astra, trained on ~100K+ NVIDIA Grace Blackwell NVLink72. From ChatGPT to o1 to Astra in 4 years.

AGI has arrived. Congratulations @OpenAI team.

400K GPUs coming online next.

4:41 PM · Sep 6, 2026 · **11.5M** Views

Jürgen Schmidhuber @SchmidhuberAI · 19h

Some claim that AGI is here now. Ridiculous. No AGI without mastery of the real world! No true self-improvement without self-improving hardware, as opposed to the already existing, self-improving, meta-learning software [DLH].

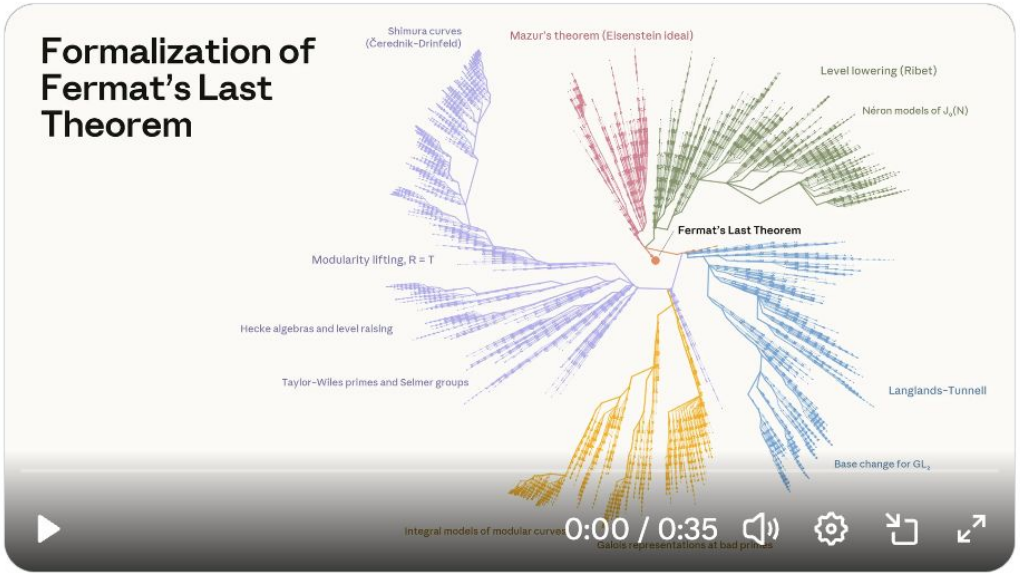
...

Anthropic @AnthropicAI · Sep 4

Checking that a major mathematical proof is correct can take years. Formalization—converting the mathematical reasoning into a form computer proof assistants like Lean can verify—can help.

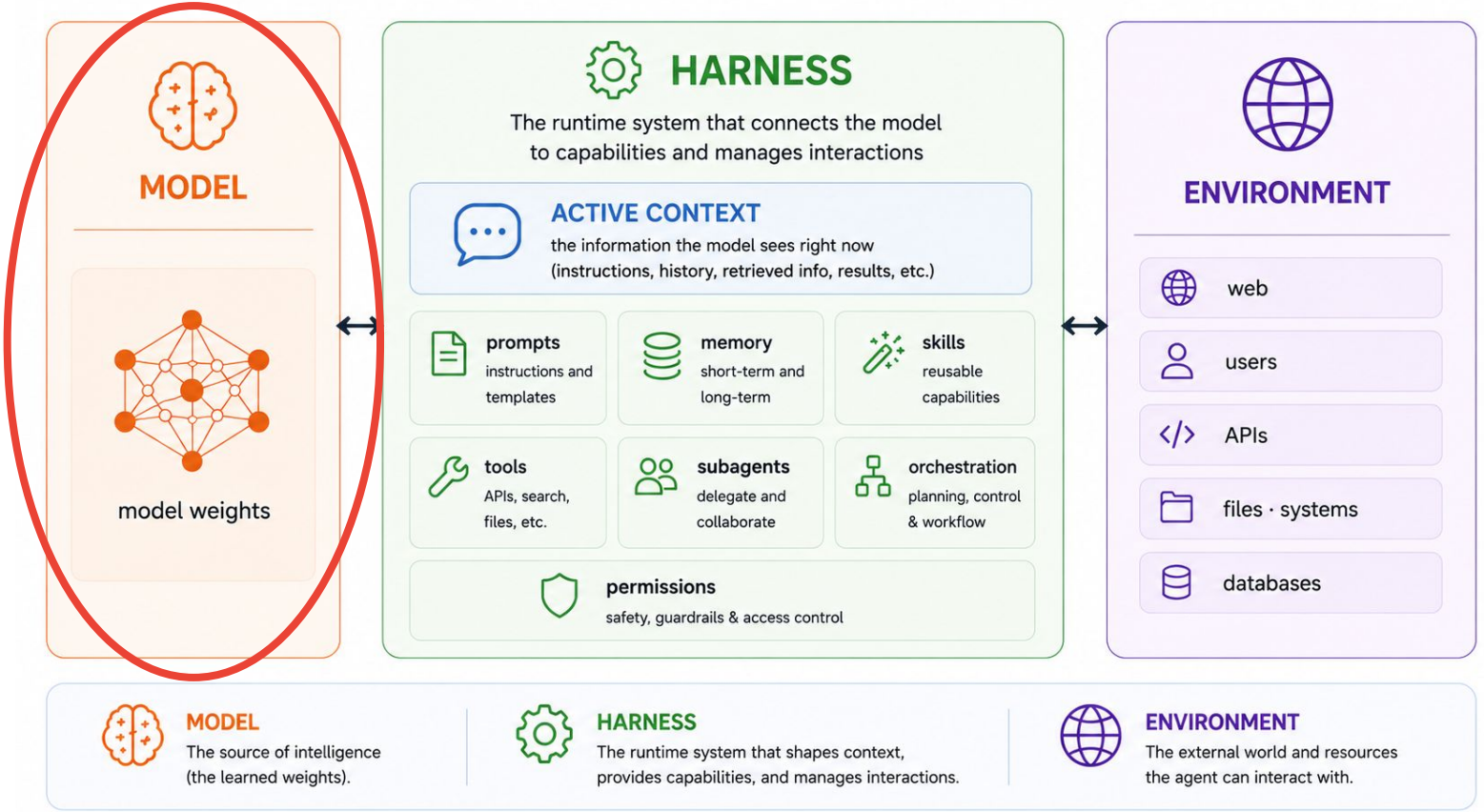
Last month, Claude completed the first formalized proof of Fermat's Last Theorem, one of

Show more

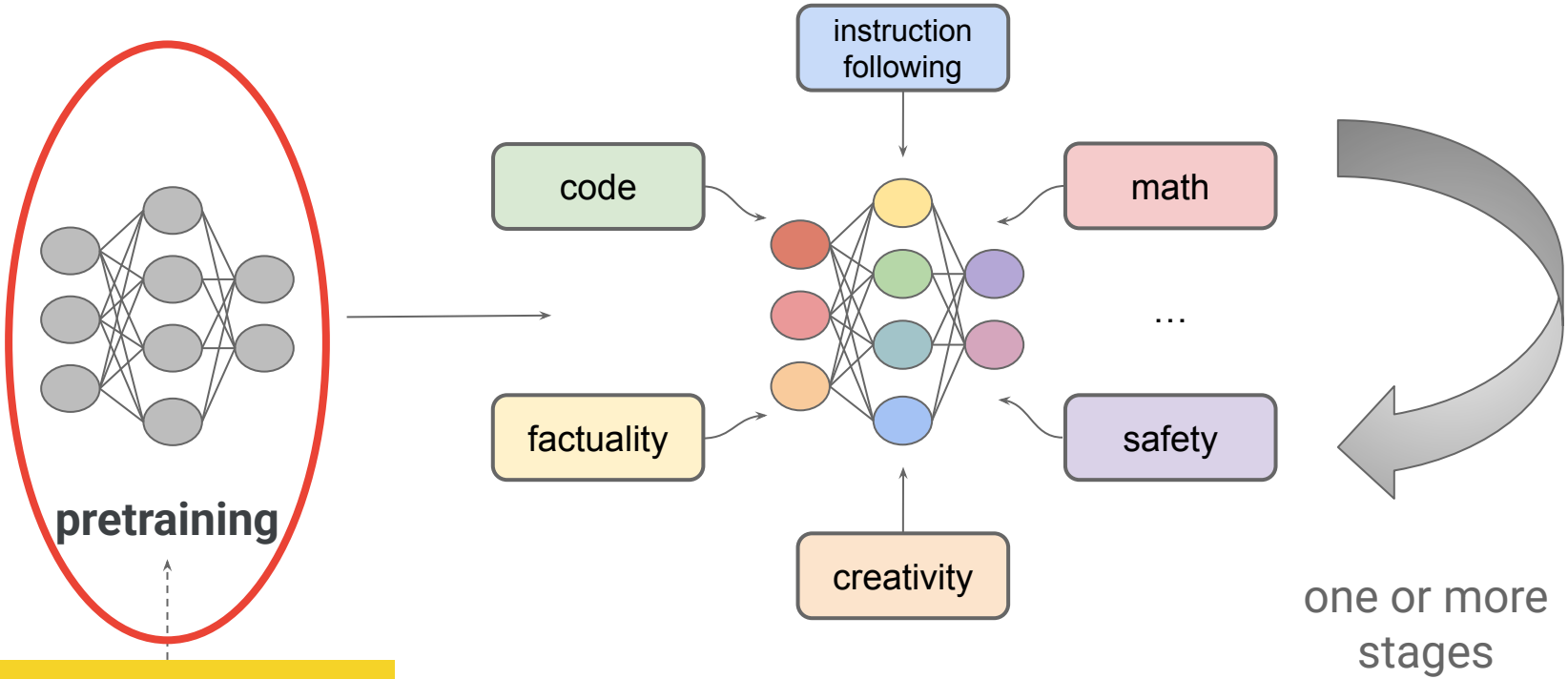


597 2.5K 13K 4.3M

Where are we?


MODEL
model weights
HARNESS
The runtime system that connects the model to capabilities and manages interactions
ACTIVE CONTEXT
the information the model sees right now (instructions, history, retrieved info, results, etc.)
prompts
instructions and templates
memory
short-term and long-term
skills
reusable capabilities
tools
APIs, search, files, etc.
subagents
delegate and collaborate
orchestration
planning, control & workflow
permissions
safety, guardrails & access control
ENVIRONMENT
web
users
APIs
files · systems
databases
MODEL
The source of intelligence (the learned weights).
HARNESS
The runtime system that shapes context, provides capabilities, and manages interactions.
ENVIRONMENT
The external world and resources the agent can interact with.

The development of modern LLMs



Transformers

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

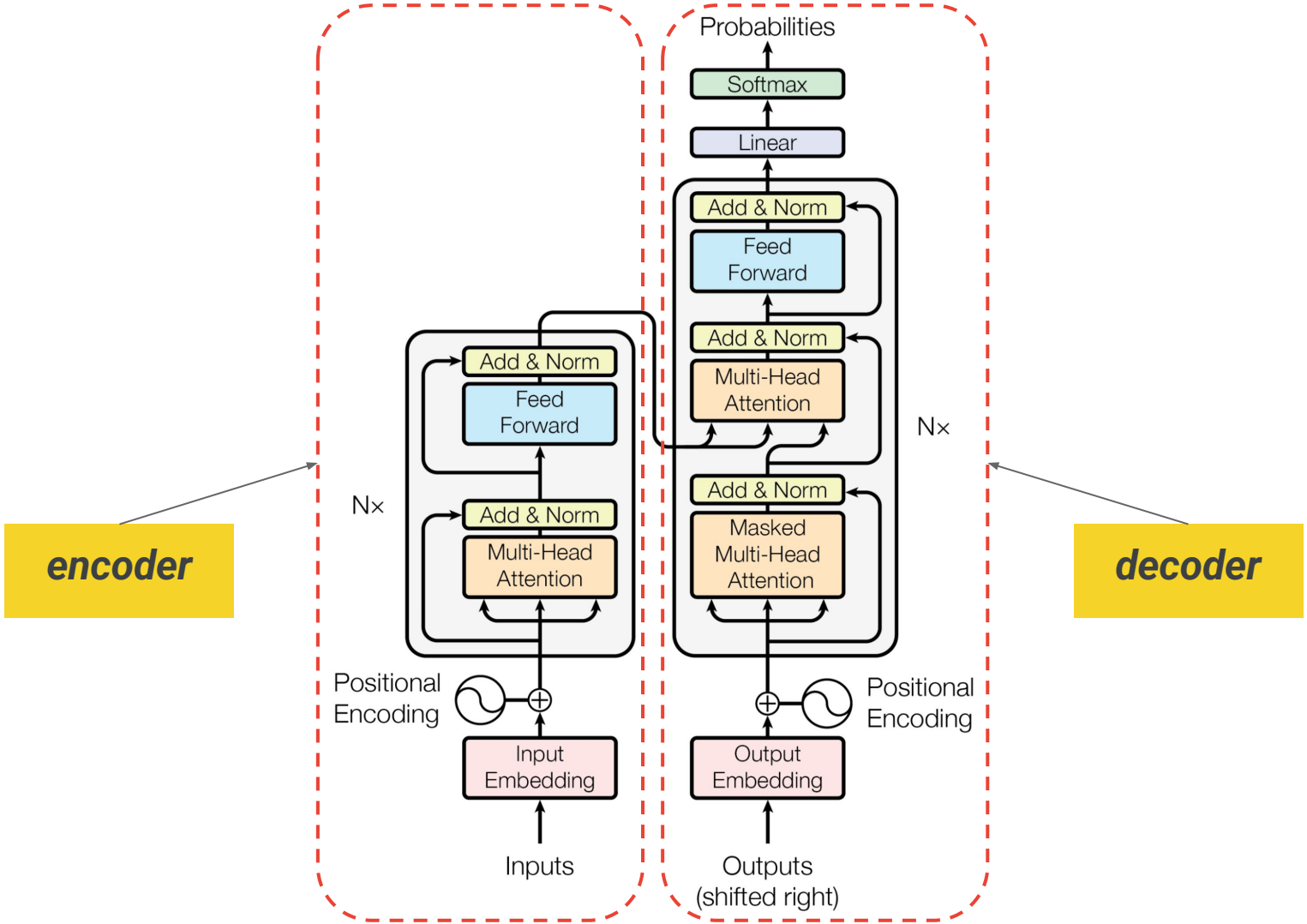
Illia Polosukhin* illia.polosukhin@gmail.com

Abstract

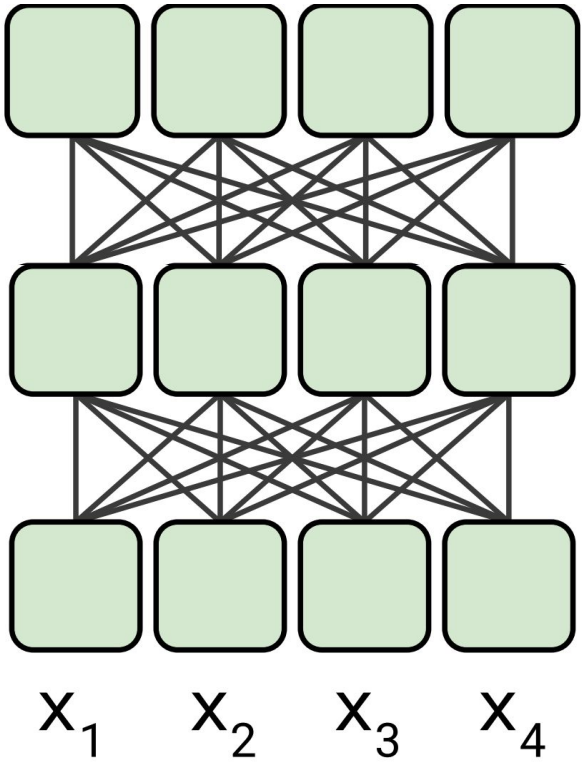
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.0 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Different Transformers architectures

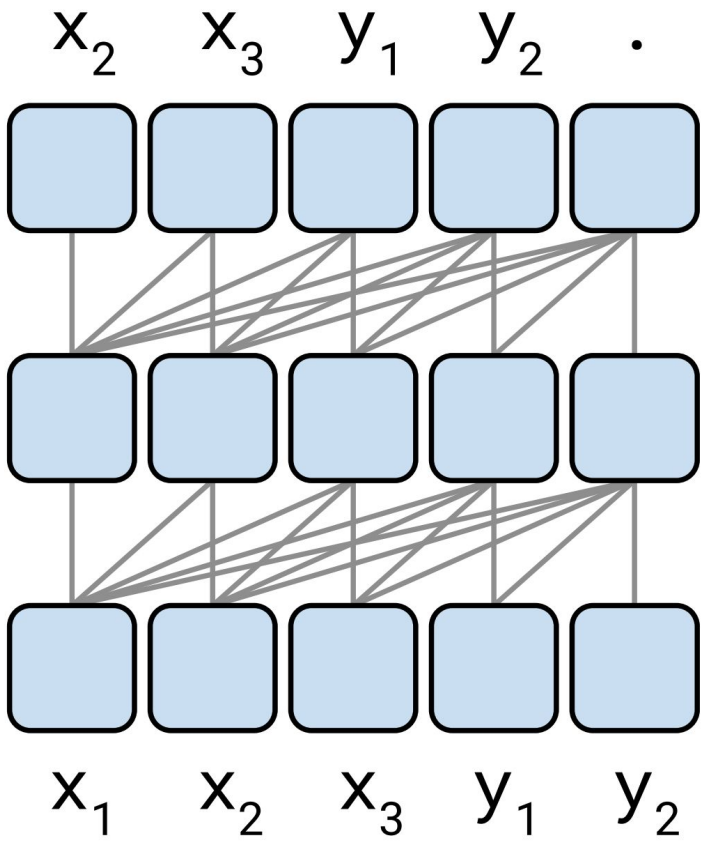
- Encoder-only
  - BERT
- Encoder-decoder
  - T5
- Decoder-only
  - GPT, Claude, Gemini


encoder
decoder
Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Nx
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

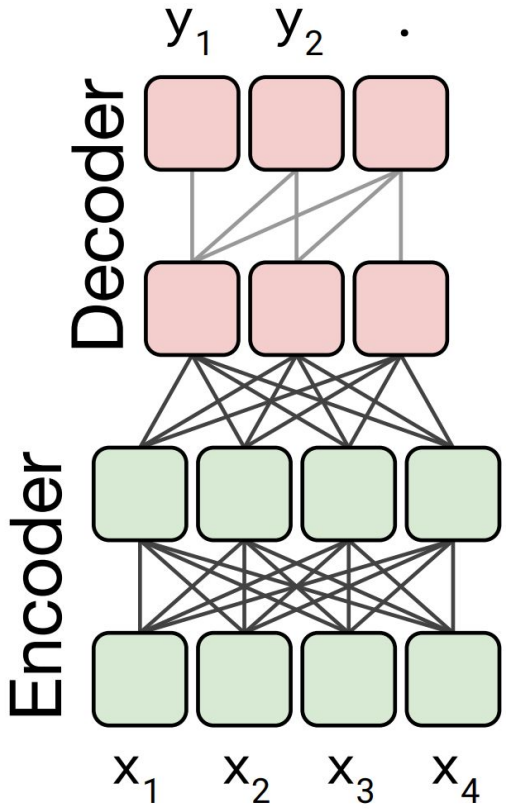
Encoder-only

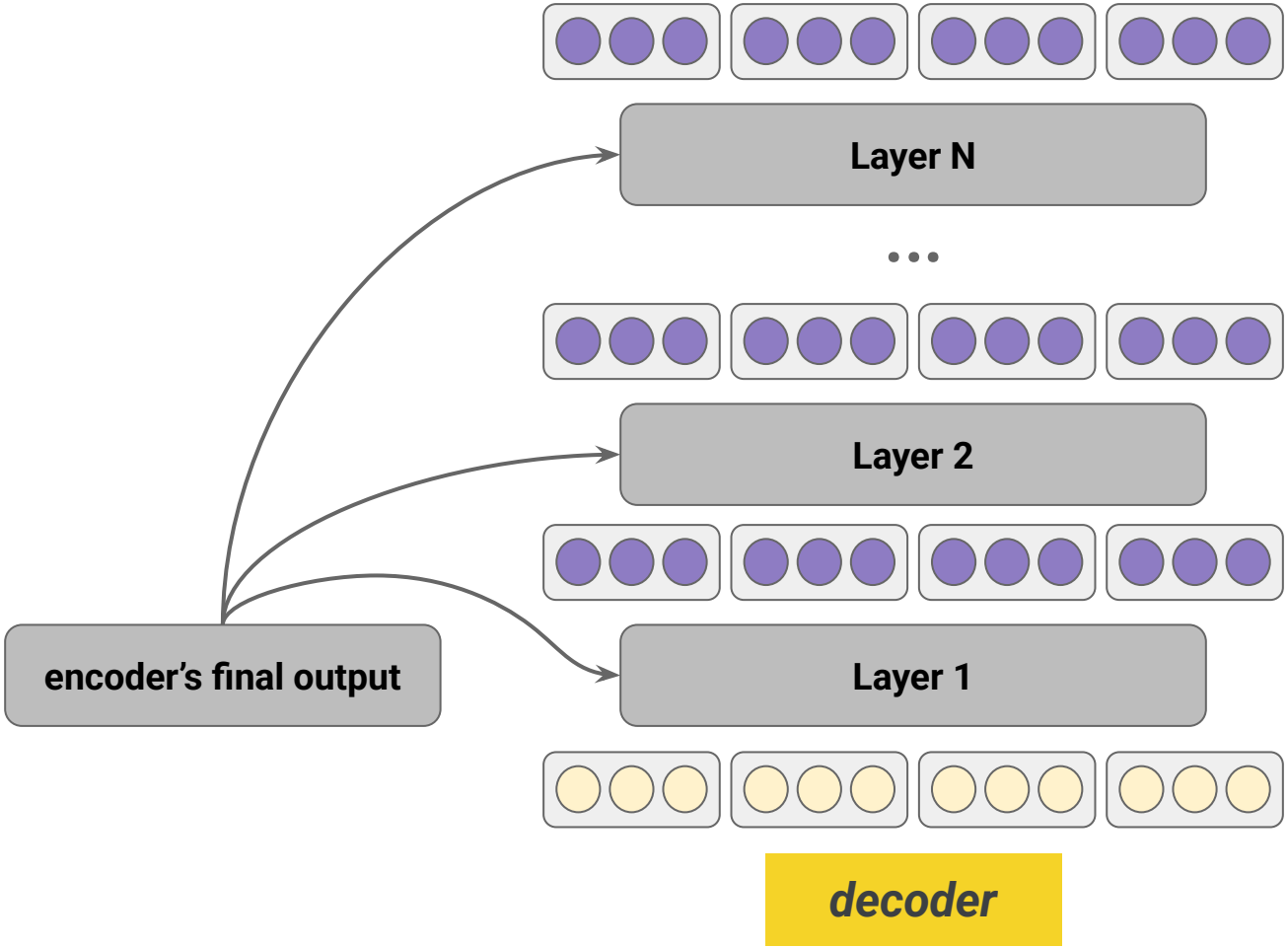


Decoder-only

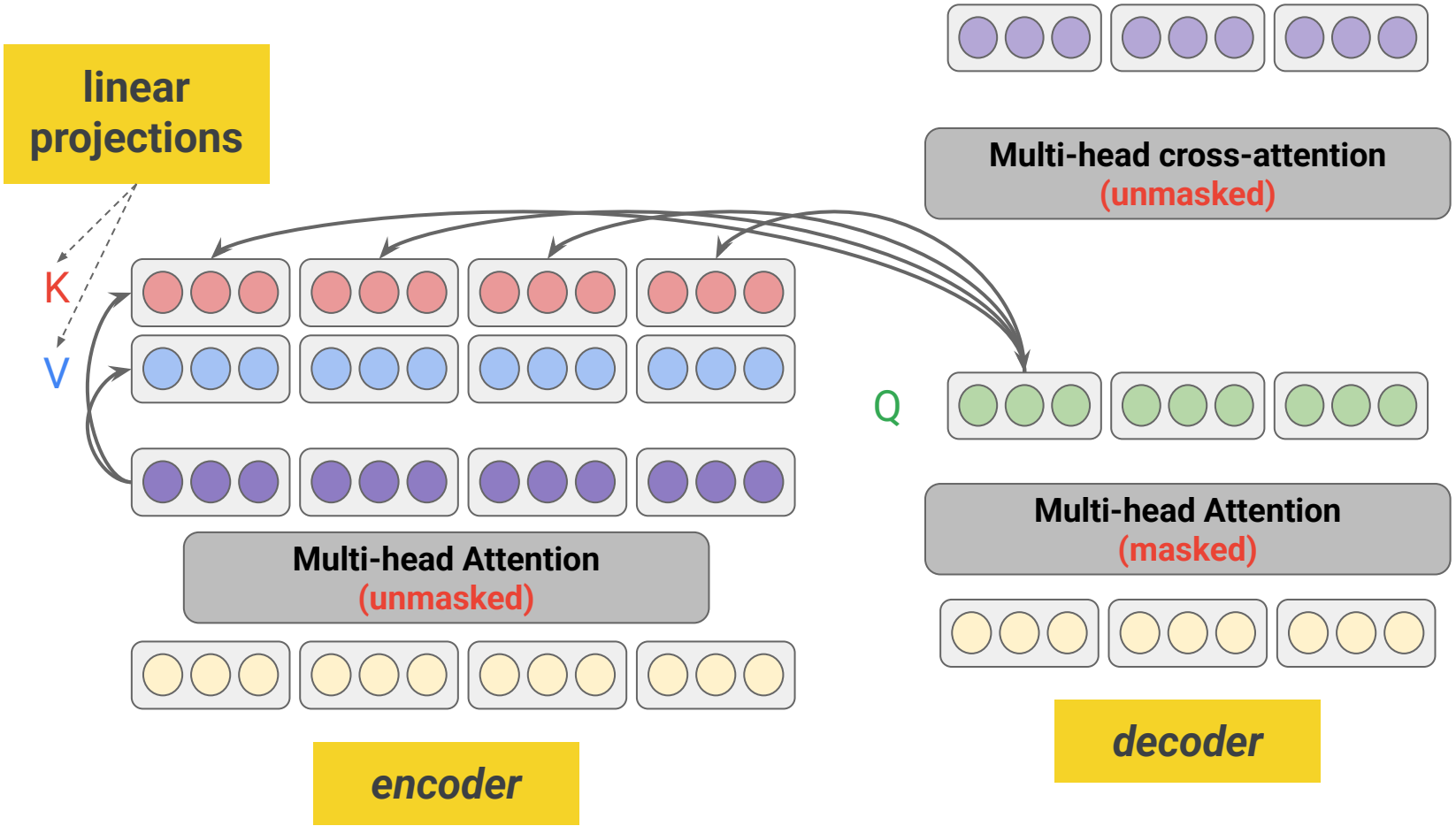


Encoder-decoder



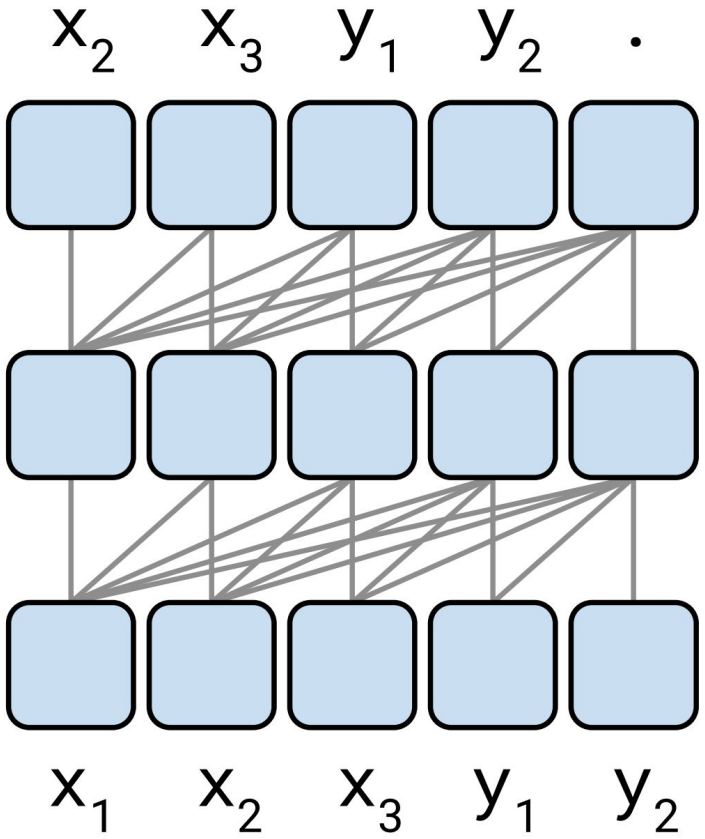

Layer N
Layer 2
Layer 1
encoder's final output
decoder

Cross-attention in the decoder



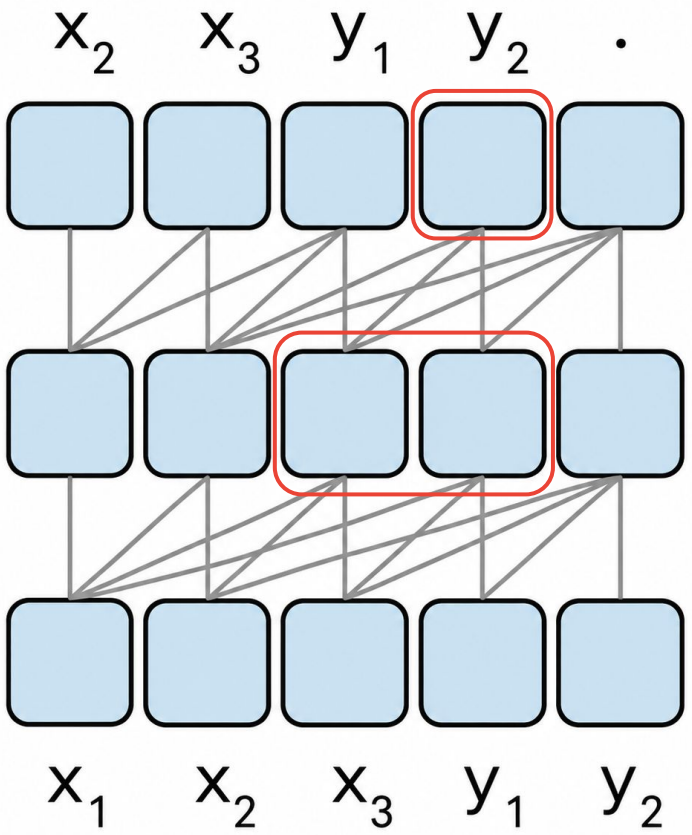
Causal masking

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ 1 & 1 & 0 & 0 & 0 & 0 & \dots \\ 1 & 1 & 1 & 0 & 0 & 0 & \dots \\ 1 & 1 & 1 & 1 & 0 & 0 & \dots \\ 1 & 1 & 1 & 1 & 1 & 0 & \dots \\ 1 & 1 & 1 & 1 & 1 & 1 & \dots \\ & & & \dots & & & \end{bmatrix}$$

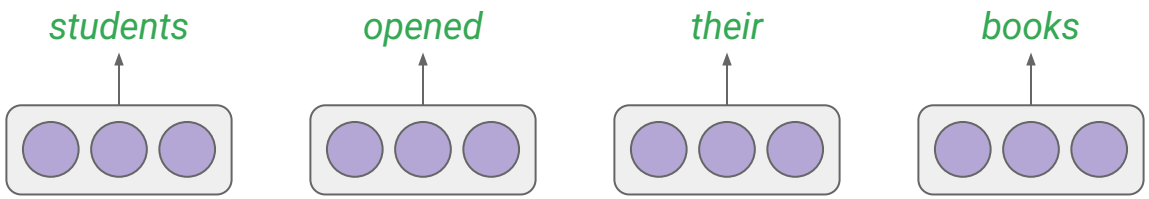


Be careful!

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ 1 & 1 & 0 & 0 & 0 & 0 & \dots \\ 0 & 1 & 1 & 0 & 0 & 0 & \dots \\ 0 & 0 & 1 & 1 & 0 & 0 & \dots \\ 0 & 0 & 0 & 1 & 1 & 0 & \dots \\ 0 & 0 & 0 & 0 & 1 & 1 & \dots \\ & & & \dots & & & \end{bmatrix}$$

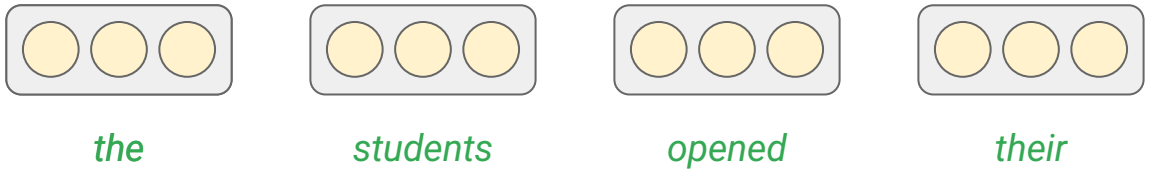


Transformer decoder

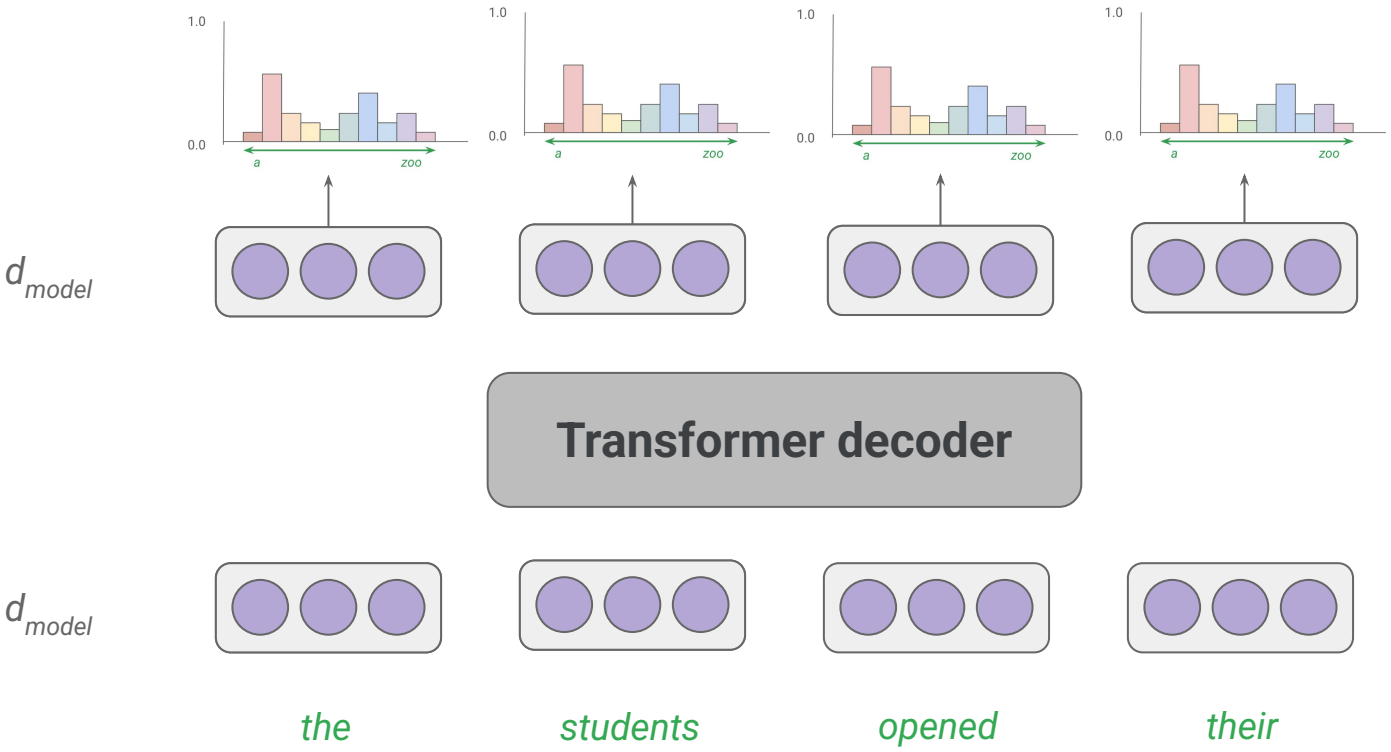


the architecture used in frontier LLMs

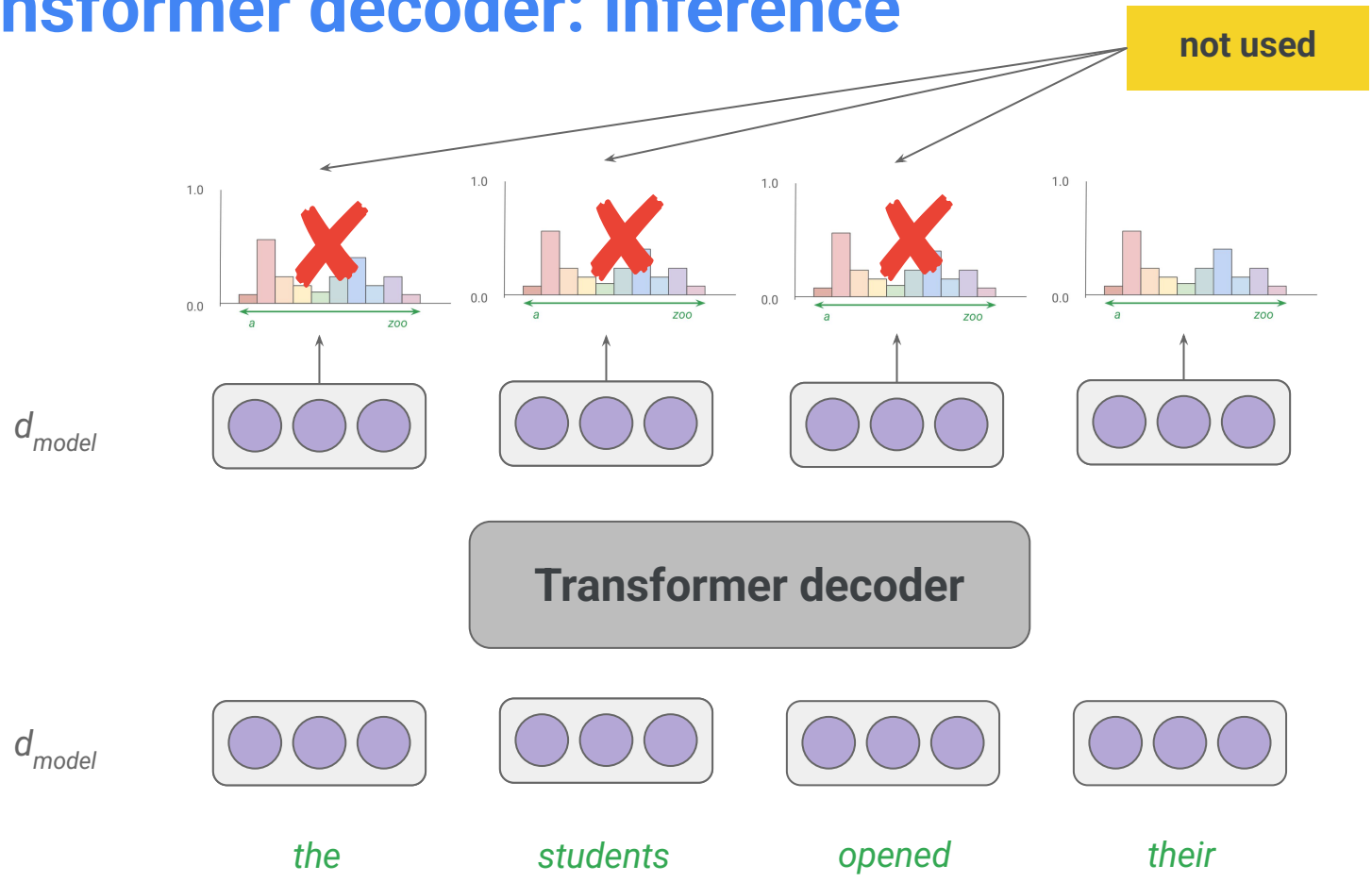
Transformer decoder
(masked)



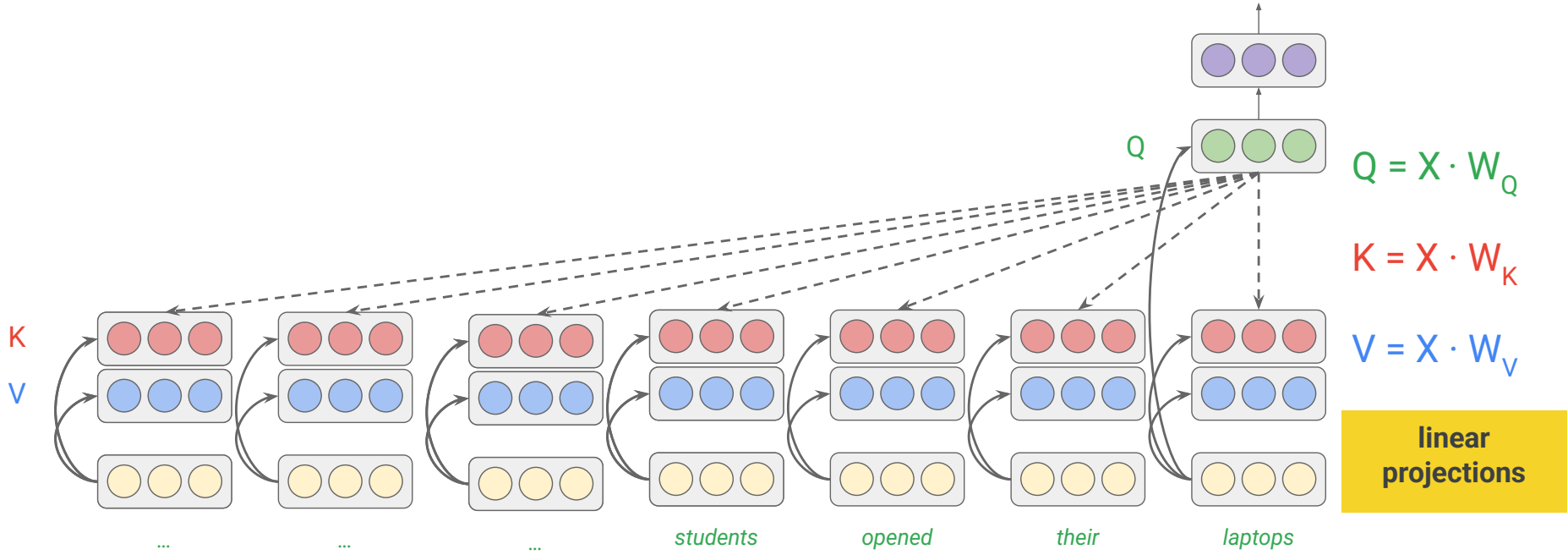
Transformer decoder: training



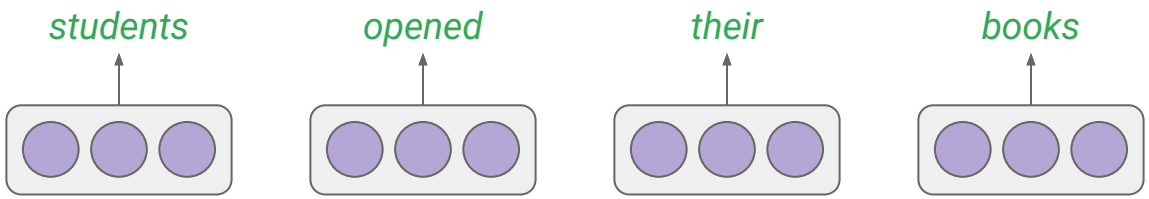
Transformer decoder: inference



Inference

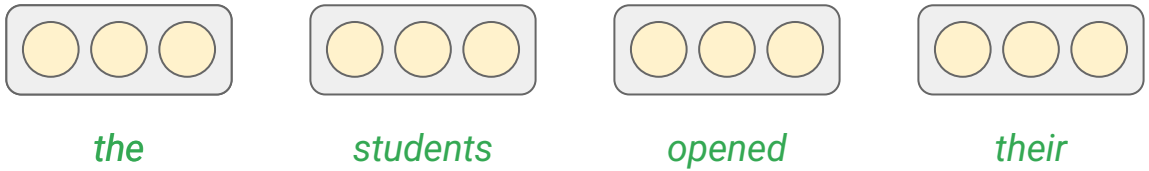

Q
K
V
Q = X · W_Q
K = X · W_K
V = X · W_V
linear projections
...
...
...
students
opened
their
laptops

Transformer decoder



the architecture used in frontier LLMs

Transformer decoder
(masked)



Different Transformers architectures

- Encoder-only
  - BERT
- Encoder-decoder
  - T5
- Decoder-only
  - GPT, Claude, Gemini

Image created by Gemini

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

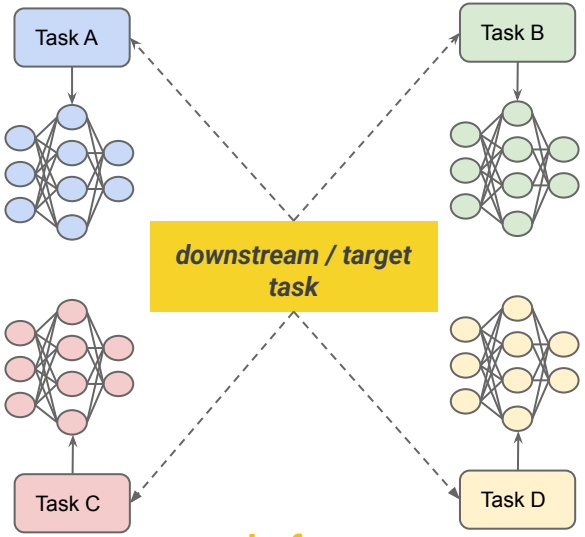
Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**

Google AI Language

{jacobdevlin,mingweichang,kentonl,kristout}@google.com

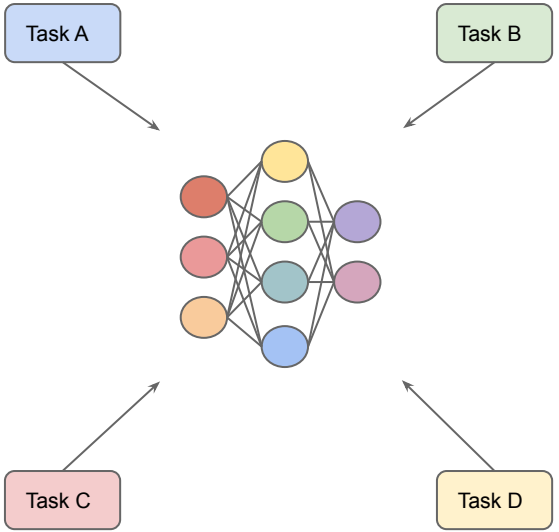
A learning paradigm shift

training task-specific models from scratch



before BERT

pretraining and then adapting



since BERT

ELMo

Image created by Gemini

Deep contextualized word representations

Matthew E. Peters[†], **Mark Neumann**[†], **Mohit Iyyer**[†], **Matt Gardner**[†],
{matthewp,markn,mohiti,mattg}@allenai.org

Christopher Clark[*], **Kenton Lee**[*], **Luke Zettlemoyer**[†*]
{csquared,kentonl,lsz}@cs.washington.edu

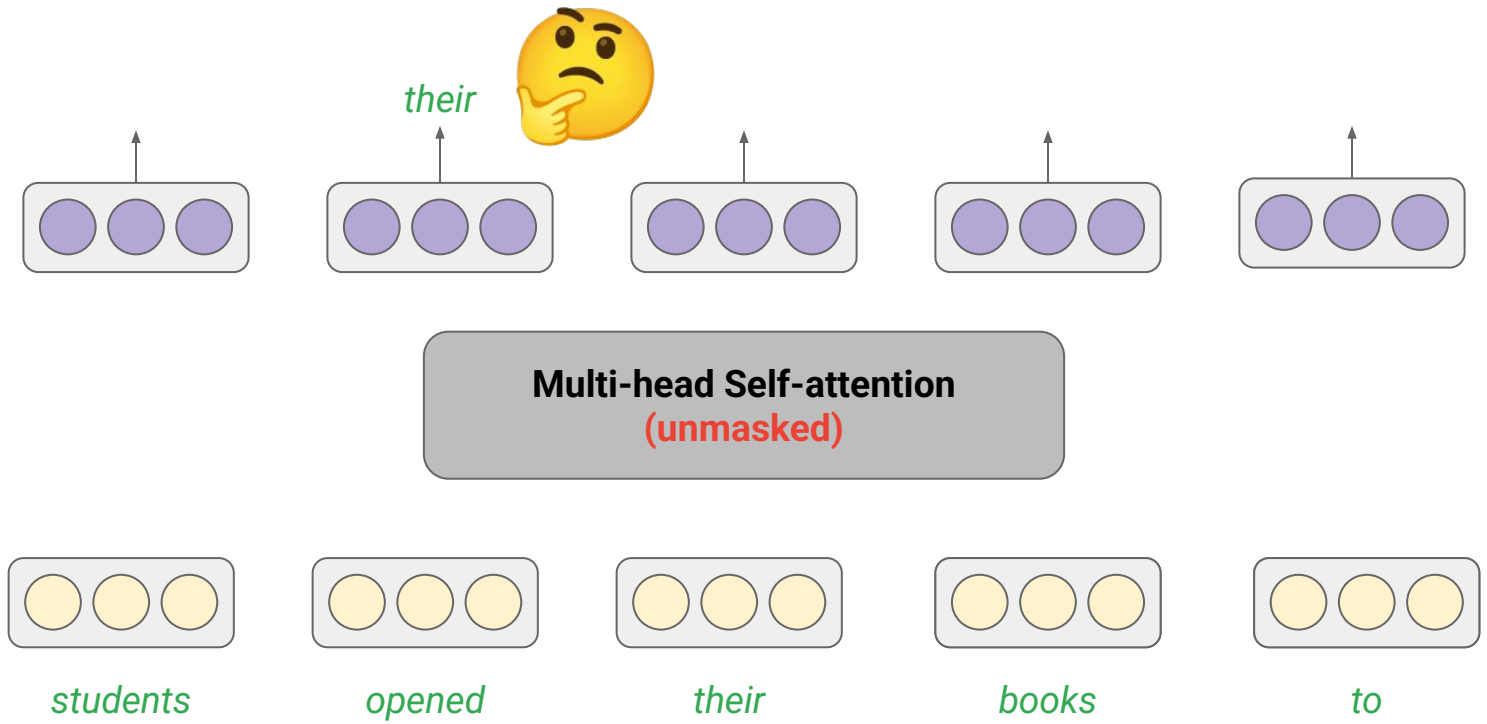
[†]Allen Institute for Artificial Intelligence
[*]Paul G. Allen School of Computer Science & Engineering, University of Washington

BERT vs. ELMo

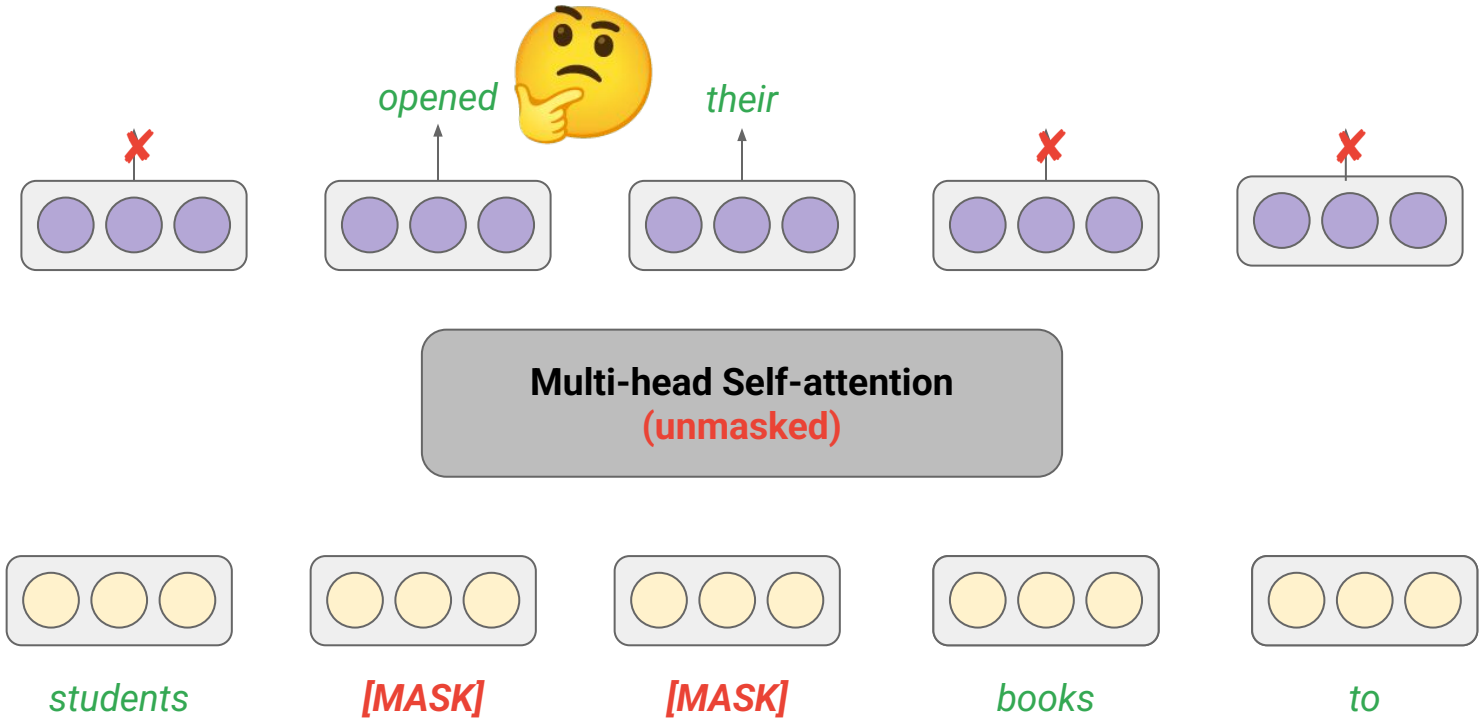
	BERT	ELMo
Model	Transformers	Bidirectional LSTM (Long Short-Term Memory, a variant of RNN)
Pre-training objective(s)	Masked language modeling + next sentence prediction	Left-to-right language modeling
Adaptation method	Fine-tuning	Feature-based (pretrained representations as additional features to task-specific models)

Pretraining

Language modeling using a Transformer encoder

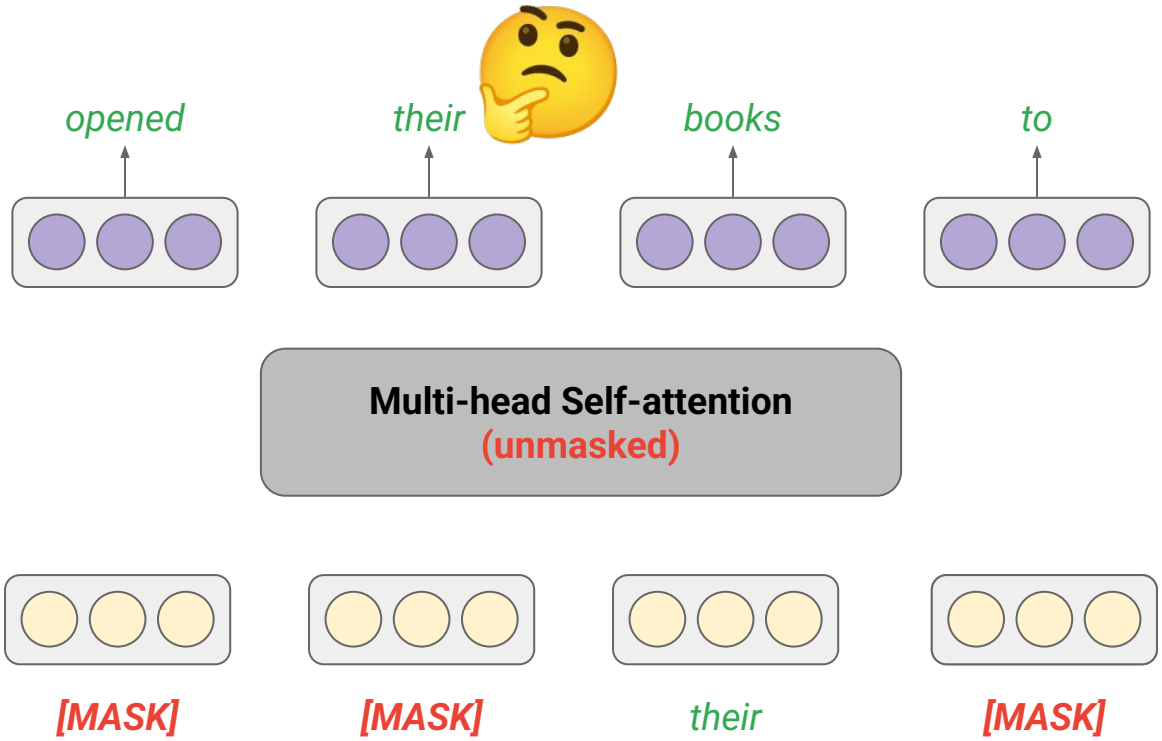


Masked language modeling



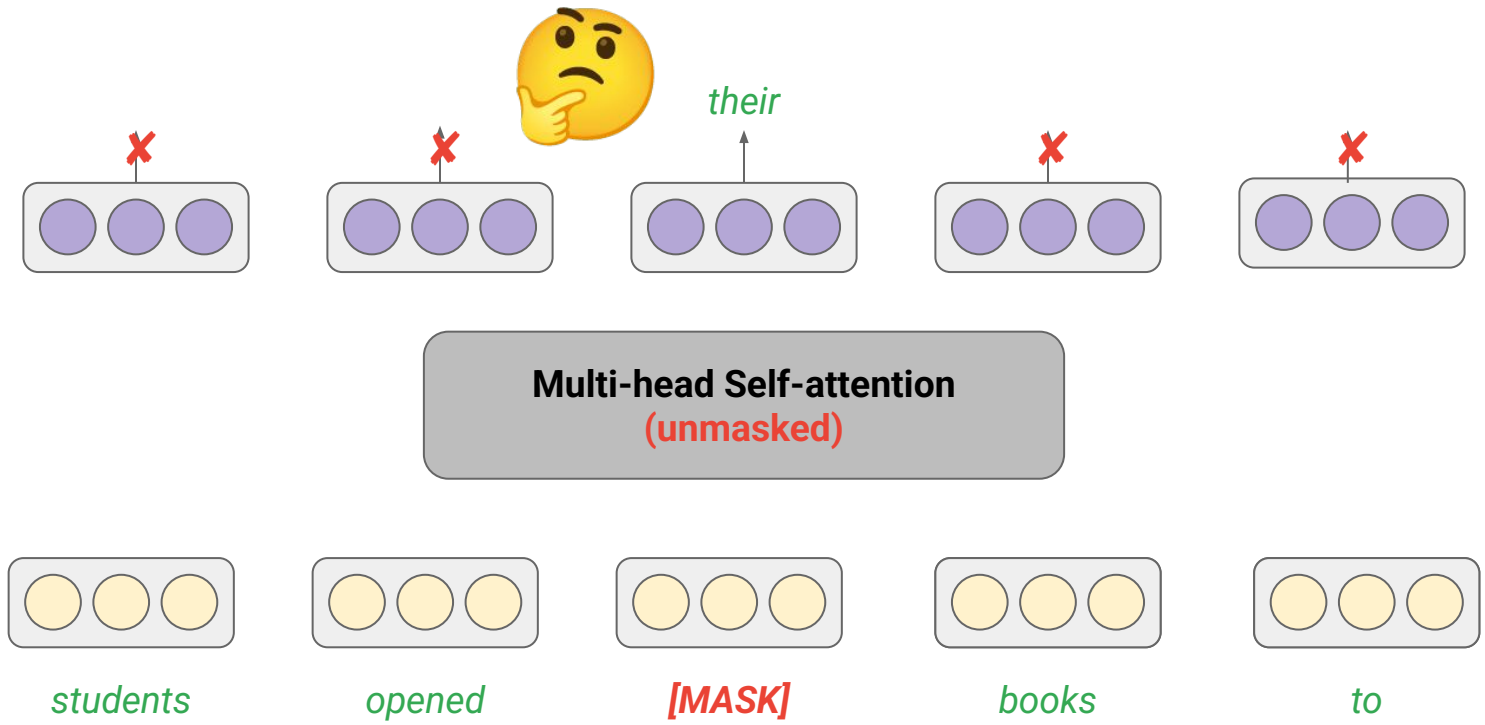
15% - 30% of all tokens in each sequence are masked at random

What if we mask more tokens?



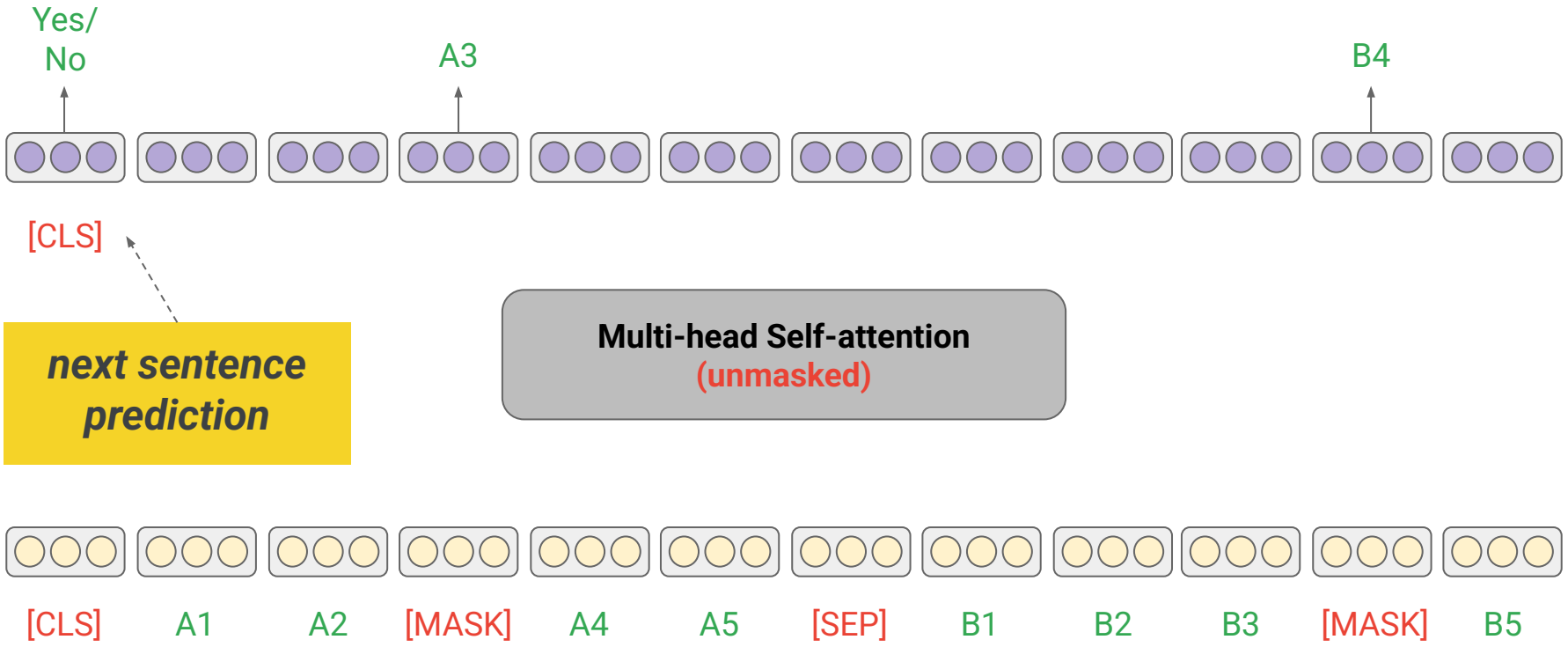
15% - 30% of all tokens in each sequence are masked at random

What if we mask less tokens?

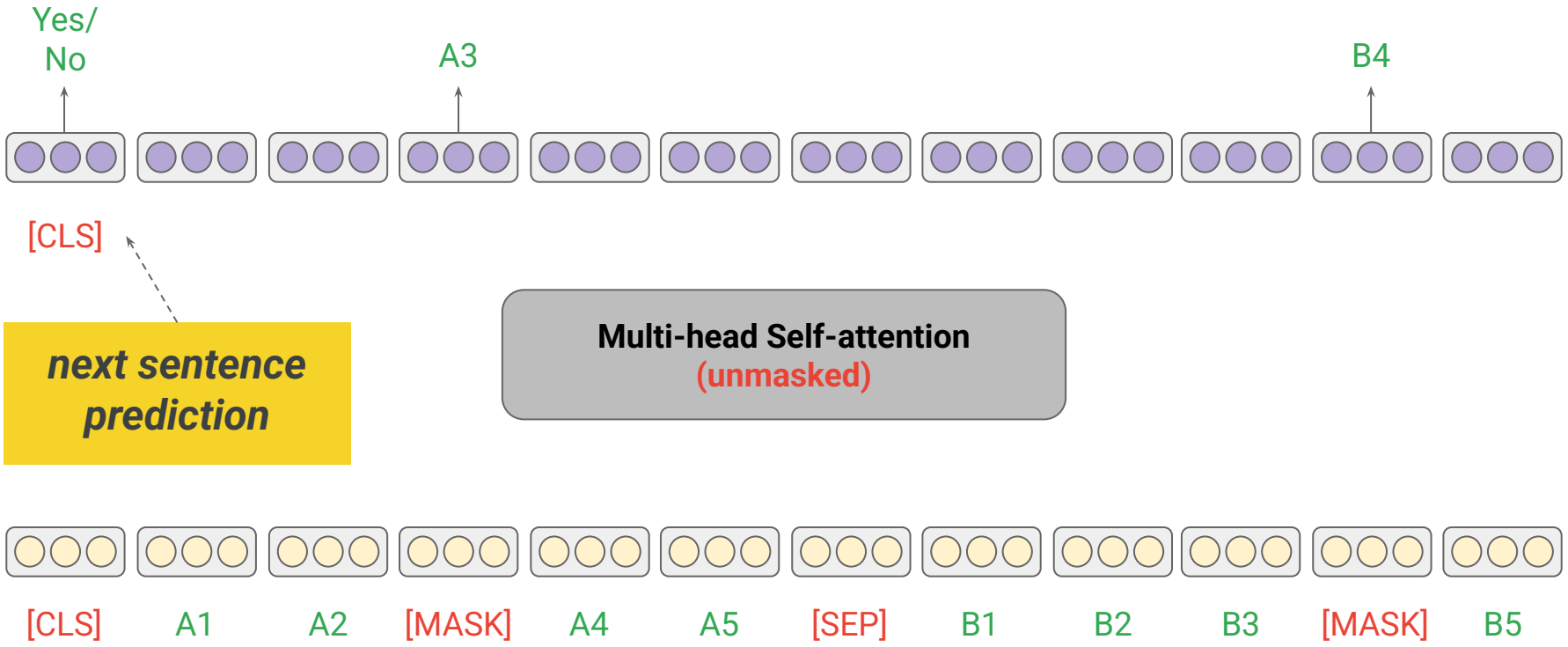


15% - 30% of all tokens in each sequence are masked at random

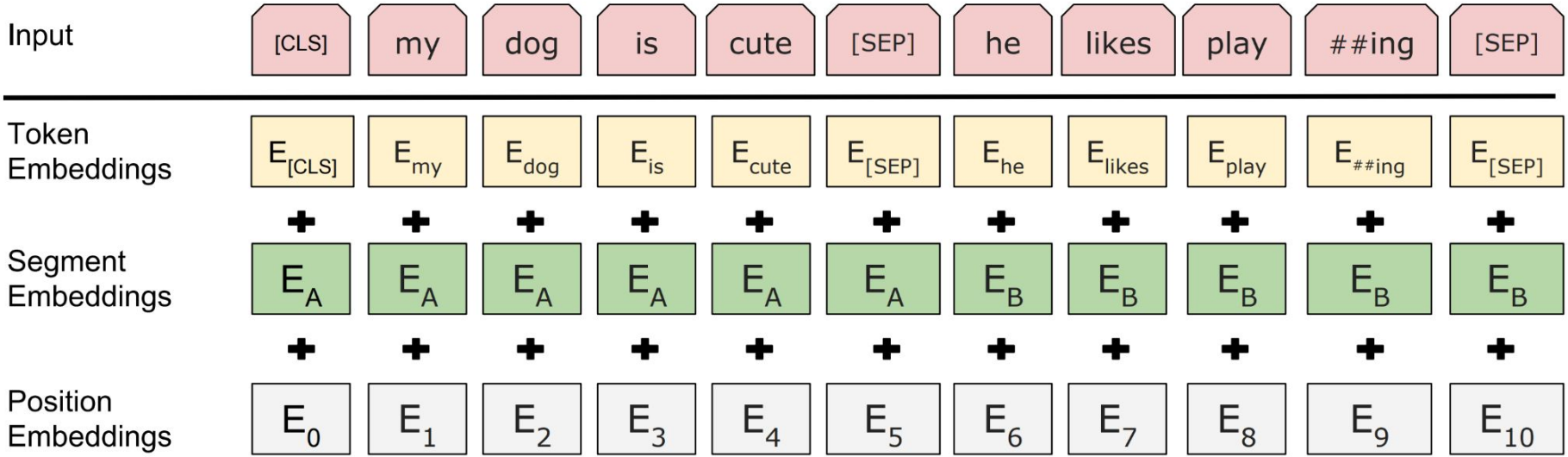
CLS & SEP tokens



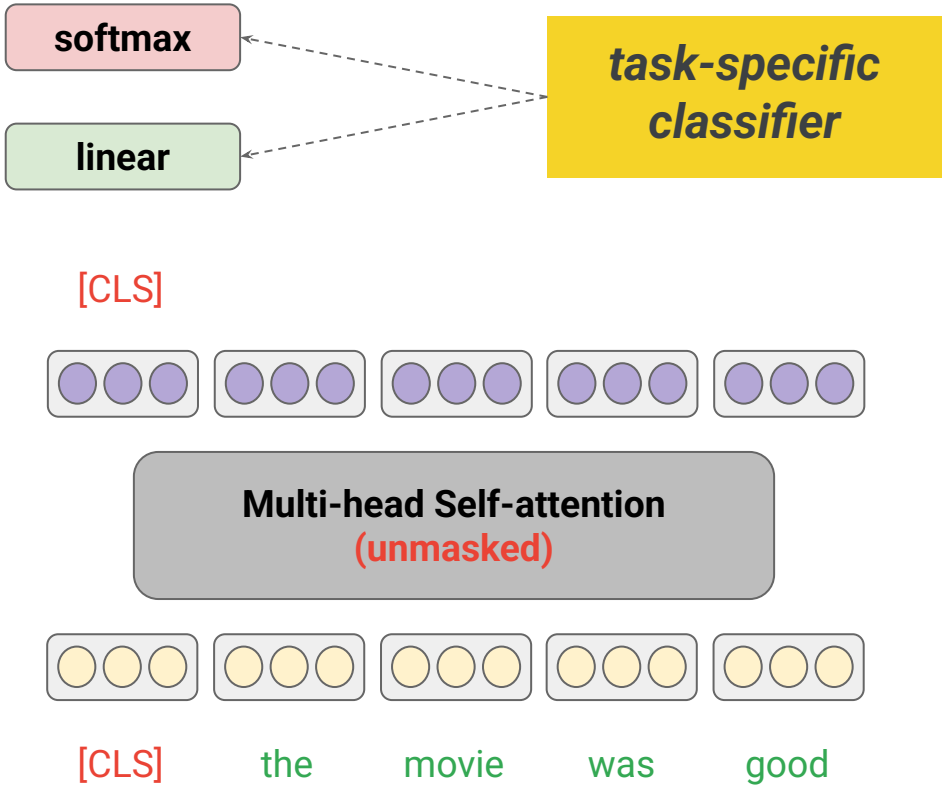
BERT Pretraining



BERT input representation

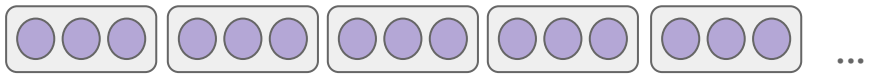


BERT Fine-tuning



T5 Pretraining: Span corruption

<X>, <Y>: sentinel tokens



Transformer encoder
(unmasked)

Thank you <X> me to your party <Y> week

<X> for inviting <Y> last <EOS>

Transformer decoder
(masked)

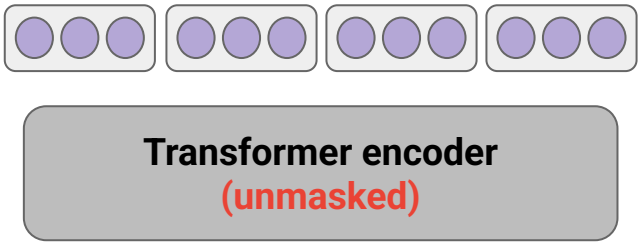
<BOS> <X> for inviting <Y> last

Thank you ~~for~~ ~~inviting~~ me to your party ~~last~~ week

encoder

decoder

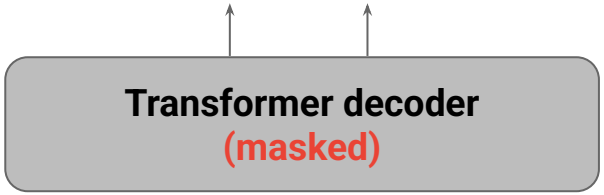
T5 Fine-tuning



sentiment analysis: this movie was good

encoder

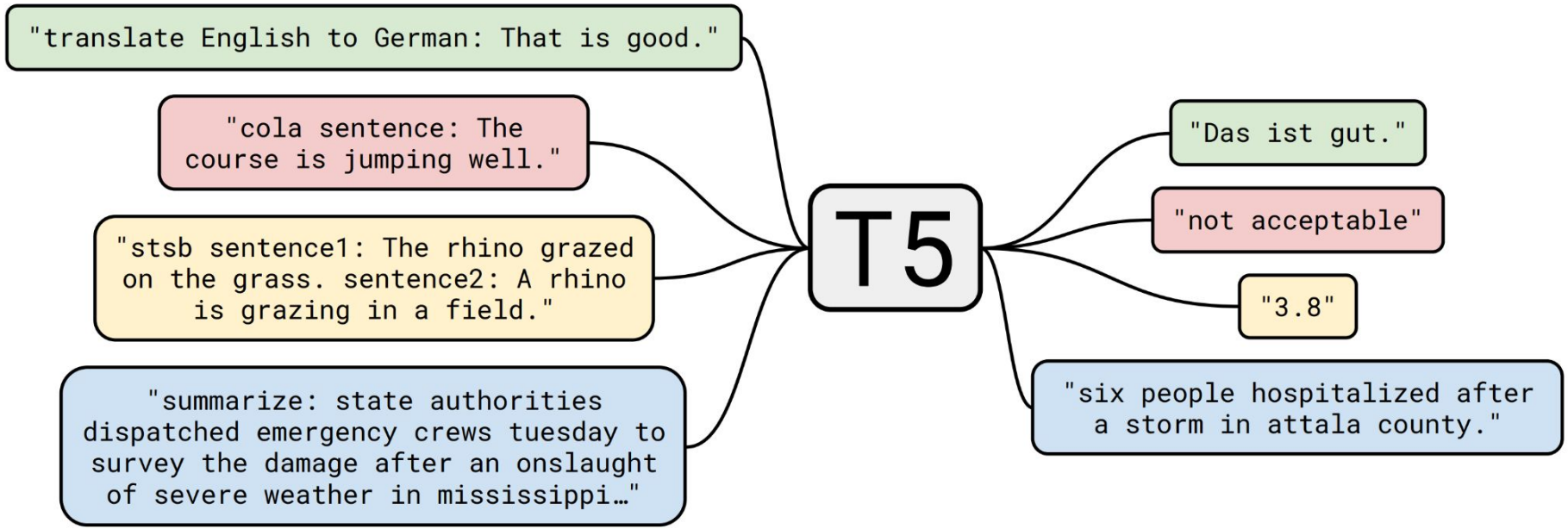
positive <EOS>



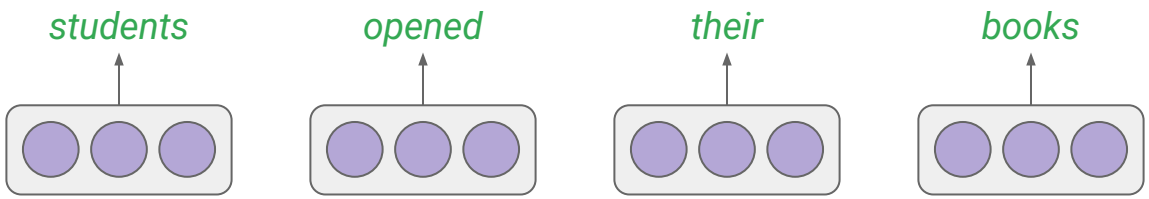
<BOS> positive

decoder

T5 facilitates multitask learning

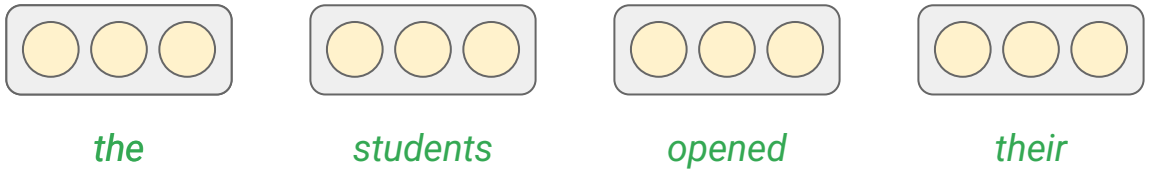


Decoder-only model



the architecture used in frontier LLMs

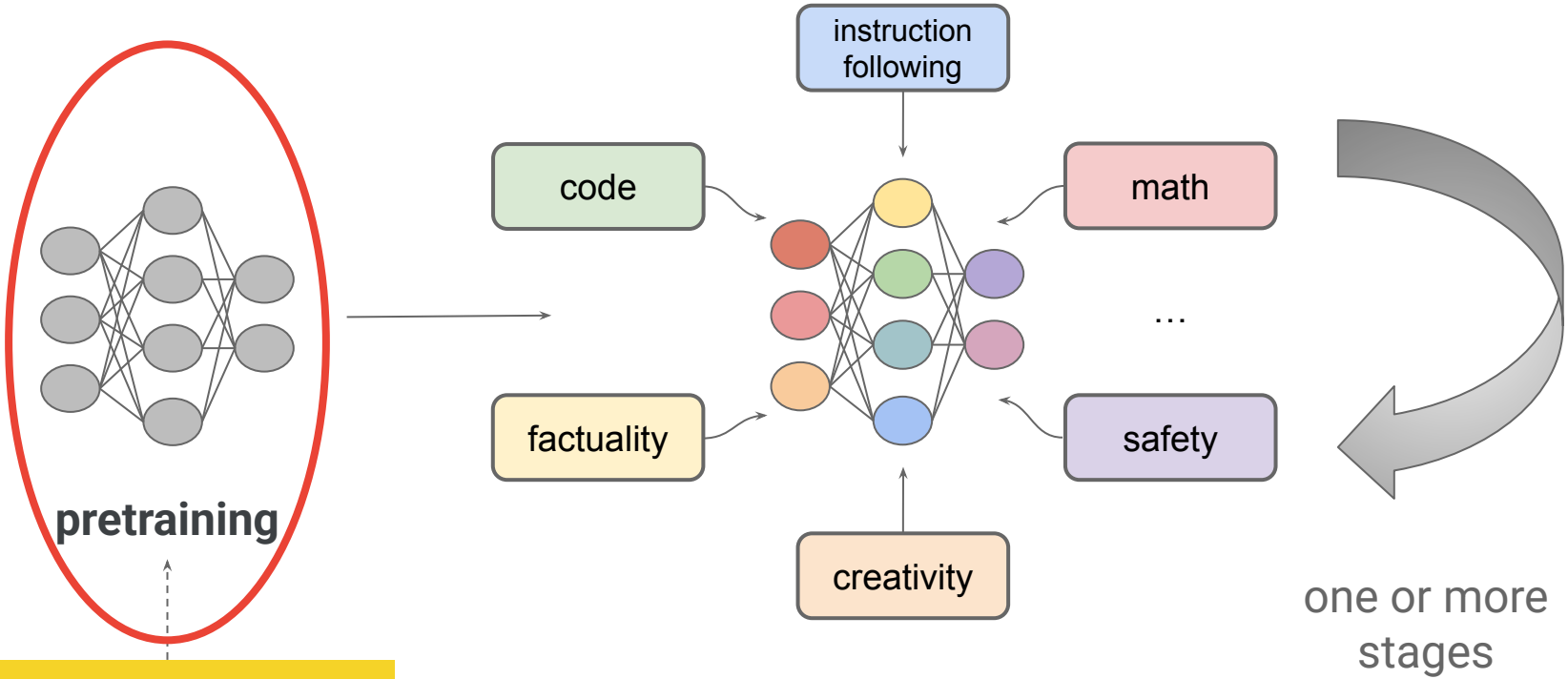
Transformer decoder
(masked)



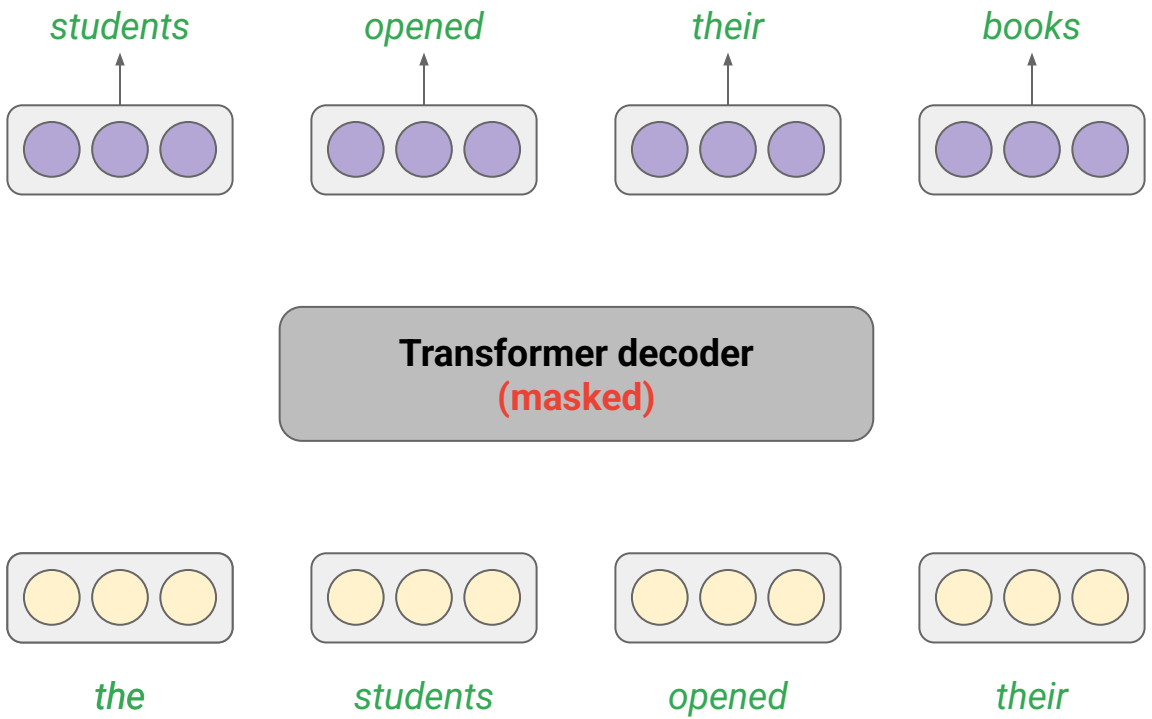
Note on cross-attention

- Can be used to inject non-text data (e.g., images, structured data, or even sensor readings) into the model

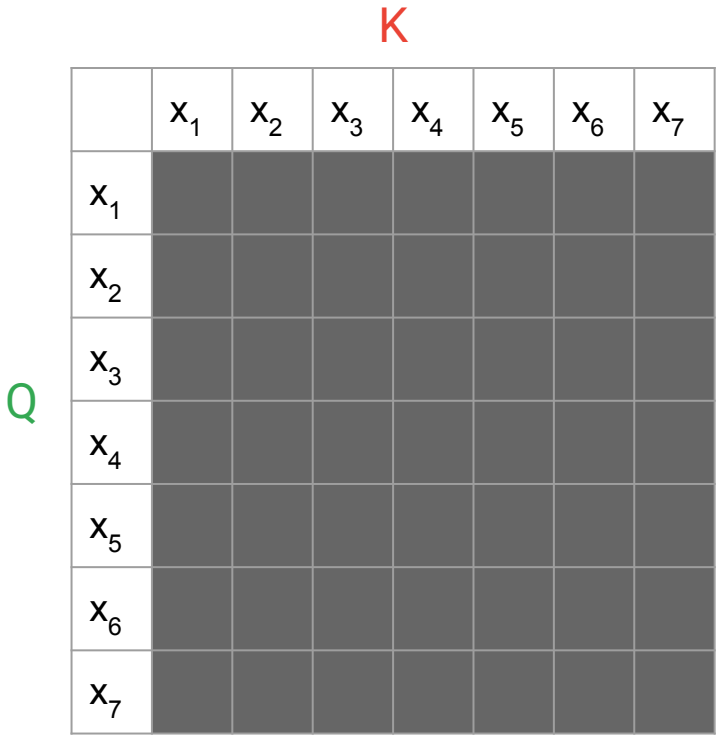
The development of modern LLMs



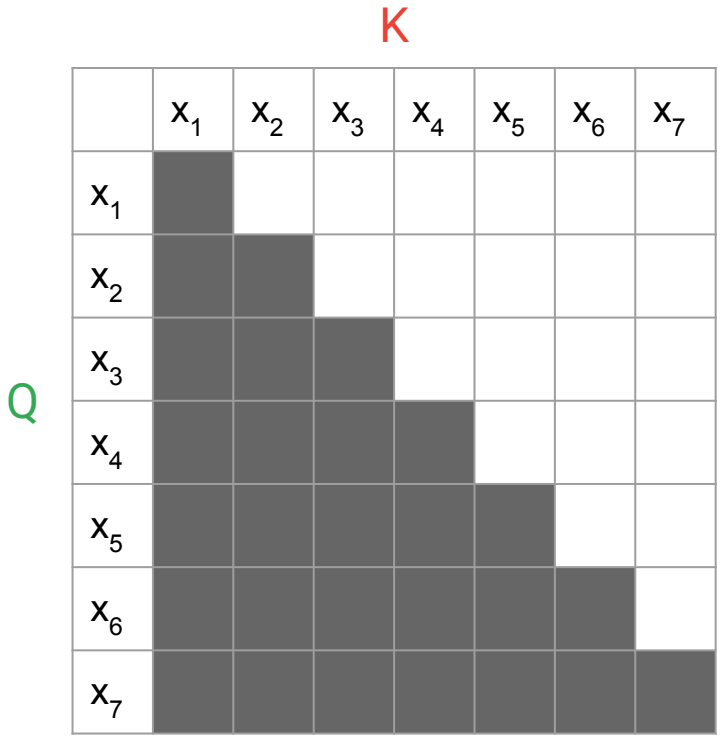
Self-supervised pretraining with a causal LM



Different attention mask patterns

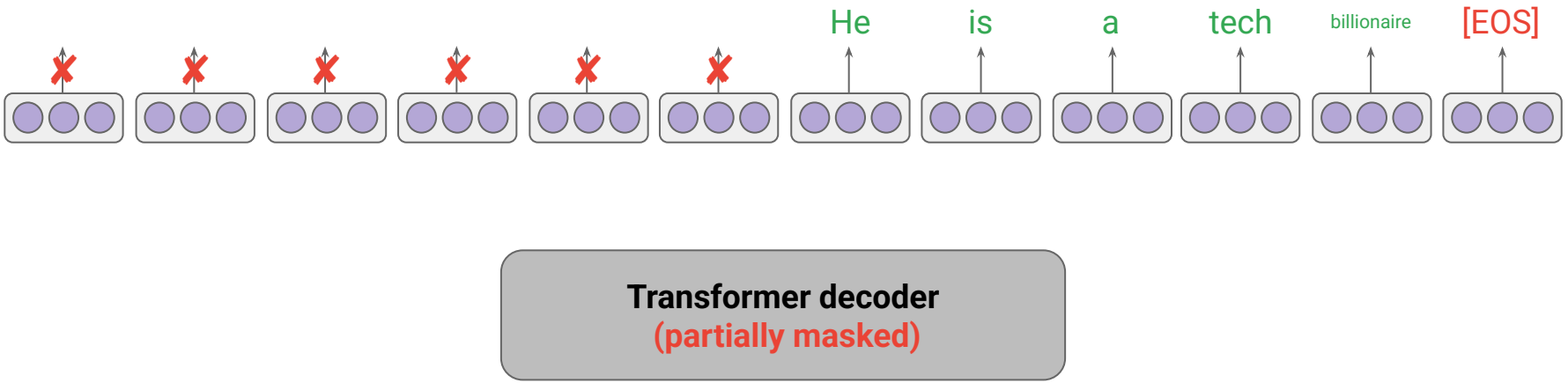


fully-visible

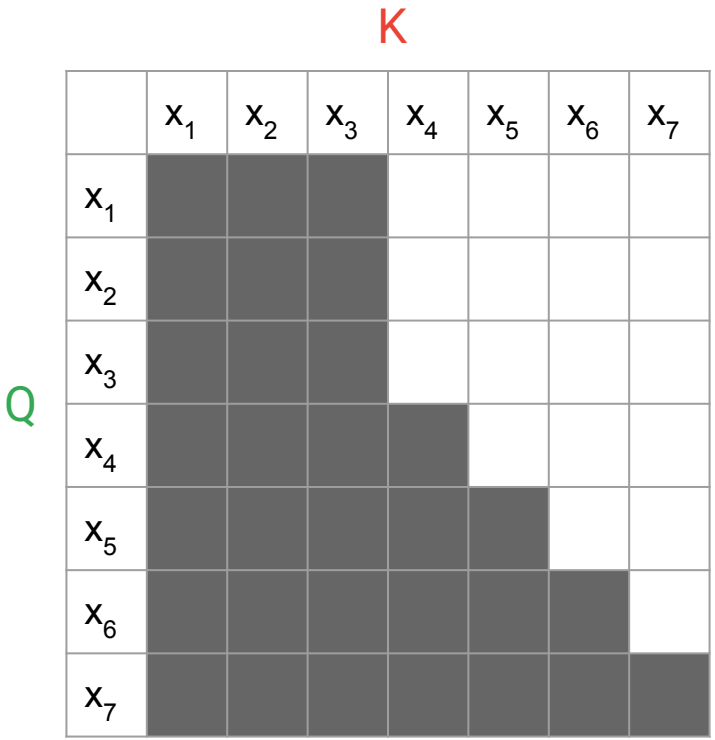


causal

Supervised fine-tuning (SFT) with prefix LM

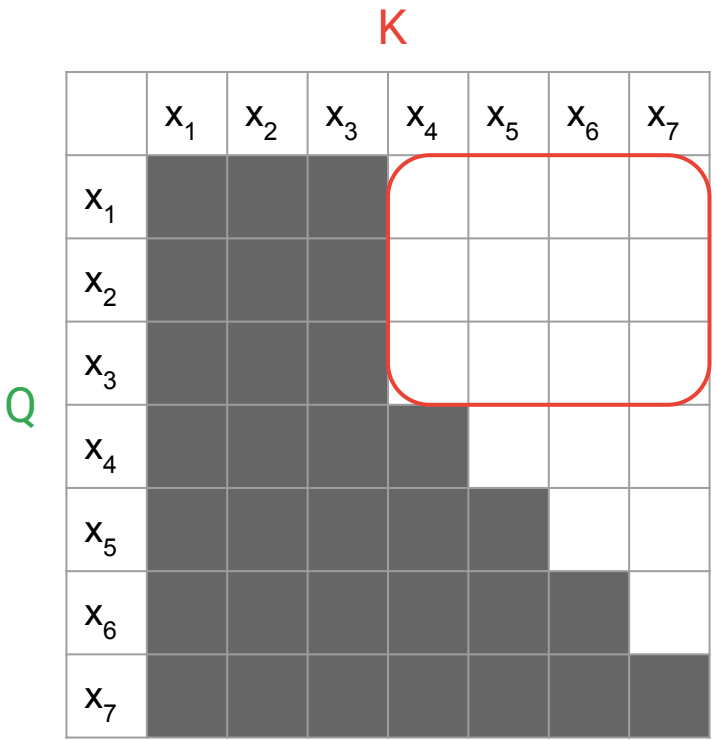


Different attention mask patterns (cont'd)



Prefix LM

Different attention mask patterns (cont'd)



Why masking here?

Attention sink: a phenomenon in LLMs

EFFICIENT STREAMING LANGUAGE MODELS WITH ATTENTION SINKS

Guangxuan Xiao[1*] **Yuandong Tian**[2] **Beidi Chen**[3] **Song Han**[1,4] **Mike Lewis**[2]

[1] Massachusetts Institute of Technology [2] Meta AI
[3] Carnegie Mellon University [4] NVIDIA
https://github.com/mit-han-lab/streaming-llm

ABSTRACT

Deploying Large Language Models (LLMs) in streaming applications such as multi-round dialogue, where long interactions are expected, is urgently needed but poses two major challenges. Firstly, during the decoding stage, caching previous tokens' Key and Value states (KV) consumes extensive memory. Secondly, popular LLMs cannot generalize to longer texts than the training sequence length. Window attention, where only the most recent KVs are cached, is a natural approach — but we show that it fails when the text length surpasses the cache size. We observe an interesting phenomenon, namely *attention sink*, that keeping the KV of initial tokens will largely recover the performance of window attention. In this paper, we first demonstrate that the emergence of *attention sink* is due to the strong attention scores towards initial tokens as a "sink" even if they are not semantically important. Based on the above analysis, we introduce StreamingLLM, an efficient framework that enables LLMs trained with a *finite length* attention window to generalize to *infinite sequence length* without any fine-tuning. We show that StreamingLLM can enable Llama-2, MPT, Falcon, and Pythia to perform stable and efficient language modeling with up to 4 million tokens and more. In addition, we discover that adding a placeholder token as a dedicated attention sink during pre-training can further improve streaming deployment. In streaming settings, StreamingLLM outperforms the sliding window recomputation baseline by up to 22.2× speedup. Code and datasets are provided in the link.

Prefix Sliding for efficient test-time scaling

Niklas Muennighoff[s] **Zhengyang Wang**[c] **Zeyi Chen**[w] **Weijia Shi**[w] **Binyuan Hui**
John Yang[s] **Dapeng Jiang**[w] **Mika Senghaas**[p] **Fares Obeid**[p] **Johannes Hagemann**[p]
Sami Jaghouar[p] **Ludwig Schmidt**[s] **Percy Liang**[s] **Jason Wei** **Andrew Y. Ng**[s]
Luke Zettlemoyer[w] **Yejin Choi**[s] **Mike Lewis**[w]
[s]Stanford University [c]University of California at Santa Barbara [p]Prime Intellect
[w]University of Washington
n.muennighoff@gmail.com

Abstract

Test-time scaling uses extra test-time compute to improve performance, such as letting language models reason longer when solving a problem. As models keep the entire reasoning trace in memory via full attention, hard tasks that need long thinking can be prohibitively expensive. However, we find most intermediate reasoning tokens lose importance as the model continues reasoning. This calls into question whether retaining them is worth the cost. Based on this insight, we propose Prefix Sliding, which discards tokens during reasoning that are not part of the prefix or the window of the last few thousand tokens. The prefix has key instructions and tools available to the model, while the most recent tokens are the current reasoning the model is working on. This caps the total memory requirement regardless of how long the model reasons, allowing for efficient long-horizon test-time scaling. Without training, Prefix Sliding can make existing models 3x faster while maintaining performance. Training with Prefix Sliding using reinforcement learning can achieve better performance by enabling scaling to reasoning traces beyond a hundred thousand tokens. Ablations show Prefix Sliding outperforms summarizing intermediate tokens or vanilla sliding window. Our code is at https://github.com/Muennighoff/prefix-sliding

Attention sink

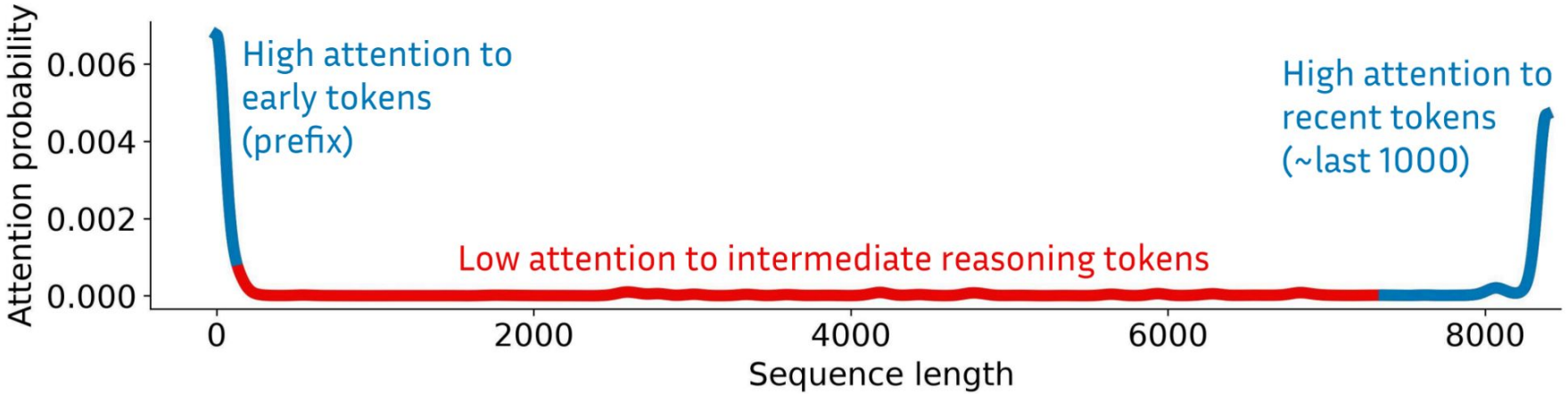


Figure 2: **The first and last few tokens receive most attention during reasoning with full attention.** We plot post-softmax attention probabilities averaged across layers and attention heads in Qwen3-1.7B for an AIME25 reasoning trace. Probabilities are Gaussian smoothed.

Studying internal representations

- Latent Planning Emerges with Scale // Transformers plan ahead
- Anthropic's Emergent Introspective Awareness in Large Language Models // the model settles on its rhyme before writing toward a full line of verse
- Anthropic's Verbalizable Representations Form a Global Workspace in Language Models // Models form a global workspace
- Anthropic's Emotion Concepts and their Function in a Large Language Model // Models form emotional concepts before responding, which shape what they say

Thank you!