The Era of BERT

CS 5624: Natural Language Processing

Spring 2025

https://tuvllms.github.io/nlp-spring-2025

Tu Vu

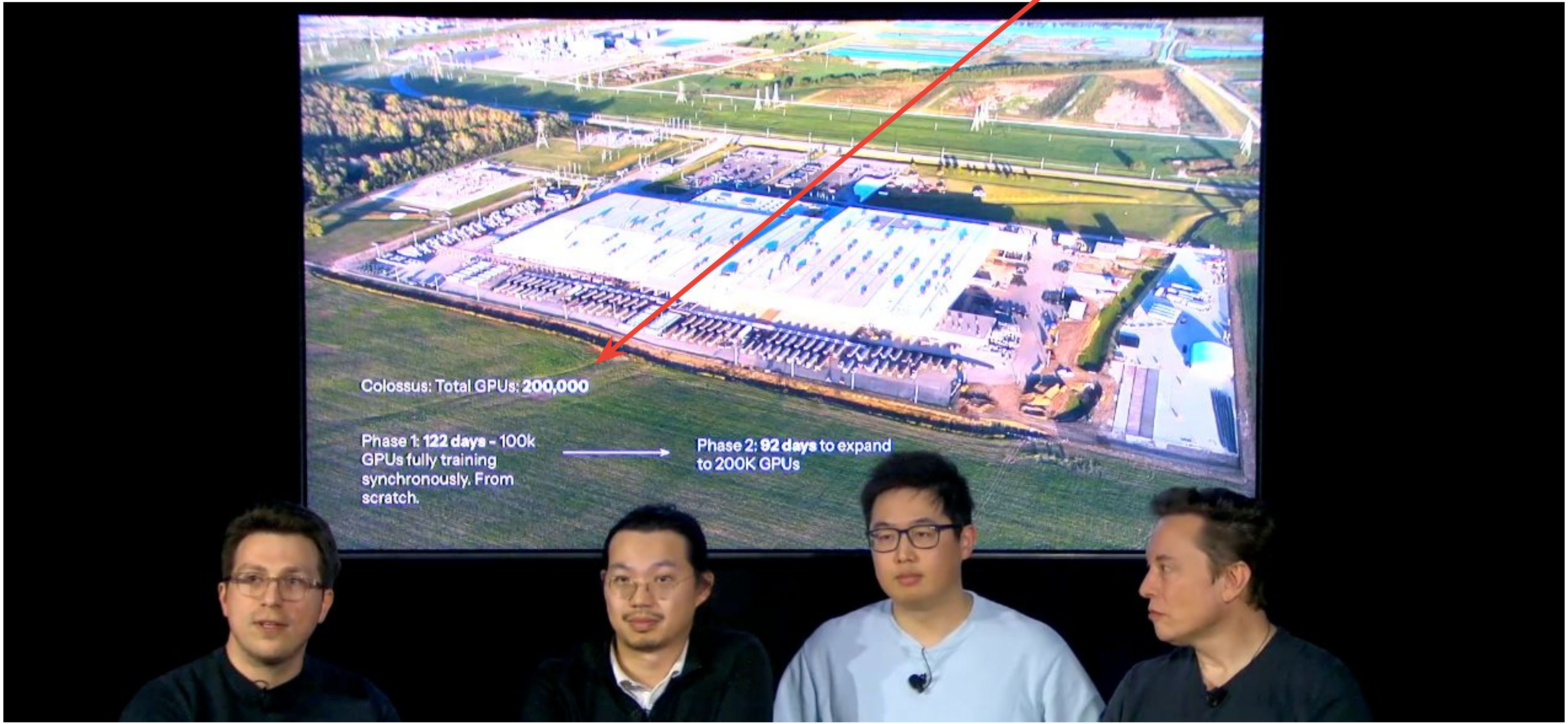
VIRGINIA TECH.

Logistics

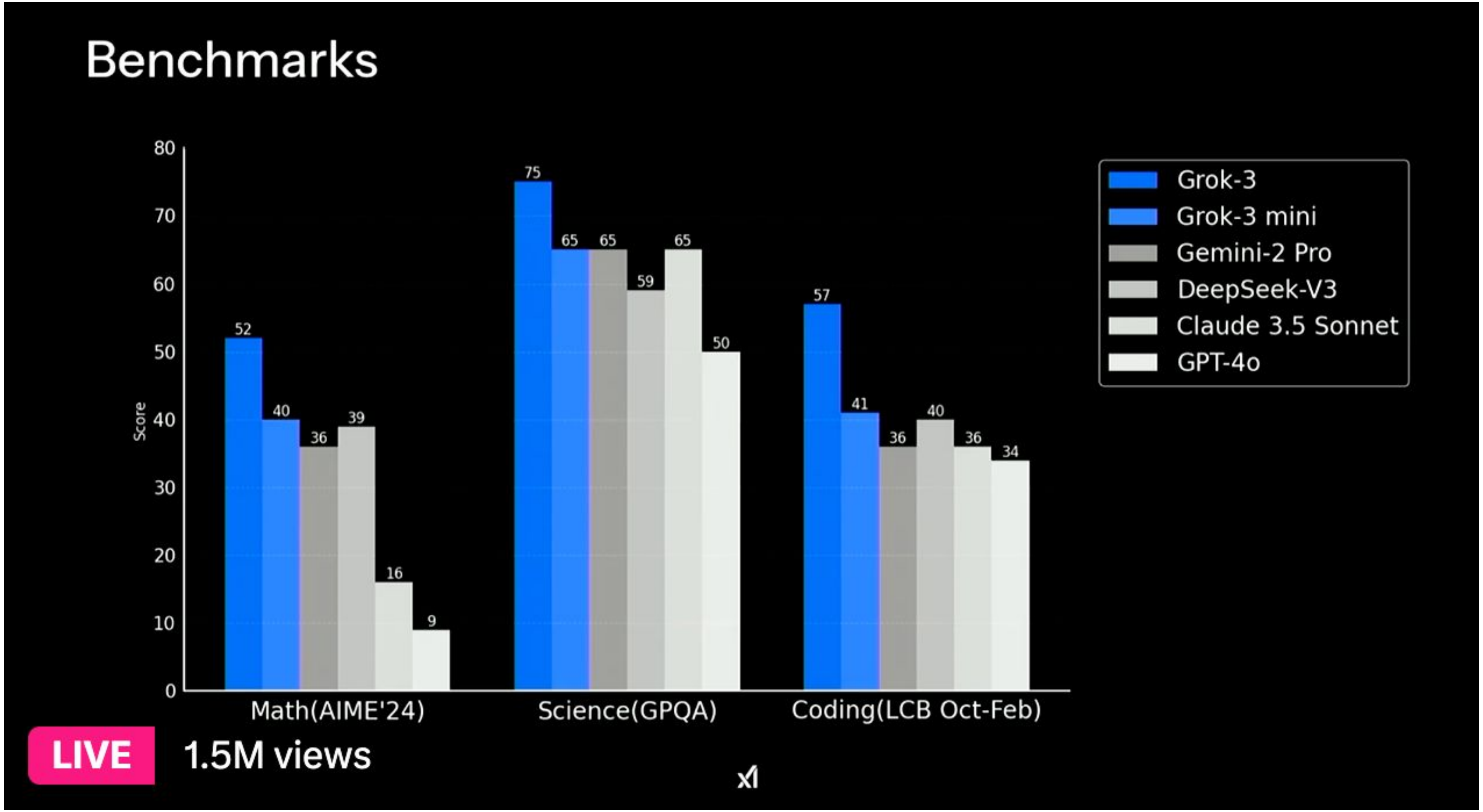
- Homework 1 & Quiz 1 are on their way
- 🚨 Final project proposal due on February 28 🚨
  - Template is on Piazza

Grok-3 came out last night

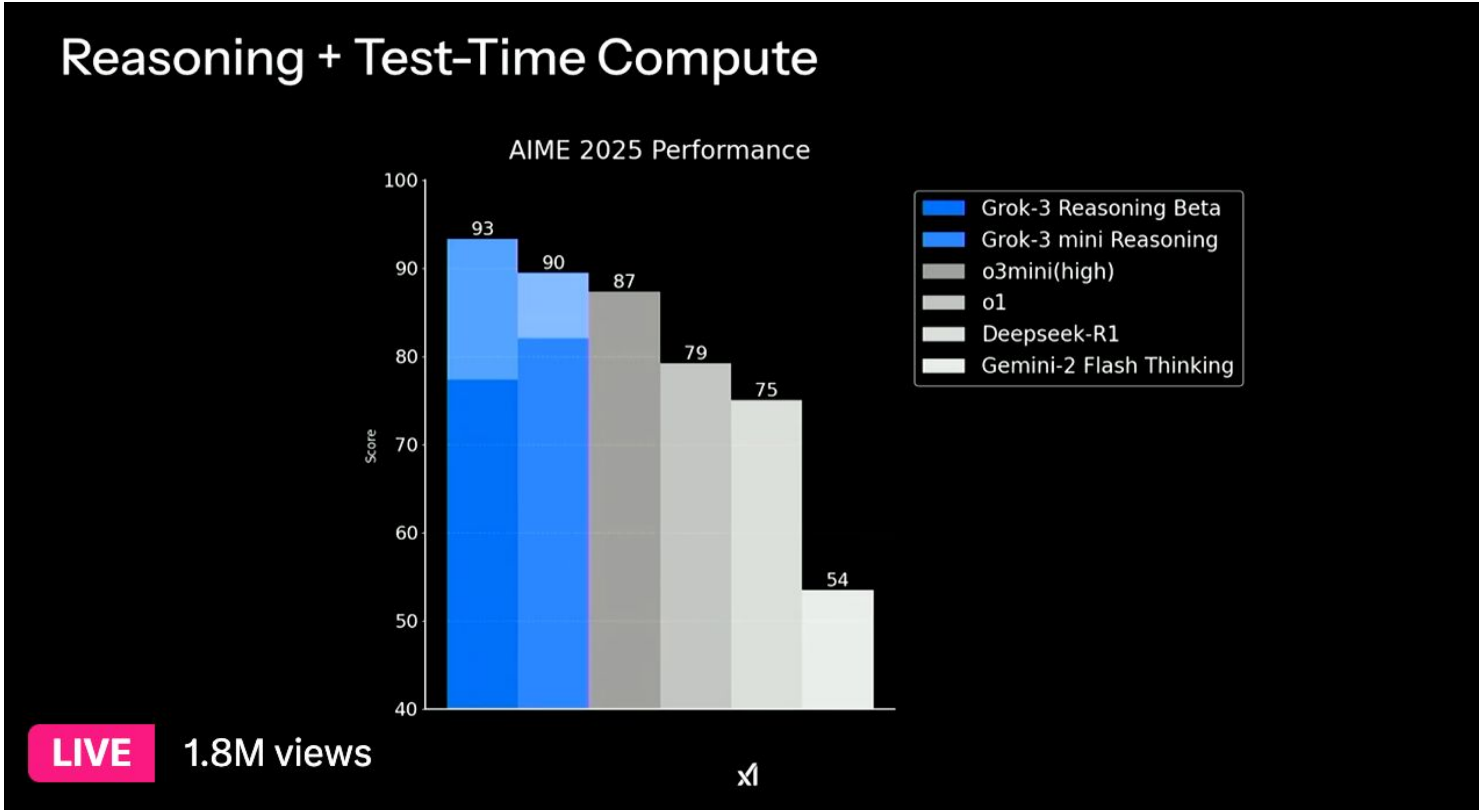
200,000 GPUs



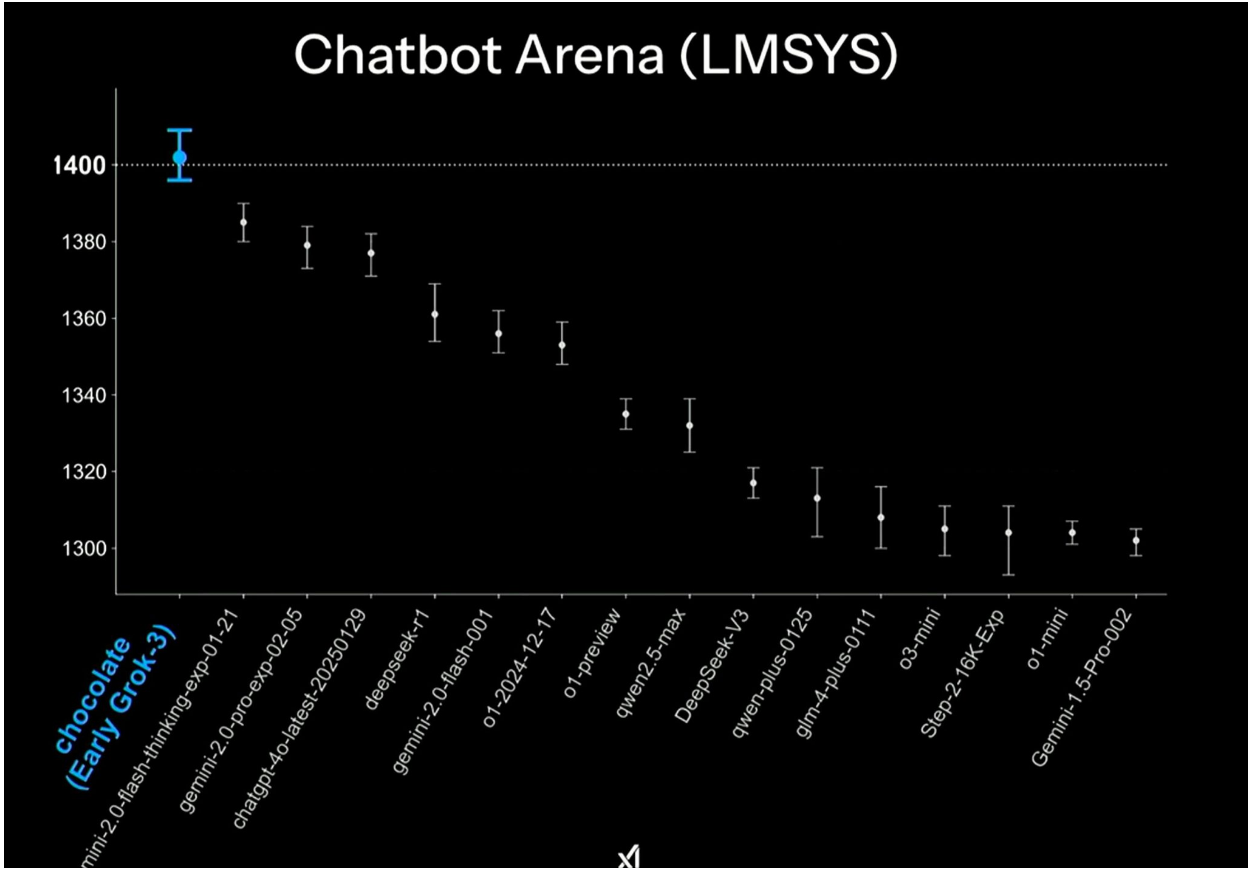
Grok-3


Benchmarks
80
70
60
50
40
30
20
10
0
Score
52
40
36
39
16
9
75
65
65
59
65
50
57
41
36
40
36
34
Math(AIME'24)
Science(GPQA)
Coding(LCB Oct-Feb)
Grok-3
Grok-3 mini
Gemini-2 Pro
DeepSeek-V3
Claude 3.5 Sonnet
GPT-4o
LIVE
1.5M views

Grok-3 (cont'd)

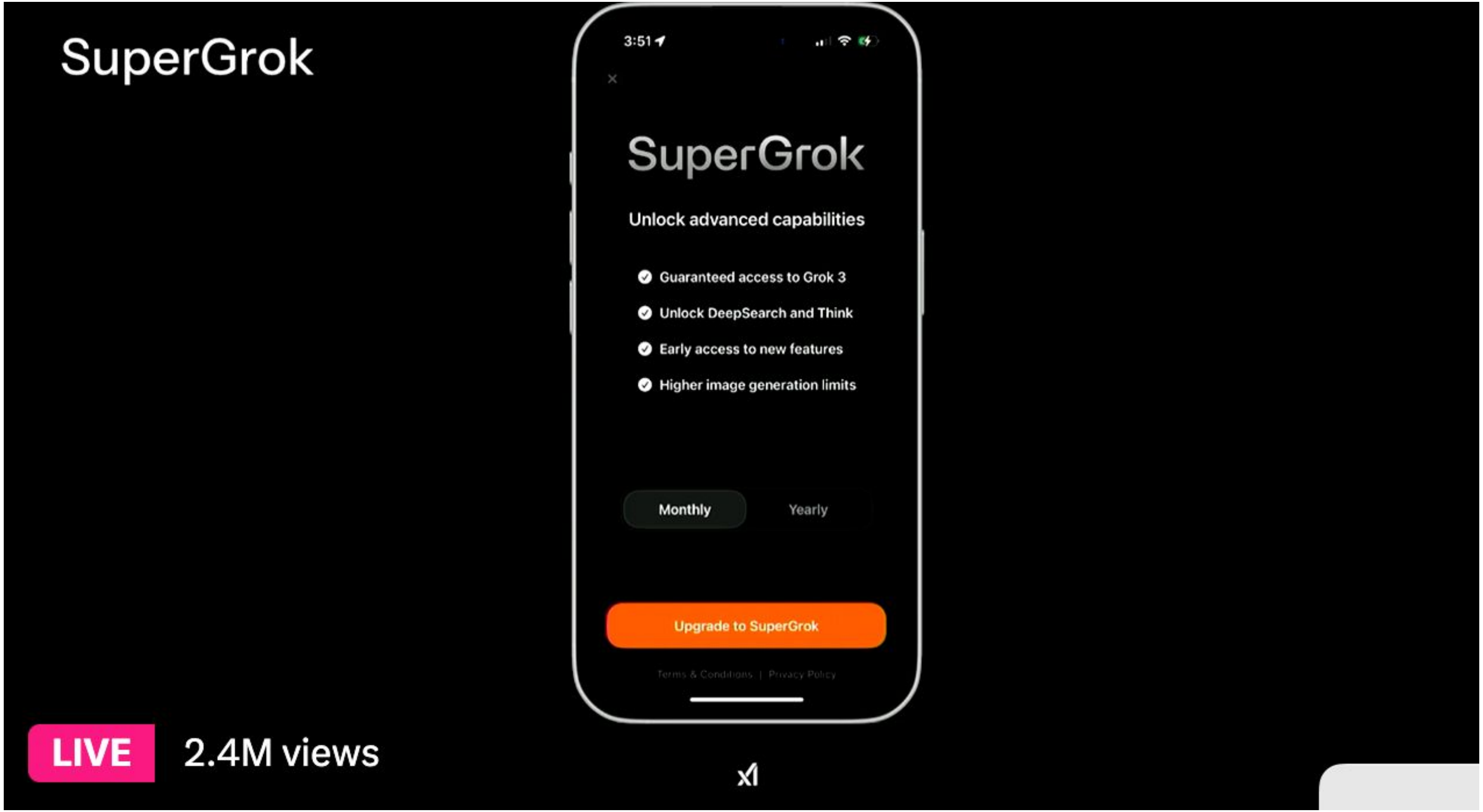

Reasoning + Test-Time Compute
AIME 2025 Performance
100
90
80
70
60
50
40
Score
93
90
87
79
75
54
Grok-3 Reasoning Beta
Grok-3 mini Reasoning
o3mini(high)
o1
Deepseek-R1
Gemini-2 Flash Thinking
LIVE
1.8M views

Grok-3 (cont'd)

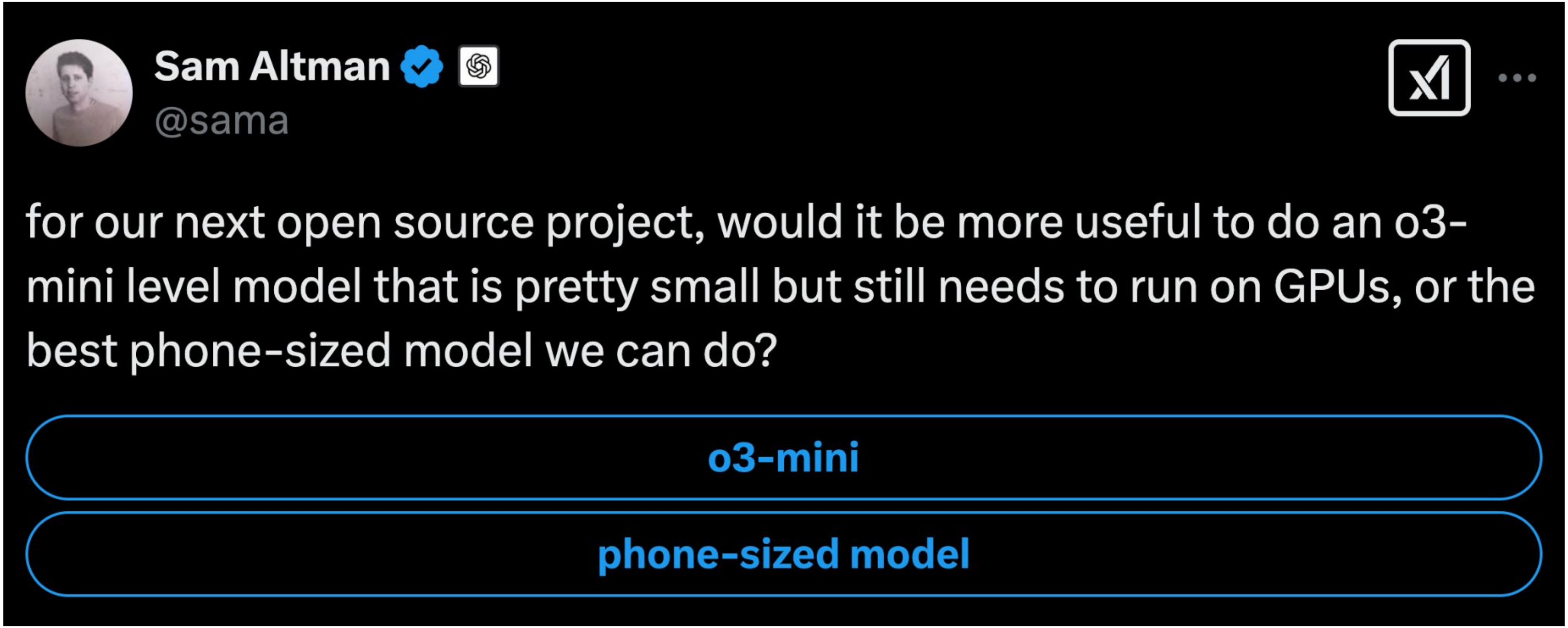


Chatbot Arena (LMSYS)
1400
1380
1360
1340
1320
1300
chocolate
(Early Grok-3)
mini-2.0-flash-thinking-exp-01-21
gemini-2.0-pro-exp-02-05
chatgpt-4o-latest-20250129
deepseek-r1
gemini-2.0-flash-001
o1-2024-12-17
o1-preview
qwen2.5-max
DeepSeek-V3
qwen-plus-0125
glm-4-plus-0111
o3-mini
Step-2-16K-Exp
o1-mini
Gemini-1.5-Pro-002

Grok-3 (cont'd)

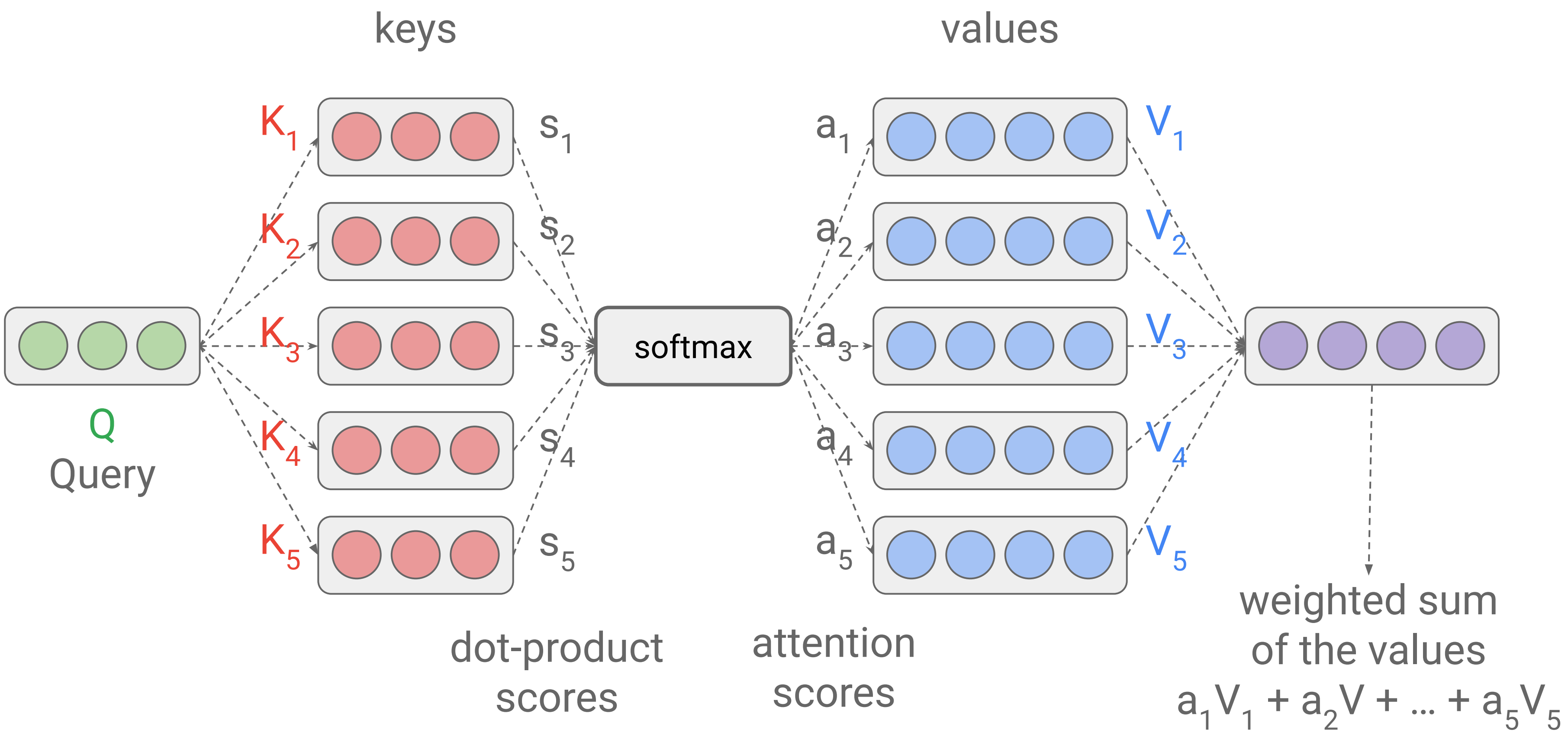

SuperGrok
3:51
SuperGrok
Unlock advanced capabilities
Guaranteed access to Grok 3
Unlock DeepSearch and Think
Early access to new features
Higher image generation limits
Monthly
Yearly
Upgrade to SuperGrok
Terms & Conditions | Privacy Policy
LIVE
2.4M views

Big tech companies are considering open-sourcing older AI models

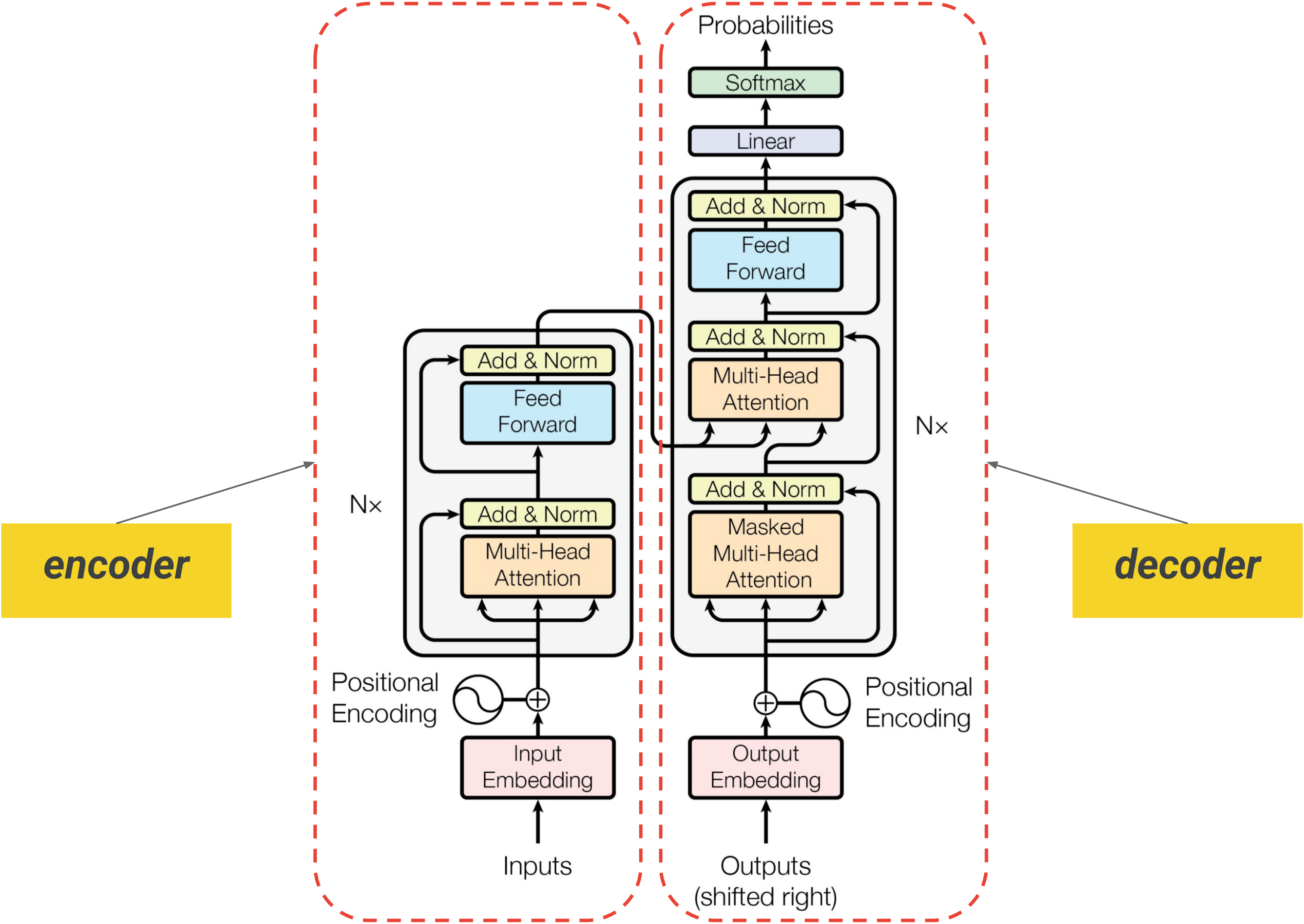


Transformers review

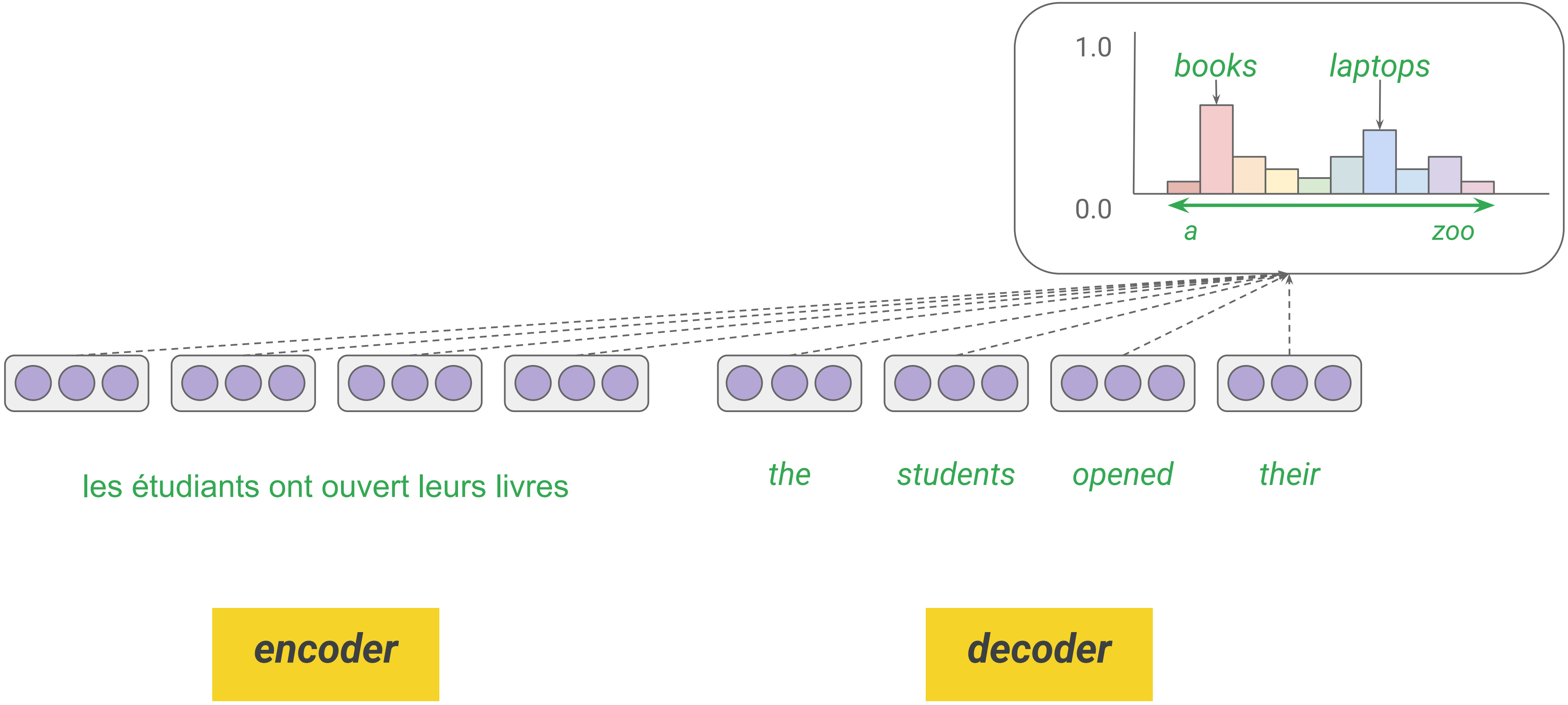
Attention mechanism



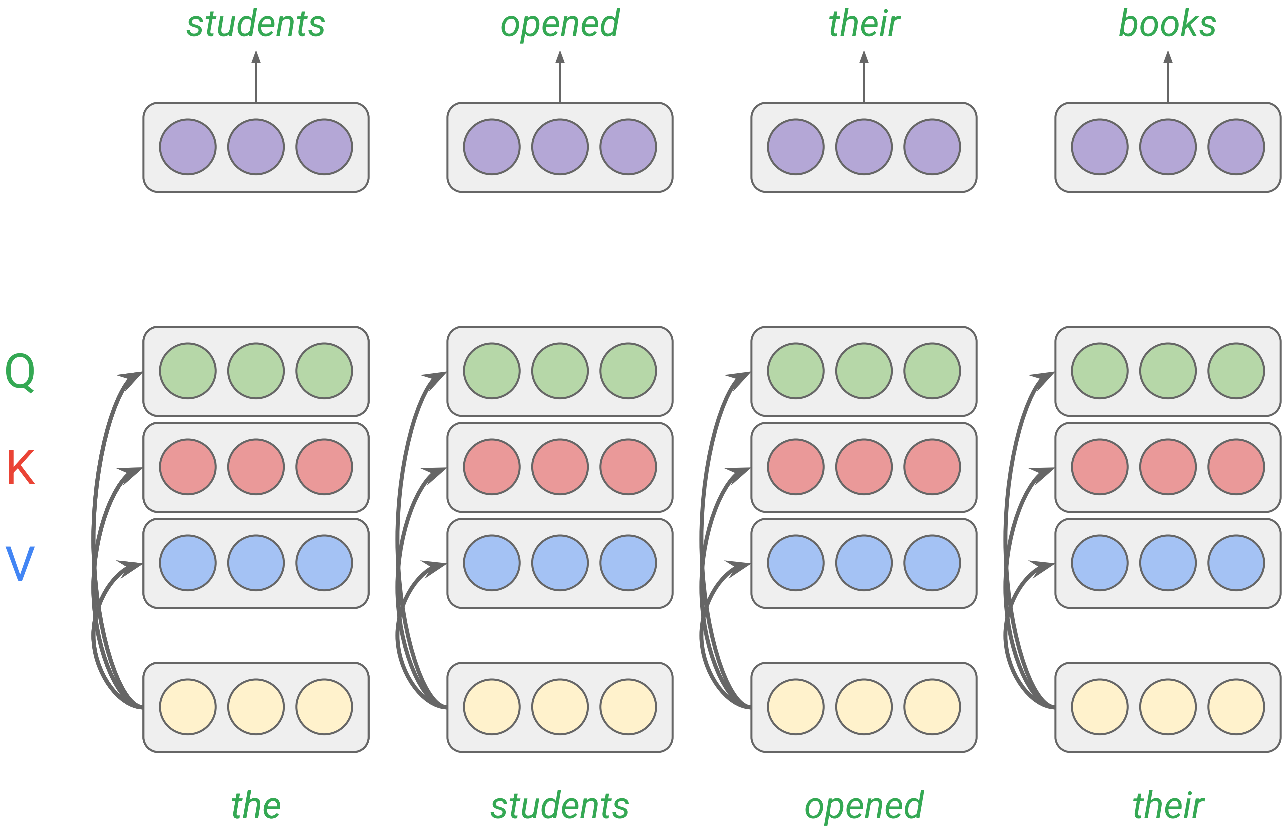
Transformers architecture



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


1.0
books
laptops
0.0
a
zoo
les étudiants ont ouvert leurs livres
the
students
opened
their
encoder
decoder

Self-attention



$$Q = X \cdot W_Q$$

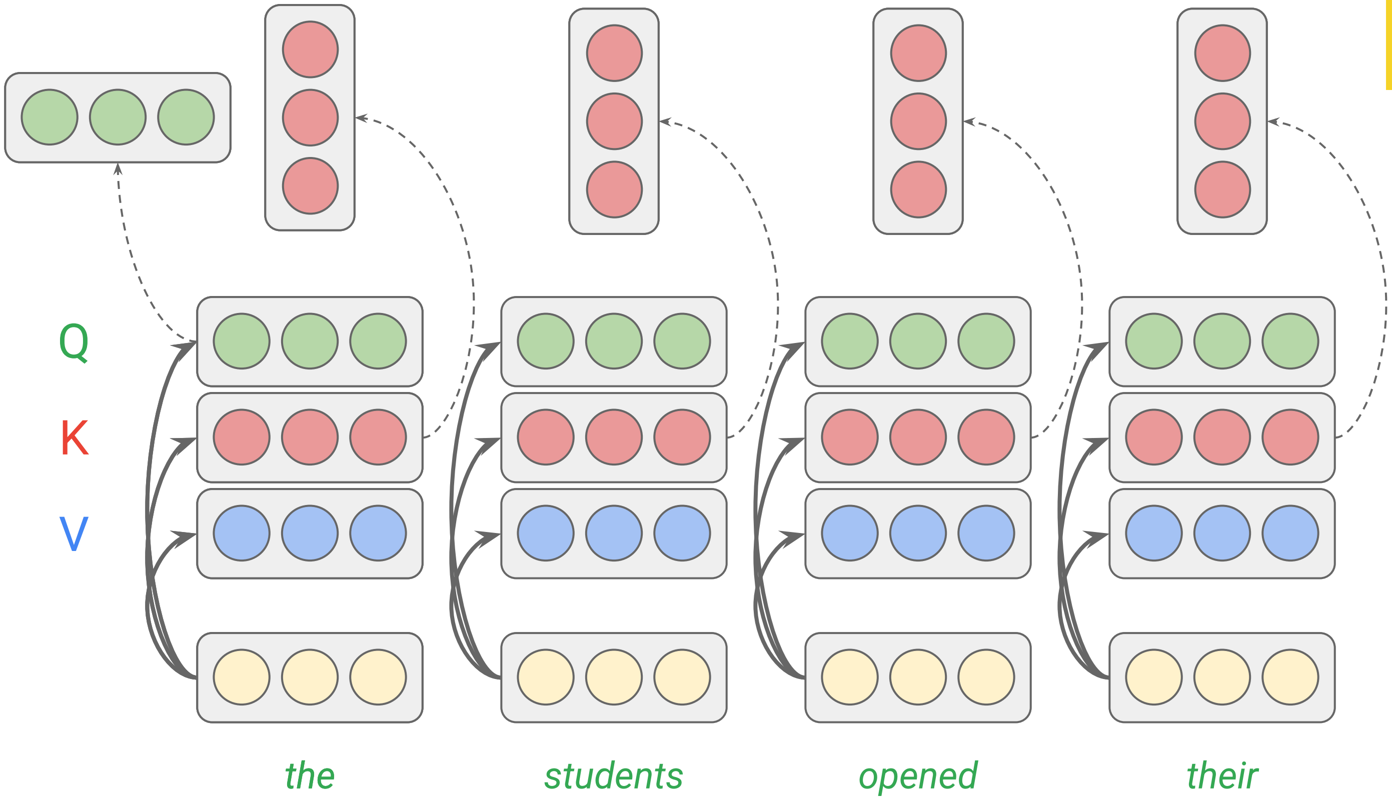
$$K = X \cdot W_K$$

$$V = X \cdot W_V$$

linear projections

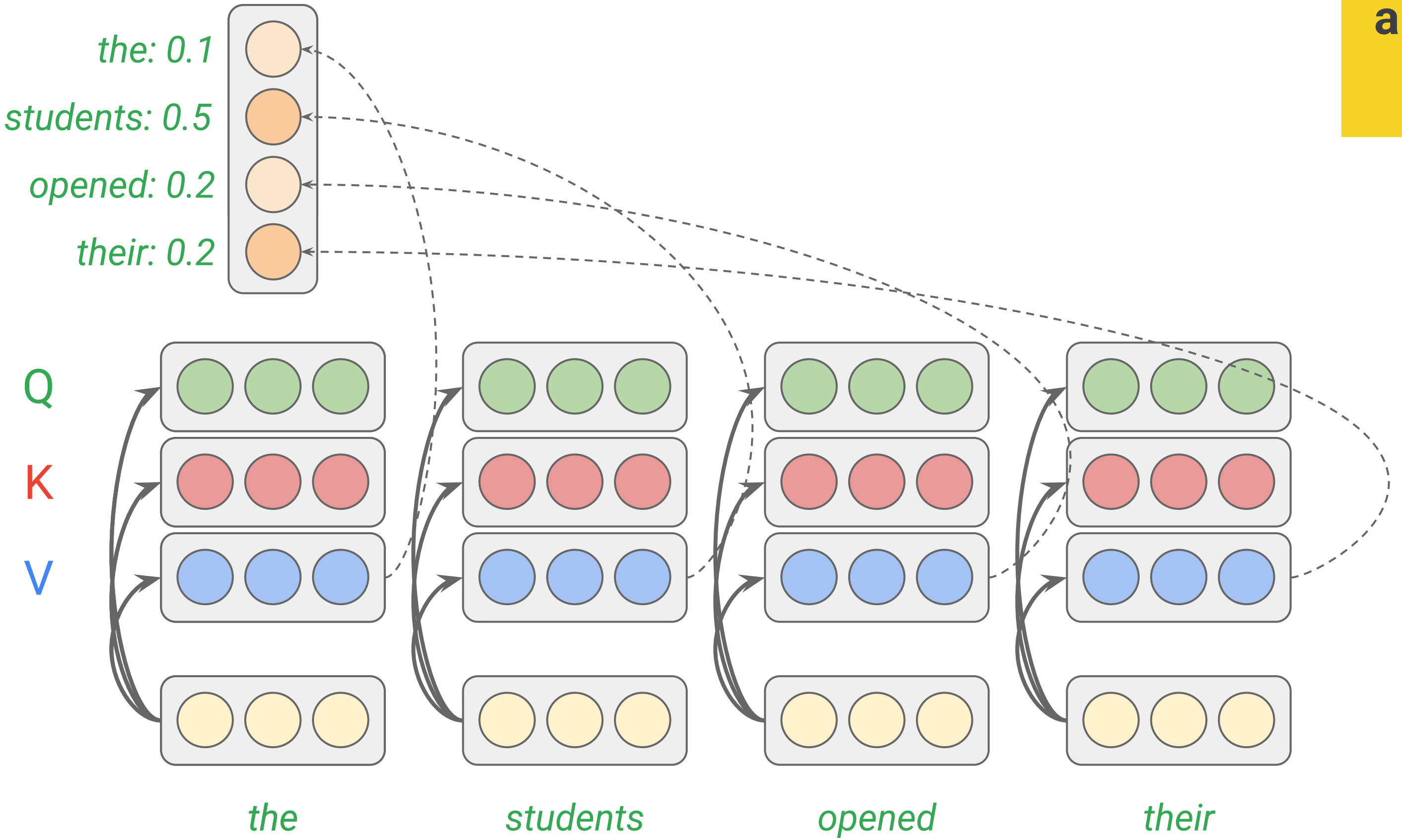
Self-attention (cont'd)

all computations are parallelized



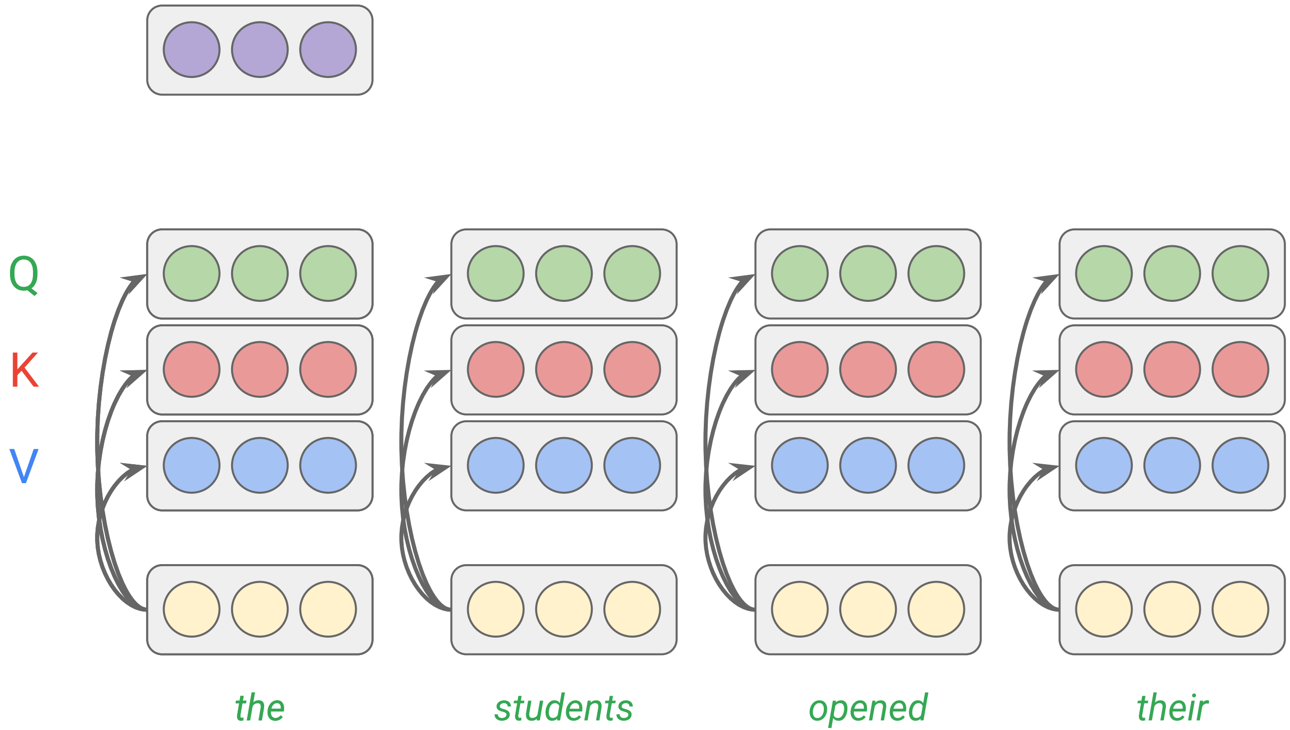
Self-attention (cont'd)

all **computations are parallelized**



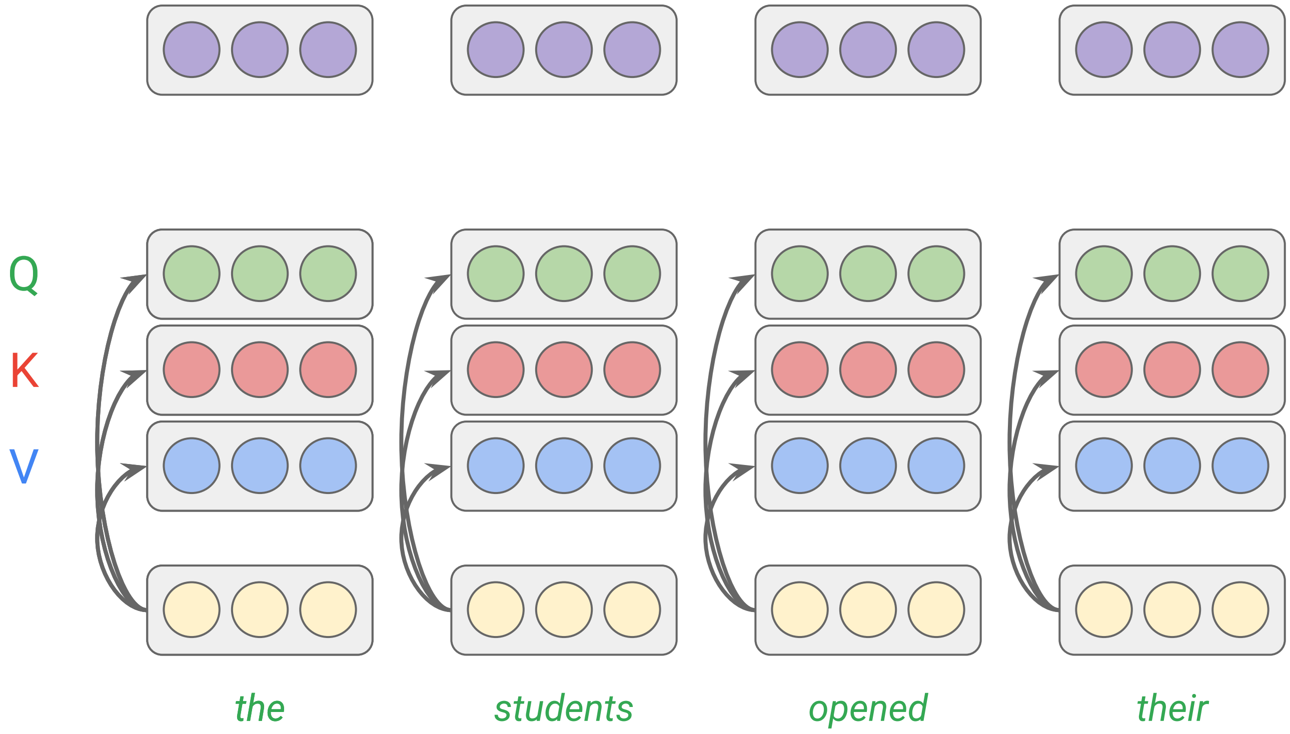
Self-attention (cont'd)

***all* computations are parallelized**



Self-attention (cont'd)

***all* computations are parallelized**



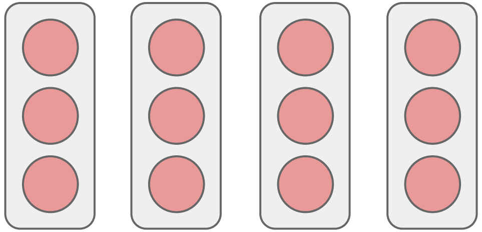
All computations are parallelized

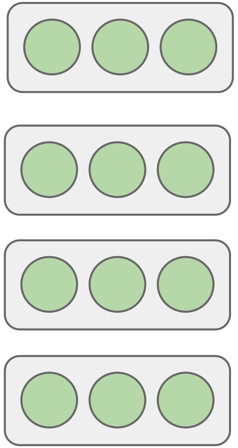
$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

d_k: scaling factor

large products push the softmax function into regions where it has extremely small gradients

Self-attention in the decoder

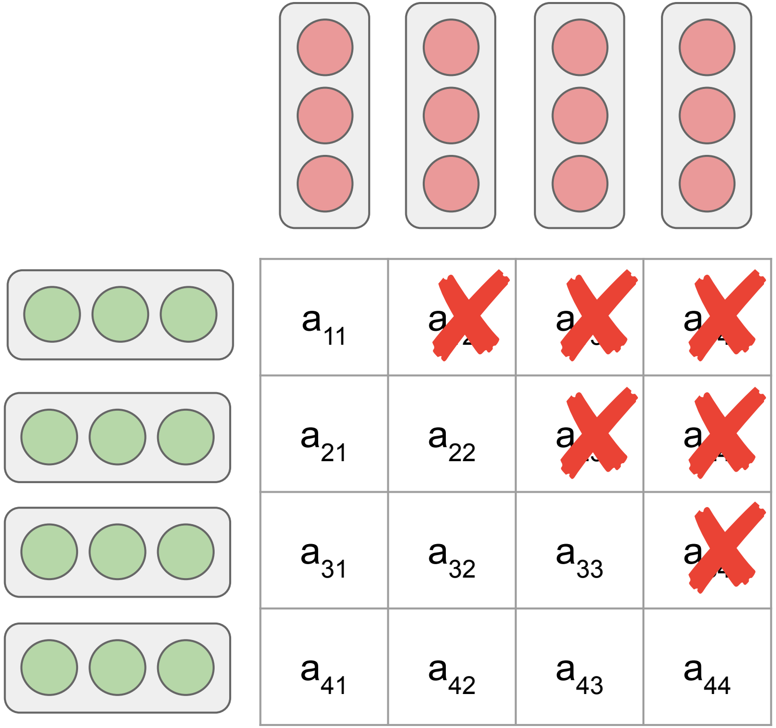




s_{11}	$-\infty$	$-\infty$	$-\infty$
s_{21}	s_{22}	$-\infty$	$-\infty$
s_{31}	s_{32}	s_{33}	$-\infty$
s_{41}	s_{42}	s_{43}	s_{44}

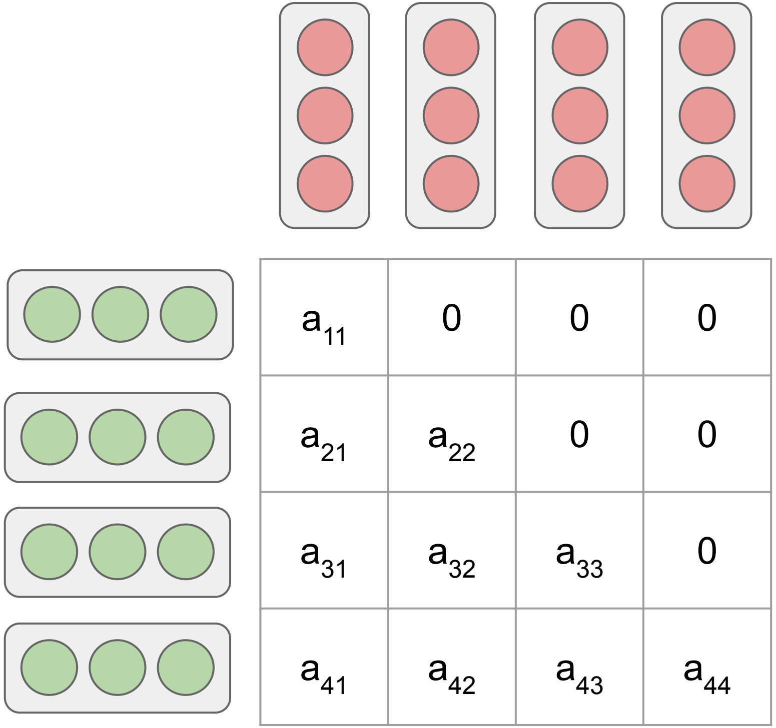
masking out (setting to $-\infty$) all values in the input of the softmax which correspond to illegal connections

Self-attention in the decoder (cont'd)



masking out all values in the input of the softmax which correspond to illegal connections

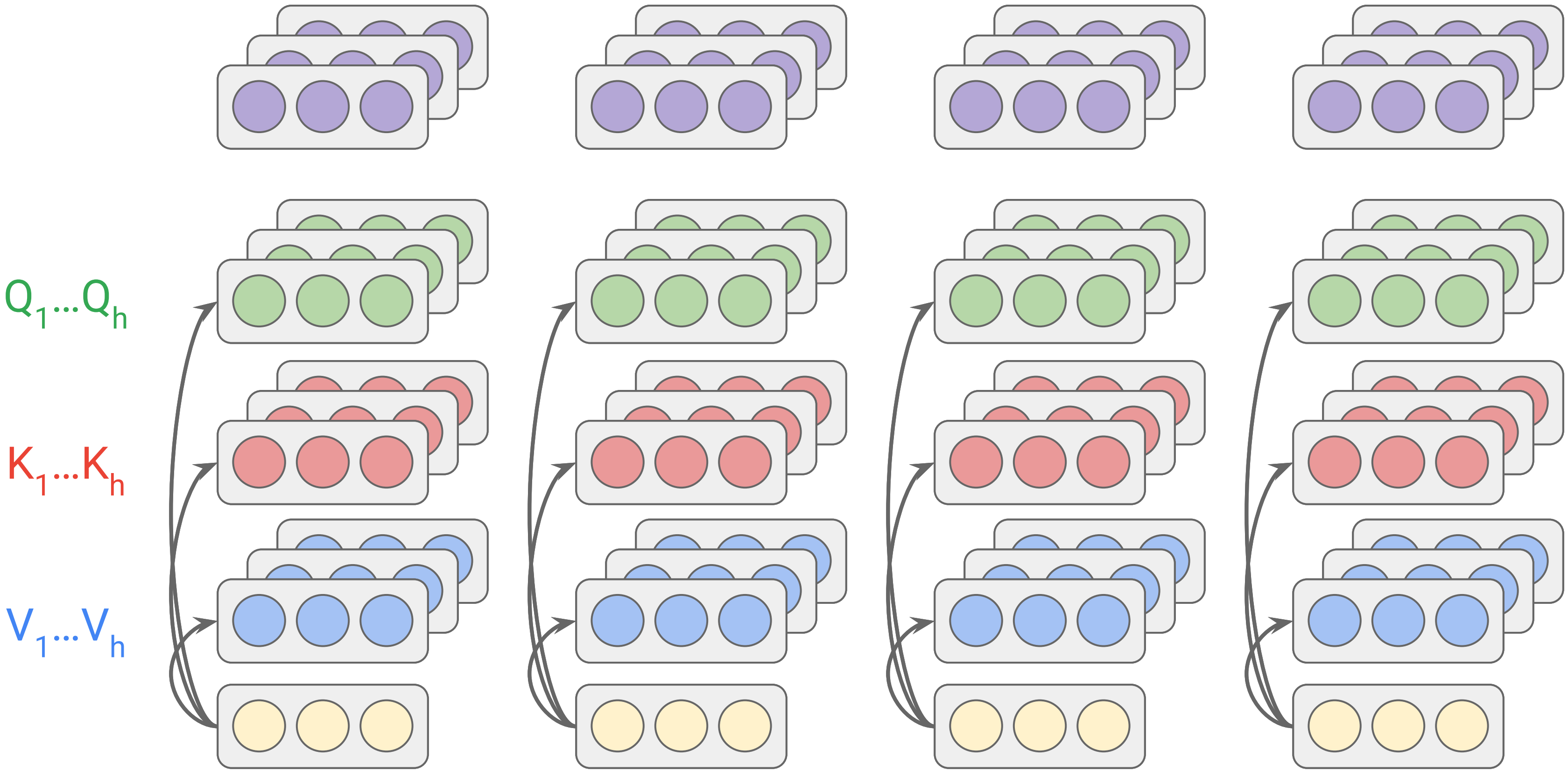
Self-attention in the decoder (cont'd)



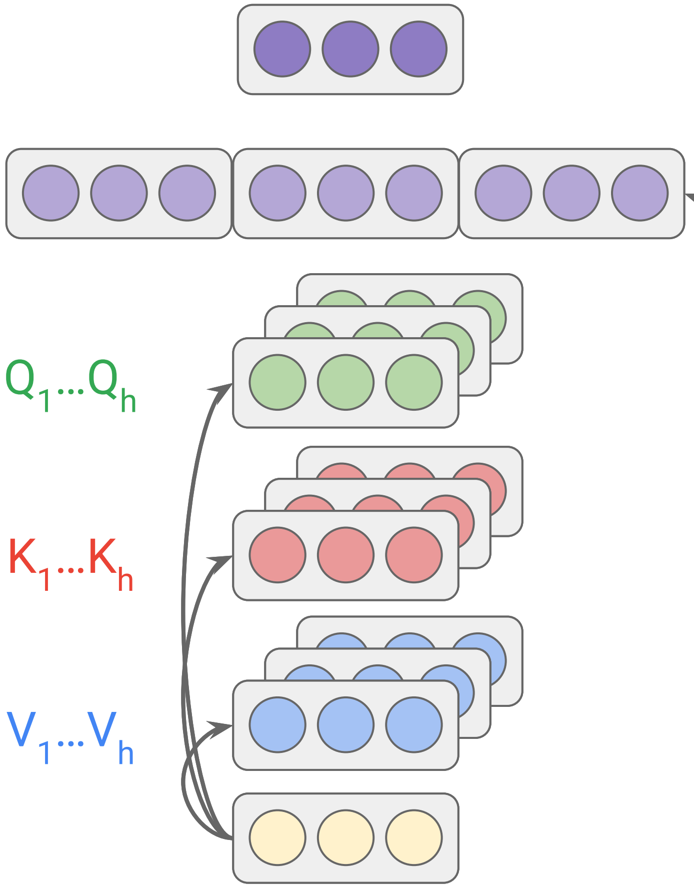
a_{11}	0	0	0
a_{21}	a_{22}	0	0
a_{31}	a_{32}	a_{33}	0
a_{41}	a_{42}	a_{43}	a_{44}

masking out all values in the input of the softmax which correspond to illegal connections

Multi-head attention

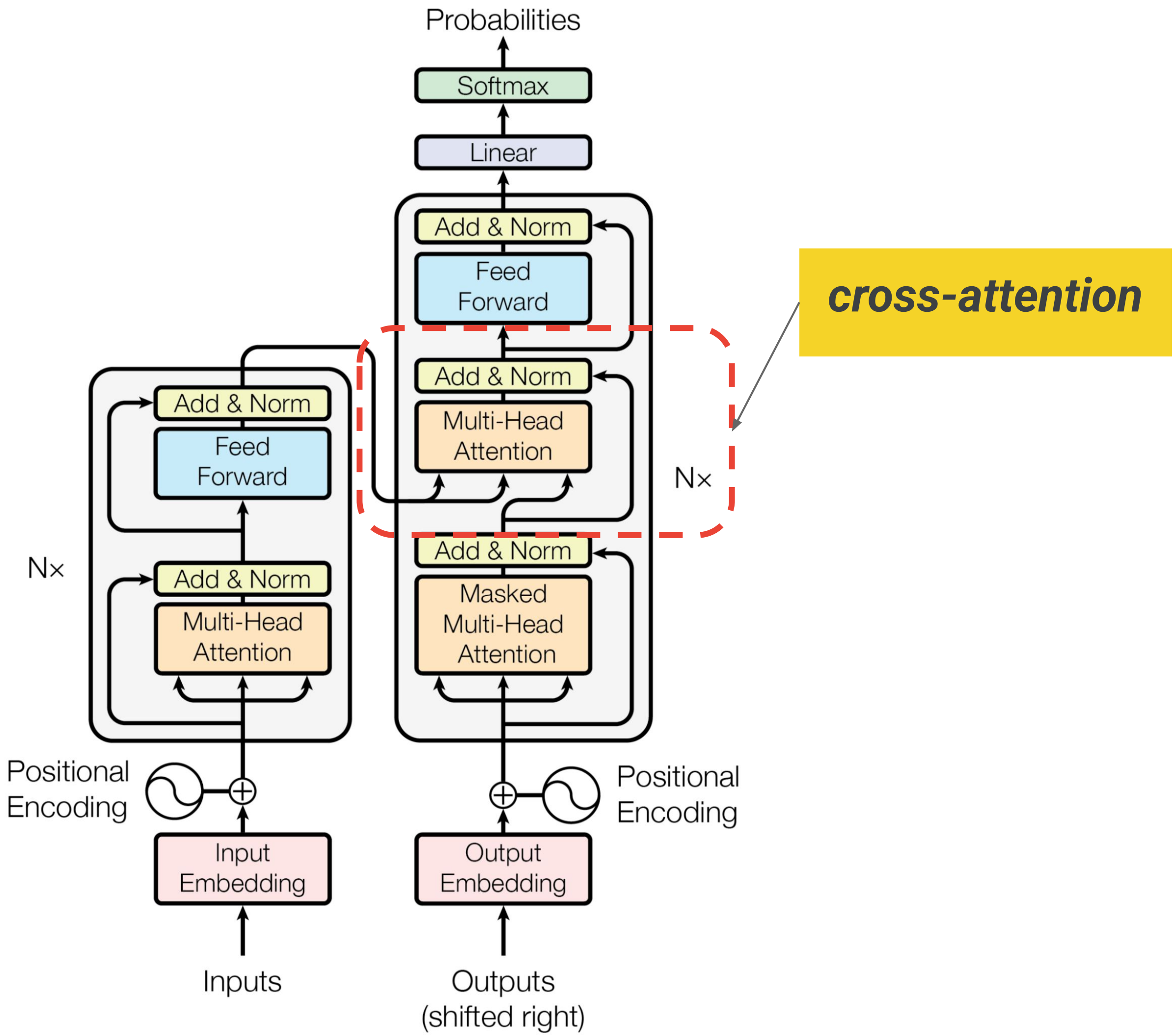


Multi-head attention (cont'd)

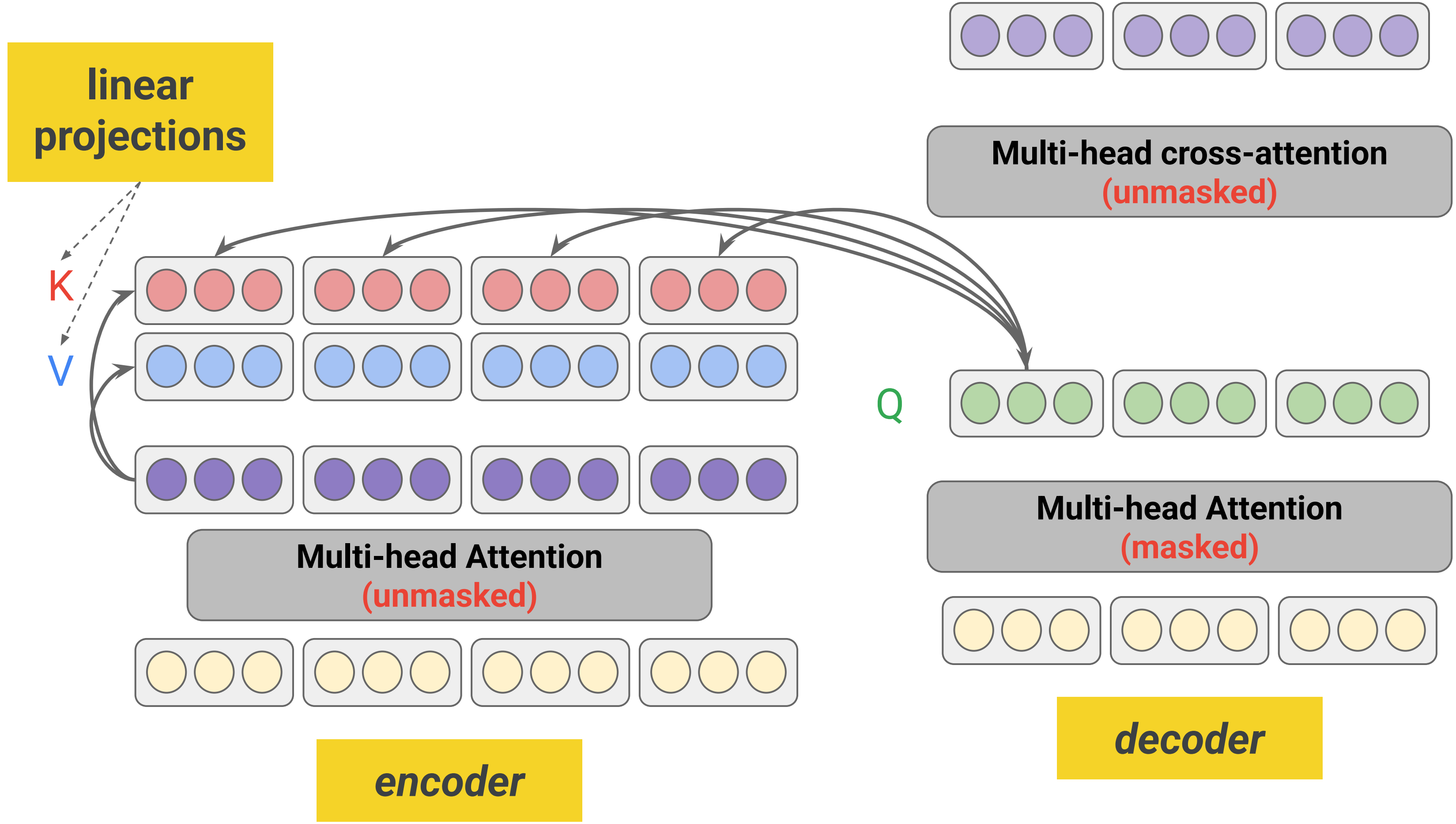


These output values are concatenated and once again projected

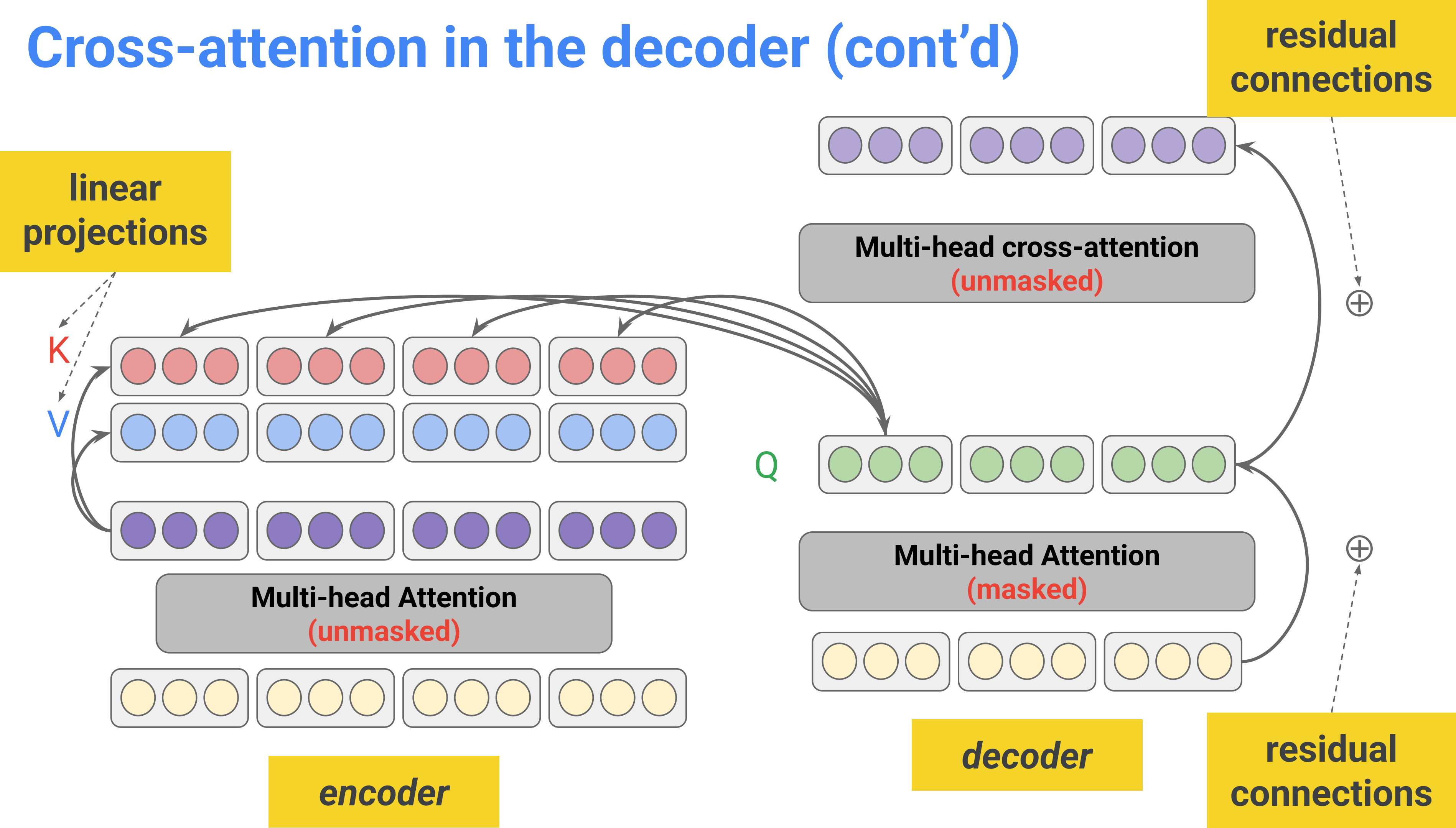
Cross-attention in the decoder



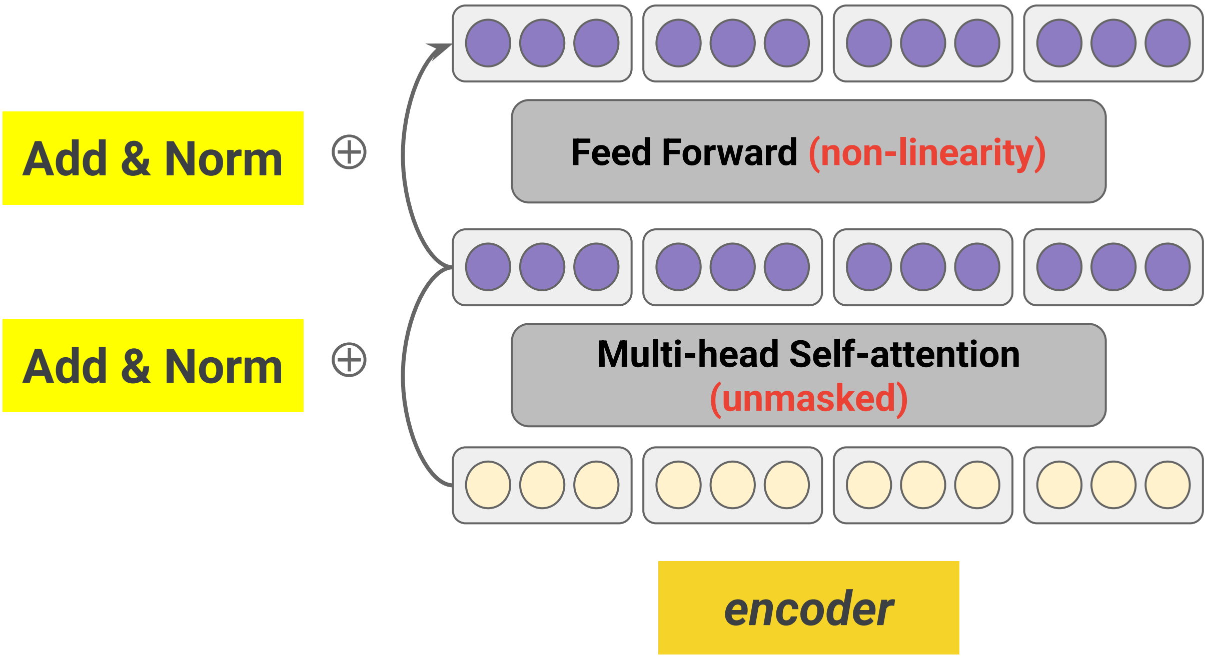
Cross-attention in the decoder



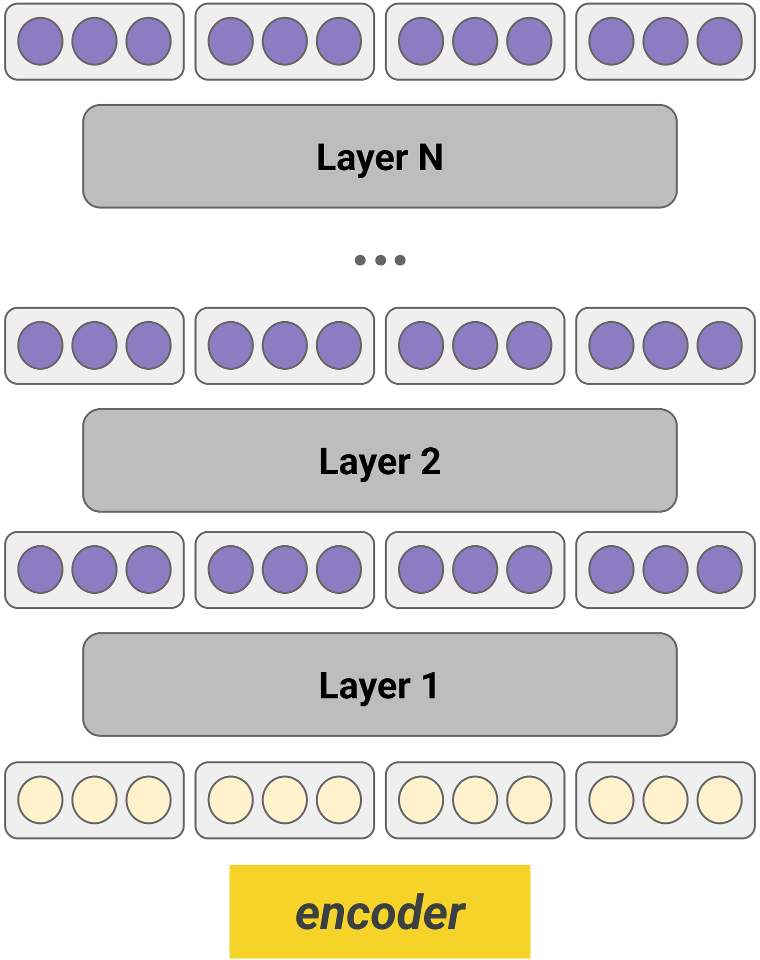
Cross-attention in the decoder (cont'd)



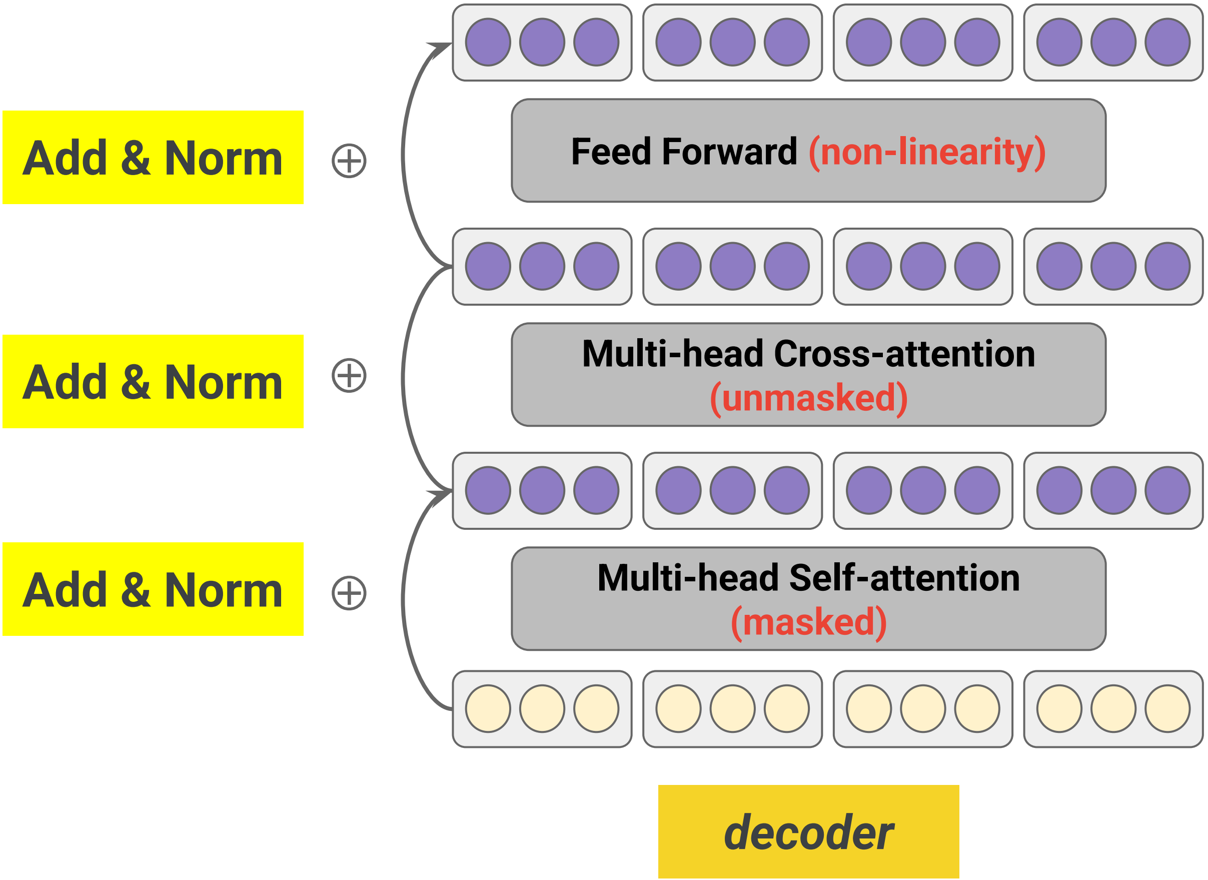
Encoder (one layer)



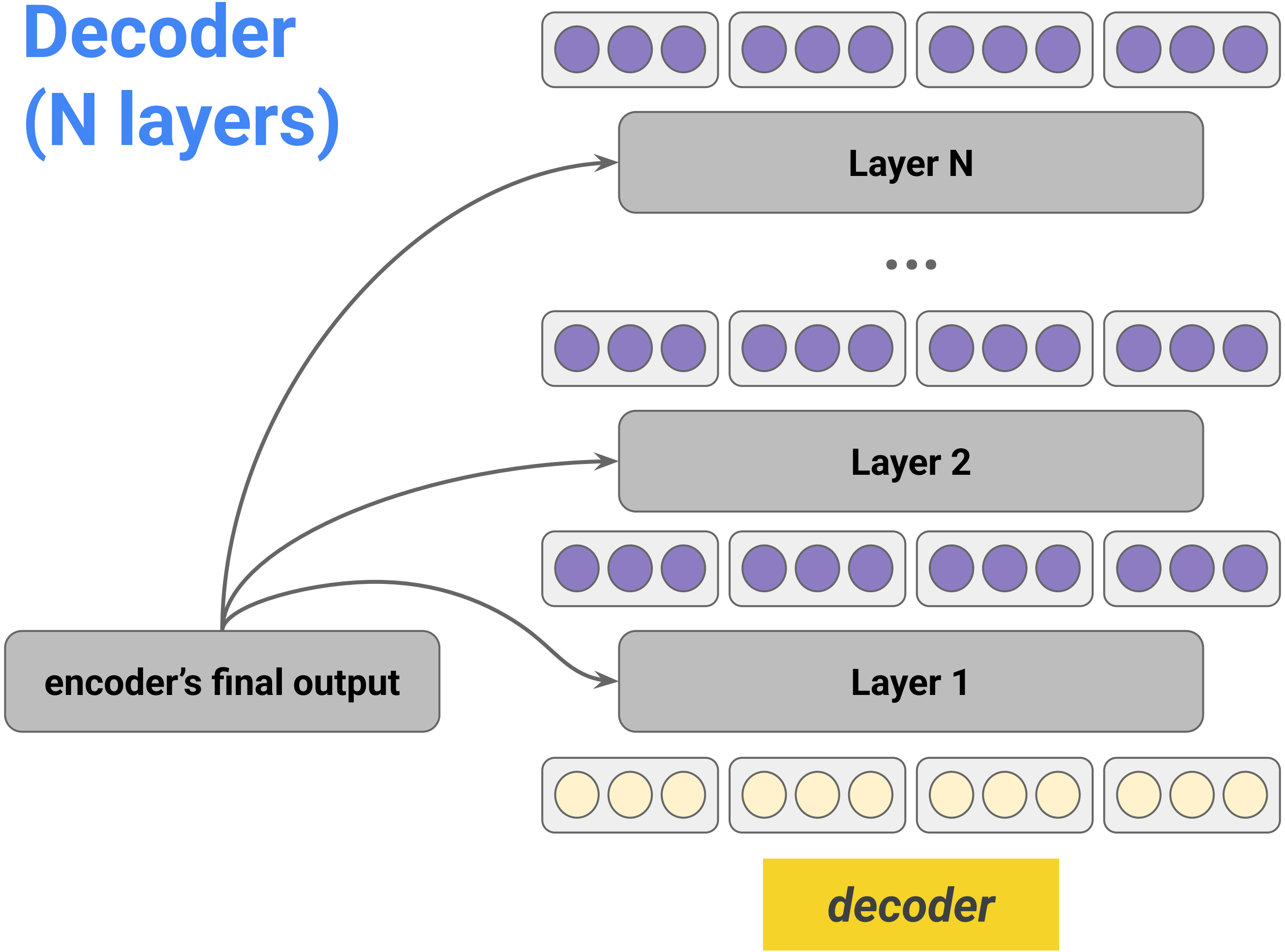
Encoder (N layers)



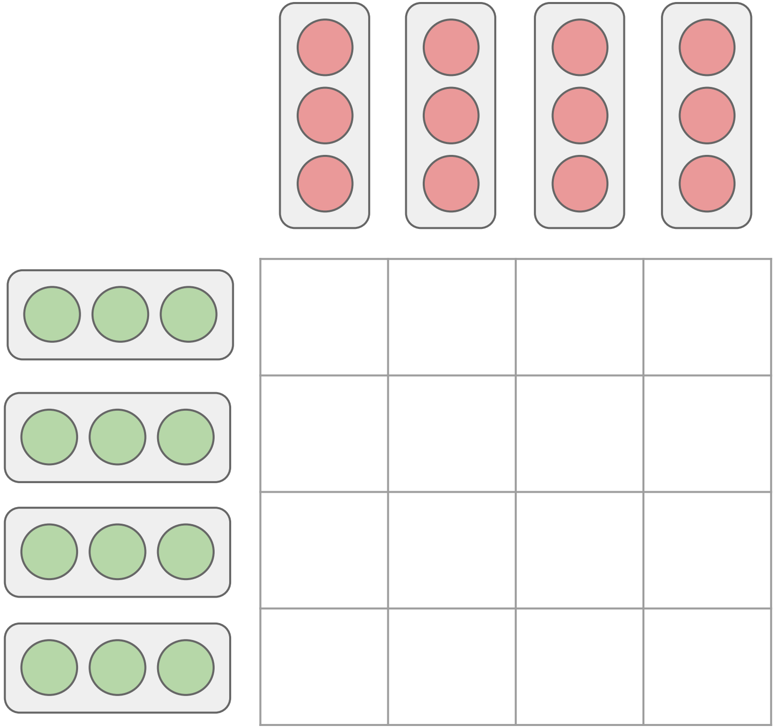
Decoder (one layer)



Decoder (N layers)



Quadratic complexity



The time complexity of self-attention is quadratic in the input length $O(n^2)$

Different model architectures

- Encoder-only
  - **BERT**
- Encoder-decoder
  - T5
- Decoder-only
  - GPT

Image created by Gemini

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

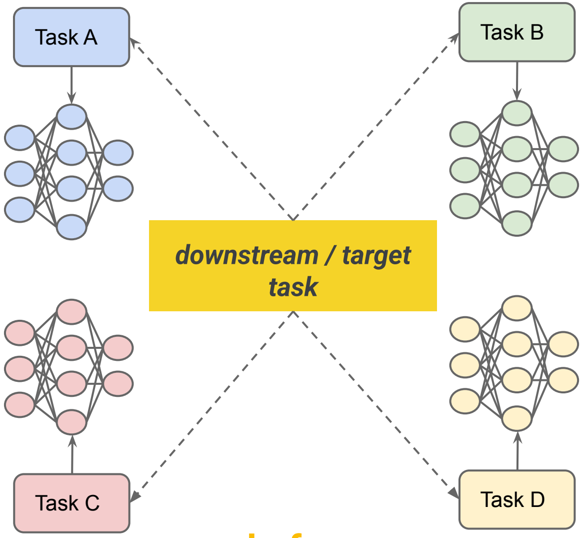
Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**

Google AI Language

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

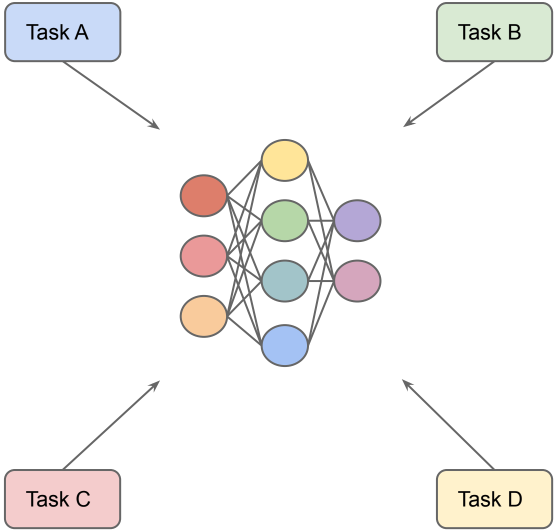
A learning paradigm shift

training task-specific models
from scratch



before
BERT

pretraining and then adapting



since
BERT

Neural network diagrams adapted from Colin Raffel's talk at Stanford MLSys Seminars

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

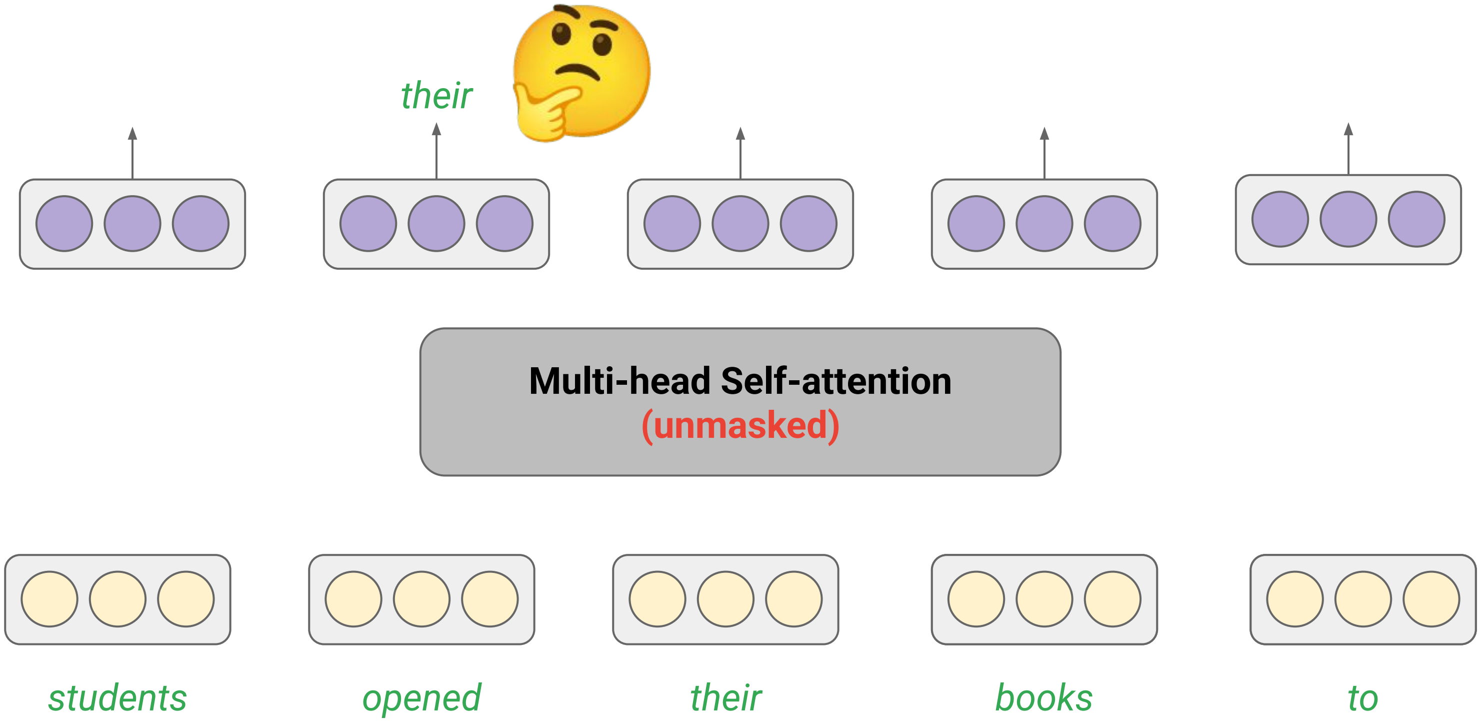
[†]Allen Institute for Artificial Intelligence
[*]Paul G. Allen School of Computer Science & Engineering, University of Washington

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)

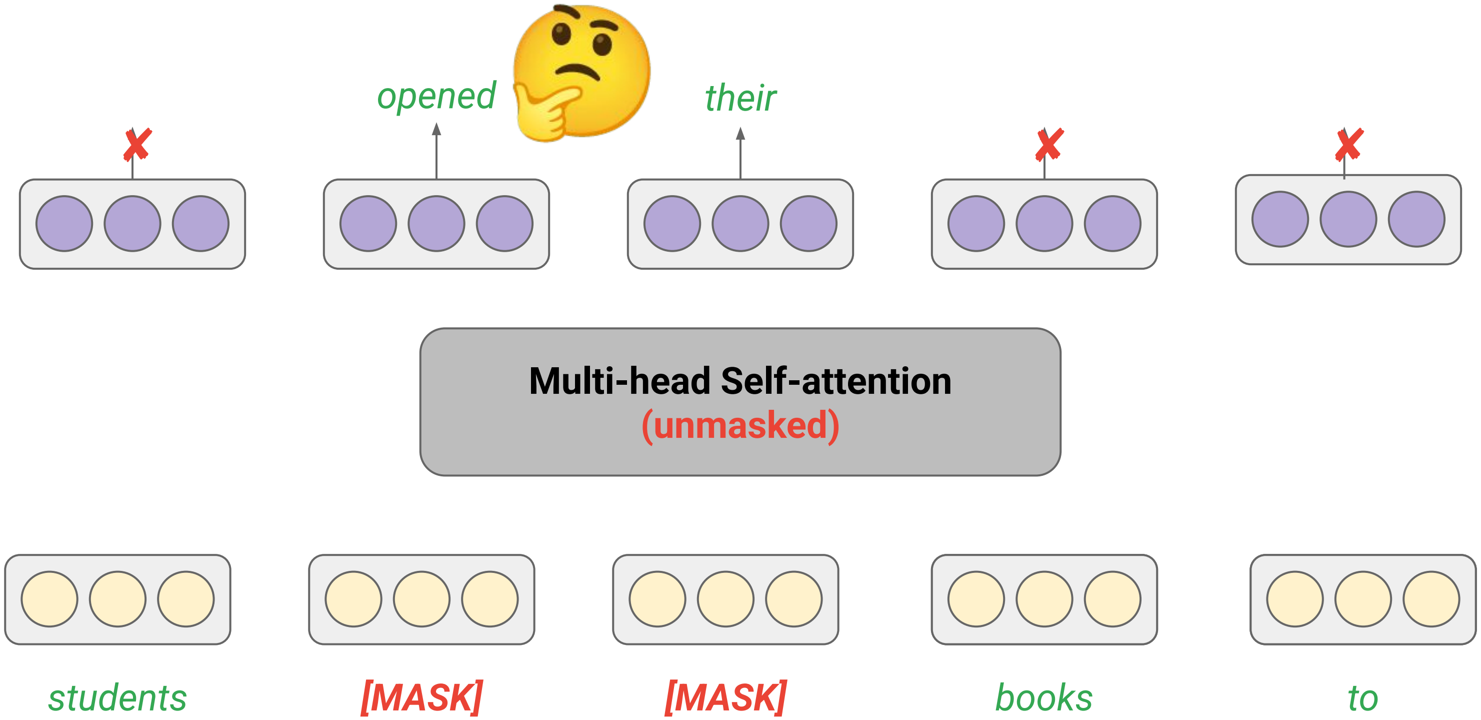
Pretraining

Language modeling using a Transformer encoder



Masked language modeling

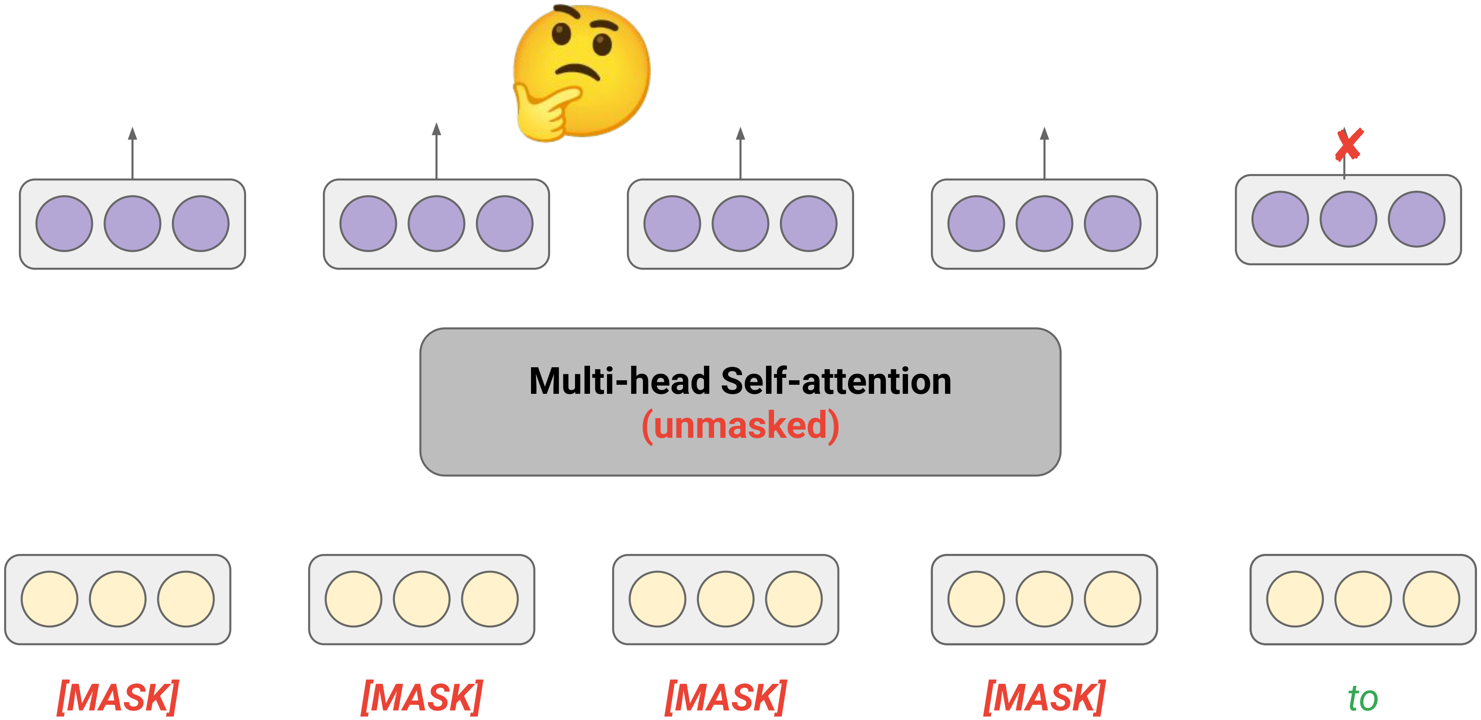
cloze task



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

What if we mask more tokens?

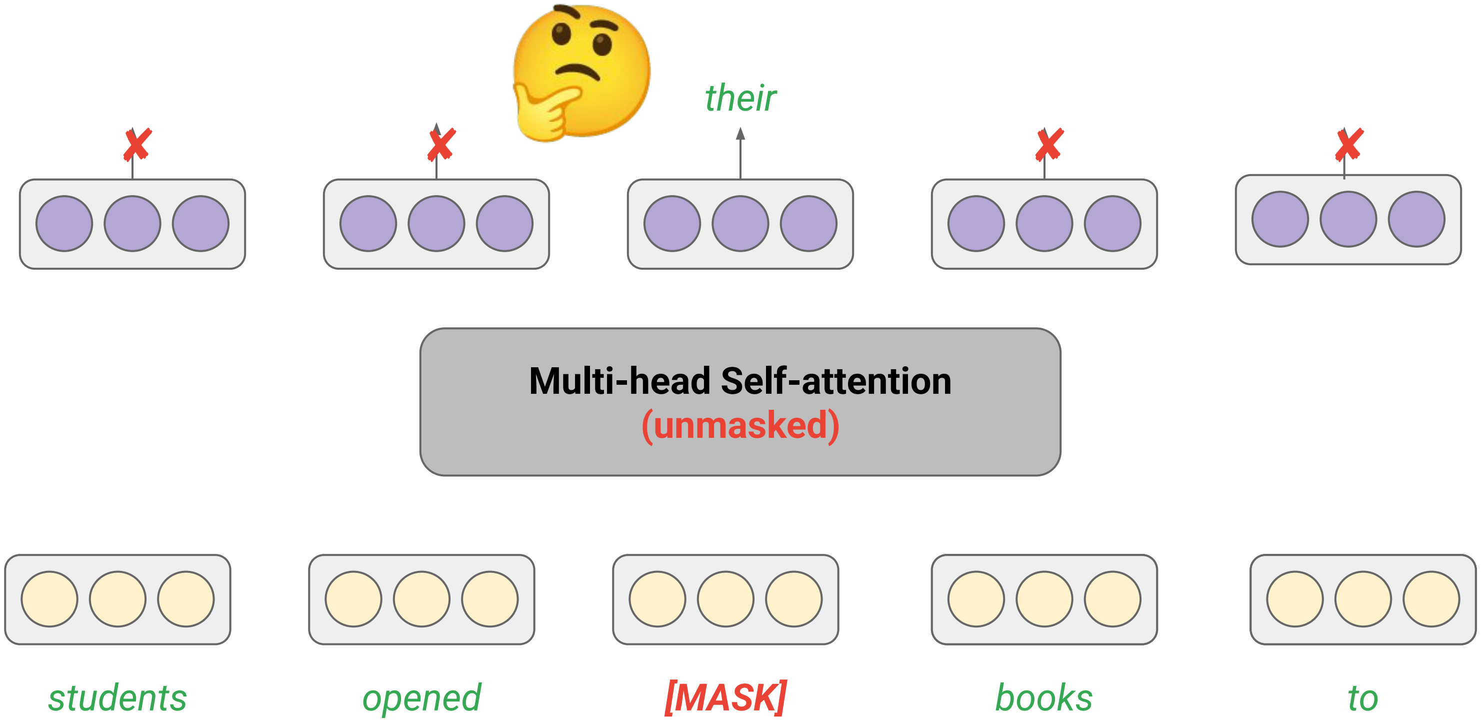
cloze task



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

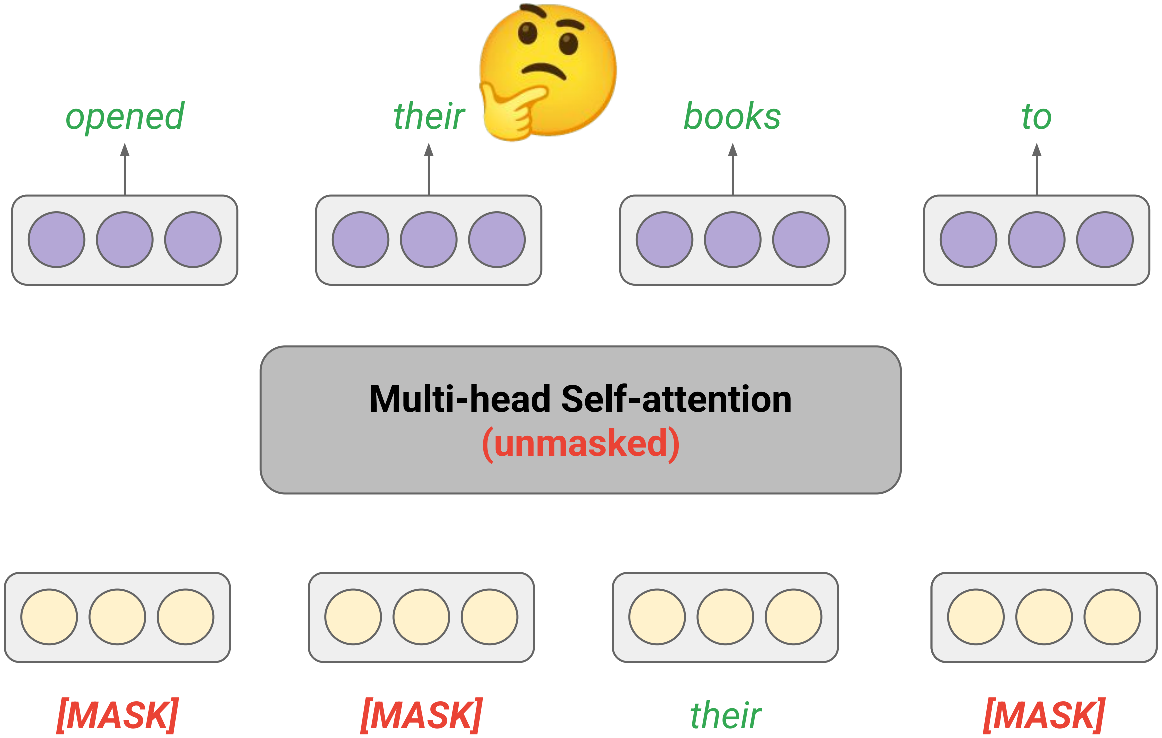
What if we mask less tokens?

cloze task



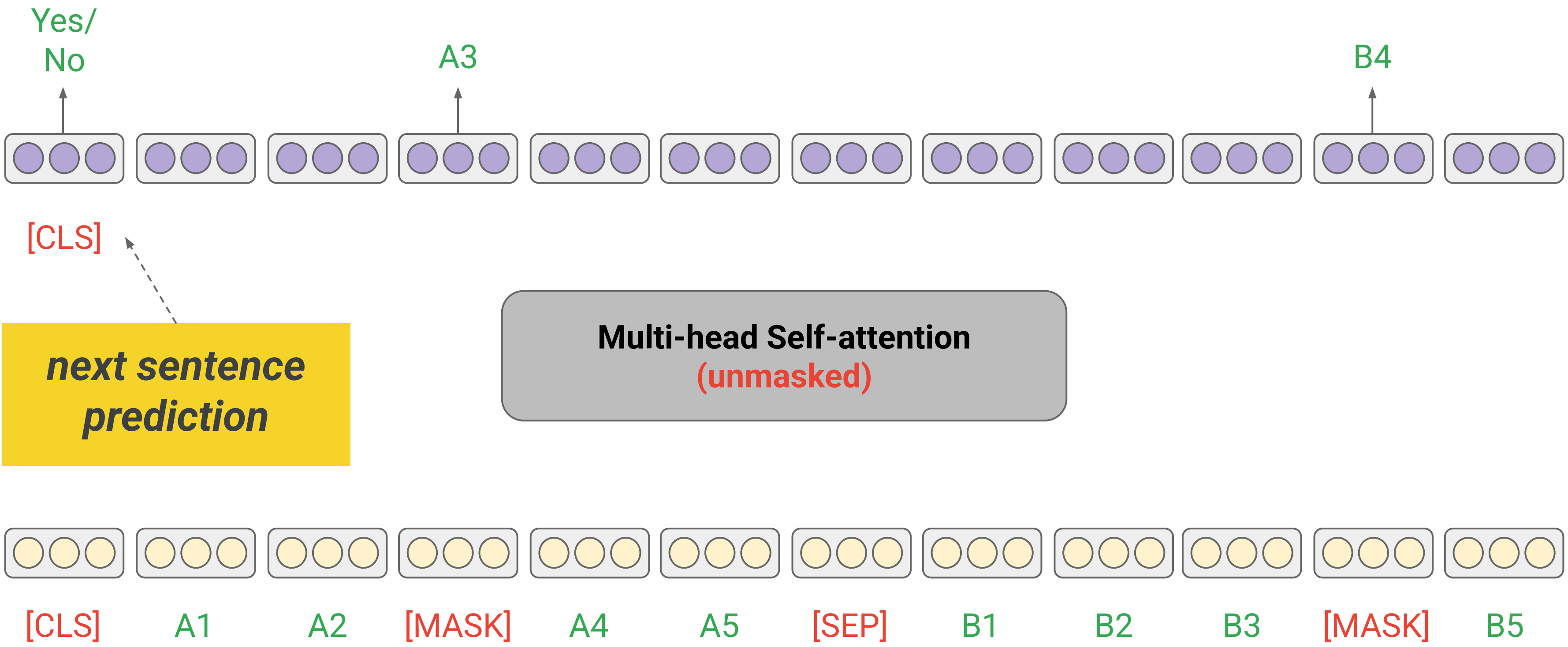
15% - 30% of all tokens in each sequence are masked at random

What if we mask more tokens?

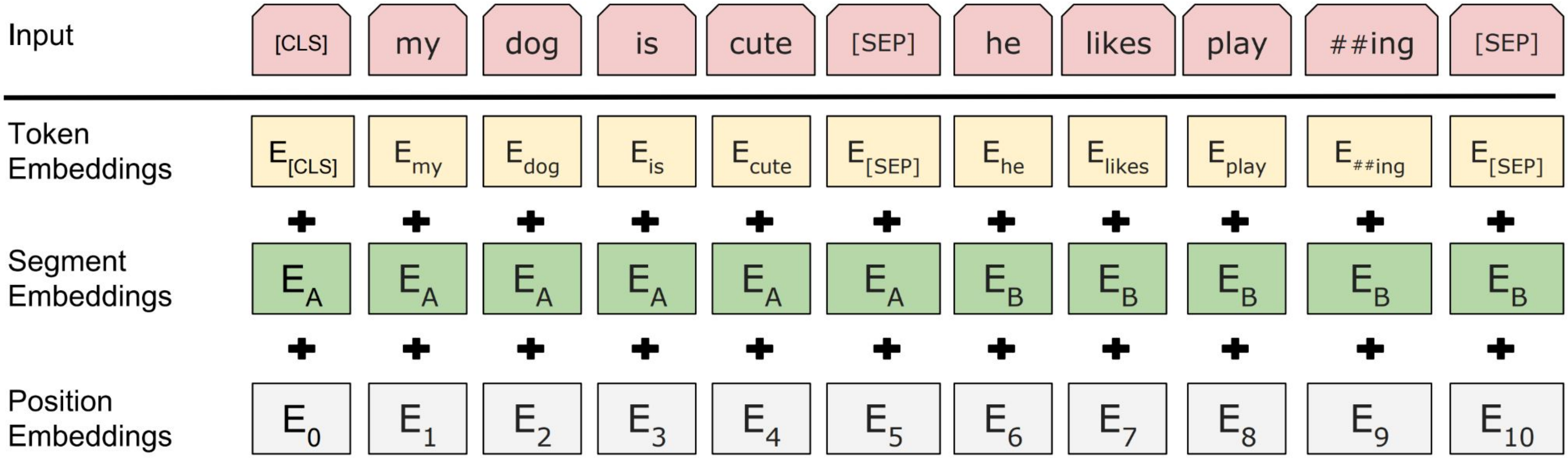


15% - 30% of all tokens in each sequence are masked at random

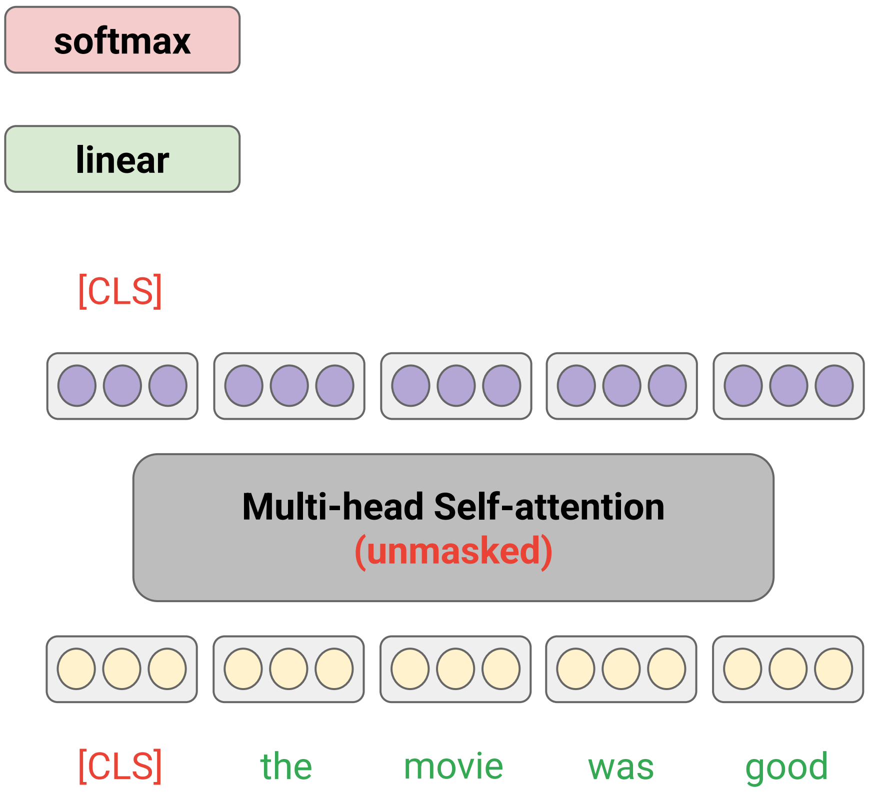
CLS & SEP tokens



BERT input representation

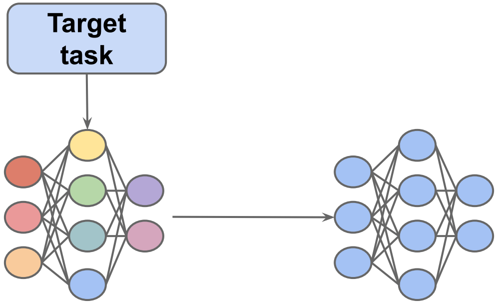


Fine-tuning


softmax
linear
[CLS]
Multi-head Self-attention
(unmasked)
[CLS]
the
movie
was
good

Intermediate-task transfer / fine-tuning

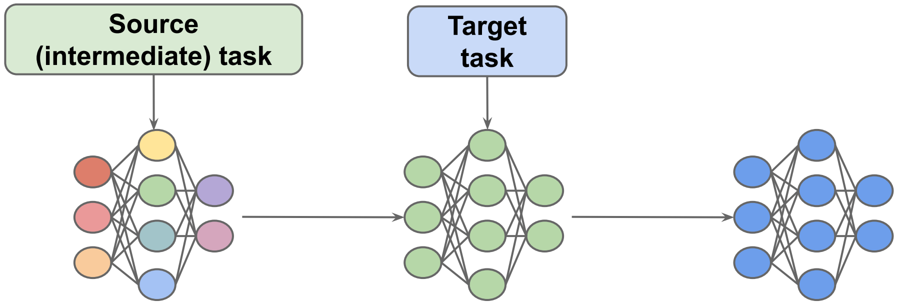
Standard Fine-tuning



BERTBASE → RTE: 63.5 ± 2.3

BERTLARGE → RTE: 68.6 ± 7.2

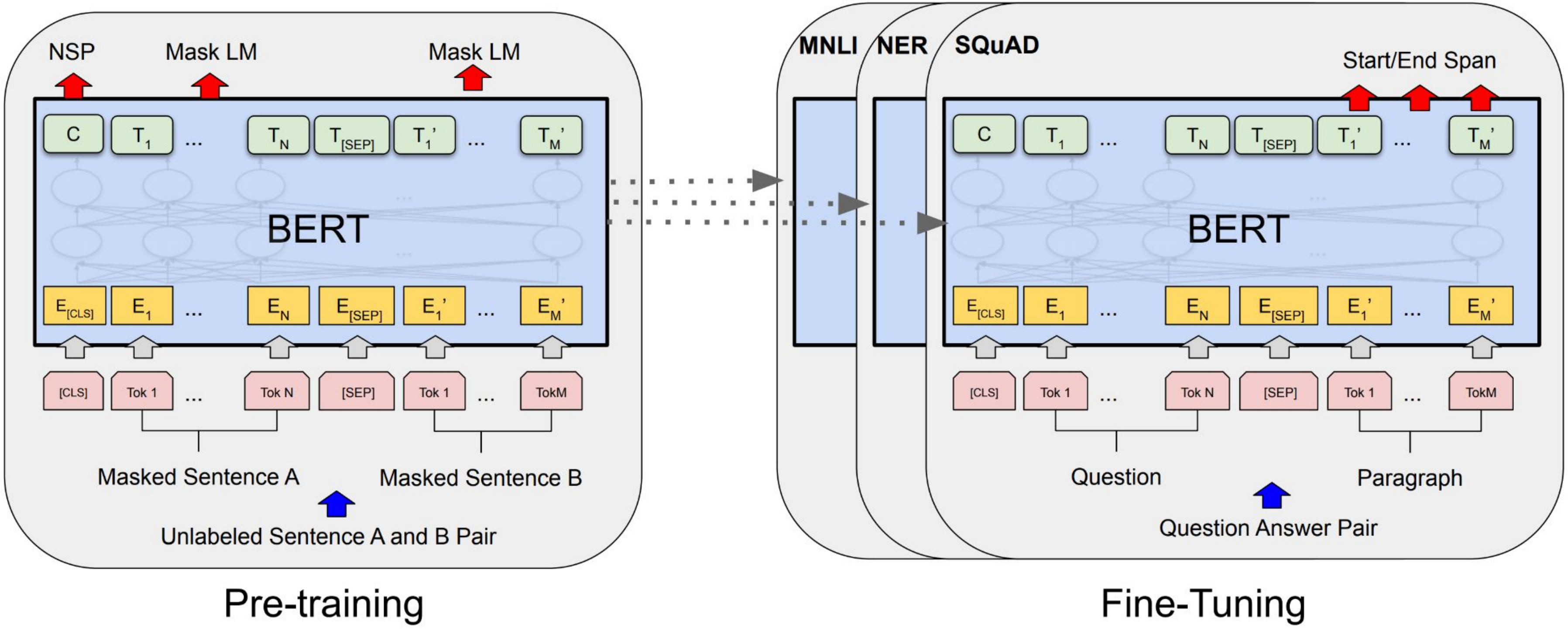
Intermediate-task transfer



BERTBASE → MNLI → RTE: 78.1 ± 1.9

BERTLARGE → MNLI → RTE: 82.3 ± 1.4

BERT Pretraining & Fine-tuning



Thank you!