SAISON 22/23

Formation

Introduction au Deep Learning

FIDLE

Ressources

https://fidle.cnrs.fr

Course materials (pdf)

Practical work environment*

Corrected notebooks

Videos (YouTube)

(*) Procedure via Docket or pip
Remember to get the latest version !

List and networks

You can also subscribe to :

http://fidle.cnrs.fr/listeinfo

Fidle information list

https://listes.services.cnrs.fr/wws/info/devlog

List of ESR* « calcul » group

https://listes.math.cnrs.fr/wws/info/calcul

List of ESR* « calcul » group

(*) ESR is Enseignement Supérieur et Recherche, french universities and public academic research organizations

Program

20 Séquences
du 17 novembre
au 14 mai 2023

1 ✓
History, Fundamental Concepts
Basic Regression DNN
Basic Classification DNN

2 3 ✓
Hight Dimensionnal Data
CNN

4 ✓
Demystify **mathematics** for neural networks.

5 ✓
Training strategies Evaluation
Sparse data (text) Embedding

6 ✓
Sequences data
RNN

7 ✓
PyTorch
A small detour with **PyTorch**.

8
«Attention is All You Need»
Transformers

9 New !
Graph Neural Network
GNN

10
Autoencoder networks
AE

11
Variational Antoencoder
VAE

12
Project session
«My project in 180 s»

13
Generative Adversarial Networks
GAN

14 New !
Diffusion Model
Text to image

15
AI, Law, Society and **Ethics**

16 New !
Model and training optimization
Resource efficiency

17
Jean-Zay
GPU acceleration

18 New !
Physics-Informed Neural Networks
PINNS

19
Deep Reinforcement Learning
RL

20
What will be tomorrow's AI
Review & perspectives !

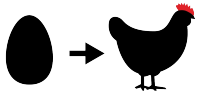
FIDLE

SAISON 22/23

Let the Transformers speak !

Roadmap

8



«Attention is All You Need»
Transformers

8.1 Introduction

- History of transformers
- Why transformers

8.2 Transformer architecture

- Vanilla transformer architecture
- Attention mechanism and multi-head attention
- Several transformer architectures

8.3 Pretraining and Fine tuning

- GPT and BERT pretraining
- Common fine tuning and prompting

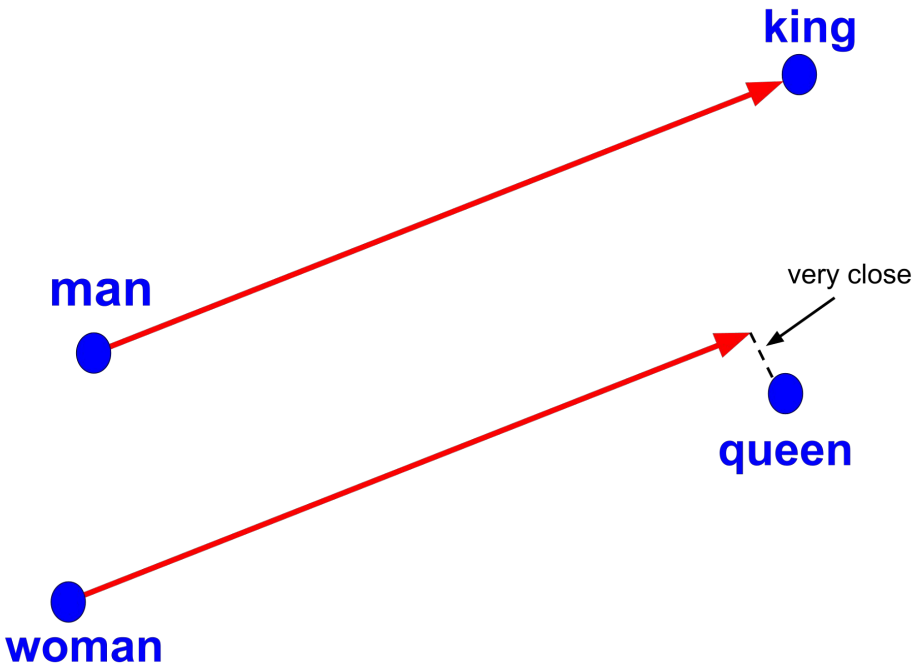
8.4 Transformers in other fields

- Visual Transformers

8.5 Example : IMDB

- IMDB Reviews

Reminder : word embedding

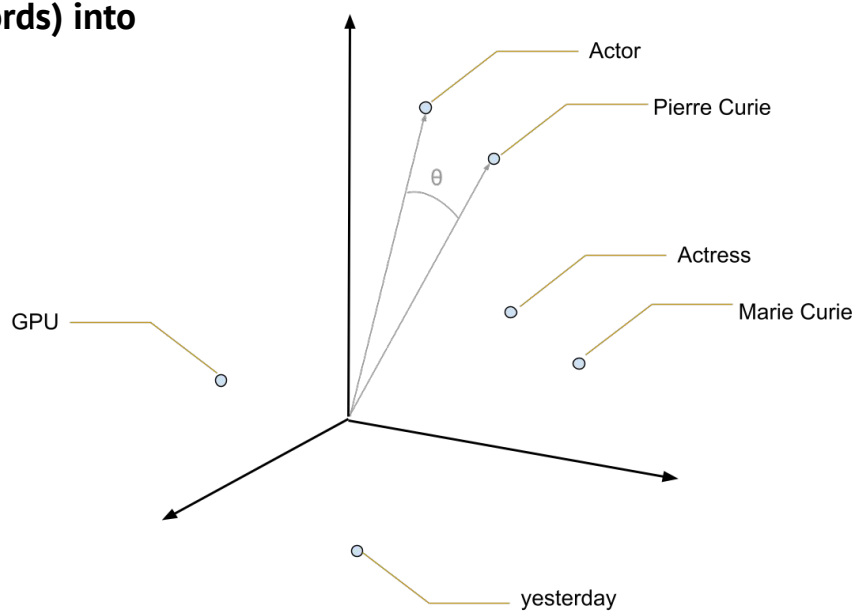


$$\text{woman} + \overbrace{(\text{king} - \text{man})}^{\approx \text{royalty}} \approx \text{queen}$$

Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron

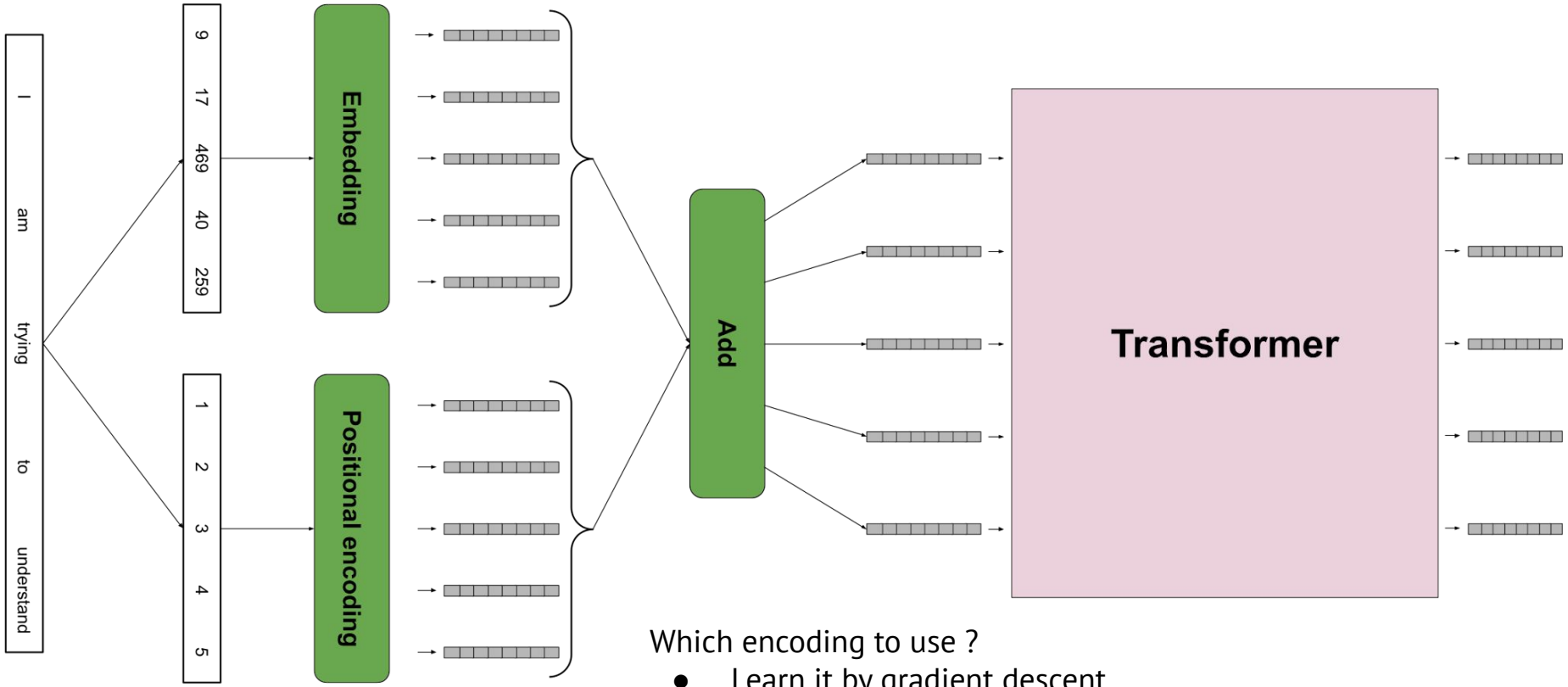
Reminder : word embedding

Turn sparse data (for instance words) into vectors



Actor - Pierre Curie + Marie Curie ≈ Actress

Positional encoding - 1



Which encoding to use ?

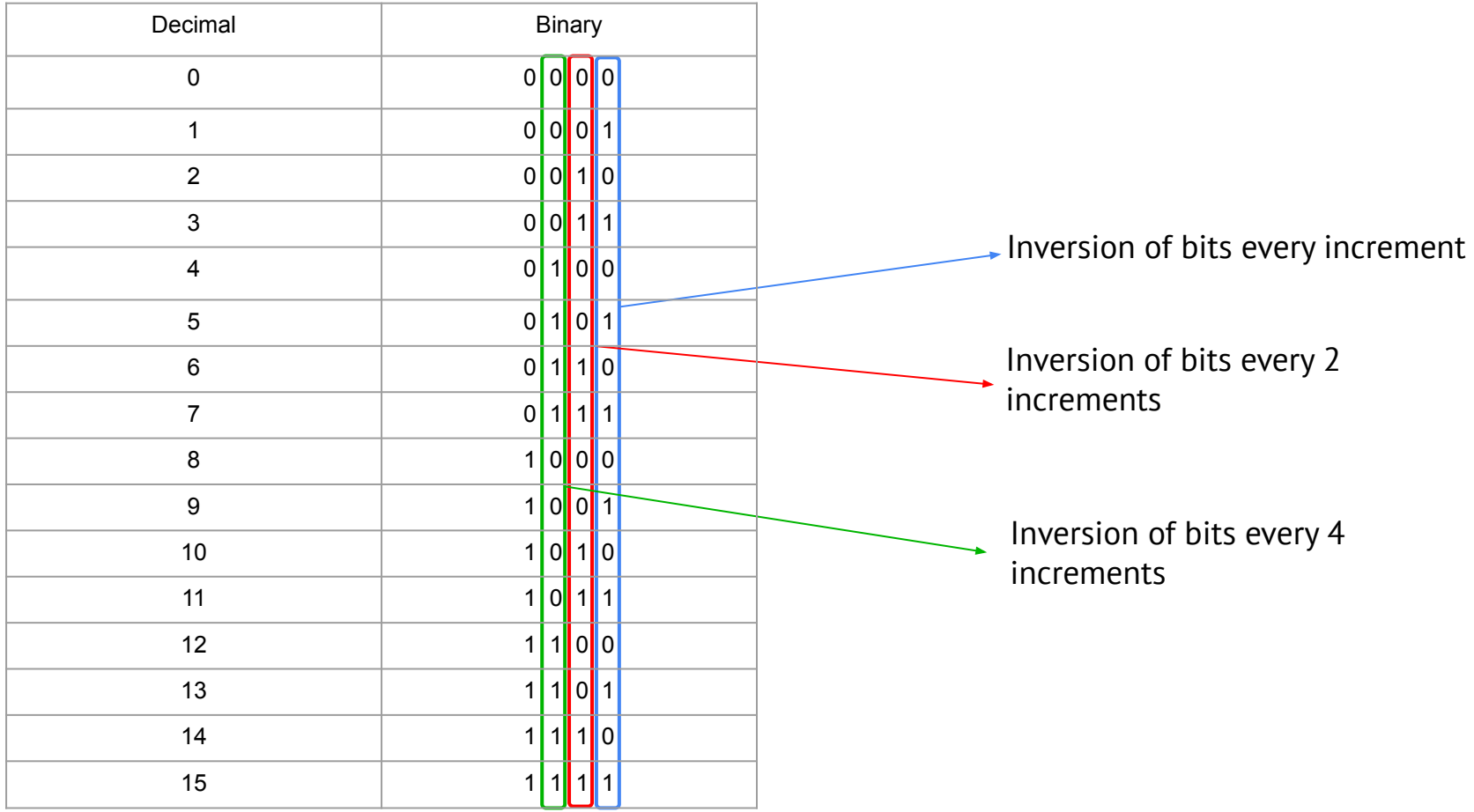
- Learn it by gradient descent
- Cosinus encoding

Positional encoding - 2

Decimal	Binary
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

Positional encoding - 3

Decimal	Binary
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1



Positional encoding - 4

Decimal	Binary
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

pos the position of the word in the sentence

i the dimension (index of a coordinate in the embedding)

d_{model} the total dimension of the embedding

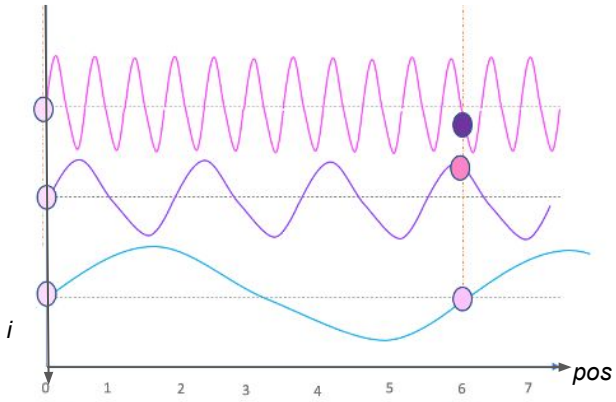
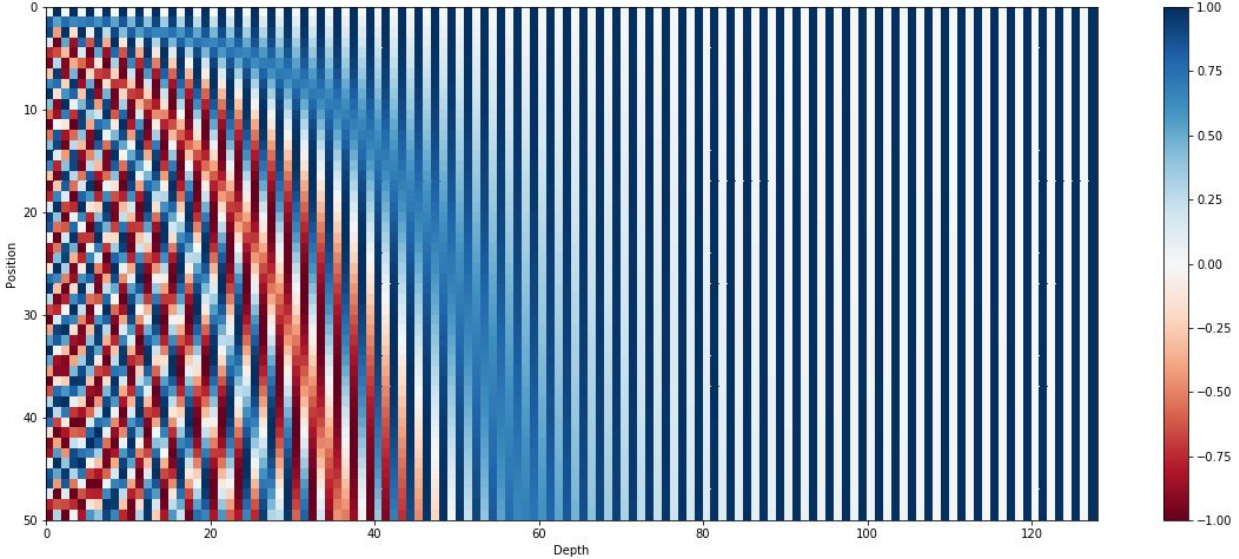


Figure from https://www.youtube.com/watch?v=dichIcUZfOw

Positional encoding - 5



$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

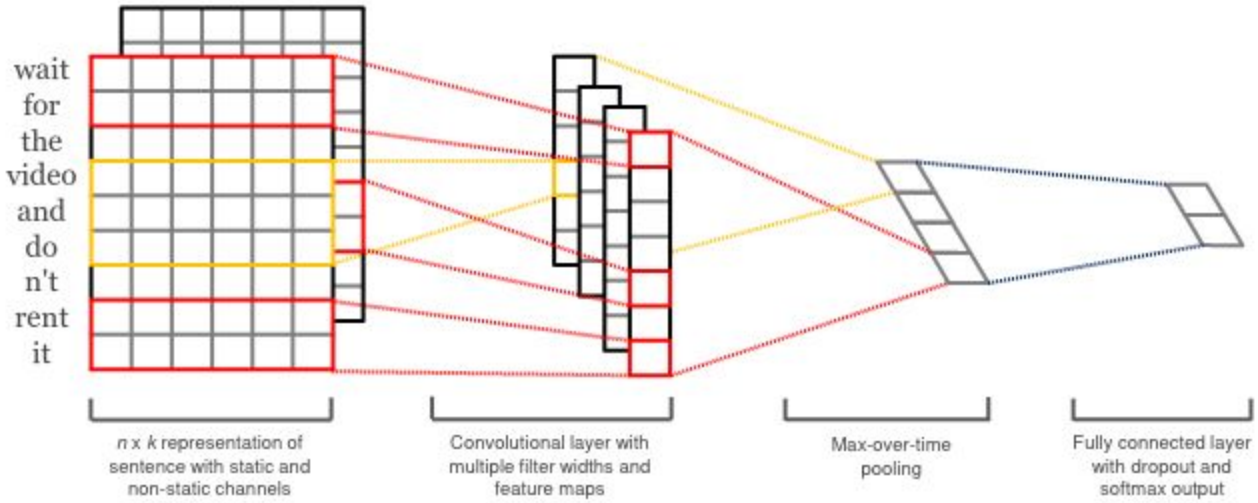
pos the position of the word in the sentence

i the dimension (index of a coordinate in the embedding)

d_{model} the total dimension of the embedding

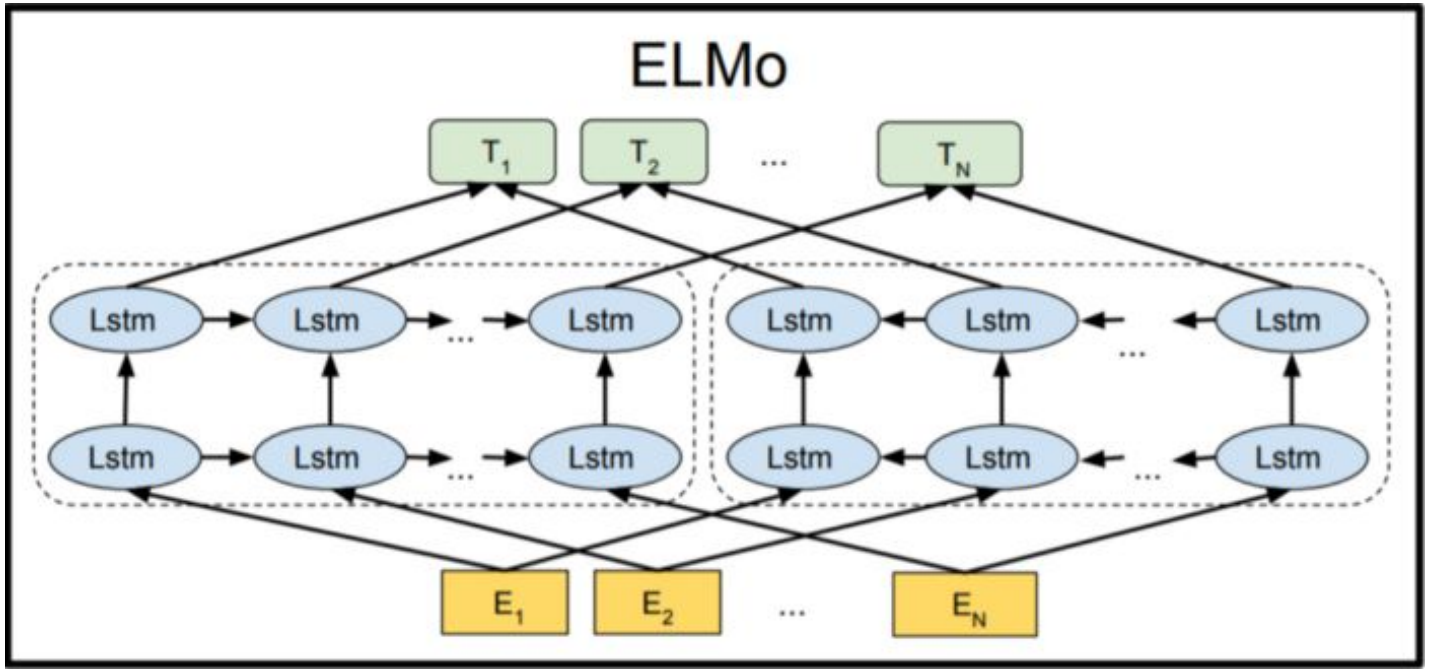
Figure from https://kazemnejad.com/blog/transformer_architecture_positional_encoding/

Architectures for NLP - CNN



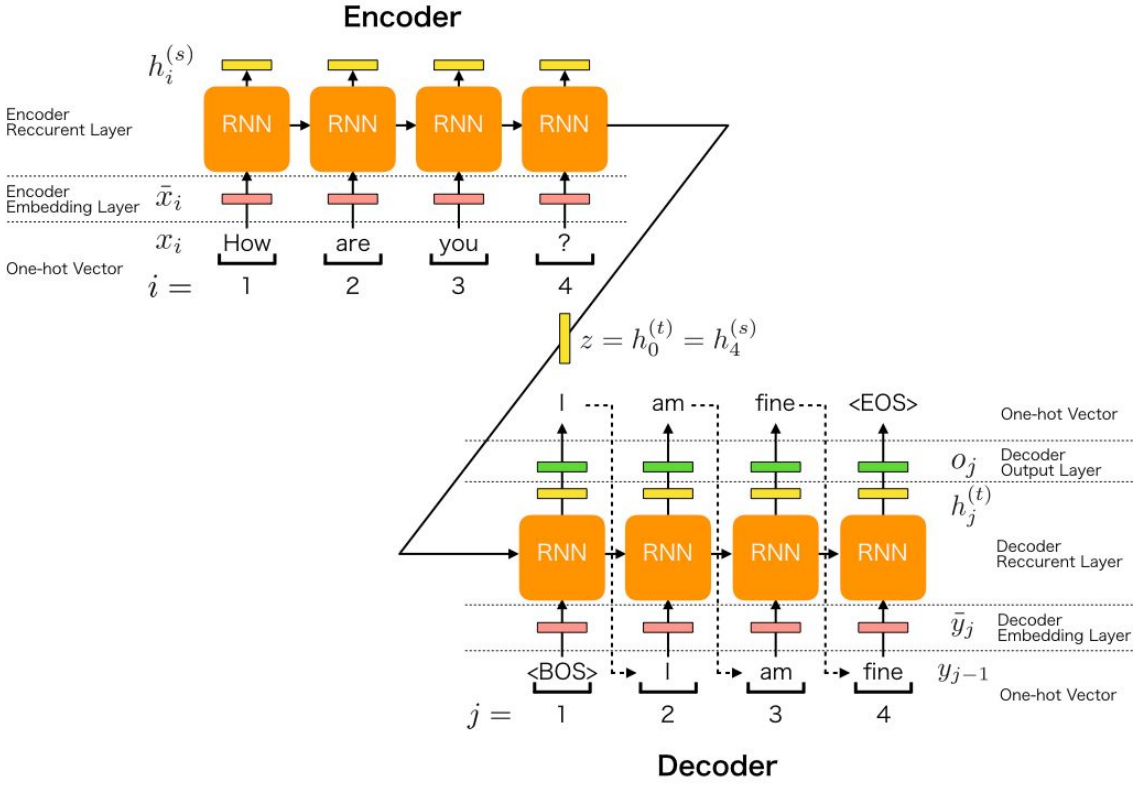
Convolutional Neural Networks for Sentence Classification, Yoon Kim, 2014, https://arxiv.org/pdf/1408.5882.pdf

Architectures for NLP - RNN



https://medium.com/saarthi-ai/elmo-for-contextual-word-embedding-for-text-classification-24c9693b0045

Architectures for NLP - RNN

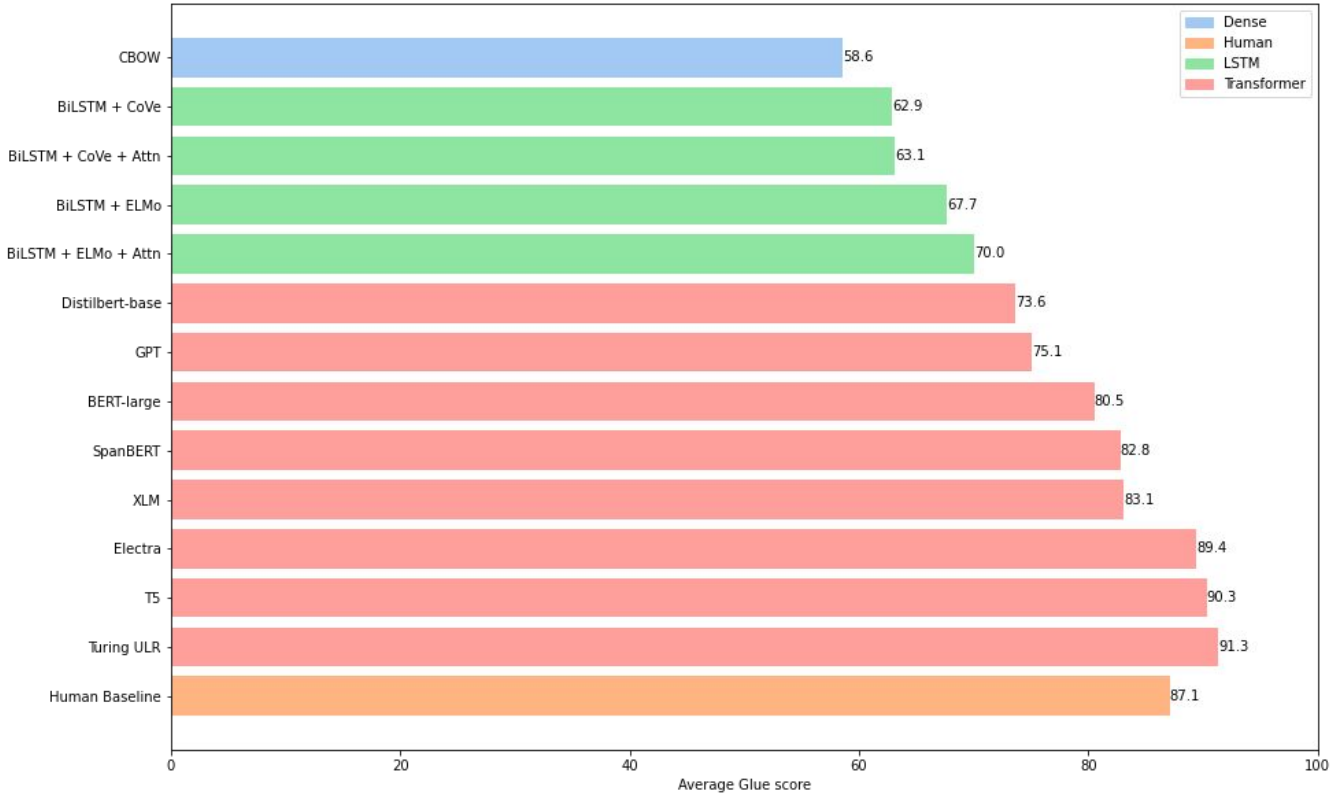


https://docs.chainer.org/en/v7.8.0/examples/seq2seq.html

What we want to achieve

- ➔ Process sequences (ideally the entire sentence)
- ➔ Easy to distribute on multiple GPUs
- ➔ Faster training than with RNN
- ➔ Initially for NLP tasks
- ➔ Allows to train huge models on gigantic datasets
- ➔ Allows for a pretraining session to pool trainings (at least partially) for multiple tasks

The King is dead, long live the King!

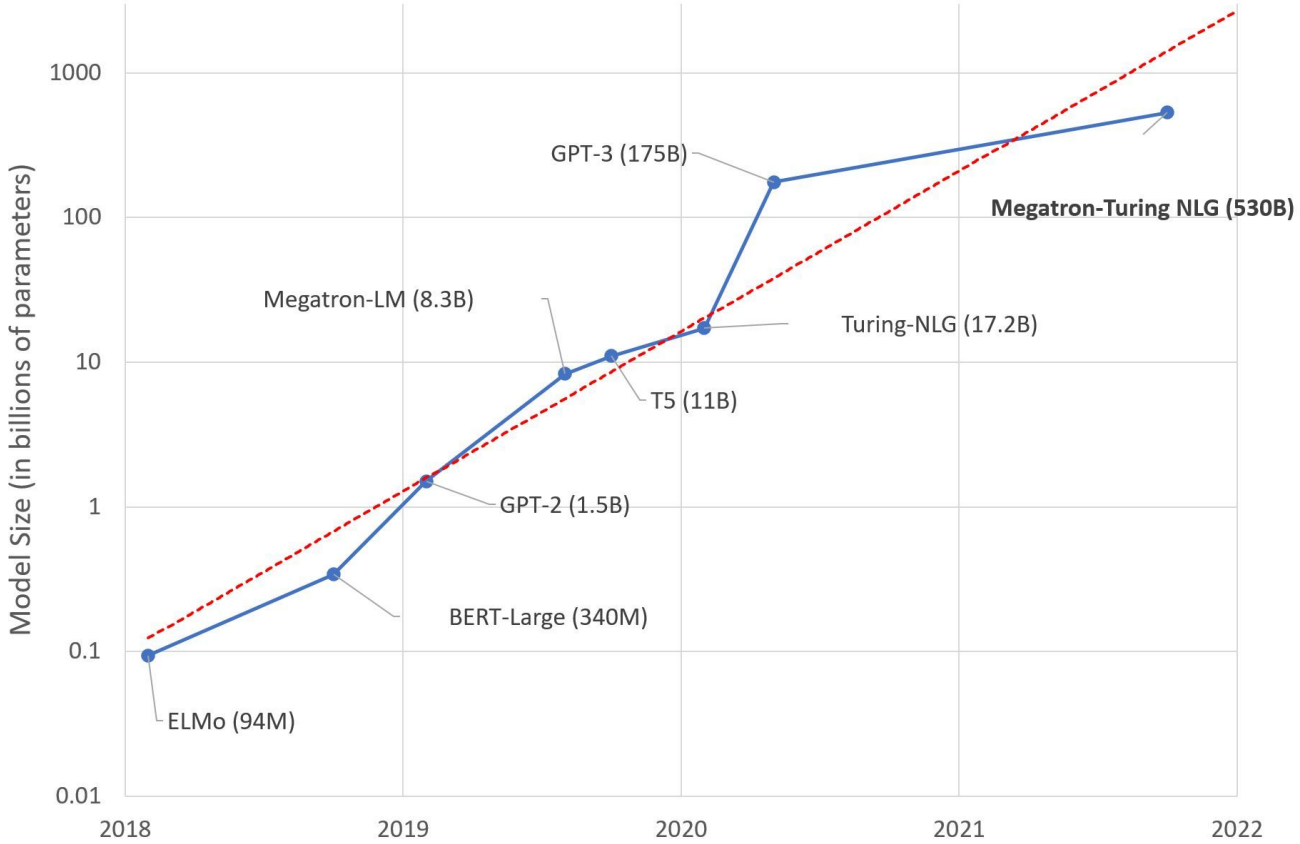


Improving Language Understanding by Generative Pre-Training, Radford et al, 2018 : https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf

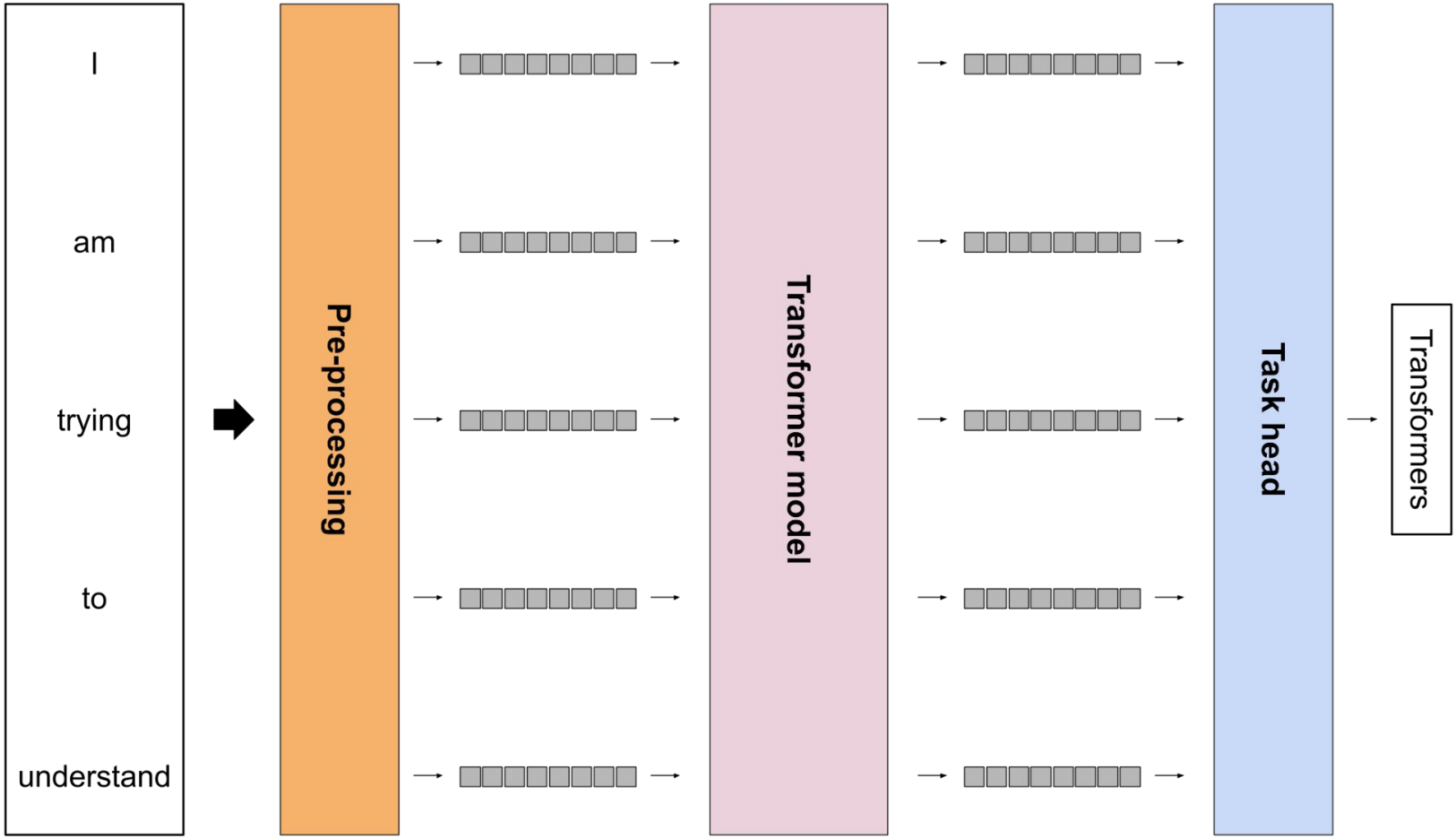
GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding, Wang A. et al, 2019 : https://arxiv.org/pdf/1804.07461.pdf

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Devlin J. et al, 2019 : https://arxiv.org/pdf/1810.04805.pdf

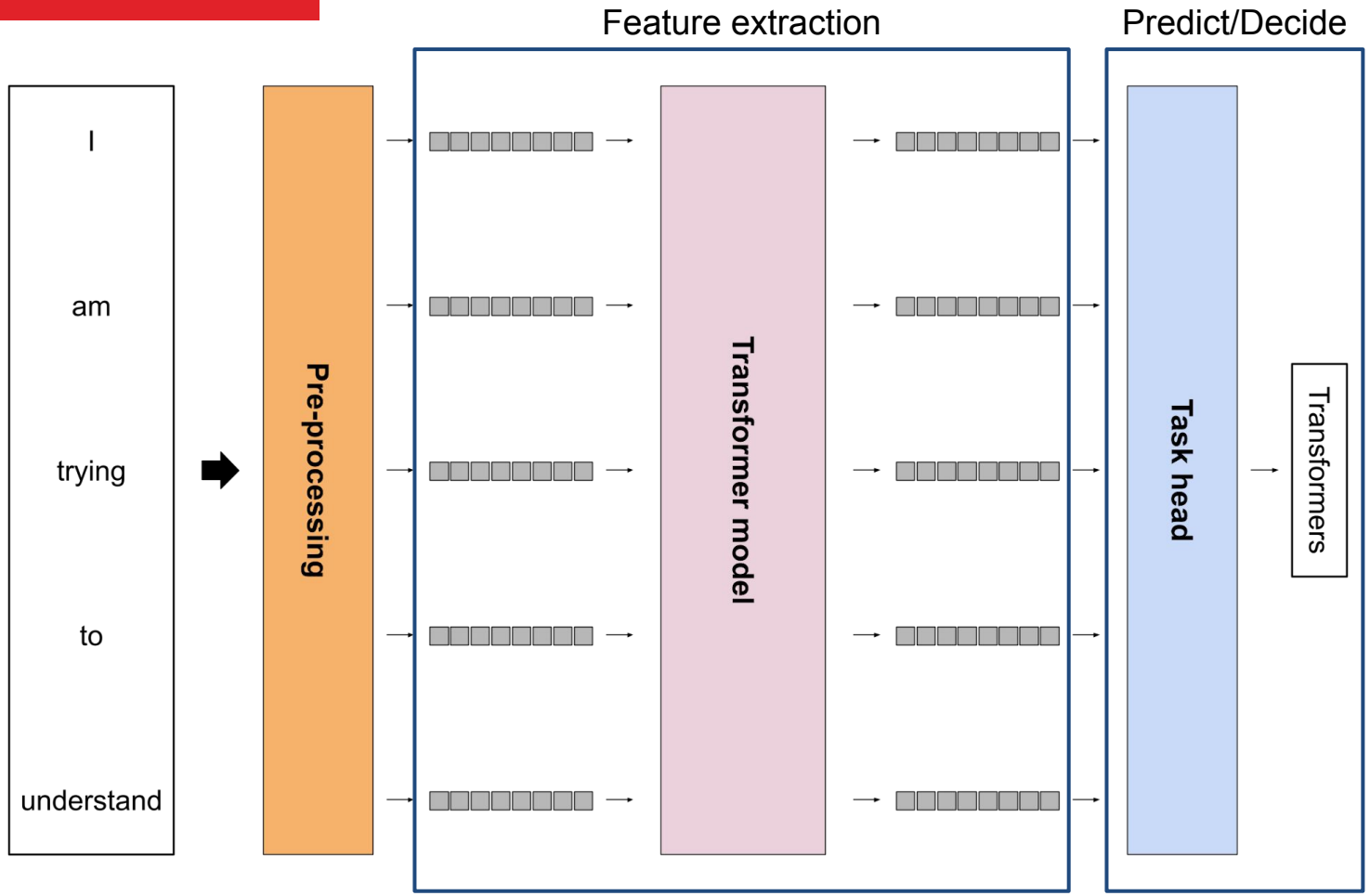
Size evolution



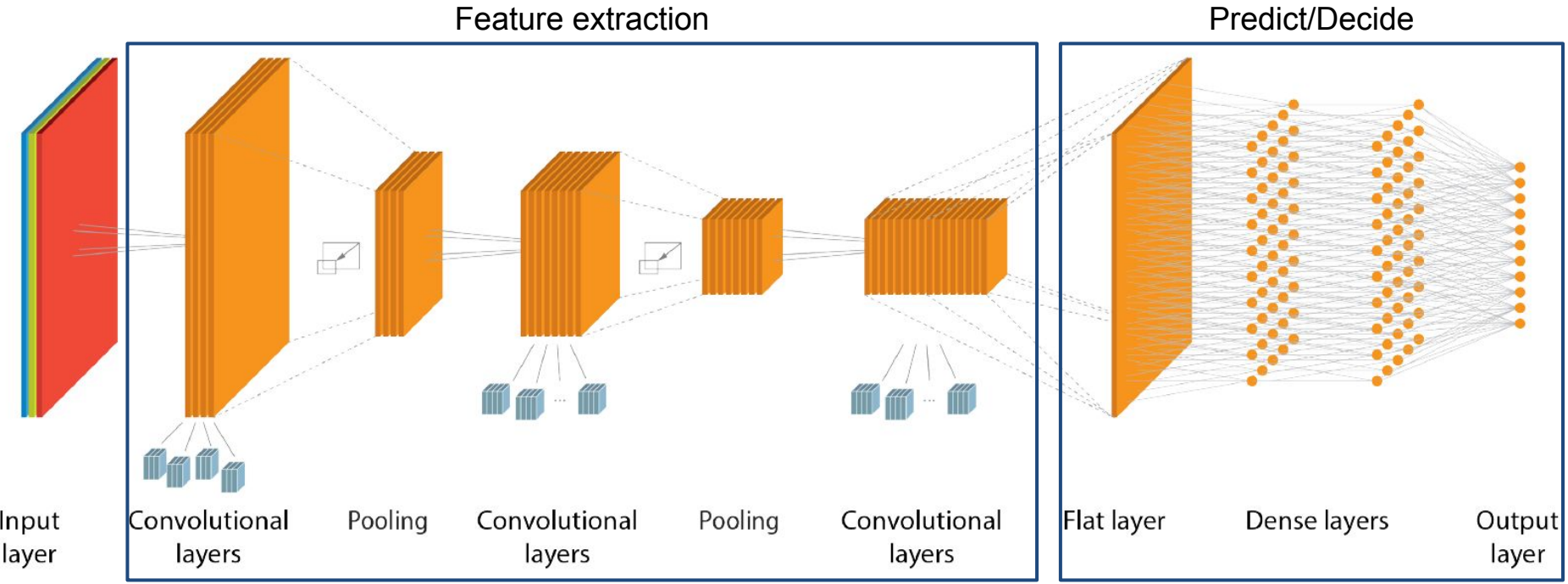
Example of NLP system



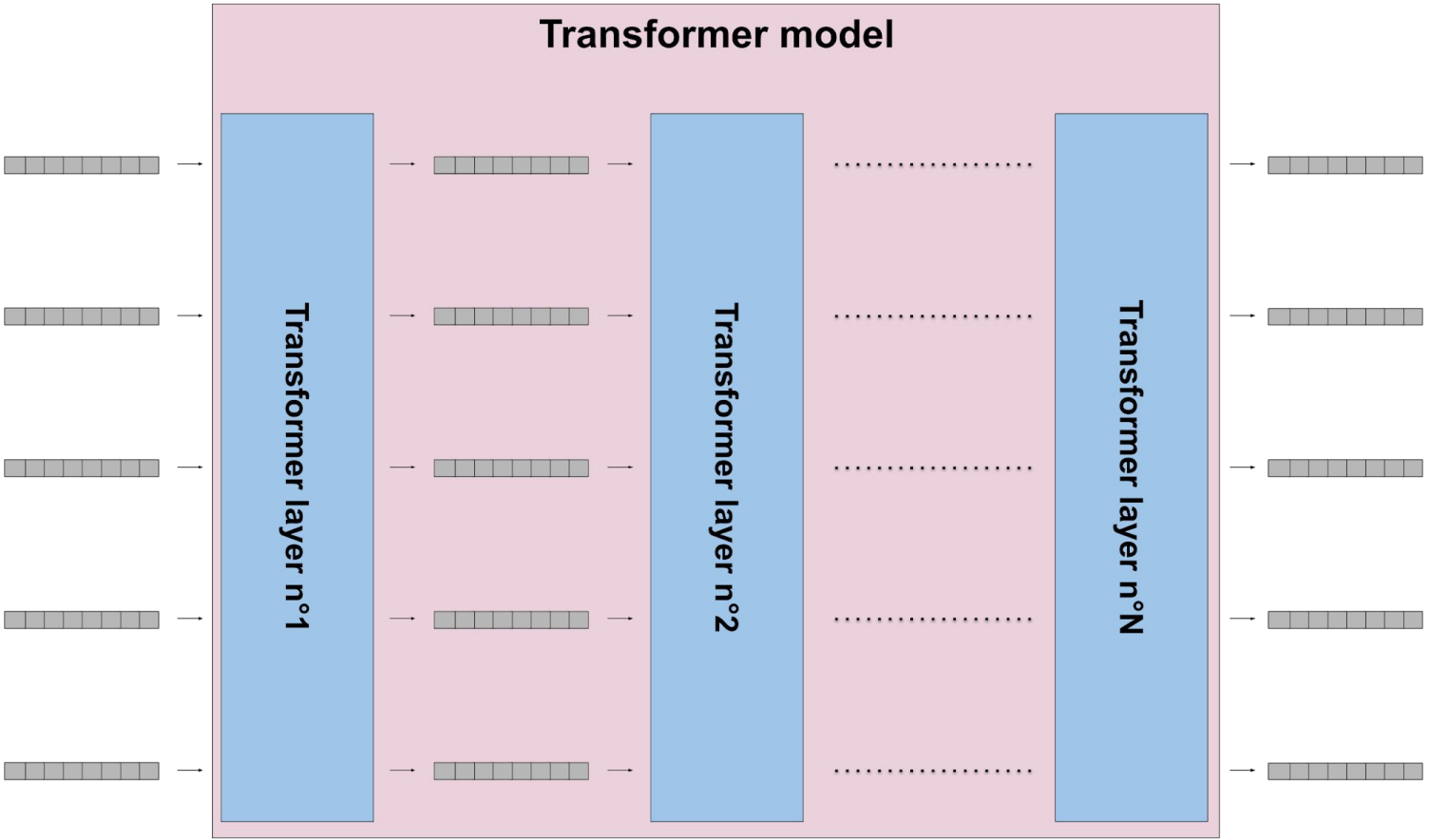
Features extraction



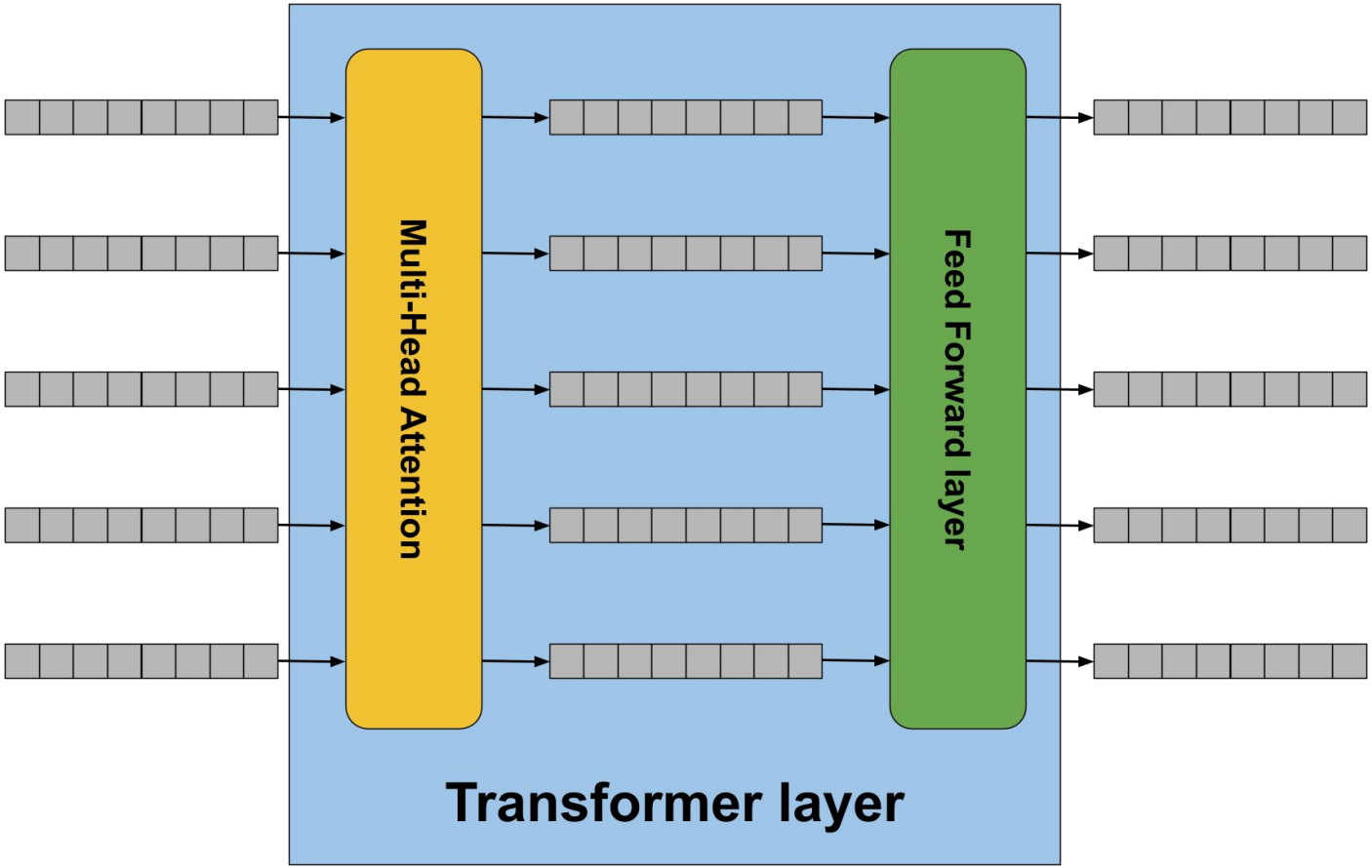
Features extraction reminder



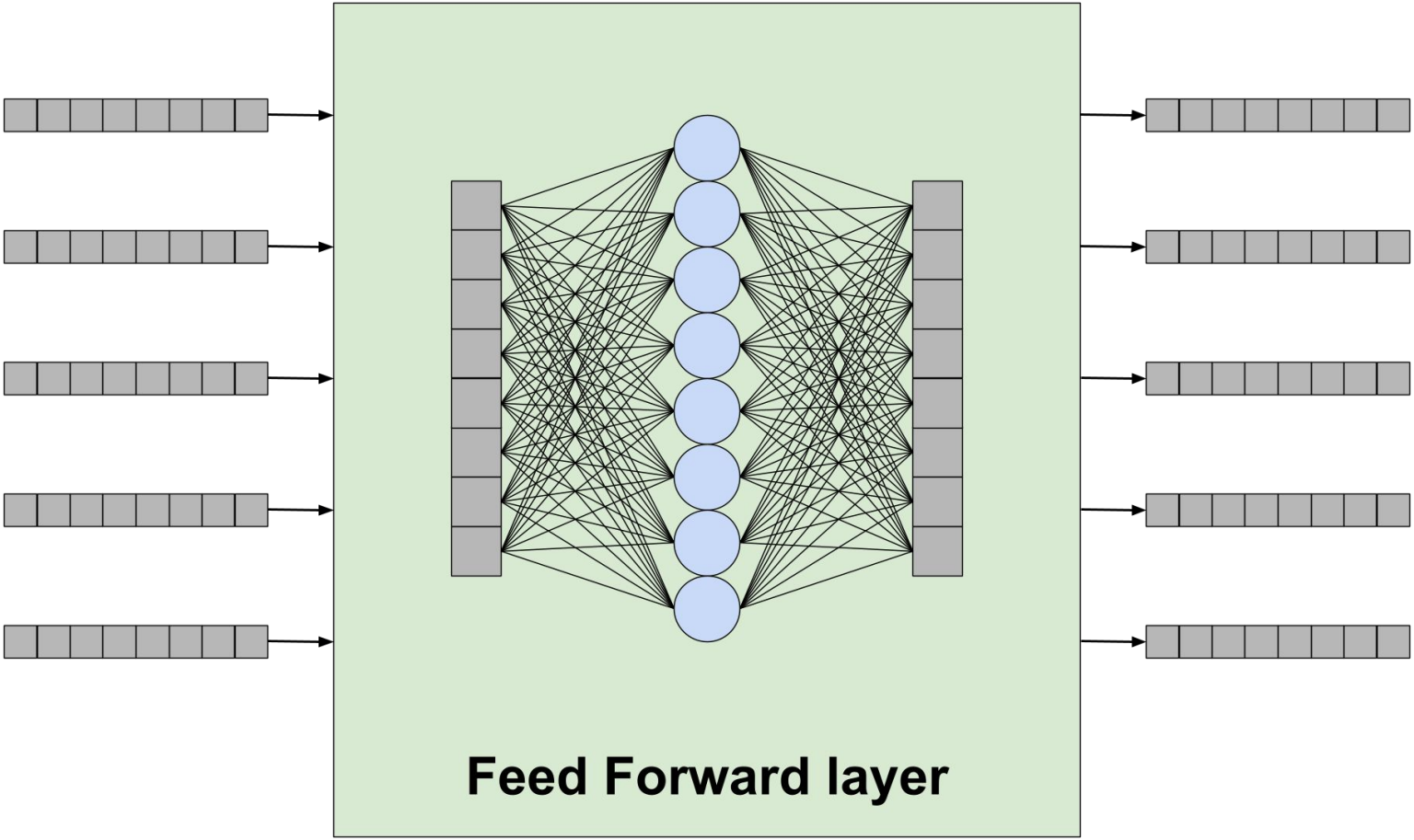
Vanilla Transformer architecture



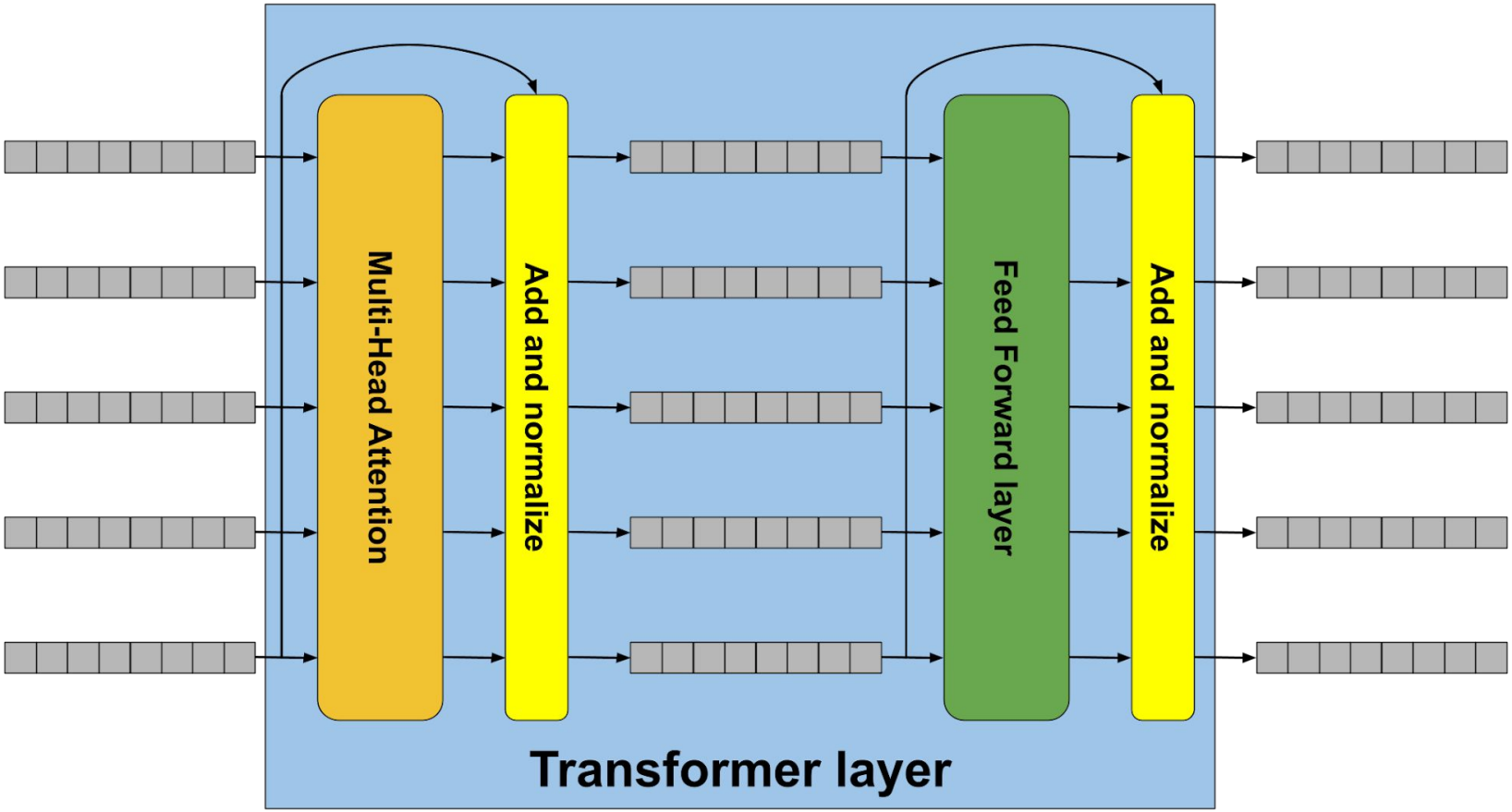
Transformer layer simplified



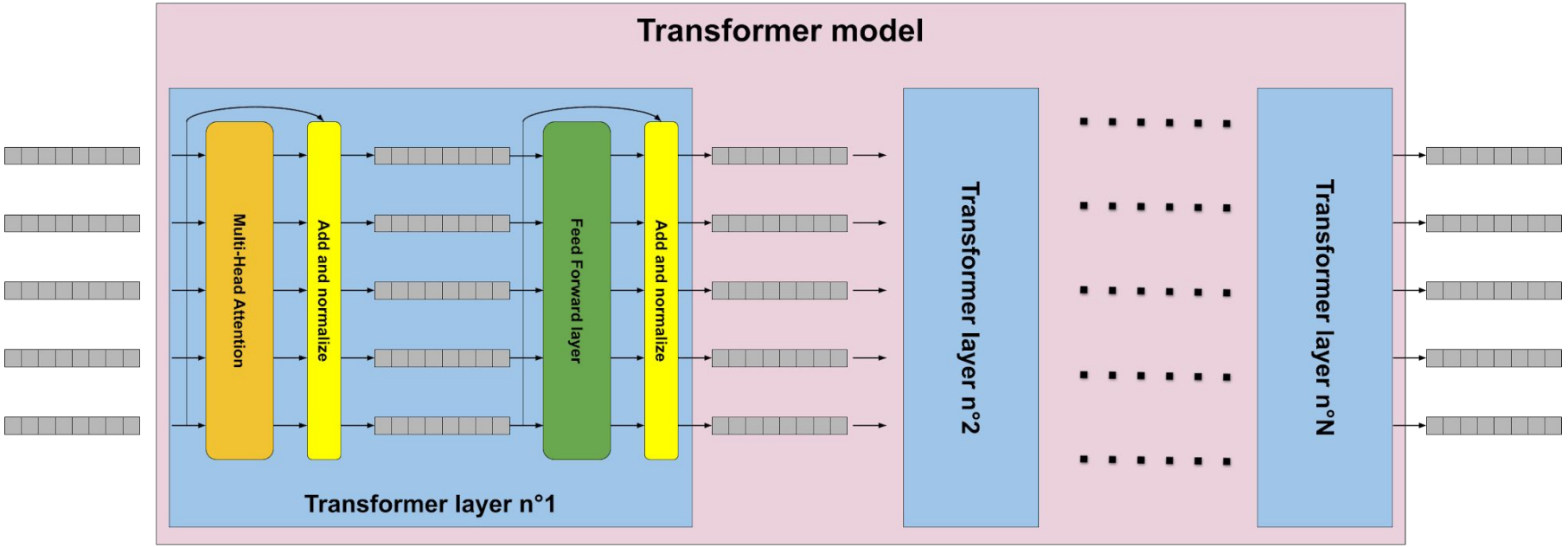
Feed Forward layer



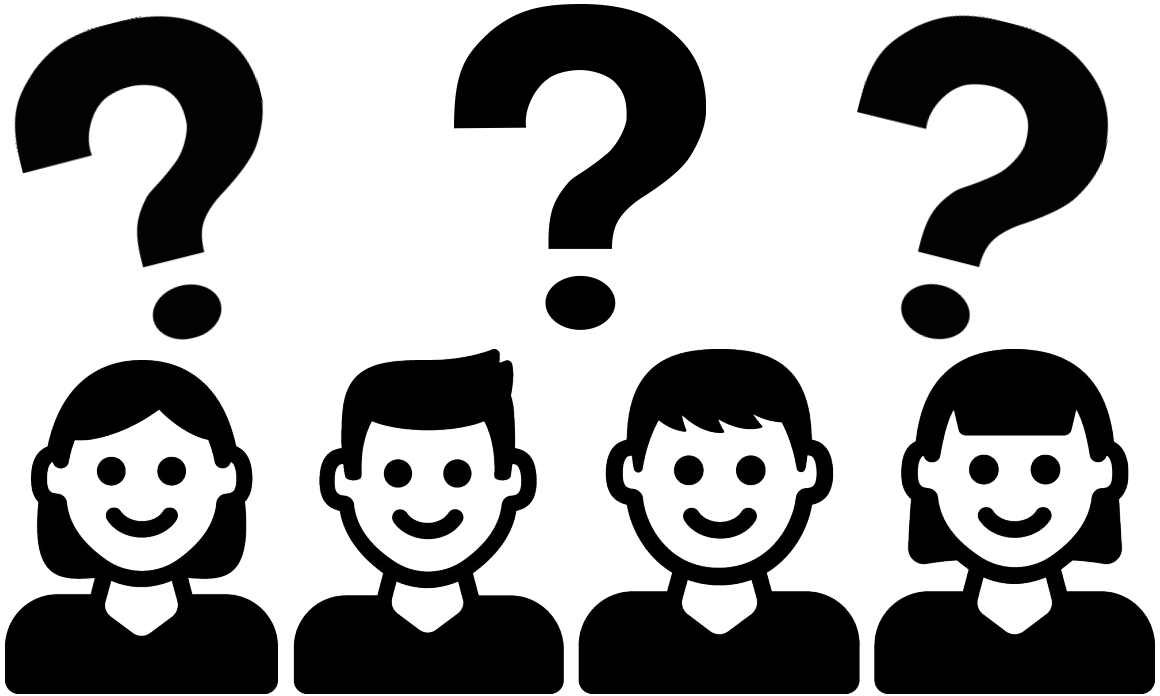
Transformer layer



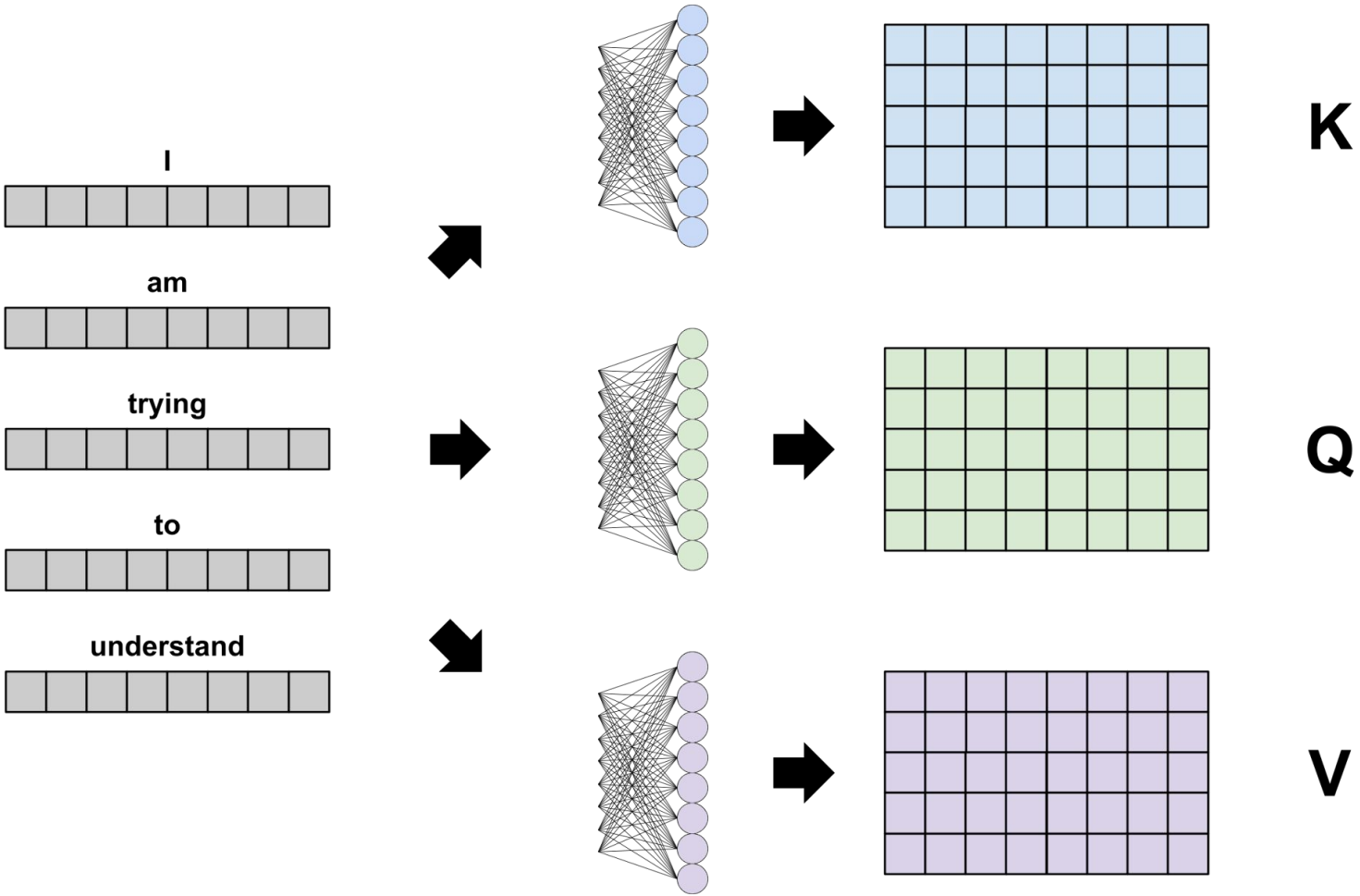
Vanilla Transformer architecture summary



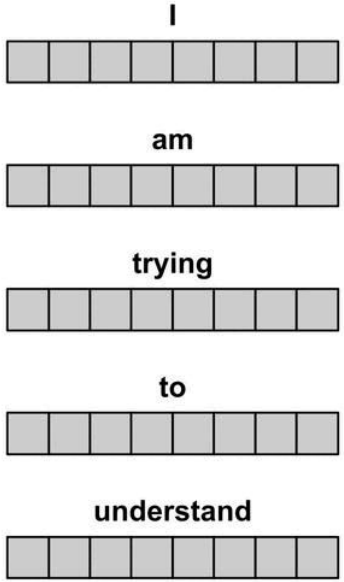
Question break #1

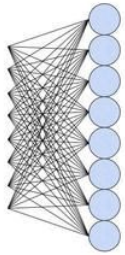


Attention explained - 1

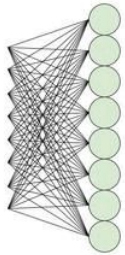


Attention explained - 1 (animated)

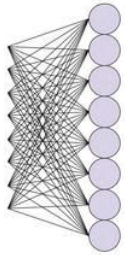




K

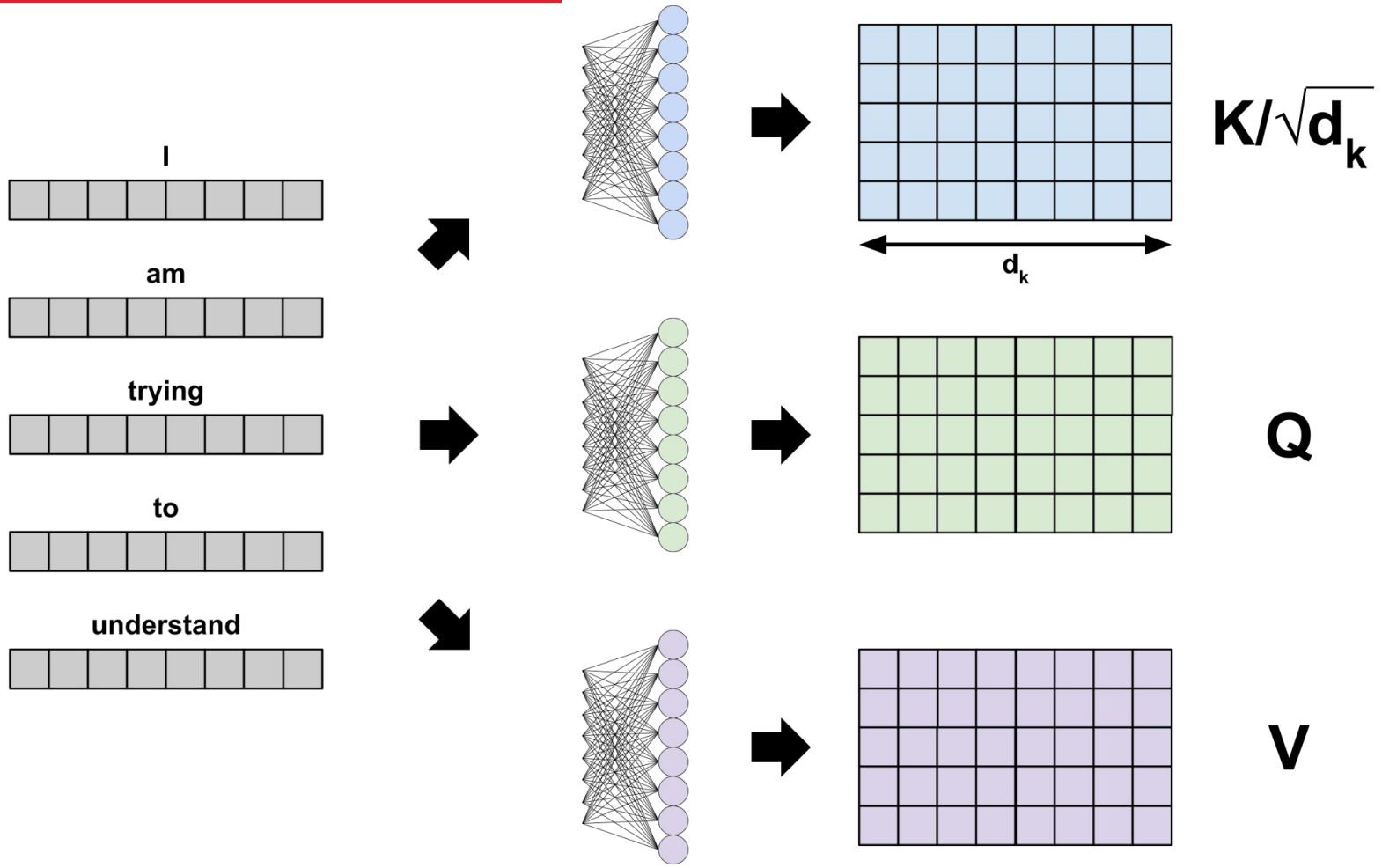


Q

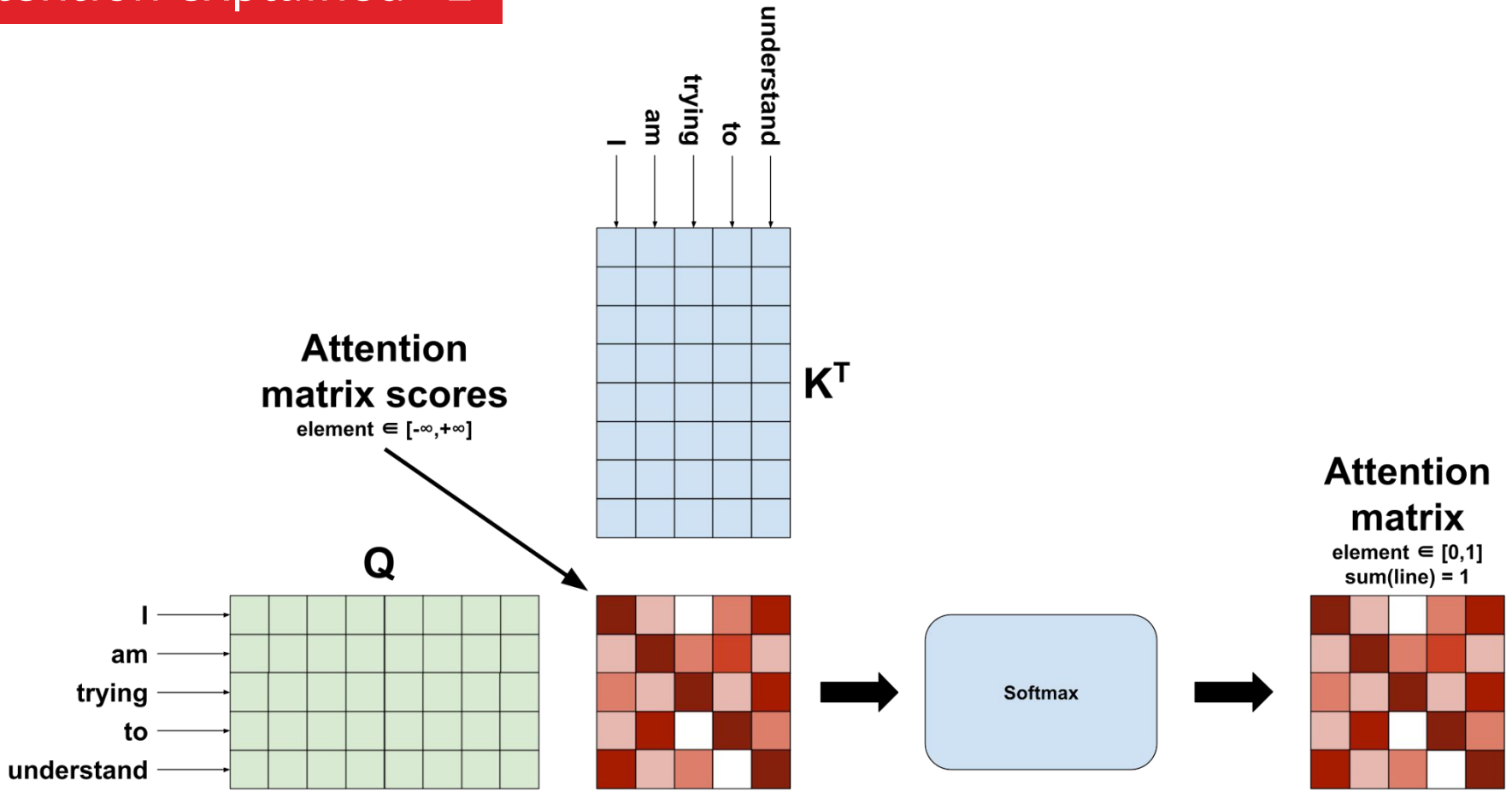


V

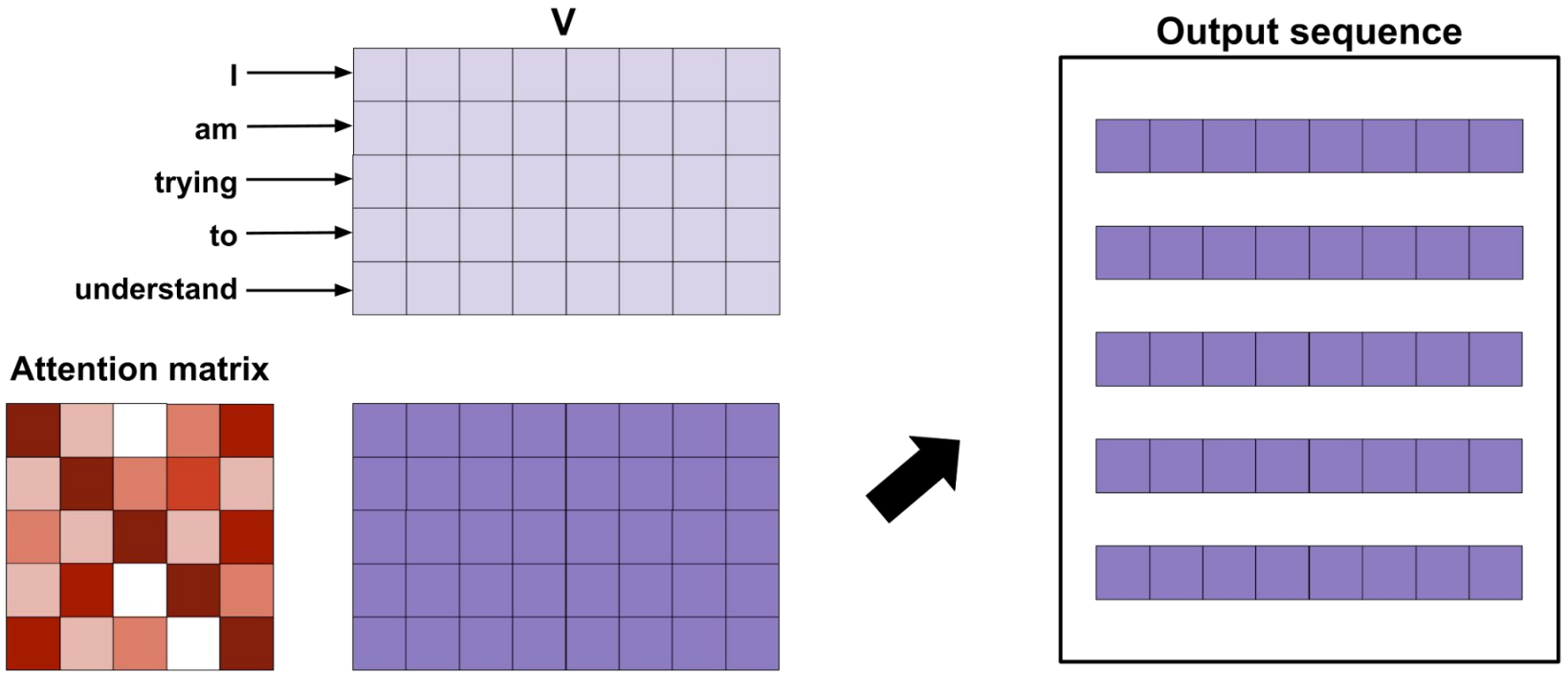
Attention explained - 1 (full)



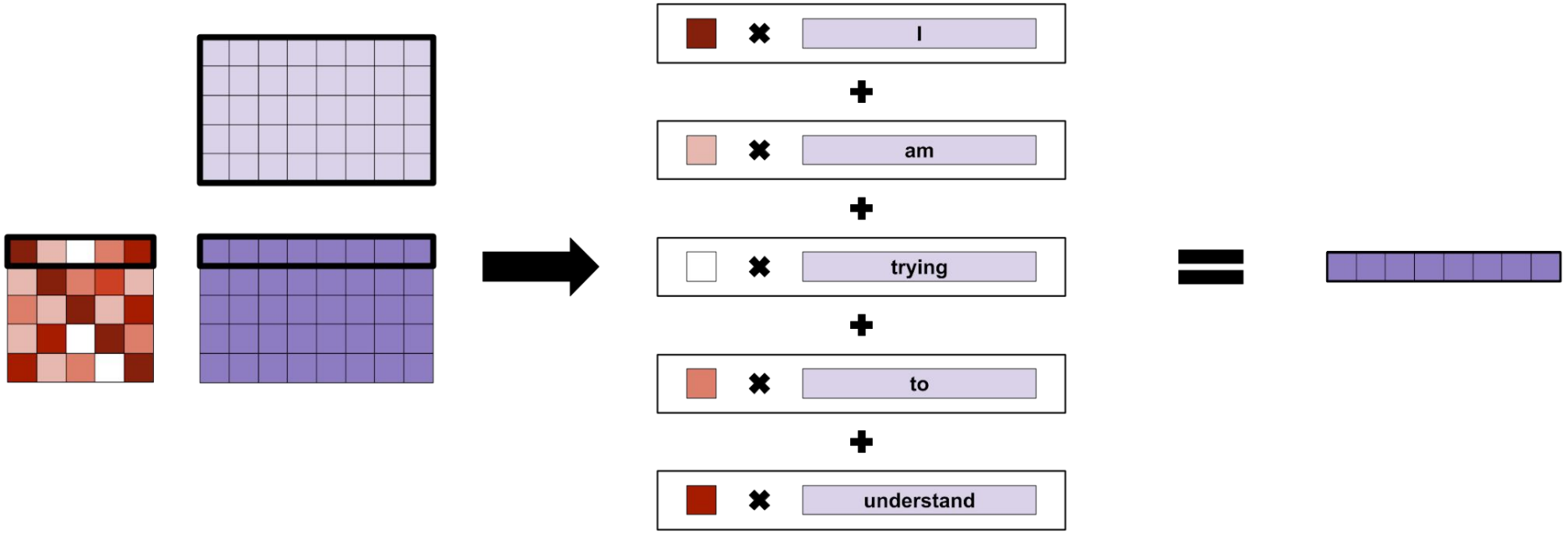
Attention explained - 2



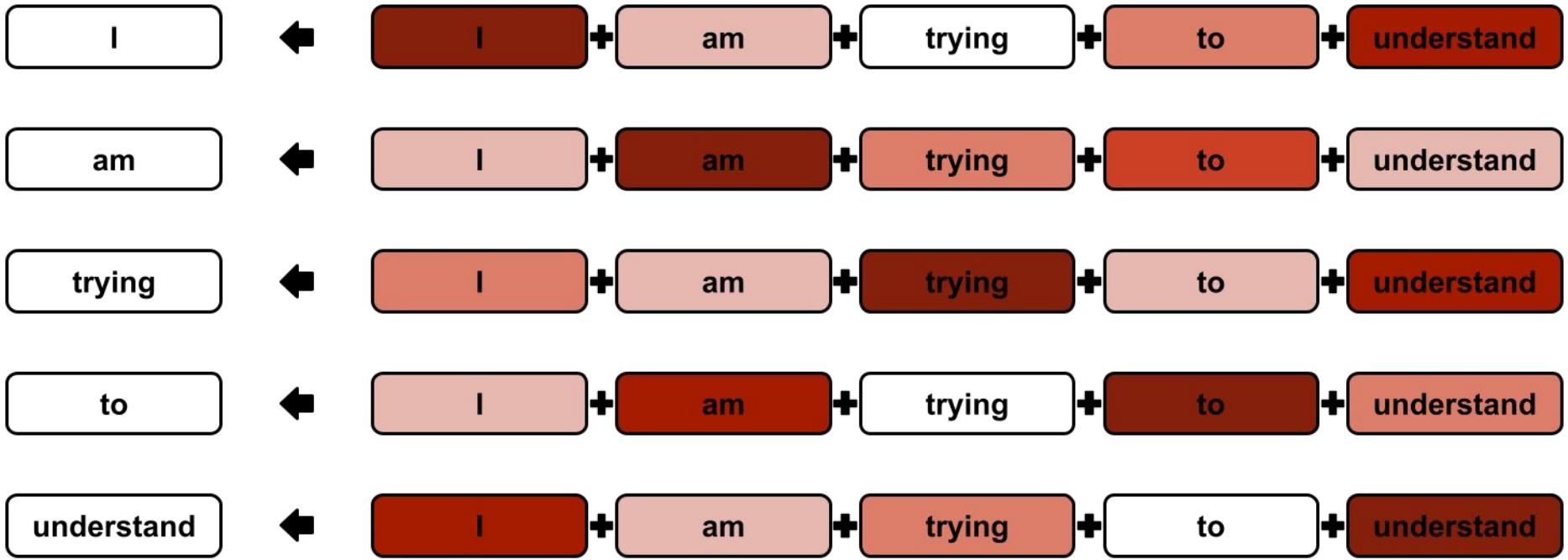
Attention explained - 3



Attention explained - 4



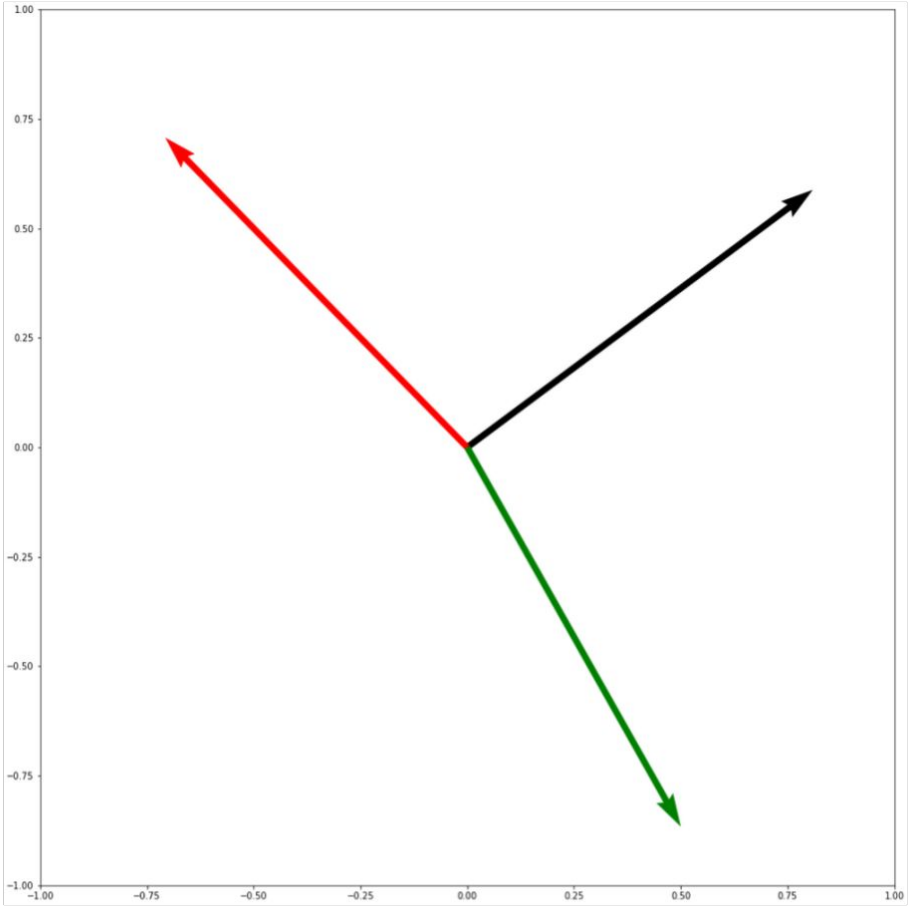
Intuition behind the attention mechanism - 0



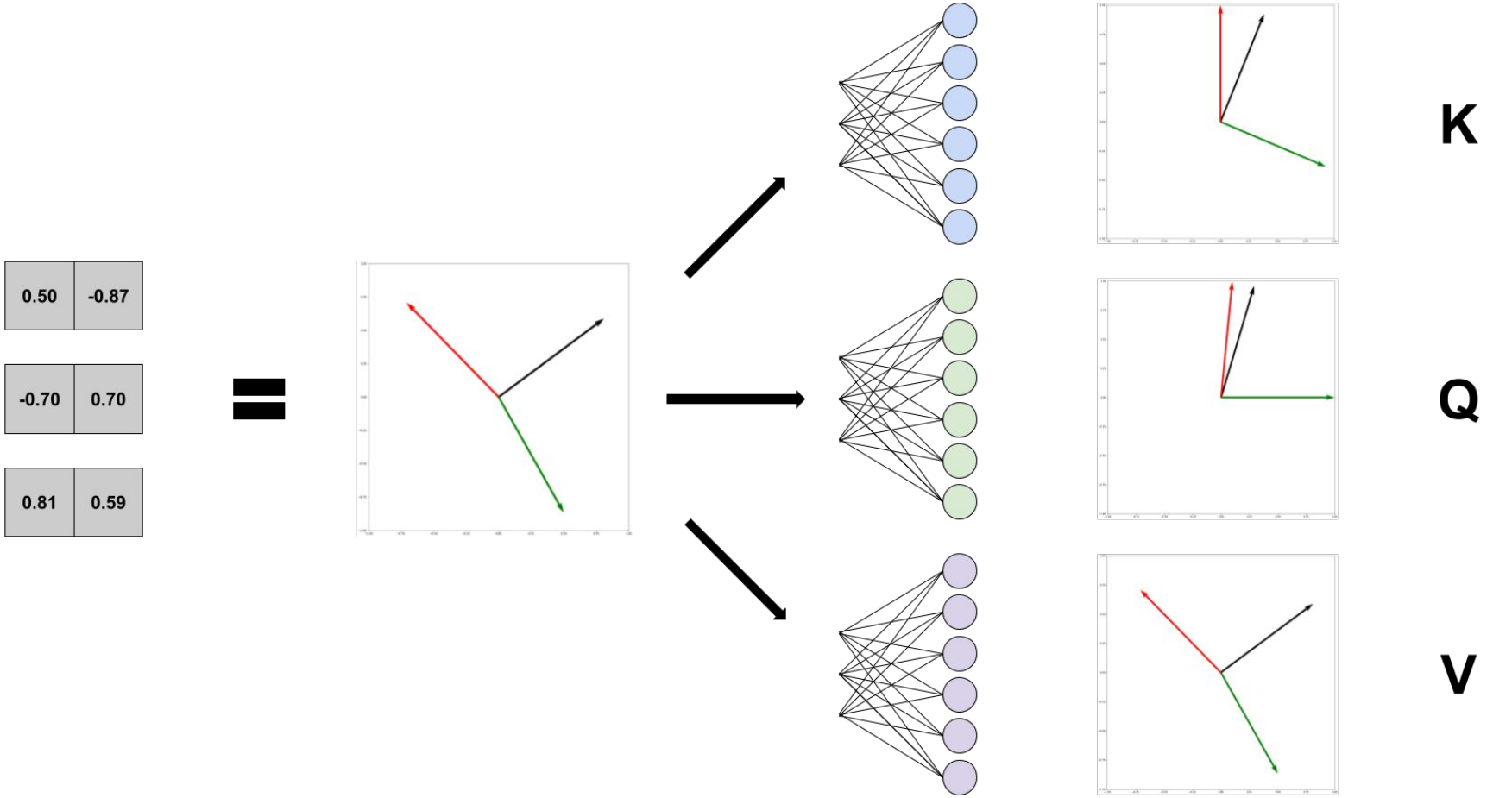
Intuition behind the attention mechanism - 1

The big dog

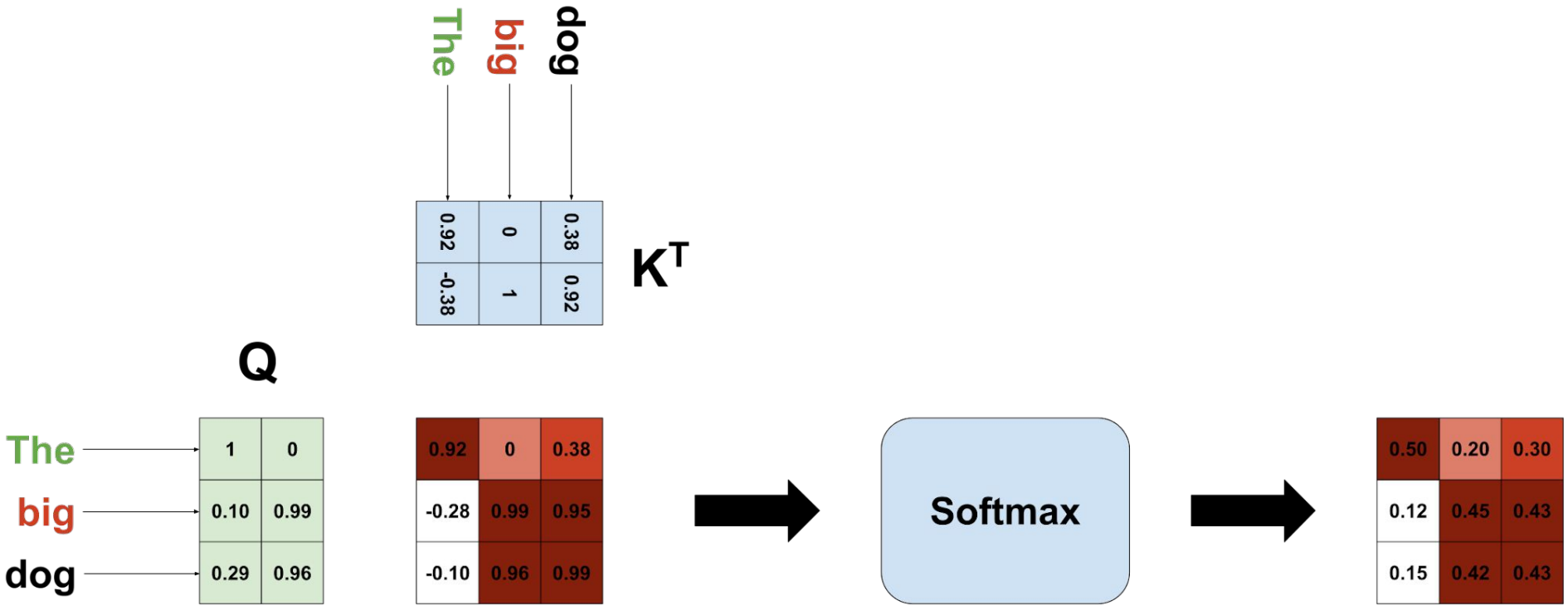
The : (0.50, -0.87)
big : (-0.70, 0.70)
dog : (0.81, 0.59)



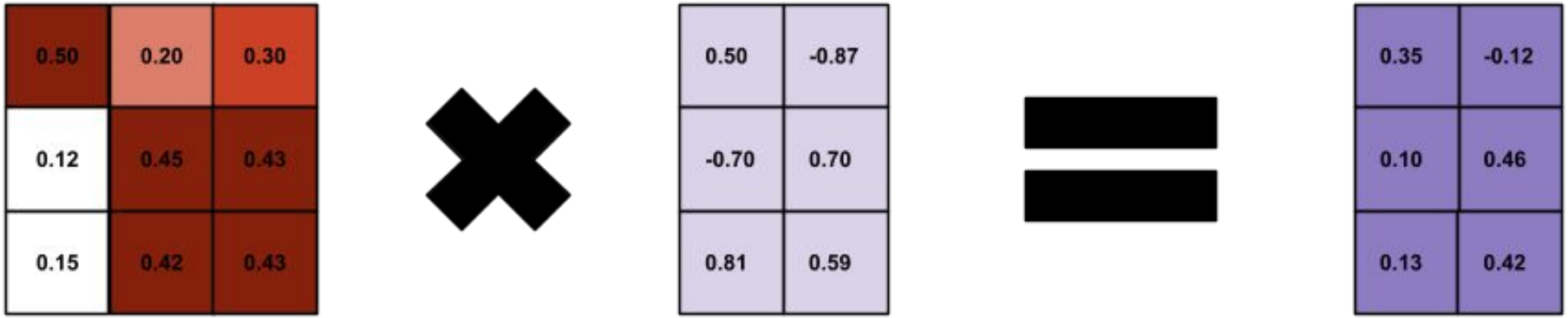
Intuition behind the attention mechanism - 2



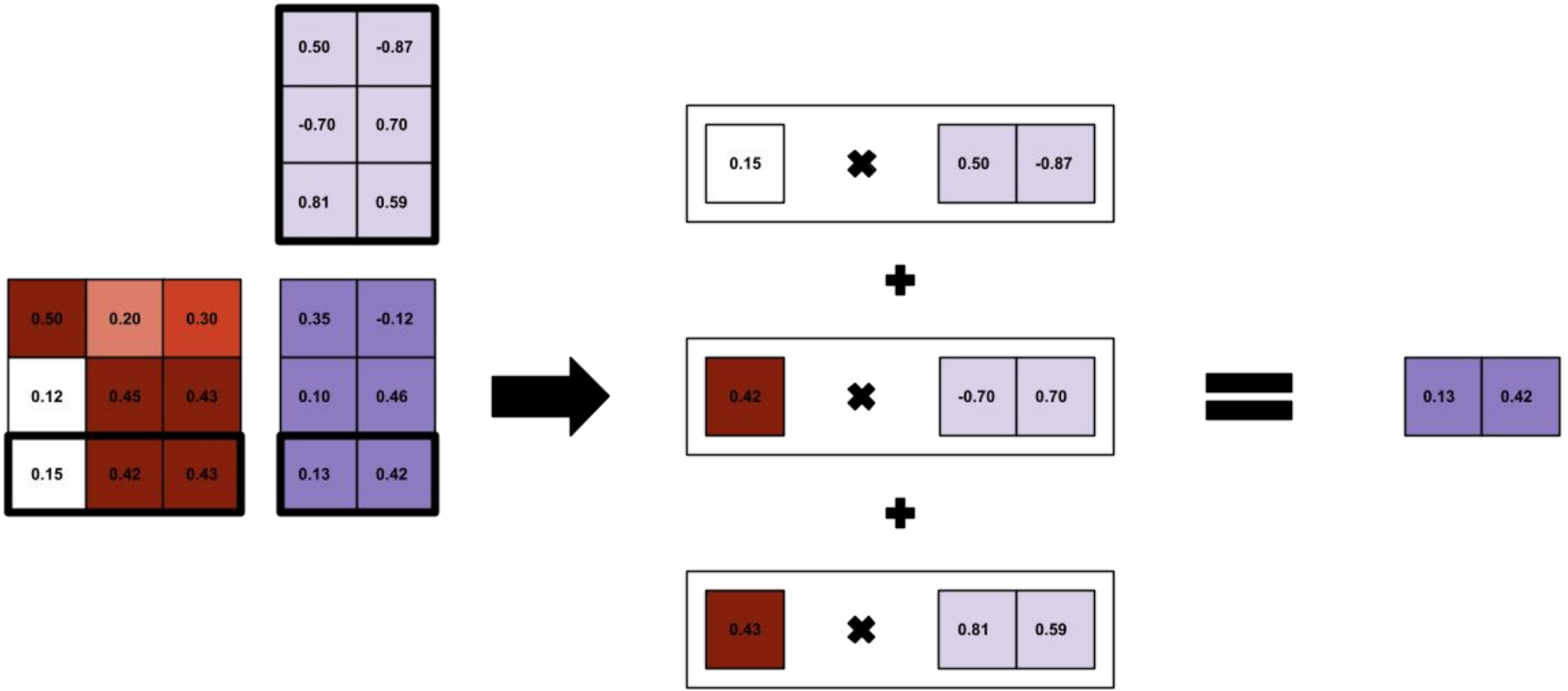
Intuition behind the attention mechanism - 3



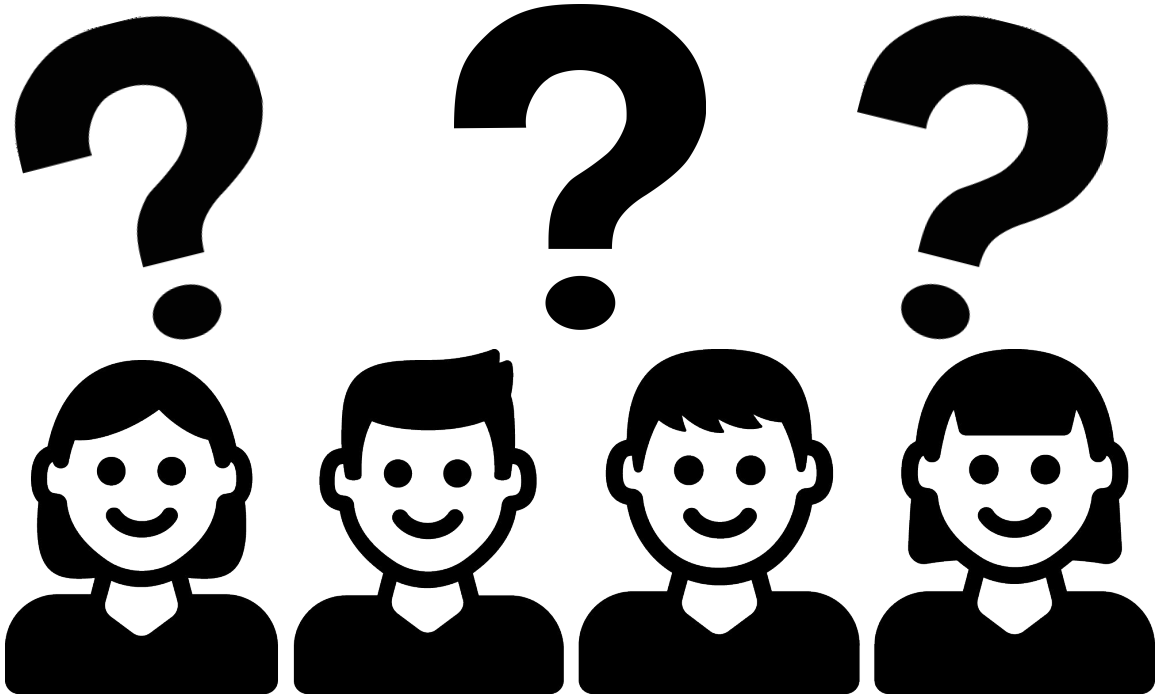
Intuition behind the attention mechanism - 4



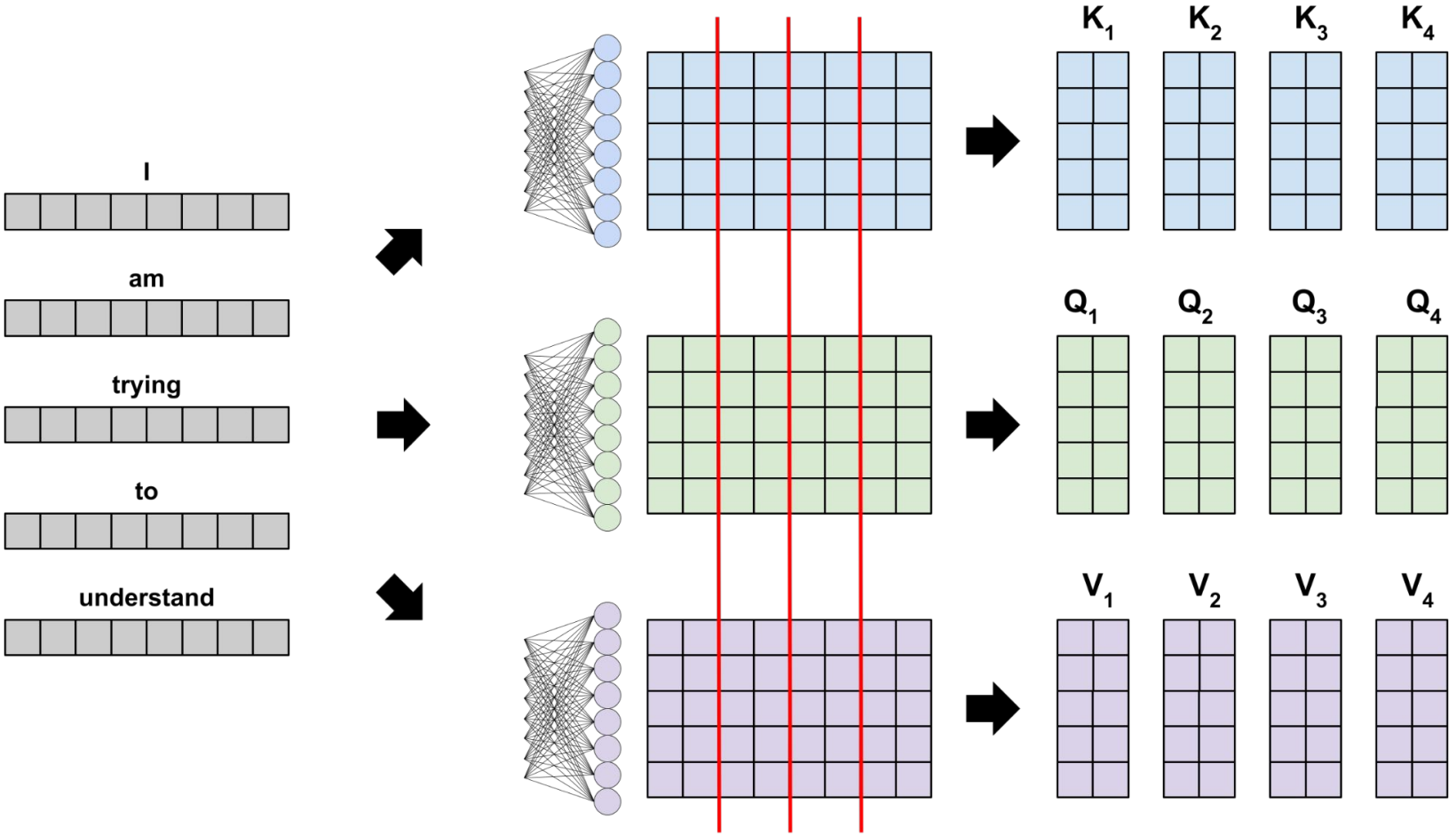
Intuition behind the attention mechanism - 4



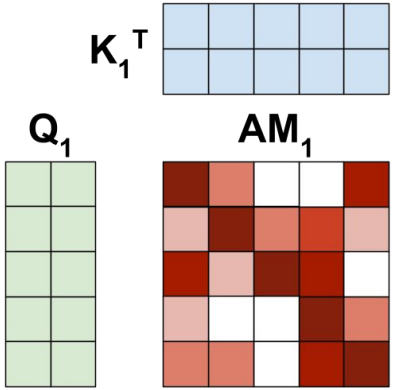
Question break #2

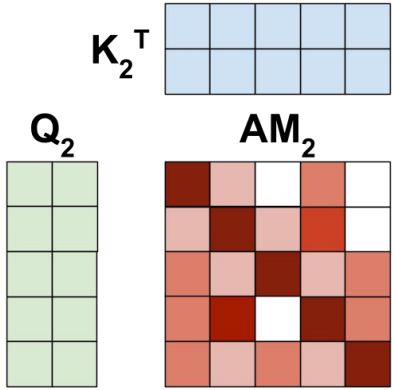


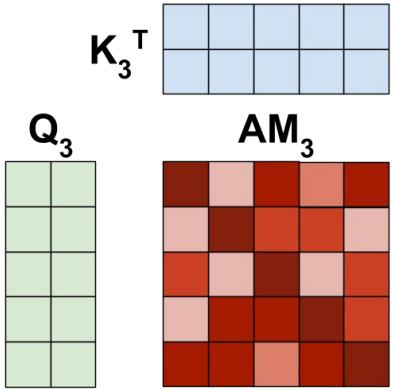
Multi-head Attention explained - 1

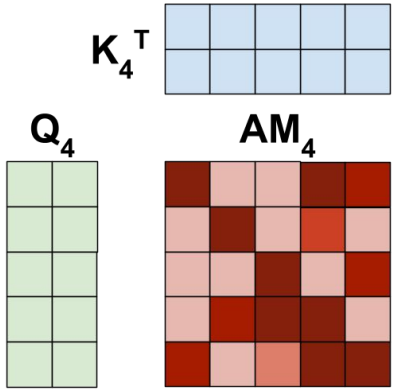


Multi-head Attention explained - 2

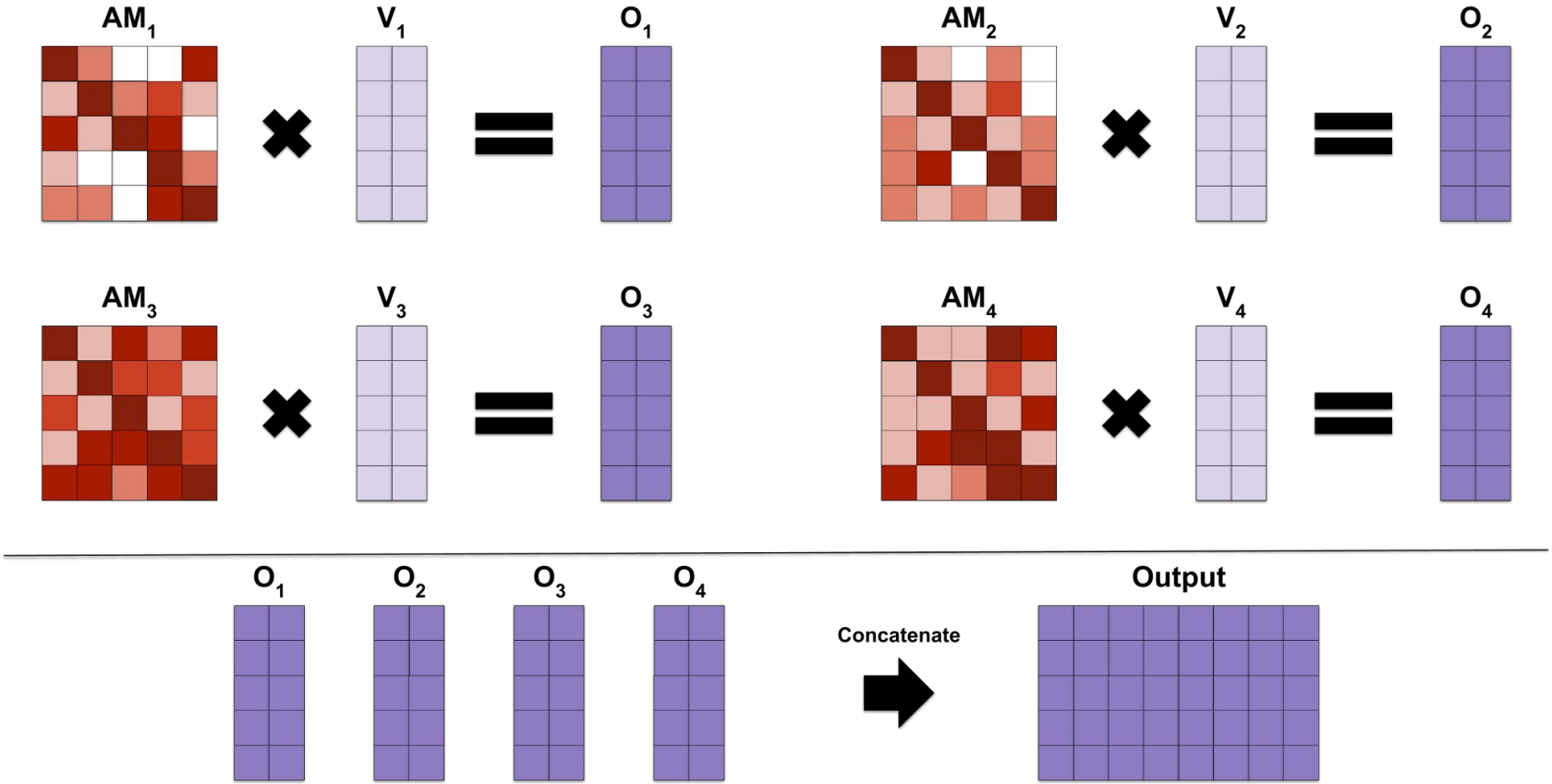




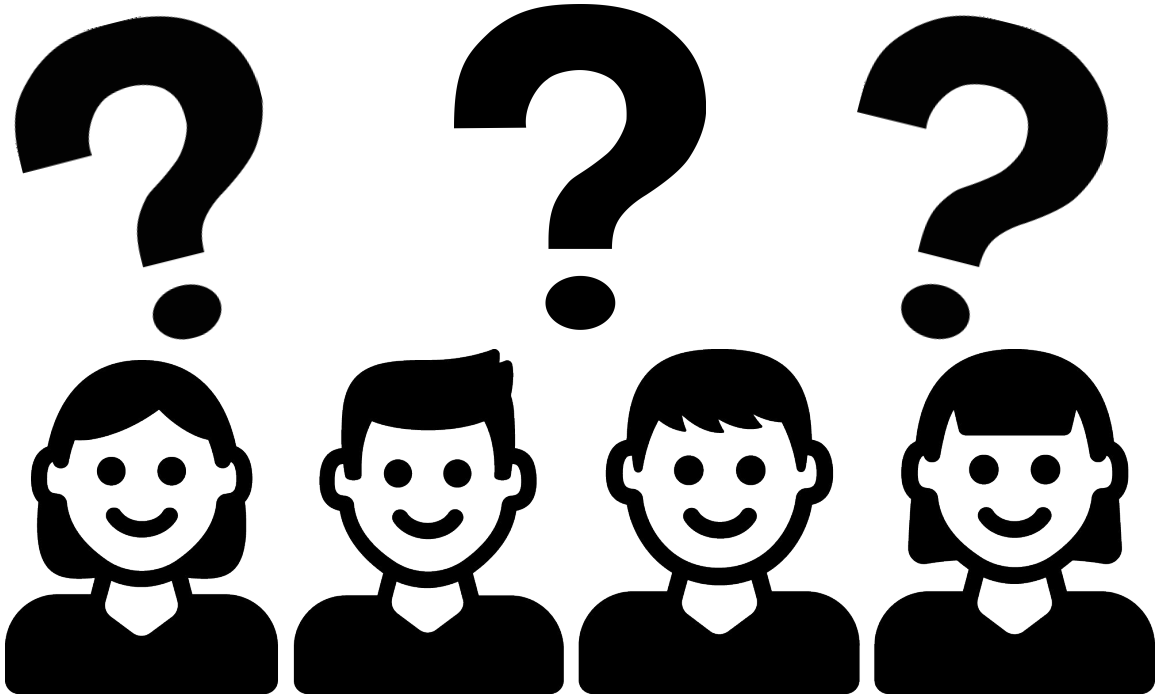




Multi-head Attention explained - 3



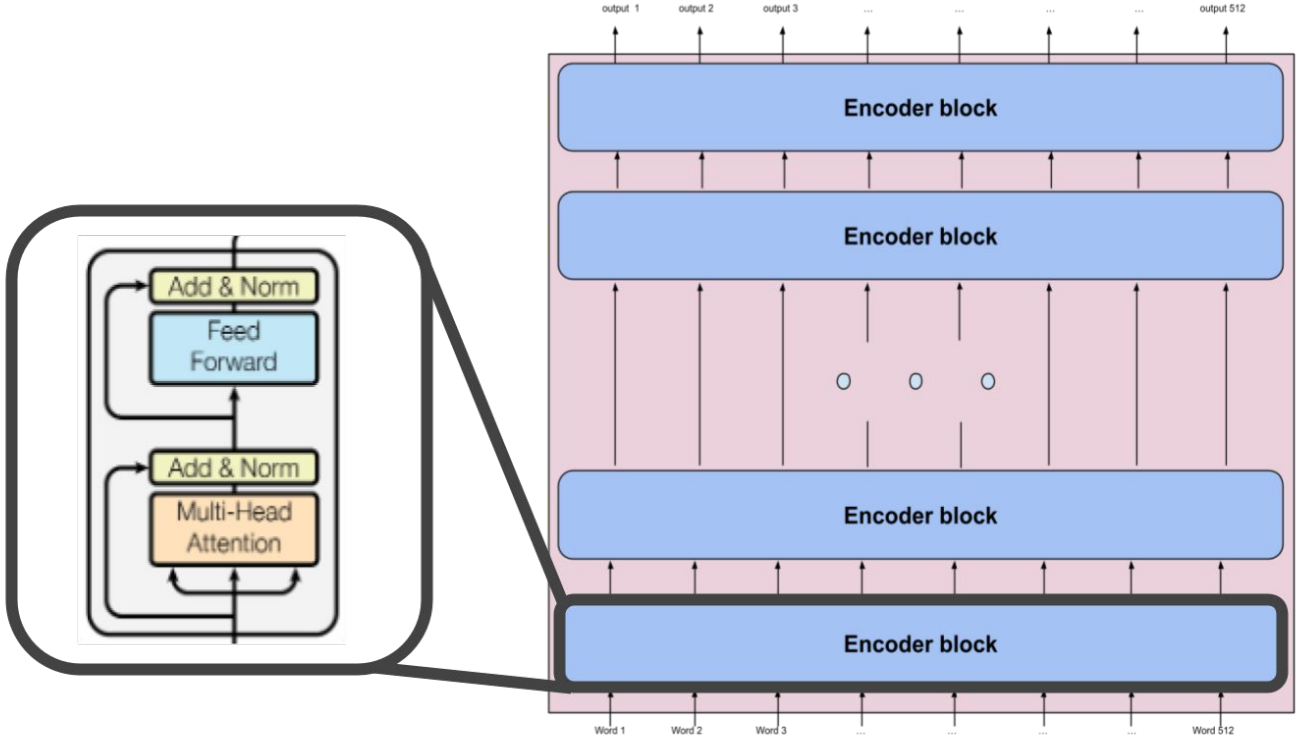
Question break #3



The most common transformer architectures

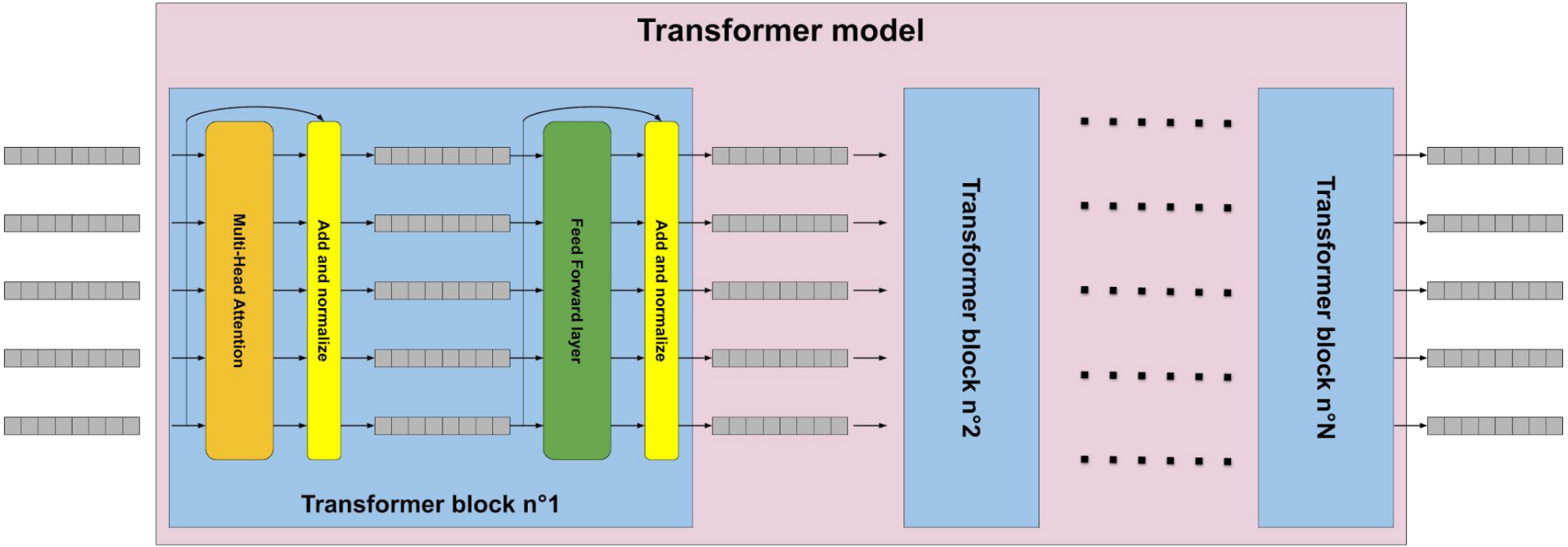
- ➔ Encoder (BERT)
- ➔ Decoder (GPT)
- ➔ Encoder-Decoder

Auto-encoding model (BERT or Encoder)



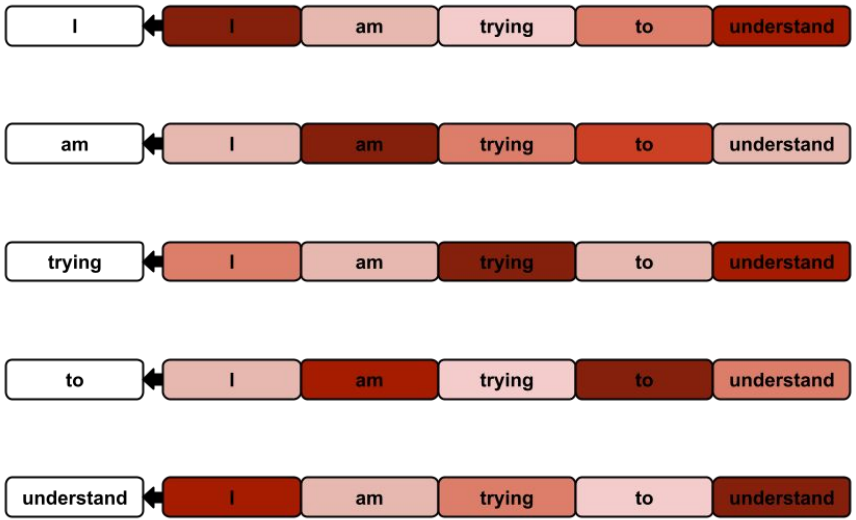
Attention is all you need, Vaswani A. et al, 2017

Auto-encoding model (BERT or Encoder)



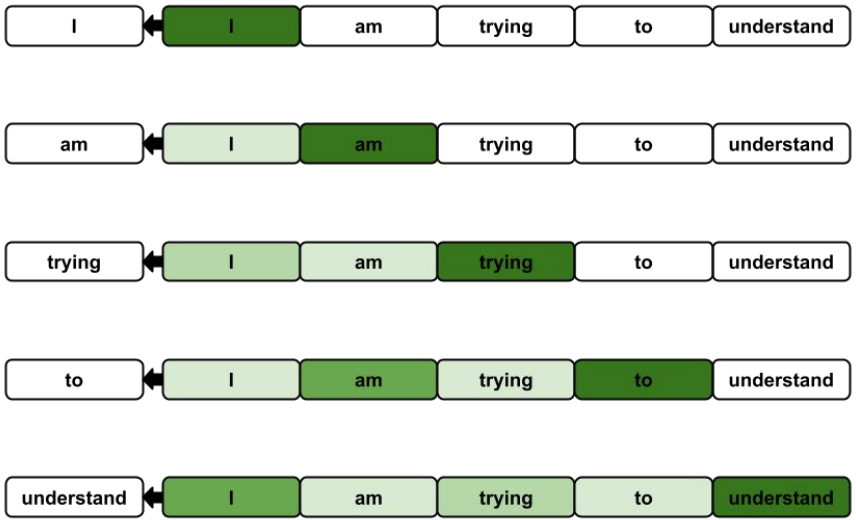
Bidirectional vs Unidirectional attention

Bidirectional attention (BERT - Encoder - Auto-encoding)

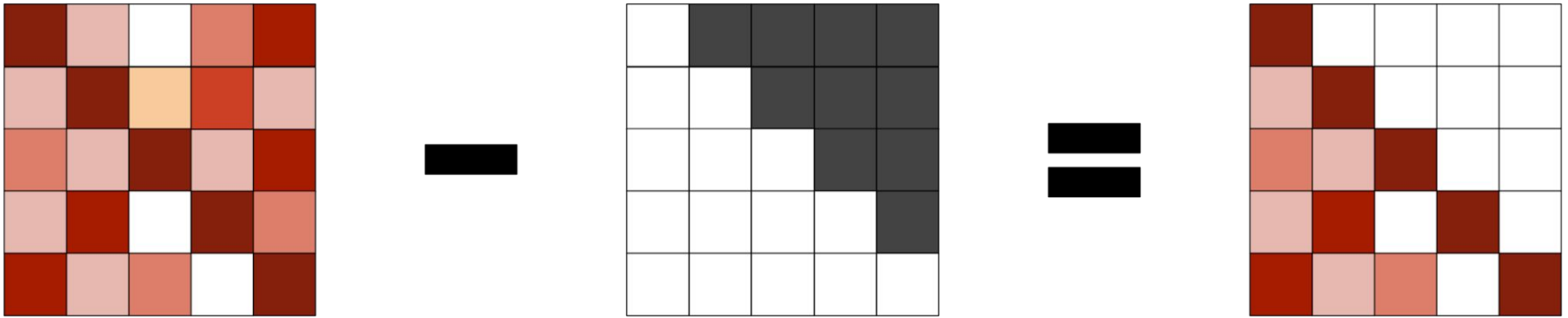


VS

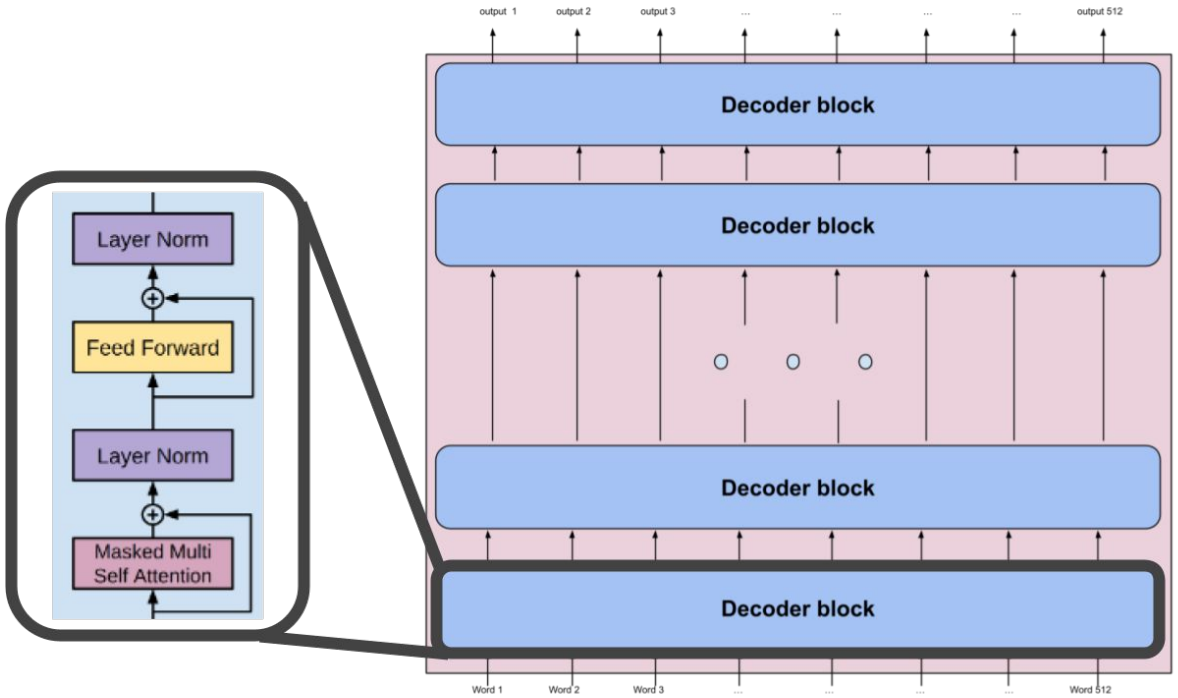
Unidirectional attention (GPT - Decoder - Auto-regressive)



Bidirectional vs Unidirectional attention

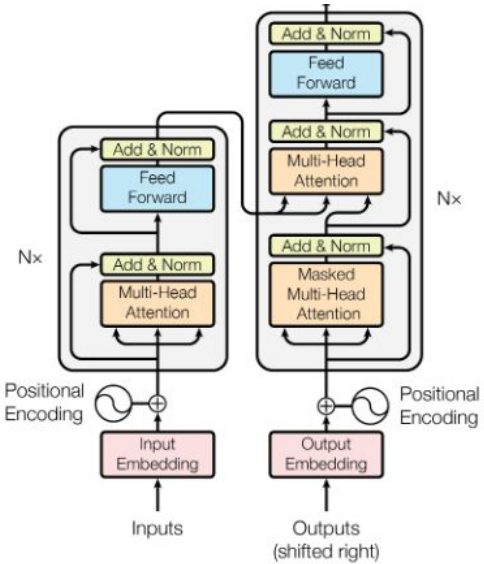


Auto-regressive model (GPT or Decoder)

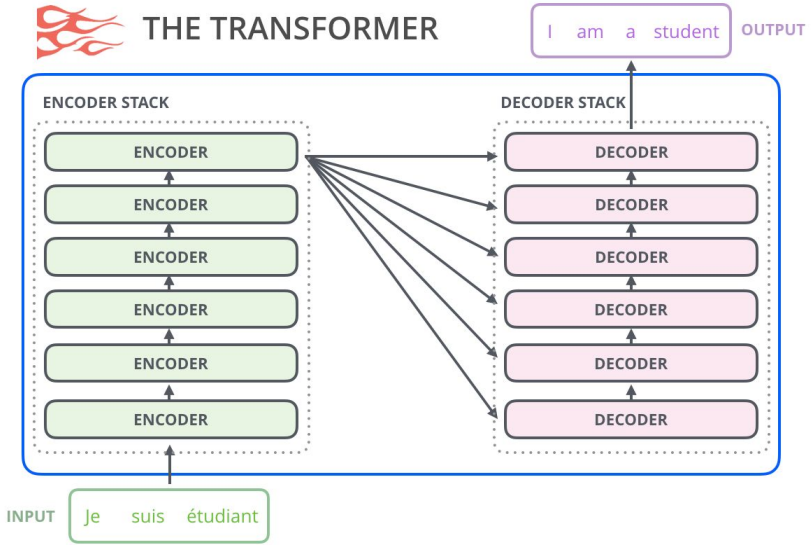


Improving Language Understanding by Generative Pre-Training, Radford et al

Encoder-Decoder model (1st



Attention is all you need, Vaswani A. et al, 2017

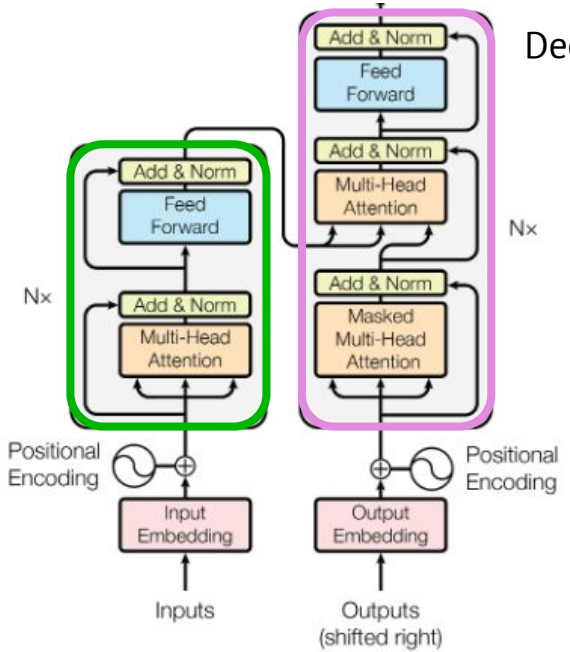


https://jalammar.github.io/illustrated-gpt2/

Small trap: misuse of language - 1

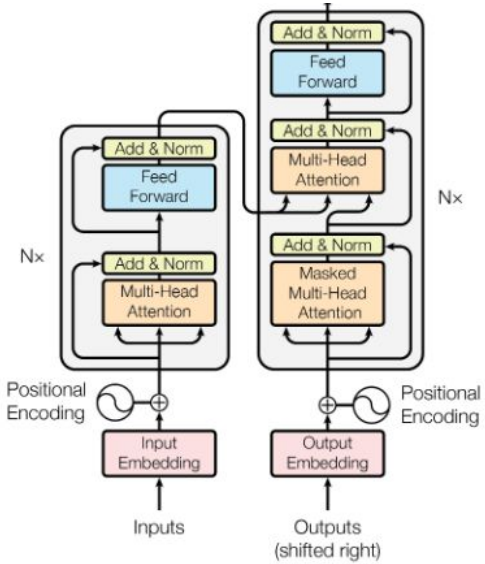
Encoder stack

Decoder stack

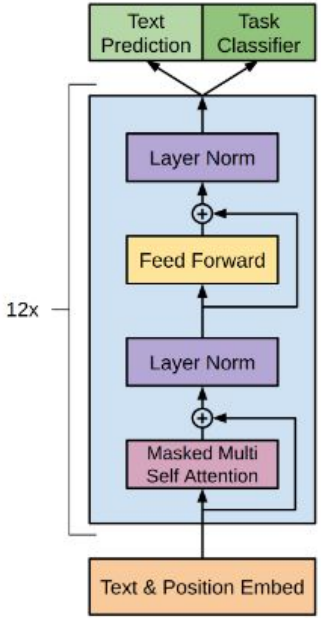


Attention is all you need, Vaswani A. et al, 2017

Small trap: misuse of language - 2

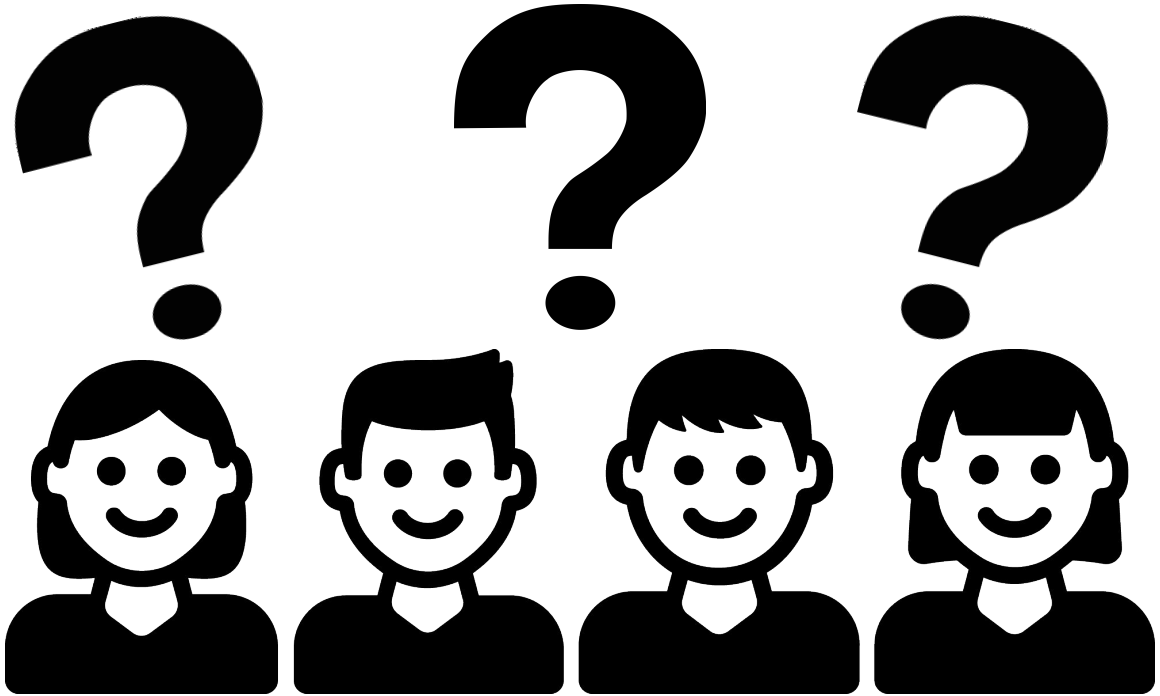


Attention is all you need, Vaswani A. et al, 2017



Improving Language Understanding by Generative Pre-Training, Radford et al

Question break #4



Pretraining

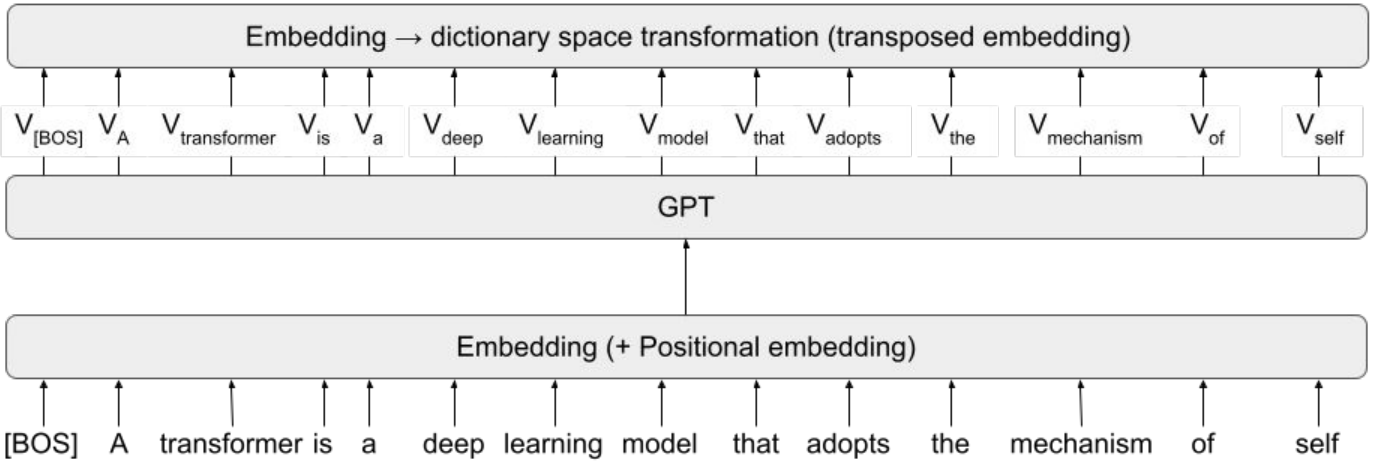
NLP covers multiple tasks:

- ➔ Question answering
- ➔ Inference
- ➔ Paraphrasing
- ➔ Grammar (is a given sentence correct grammatically?)
- ➔ Sentiment analysis
- ➔ etc

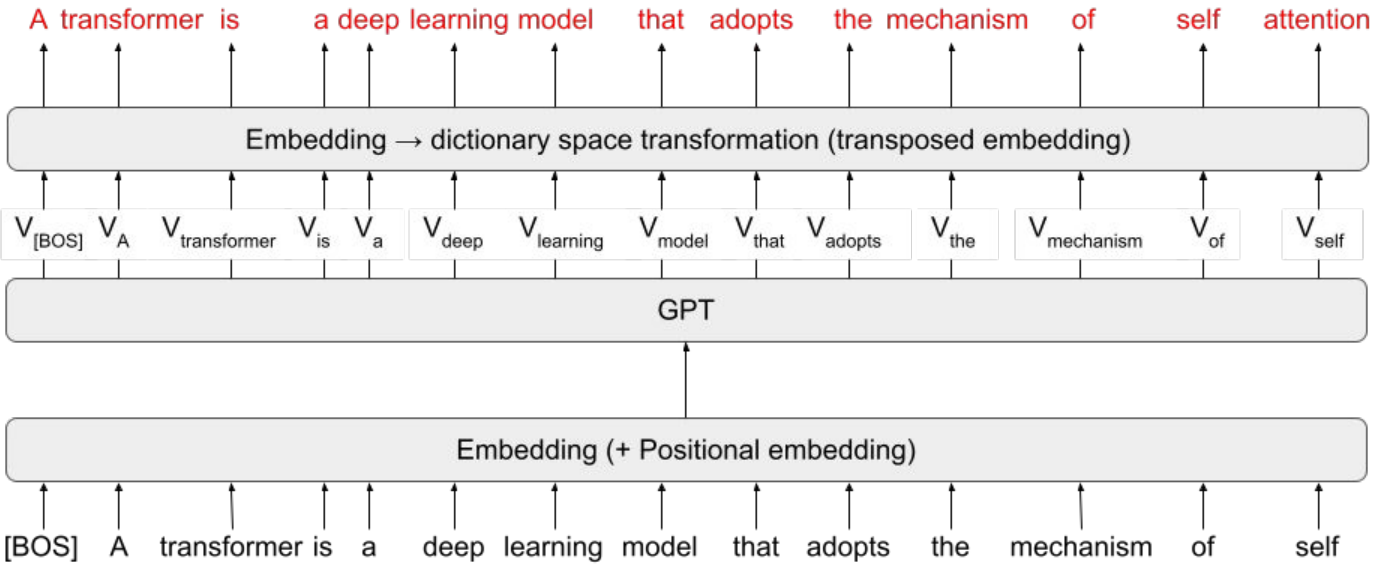
But the language is always the same

Can we pool trainings (at least partially)?

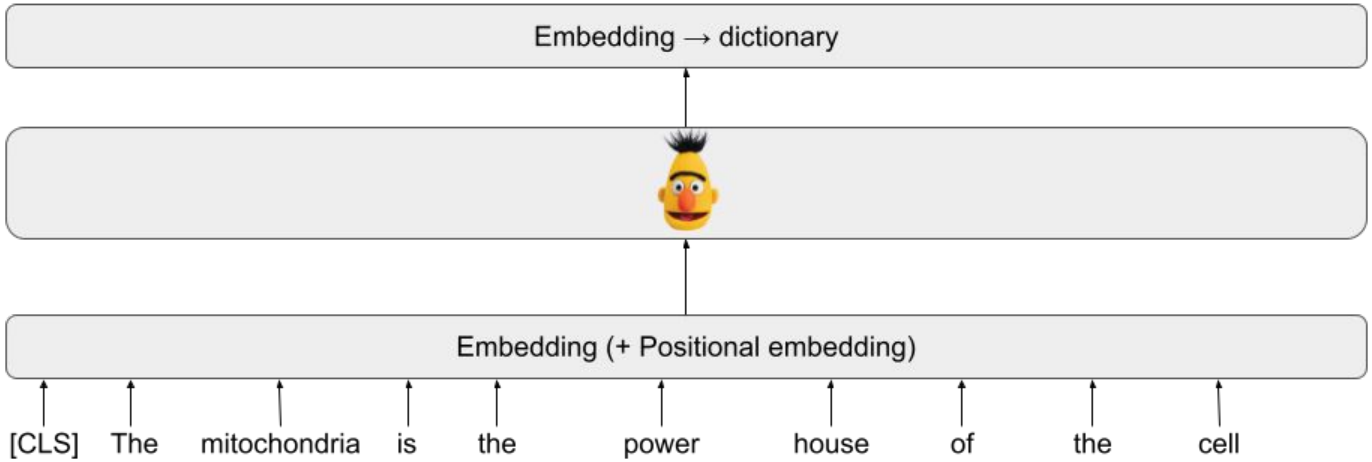
GPT pretraining



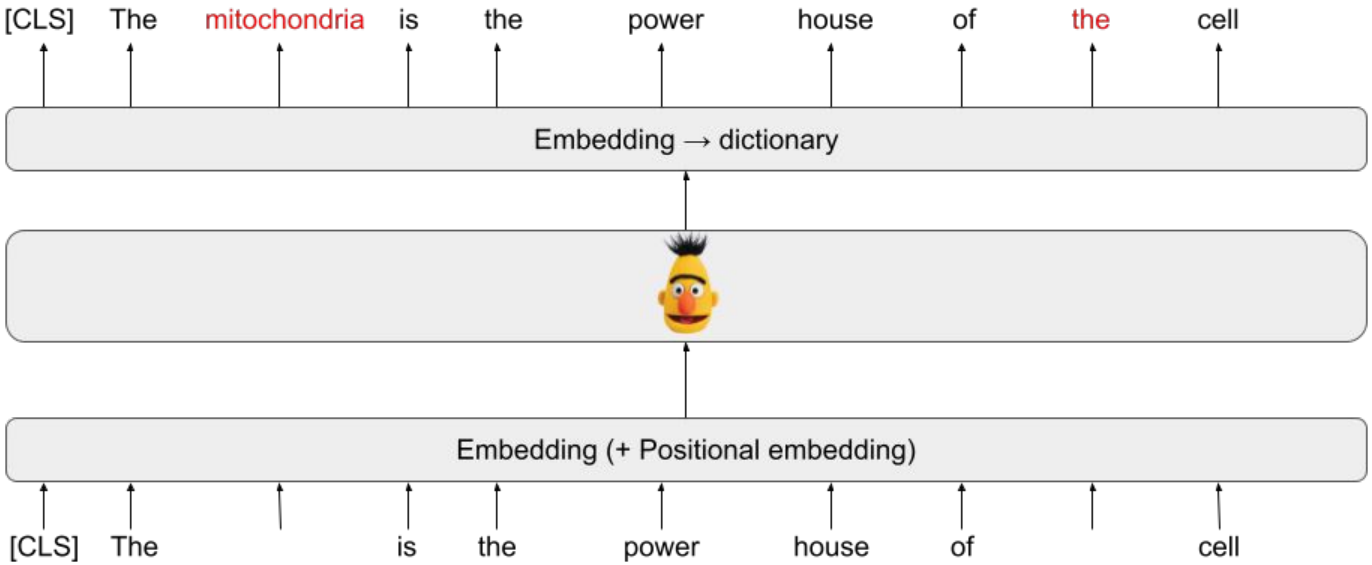
GPT pretraining: predict the next word



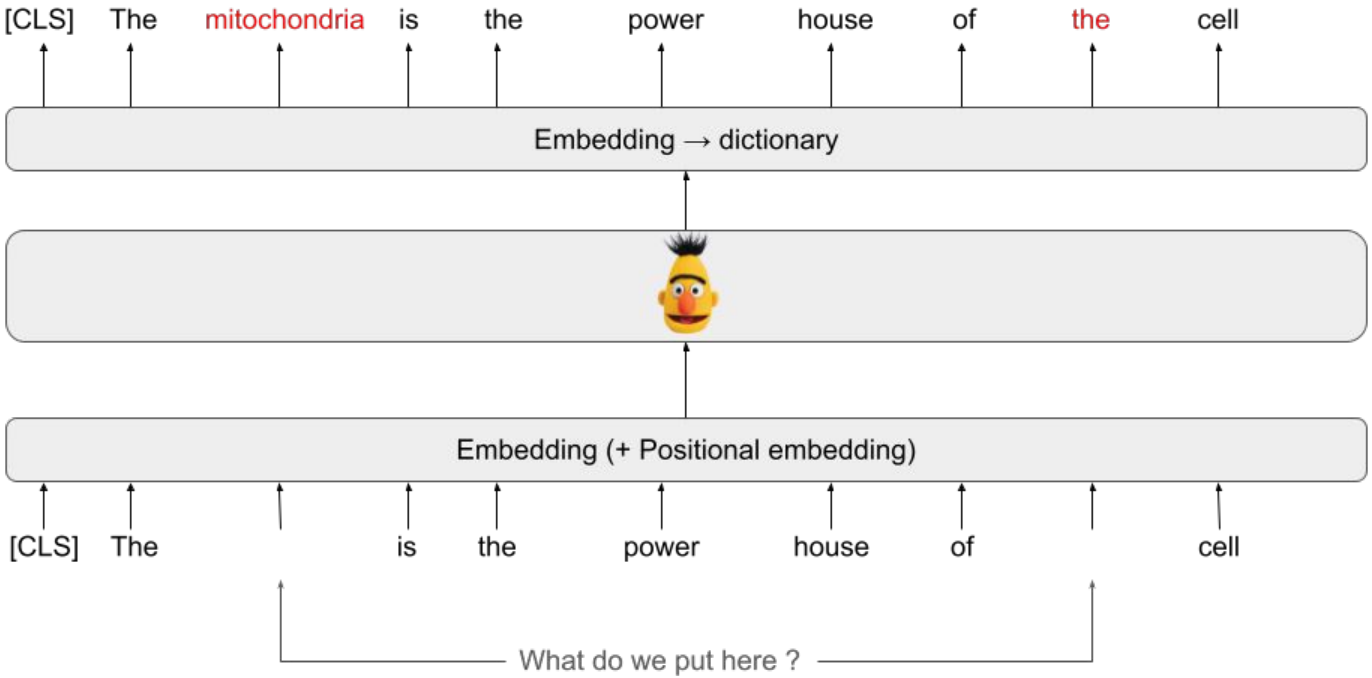
BERT pretraining



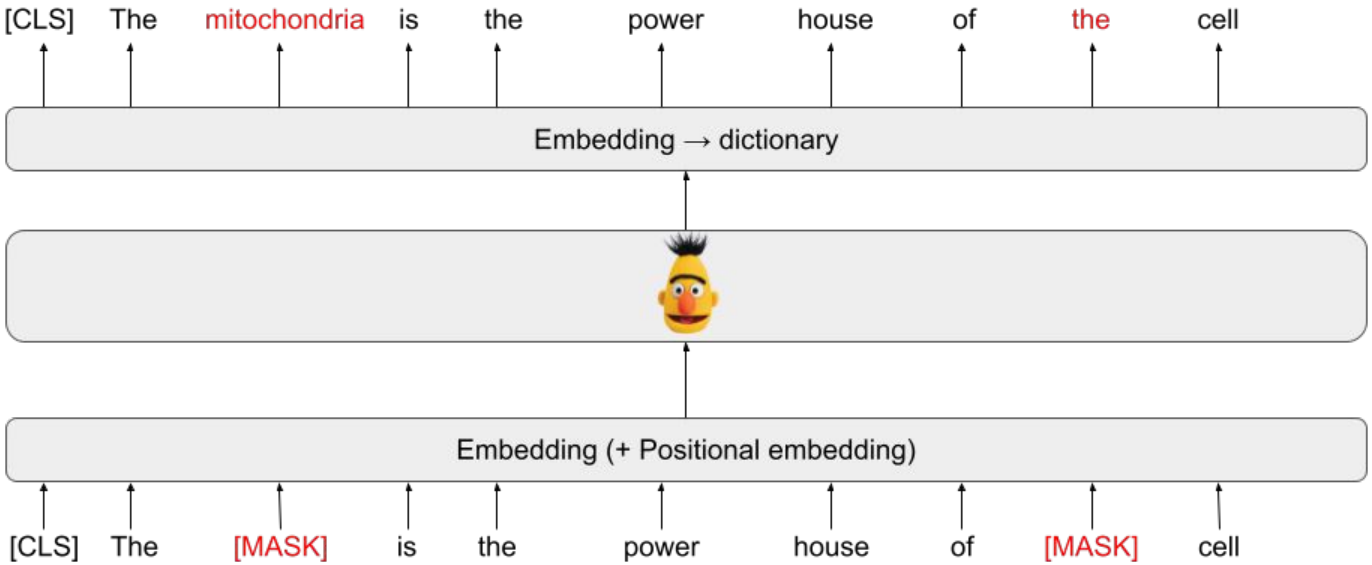
BERT pretraining: Masked Language Model 1



BERT pretraining: Masked Language Model 2

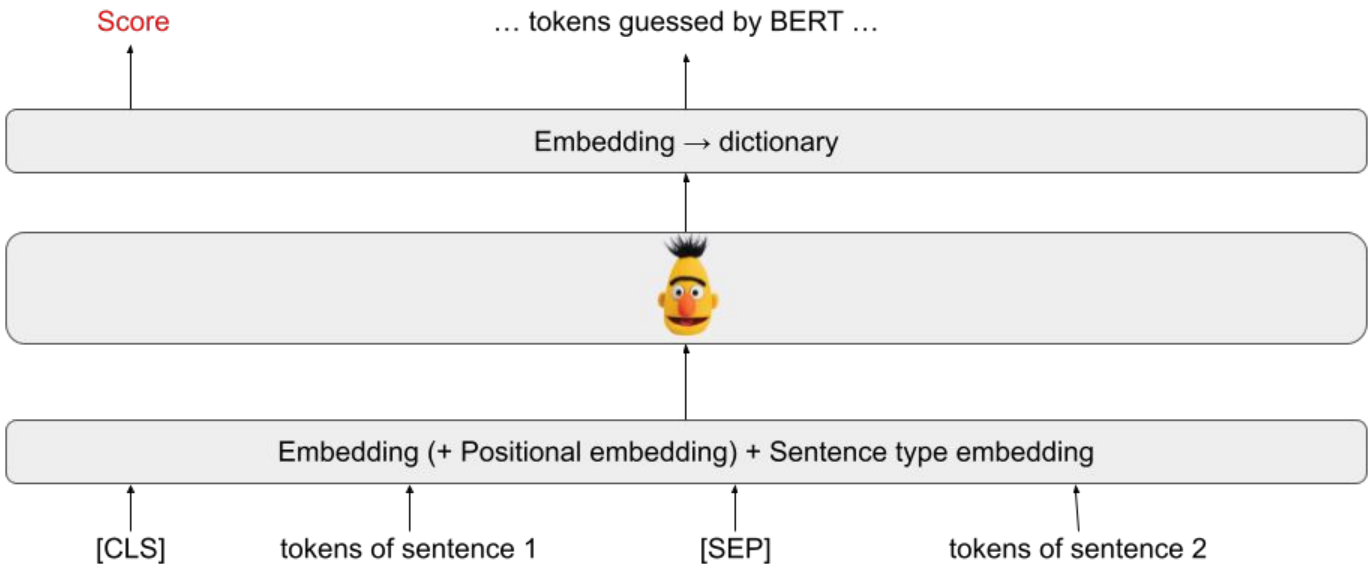


BERT pretraining: Masked Language Model 3



BERT pretraining: Next Sentence Prediction

Classification score : does sentence 2 comes right after sentence 1 ?



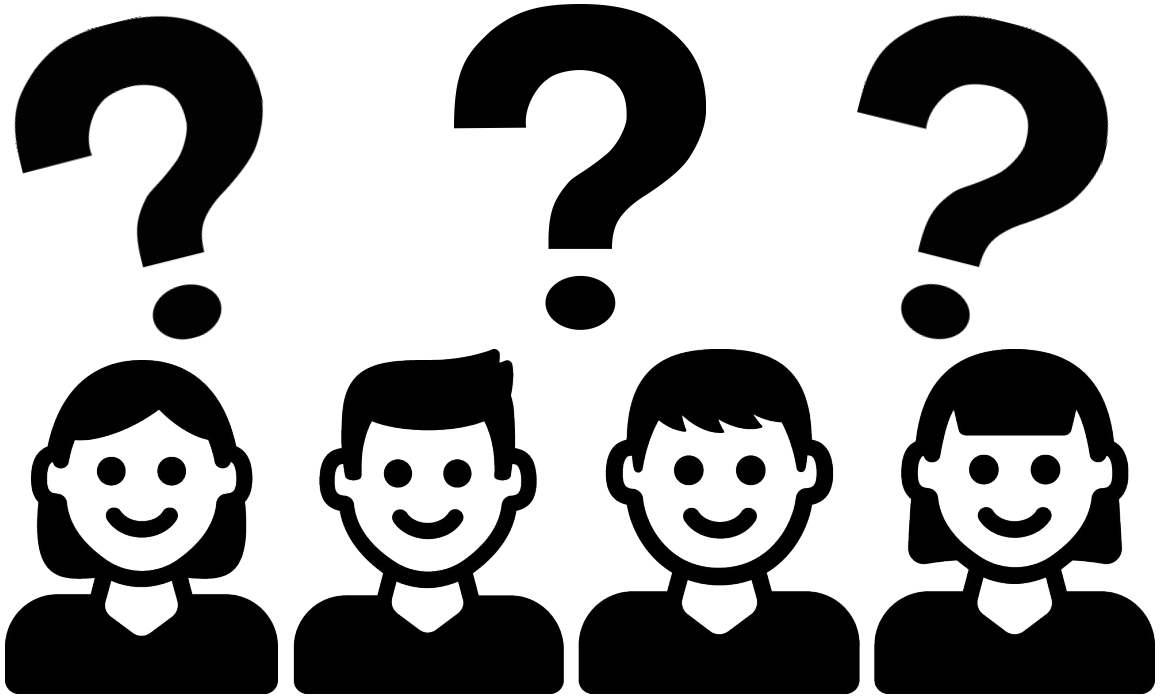
BERT pretraining

- ➔ Next Sentence Prediction is actually detrimental

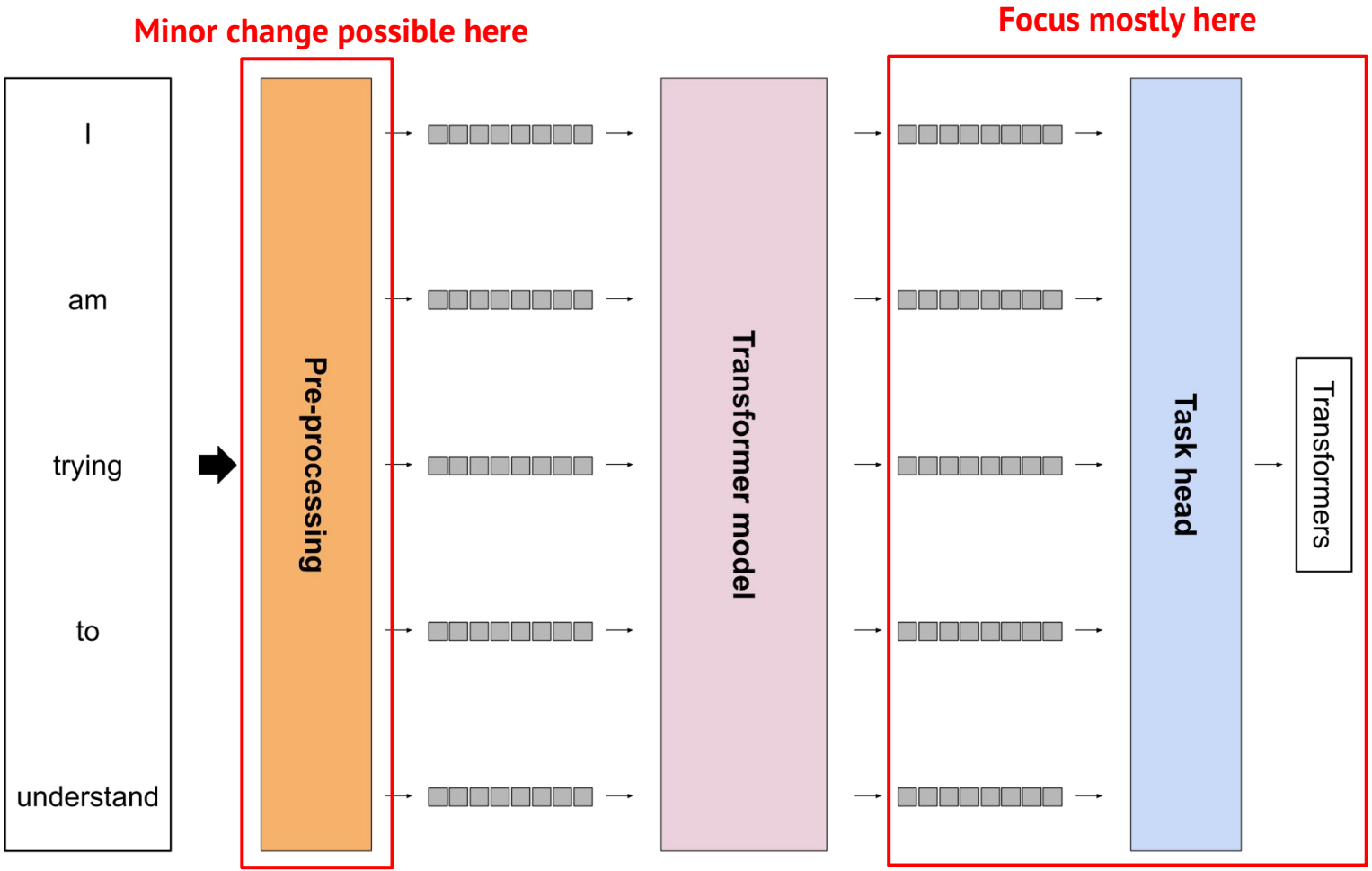
  Source: RoBERTa: A Robustly Optimized BERT Pretraining Approach, Liu Y. et al, 2019, https://arxiv.org/pdf/1907.11692.pdf

- ➔ You can use multiple objectives (as BERT did, or other variants like SpanBERT)

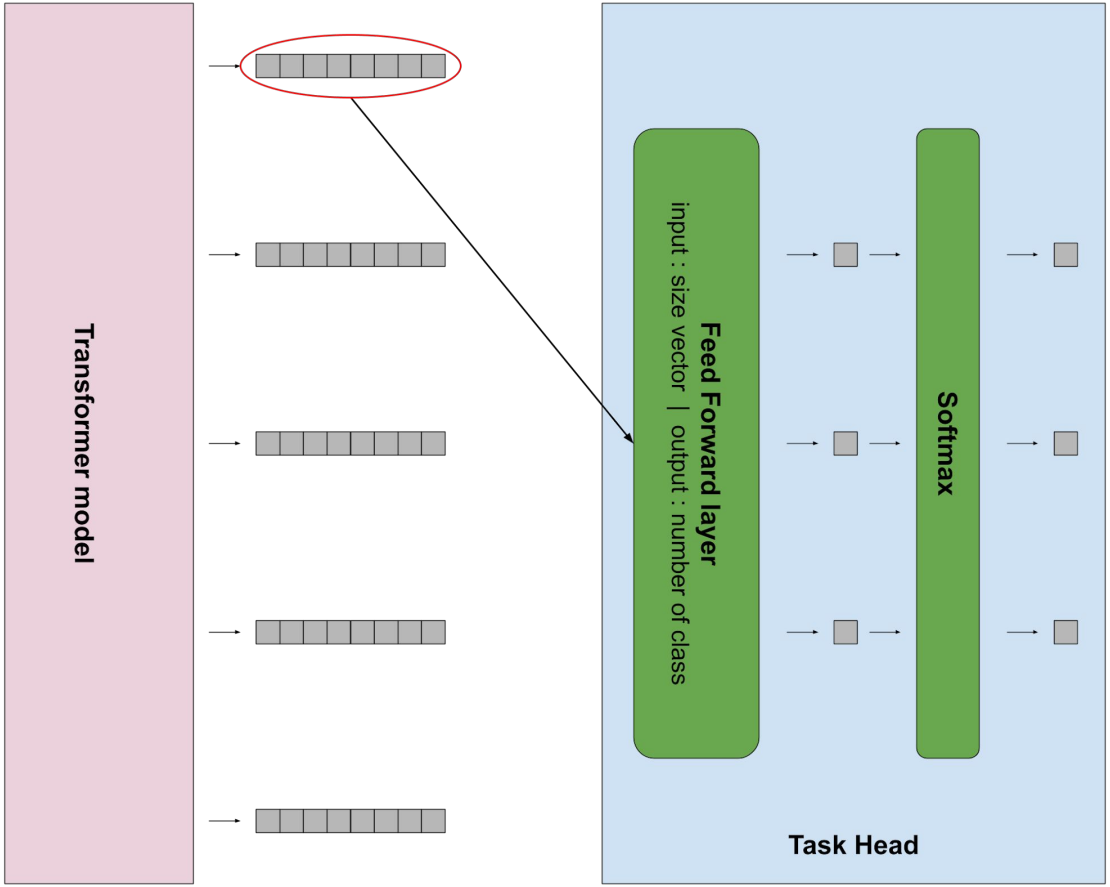
Question break #5



Classical fine tuning



Example of a classical fine tuning - classification



Text classification dataset

Input: reviews (text)
Label: 0, 1 or 2 (negative, neutral, positive)

Prompting example

Dataset

Input: Reviews
Label: Positive, neutral or negative

Template

The comment is

Prompting

Dataset

Input: X (text)
Label: Z (can be formulate as text)

Template

T (text)

Exemple de prompting

Type	Task	Input ([X])	Template	Answer ([Z])
Text CLS	Sentiment	I love this movie.	[X] The movie is [Z].	great fantastic ...
	Topics	He prompted the LM.	[X] The text is about [Z].	sports science ...
	Intention	What is taxi fare to Denver?	[X] The question is about [Z].	quantity city ...
Text-span CLS	Aspect Sentiment	Poor service but good food.	[X] What about service? [Z].	Bad Terrible ...
Text-pair CLS	NLI	[X1]: An old man with ... [X2]: A man walks ...	[X1]? [Z], [X2]	Yes No ...
Tagging	NER	[X1]: Mike went to Paris. [X2]: Paris	[X1][X2] is a [Z] entity.	organization location ...
Text Generation	Summarization	Las Vegas police ...	[X] TL;DR: [Z]	The victim ... A woman ... ...
	Translation	Je vous aime.	French: [X] English: [Z]	I love you. I fancy you. ...

Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing, Pengfei Liu and al

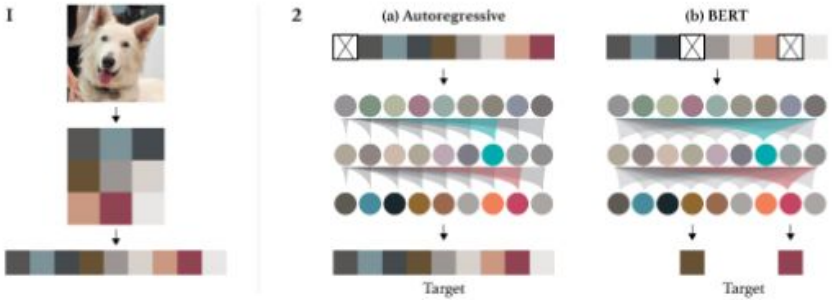
Transformers with images

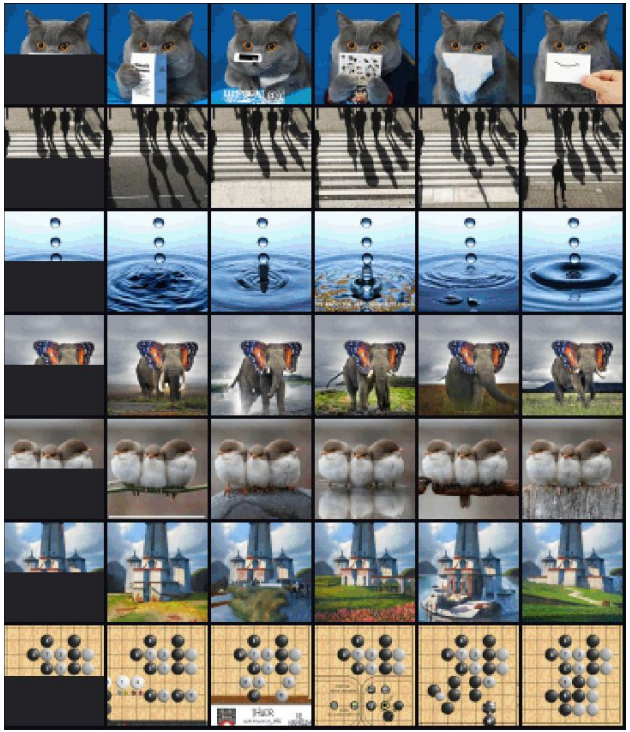
Not only for words but any 1D input.
What about 2D inputs (for instance pictures) ?

➔ ImageGPT
➔ Visual Transformer

Also useful in other fields: chemistry, biology…

ImageGPT



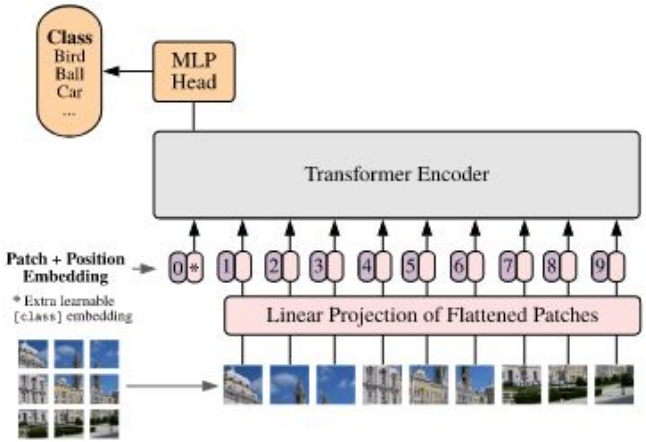


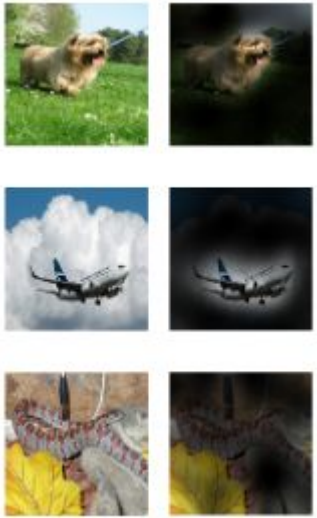
Generative Pretraining from Pixels, Chen M. et al, 2020

https://cdn.openai.com/papers/Generative_Pretraining_from_Pixels_V2.pdf

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

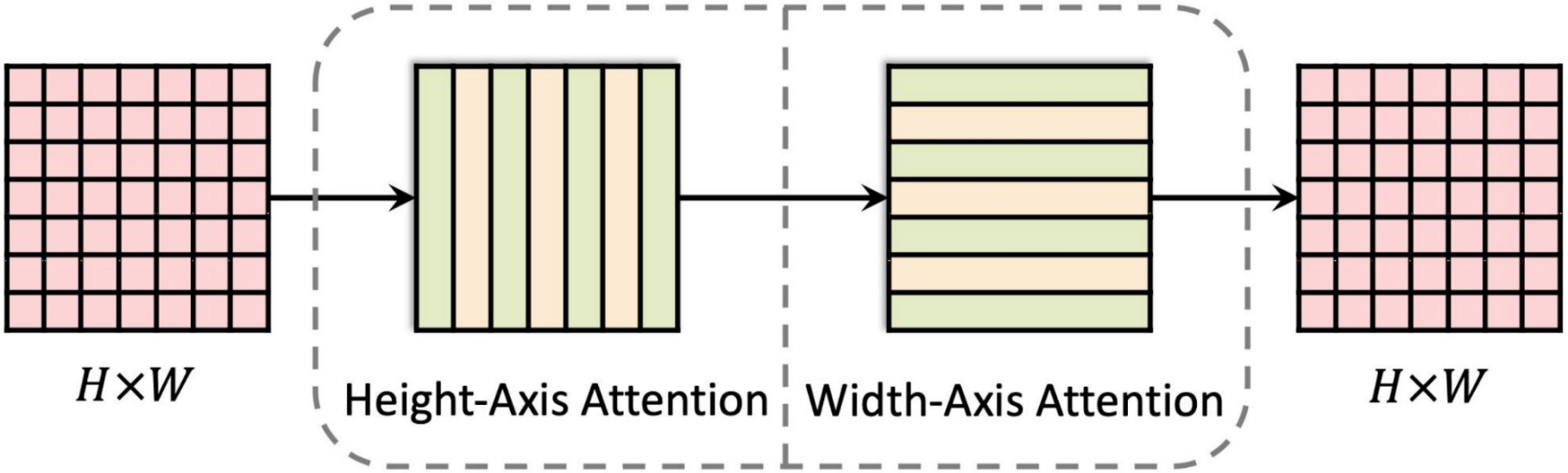
Visual Transformer





An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, Dosovitskiy A et al, 2021: https://arxiv.org/pdf/2010.11929.pdf

Axial Attention



https://ai.googleblog.com/2020/08/axial-deeplab-long-range-modeling-in.html

Bibliography

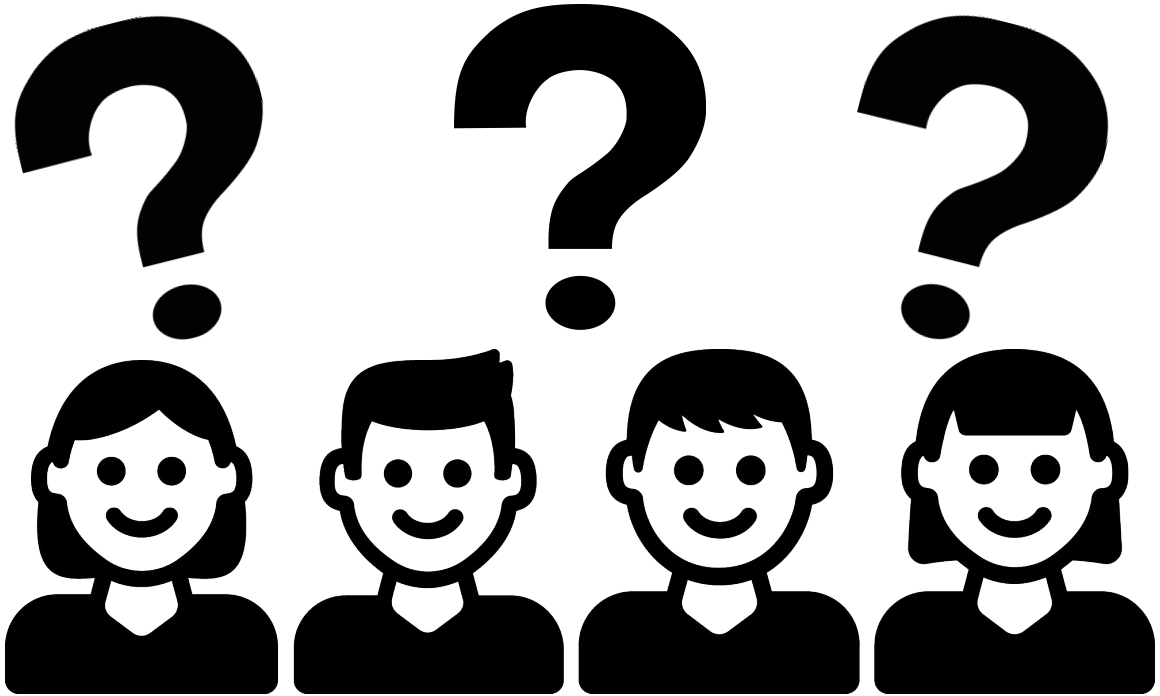
Papers:

- Attention Is All You Need (https://arxiv.org/abs/1706.03762)
- BERT (https://arxiv.org/abs/1810.04805)
- GPT (https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf)
- GPT3 (https://arxiv.org/abs/2005.14165)
- T5 (https://arxiv.org/abs/1910.10683)
- BLOOM (https://arxiv.org/abs/2211.05100)
- Foundation Models (https://arxiv.org/abs/2108.07258)

Book:

- Natural Language Processing with Transformers

Question break #6 &
In practice



Next, on Fidle :

9 New !

Graph Neural Network

GNN

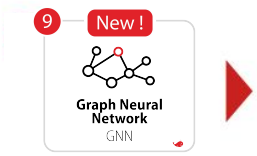
Jeudi 26 janvier, 14h00

Séquence 9 :

Travailler avec des données structurées : Graph Neural Network (GNN)

Omniprésence et problématique des graphes
Approches classiques
Les GNN
Cas d'usages
Durée : 2h

Next on Fidle :



Jeudi 26 janvier, 14h00

Séquence 9 :

Travailler avec des données structurées : Graph Neural Network (GNN)

To be continued...

Contact@fidle.cnrs.fr
FIDLE https://fidle.cnrs.fr
YouTube https://fidle.cnrs.fr/youtube

FIDLE