Transformer-based sentence encoding and decoding

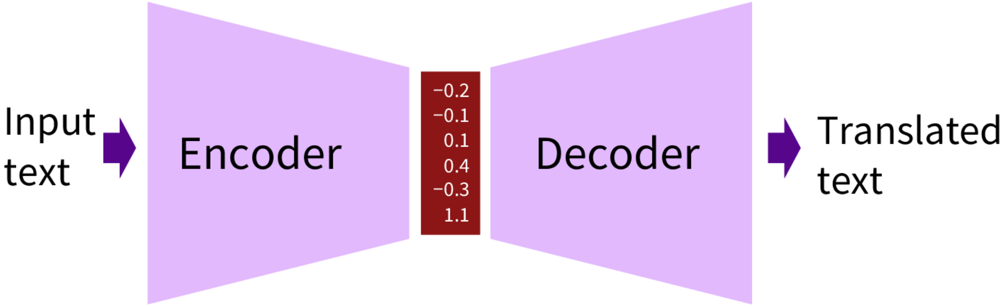
Prof. Luca Cagliero
Dipartimento di Automatica e Informatica
Politecnico di Torino

Lecture goal

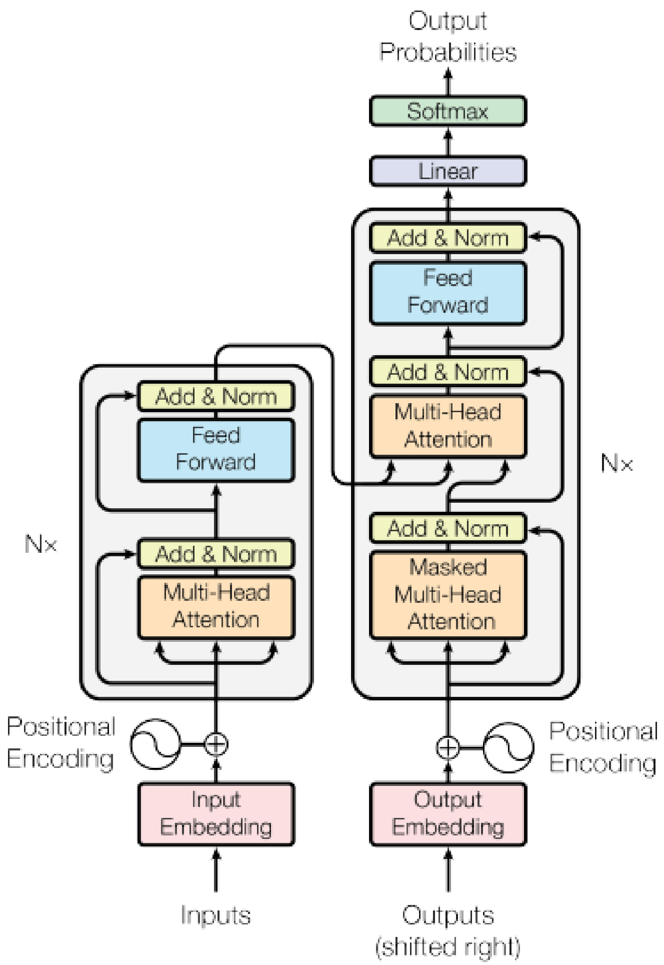
- Transformers recap
- Sentence encoding & decoding
- The GPT decoders
- The BERT encoder

The encoder-decoder mechanism

- The encoder reads a variable-length sequence and maps it to a fixed-size vector
- The decoder reads the vector and produces a variable-length output sequence

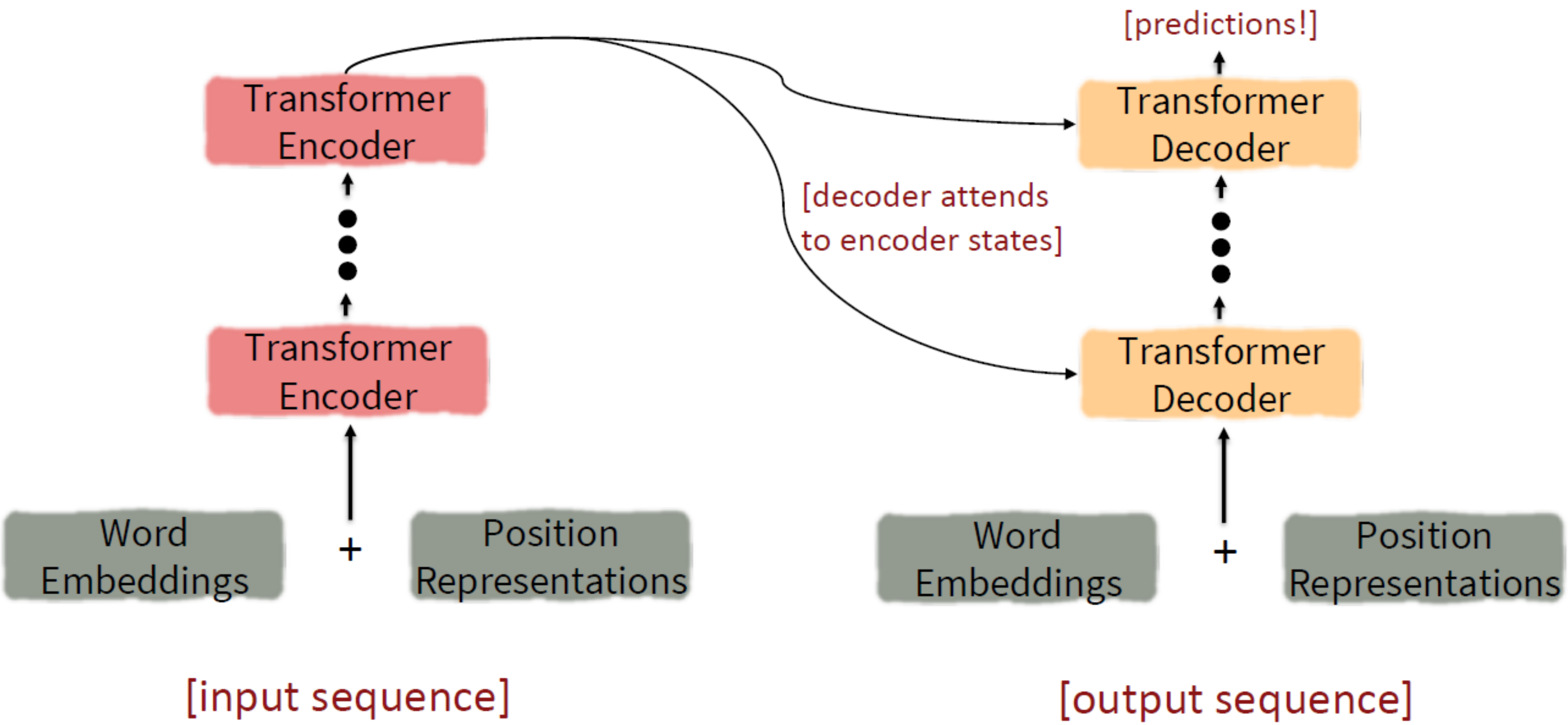


The Transformer architecture



Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. NIPS 2017

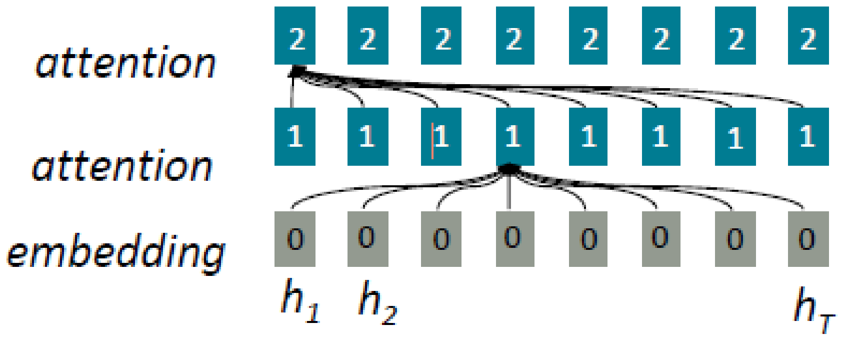
The Transformer architecture



Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The attention mechanism

- Treat each word's representation as a query to access and incorporate information from a set of values
  - From the decoder to the encoder -> cross-attention
  - Within a single sentence -> self-attention
- Highly parallelizable
  - All words interact at every layer



Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. NIPS 2017

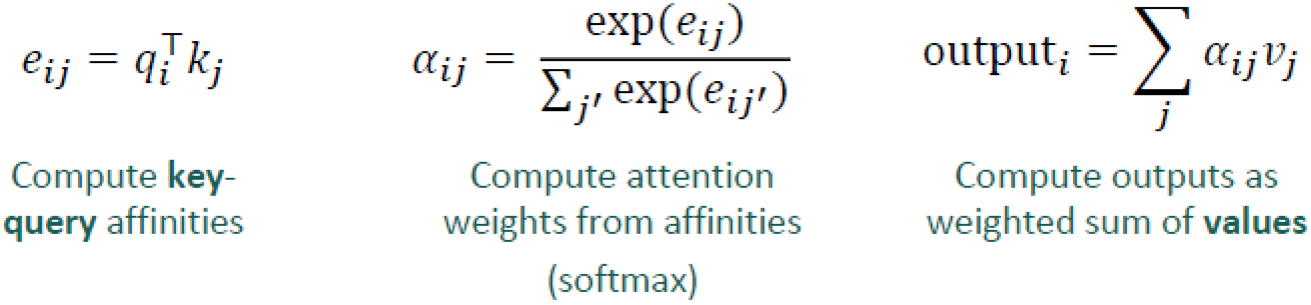
Self-attention

- We operate on queries, keys, and values
  - Queries q_1, q_2, ..., q_T where $q_i \in \mathbb{R}^d$
  - Keys k_1, k_2, ..., k_T where $k_i \in \mathbb{R}^d$
  - Values v_1, v_2, ..., v_T where $v_i \in \mathbb{R}^d$
- In practice
  - The number of queries can differ from the number of keys and values

Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The self-attention mechanism

- In self attention the queries, keys, and values are drawn from the same source
  - If the output of the previous layer is $x_1,\ldots,x_T$ (one vector per word) then we could let $v_i=k_i=q_i=x_i$
  - Use the same vectors for all of them
- (Dot-product) self-attention operation



$$e_{ij} = q_i^{\mathsf{T}} k_j$$

Compute **key-query** affinities

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{j'} \exp(e_{ij'})}$$

Compute attention weights from affinities (softmax)

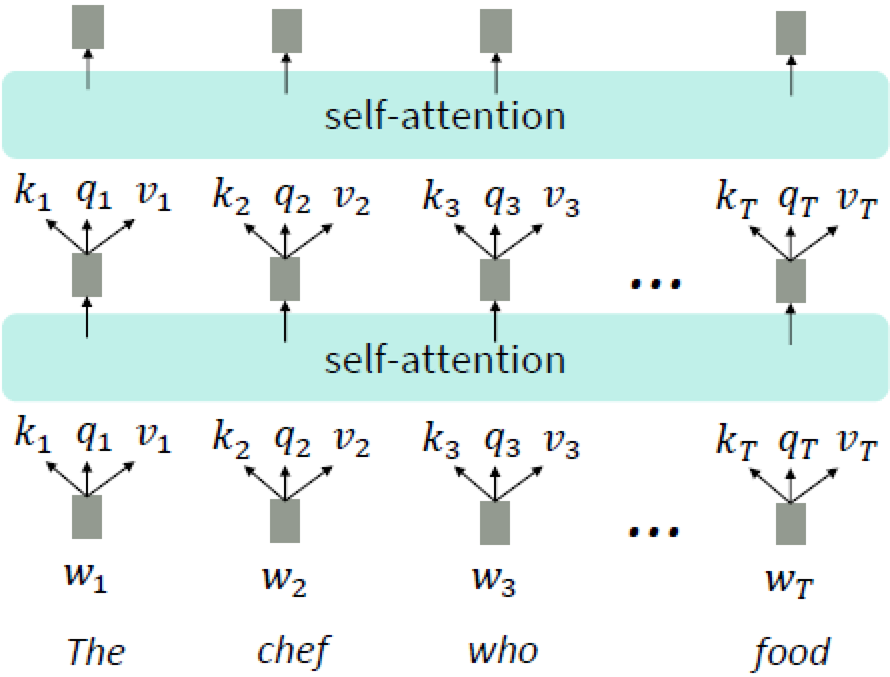
$$\text{output}_i = \sum_j \alpha_{ij} v_j$$

Compute outputs as weighted sum of **values**

Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The self-attention mechanism

- Input order is unknown
- Non-sequential approach



Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The self-attention mechanism

- Align words in the sequence with other words in the same sequence
- More effective than LSTMs in avoiding locality bias
- More efficient than recurrent/convolutional models

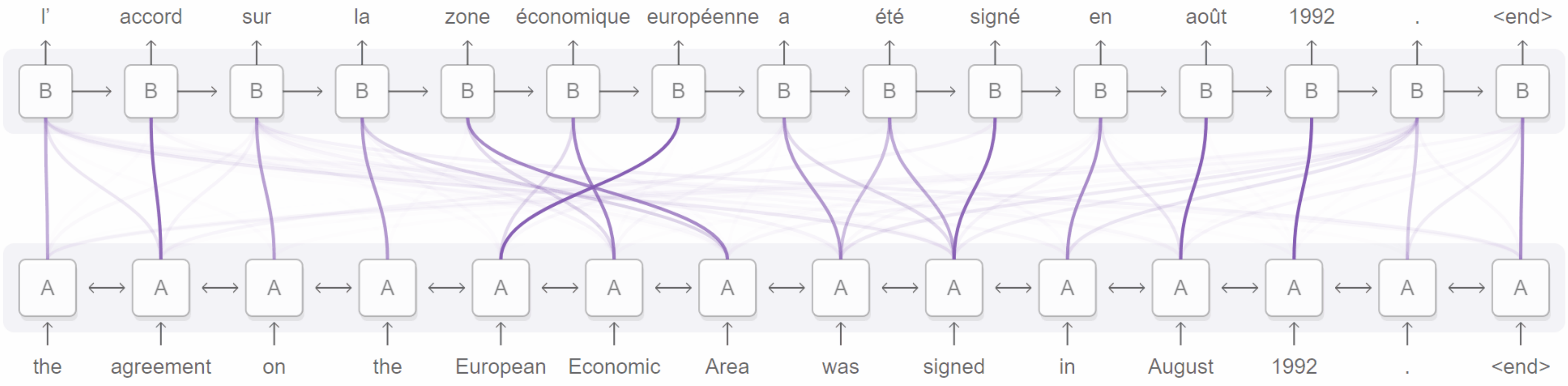


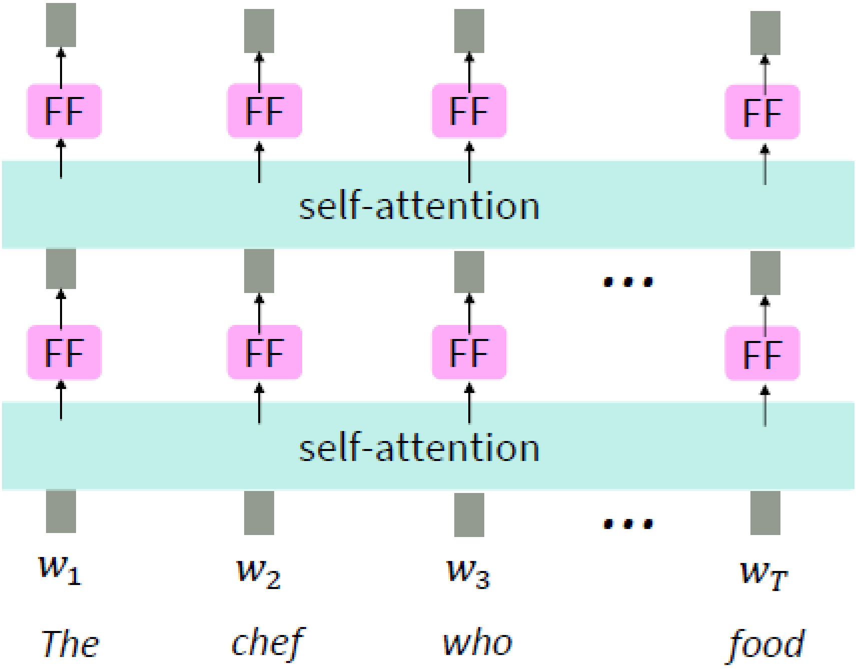
Diagram derived from Fig. 3 of Bahdanau, *et al.* 2014

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. NIPS 2017

The self-attention mechanism

- Add nonlinearities in the self-attention mechanism

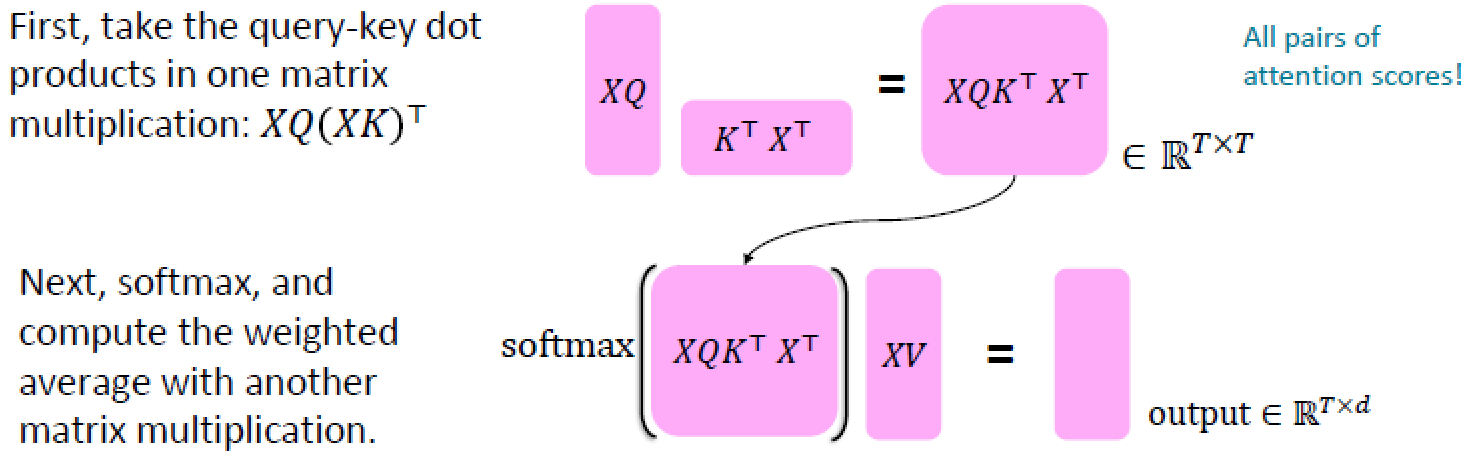
$$
\begin{aligned}
m_i &= MLP(\text{output}_i) \\
&= W_2 * \text{ReLU}(W_1 \times \text{output}_i + b_1) + b_2
\end{aligned}
$$



Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The Transformer encoder

- The input is a sequence of tokens
- Tokens are
  - Mapped into a sequence of numbers (entries in the vocabulary)
  - processed by the neural network
- The output is a sequence of vectors
  - Each vector corresponds to an input token with the same index



First, take the query-key dot products in one matrix multiplication: $XQ(XK)^\top$

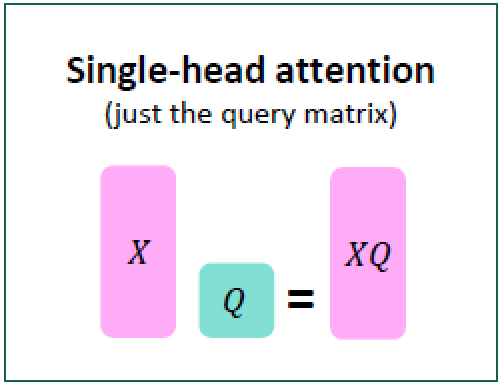
Next, softmax, and compute the weighted average with another matrix multiplication.

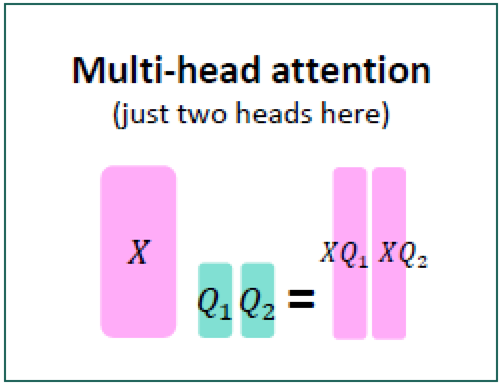
XQ
$K^\top X^\top$
$=$
$XQK^\top X^\top$
$\in \mathbb{R}^{T \times T}$
All pairs of attention scores!
softmax
$XQK^\top X^\top$
XV
$=$
output $\in \mathbb{R}^{T \times d}$

Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

The Transformer encoder

- Multi-head attention
  - For a generic word attend to multiple places in the sentence

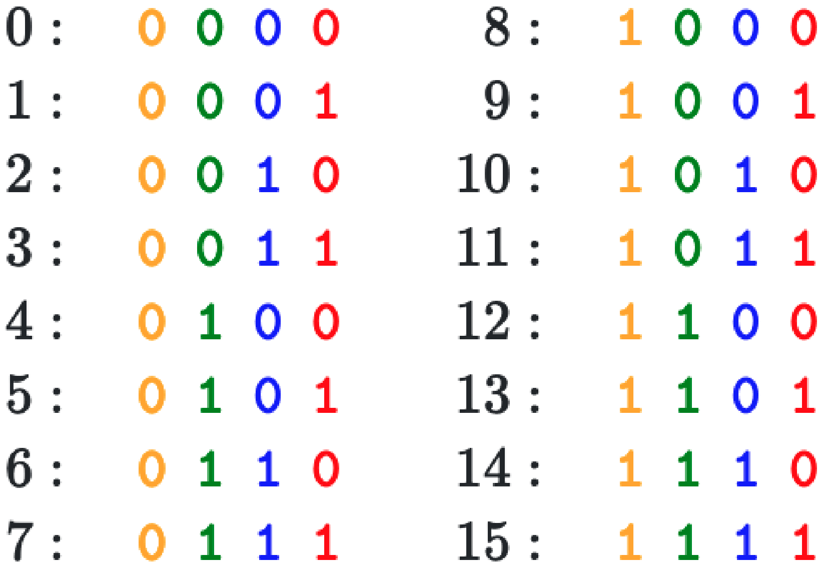




Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

Positional encoding

- Finite dimensional representation of the location or “position” of the units in a sequence
- Positions are just the indices in the sequence encoded by means of sine/cosine functions



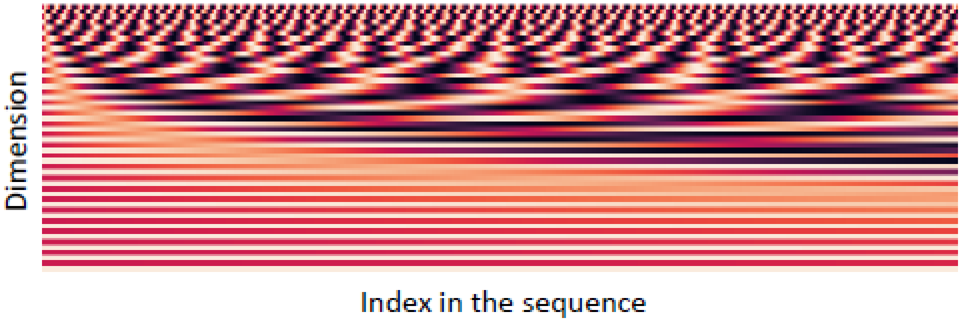
0 :	0	0	0	0	8 :	1	0	0	0
1 :	0	0	0	1	9 :	1	0	0	1
2 :	0	0	1	0	10 :	1	0	1	0
3 :	0	0	1	1	11 :	1	0	1	1
4 :	0	1	0	0	12 :	1	1	0	0
5 :	0	1	0	1	13 :	1	1	0	1
6 :	0	1	1	0	14 :	1	1	1	0
7 :	0	1	1	1	15 :	1	1	1	1

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. NIPS 2017

Positional encoding

- The absolute position is not important
  - Periods restart when the sequence is too long
- Relative positions really matter!

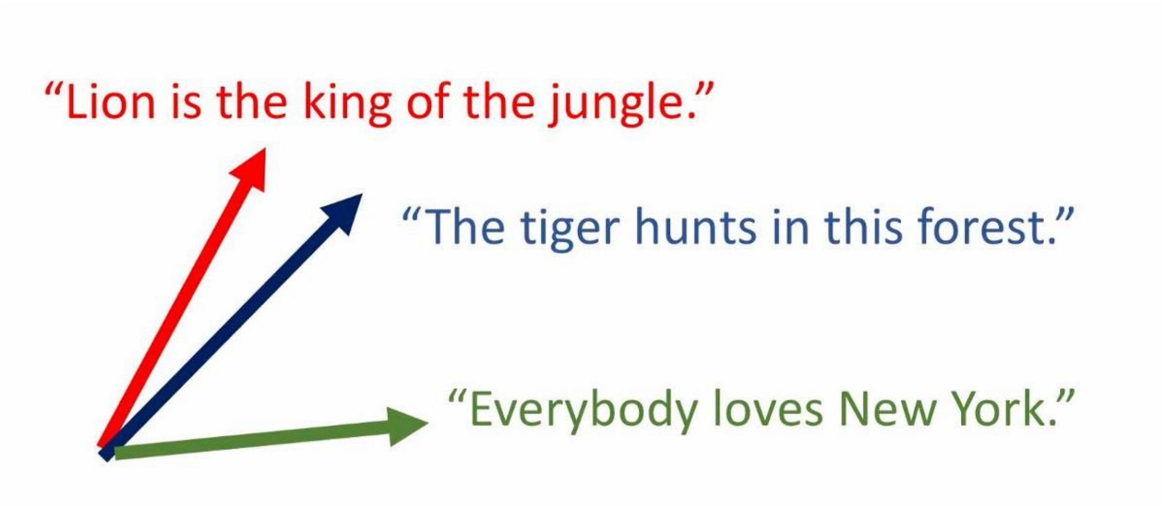
$$p_i = \begin{pmatrix} \sin(i/10000^{2*1/d}) \\ \cos(i/10000^{2*1/d}) \\ \vdots \\ \sin(i/10000^{2*\frac{d}{2}/d}) \\ \cos(i/10000^{2*\frac{d}{2}/d}) \end{pmatrix}$$



Natural Language Processing with Deep Learning. CS224N/Ling284 John Hewitt Lecture 9: Self-Attention and Transformers

Use transformers for sentence encoding

- Map variable-length sequences to fixed-size vectors
- Build a semantically sensitive, contextualized sentence representation


"Lion is the king of the jungle."
"The tiger hunts in this forest."
"Everybody loves New York."

GPT vs. BERT

- A transformer uses the encoder stack to model the input and uses the decoder stack to model the output
  - using input information from encoder side
- What if you are interested in training a language model for the input for some other tasks?
  - then we do not need the decoder of the transformer
  - Use **BERT encoder** (or similar)

Jay Alammar, The illustrated GPT-2, https://jalammar.github.io/illustrated-gpt2/ (latest access: April 2021)

GPT vs. BERT

- A transformer uses the encoder stack to model the input and uses the decoder stack to model output
  - using input information from encoder side
- What if we just want to model the “next word”?
  - Forward LM
  - Get rid of the encoder side of a transformer
  - output “next word” one by one
  - Use the **GPT decoder**

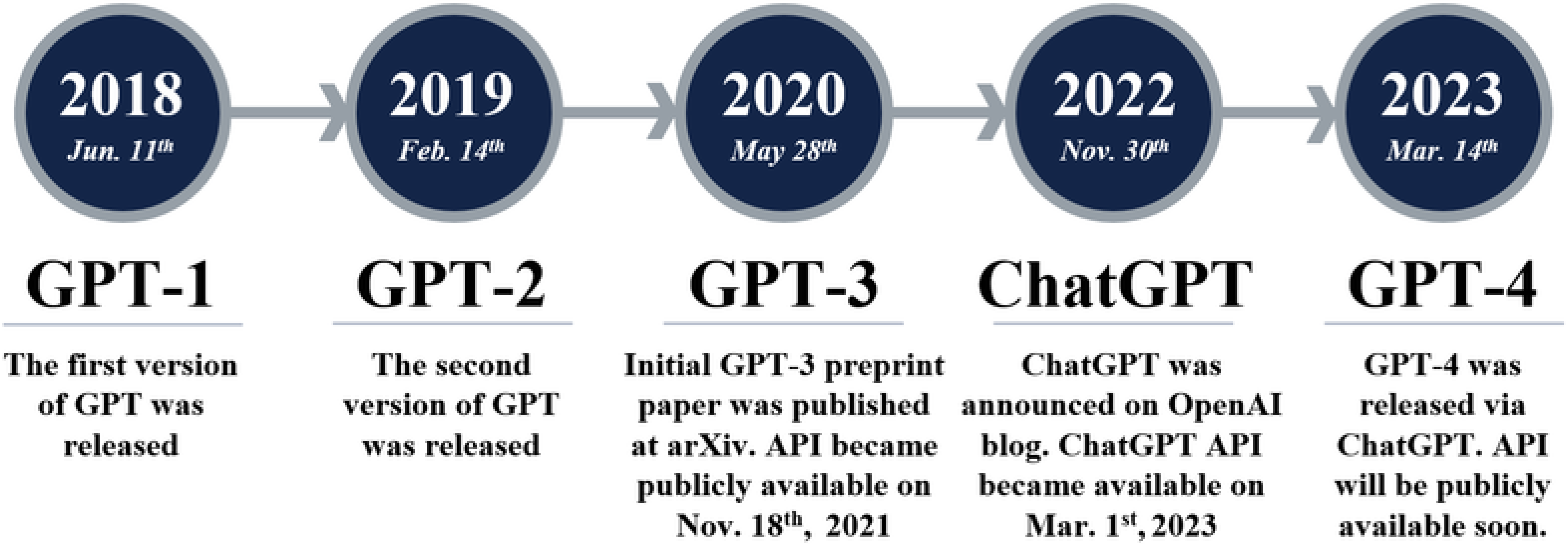
Jay Alammar, The illustrated GPT-2, https://jalammar.github.io/illustrated-gpt2/ (latest access: April 2021)

GPT

- Key idea
  - Use transformers for sentence decoding
  - Unsupervised learning approach to train a language model
- Key properties
  - Masked self-attention
  - Byte Pair Encoding
- Common applications
  - Machine Translation
  - Abstractive Summarization
- Github projects
  - https://github.com/openai/gpt-2
  - https://github.com/openai/gpt-3

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

GPT history



2018
Jun. 11th

GPT-1

The first version of GPT was released

2019
Feb. 14th

GPT-2

The second version of GPT was released

2020
May 28th

GPT-3

Initial GPT-3 preprint paper was published at arXiv. API became publicly available on Nov. 18th, 2021

2022
Nov. 30th

ChatGPT

ChatGPT was announced on OpenAI blog. ChatGPT API became available on Mar. 1st, 2023

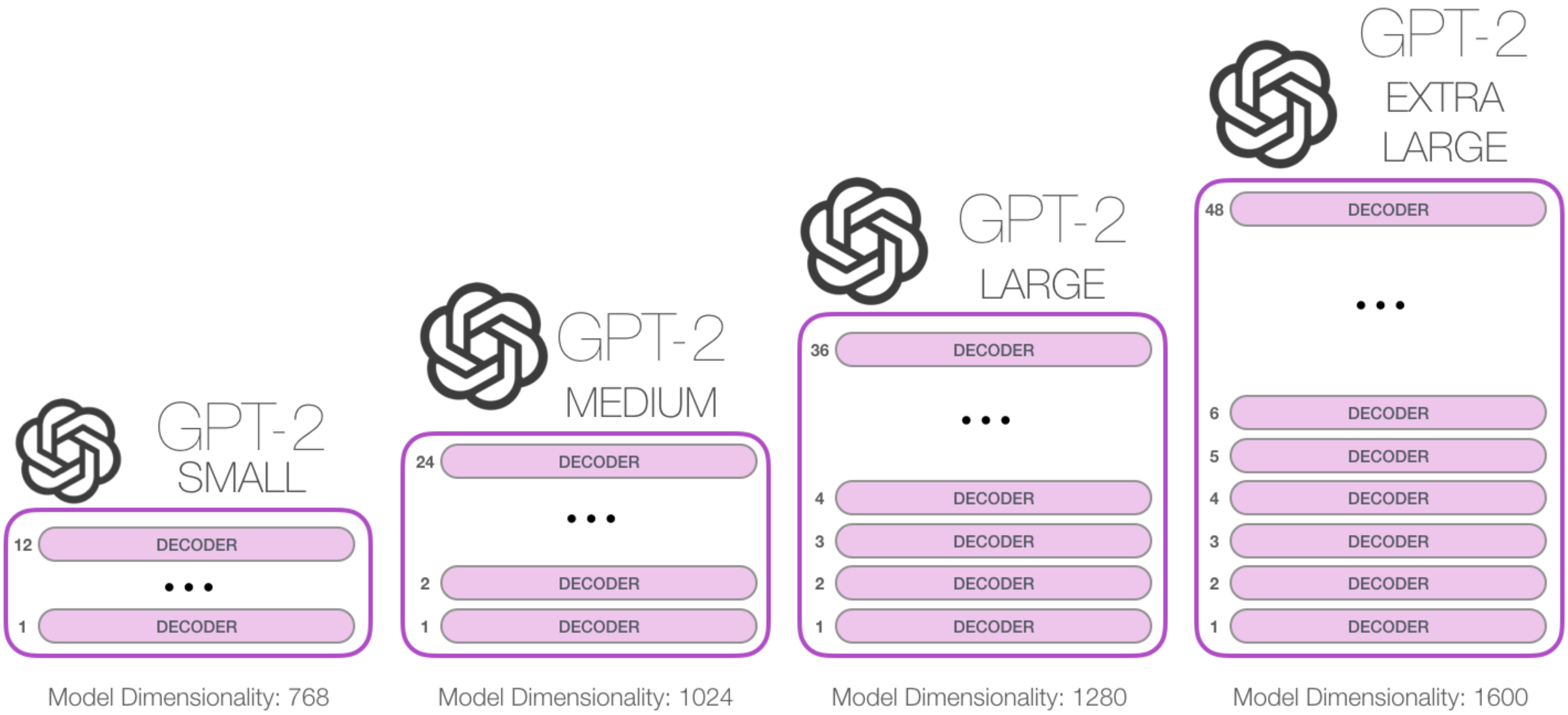
2023
Mar. 14th

GPT-4

GPT-4 was released via ChatGPT. API will be publicly available soon.

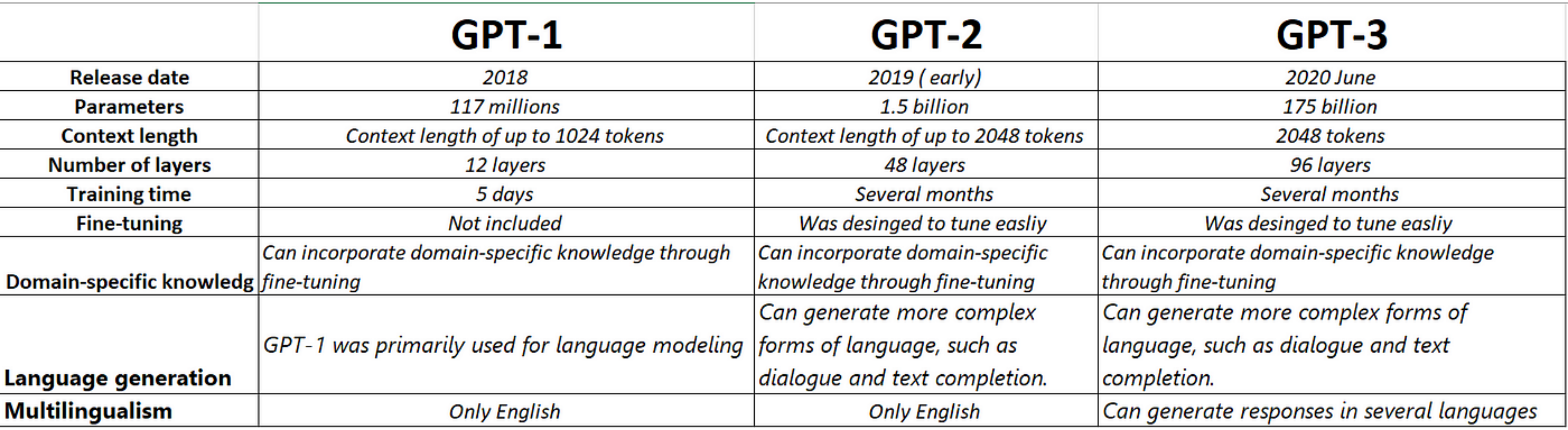
Hasin Rehana et al. 2023 Evaluation of GPT and BERT-based models on identifying protein-protein interactions in biomedical text

GPT-2 versions



Jay Alammar, The illustrated GPT-2, https://jalammar.github.io/illustrated-gpt2/ (latest access: April 2021)

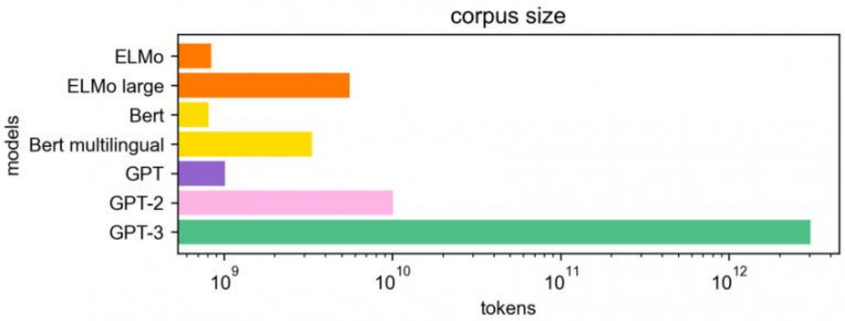
GPT model comparisons

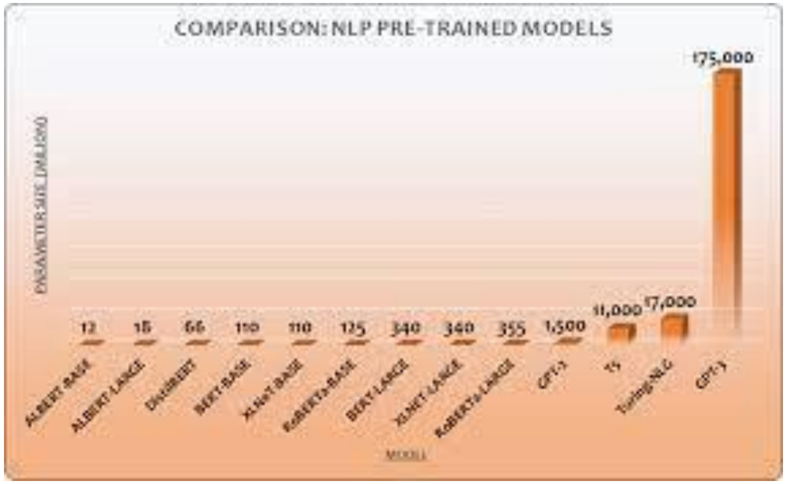


	GPT-1	GPT-2	GPT-3
Release date	*2018*	*2019 (early)*	*2020 June*
Parameters	*117 millions*	*1.5 billion*	*175 billion*
Context length	*Context length of up to 1024 tokens*	*Context length of up to 2048 tokens*	*2048 tokens*
Number of layers	*12 layers*	*48 layers*	*96 layers*
Training time	*5 days*	*Several months*	*Several months*
Fine-tuning	*Not included*	*Was desinged to tune easliy*	*Was desinged to tune easliy*
Domain-specific knowledg	*Can incorporate domain-specific knowledge through fine-tuning*	*Can incorporate domain-specific knowledge through fine-tuning*	*Can incorporate domain-specific knowledge through fine-tuning*
Language generation	*GPT-1 was primarily used for language modeling*	*Can generate more complex forms of language, such as dialogue and text completion.*	*Can generate more complex forms of language, such as dialogue and text completion.*
Multilingualism	*Only English*	*Only English*	*Can generate responses in several languages*

Hasin Rehana et al. 2023 Evaluation of GPT and BERT-based models on identifying protein-protein interactions in biomedical text

GPT-3





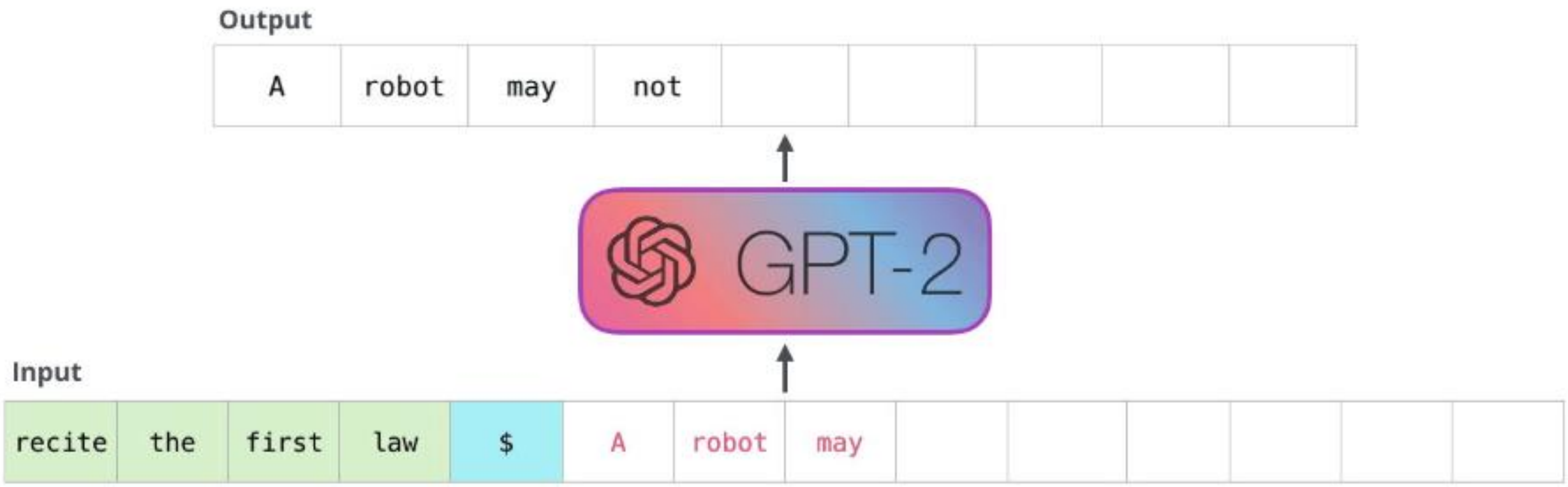
https://medium.com/@stuart.richards.54/blame-the-robot-consumer-personalization-with-gpt-3-part-2-817b35f37322
https://hellofuture.orange.com/en/the-gpt-3-language-model-revolution-or-evolution/ (latest access: April 2021)

GPT-4

- Newer and improved version of GPT-3.5
    - 10 times more advanced
    - better understanding of context and differentiate nuances, leading to more precise and logical answers.
- Higher memory limit
    - It can process up to 25,000 words.
        - Support longer conversations
        - Provide lengthier responses
        - search through and analyze large volumes of text in documents.
- Multimodal feature
    - combining both language and vision models to understand images.
        - Aid to visually impaired, enhancing accessibility, moderation, and more.

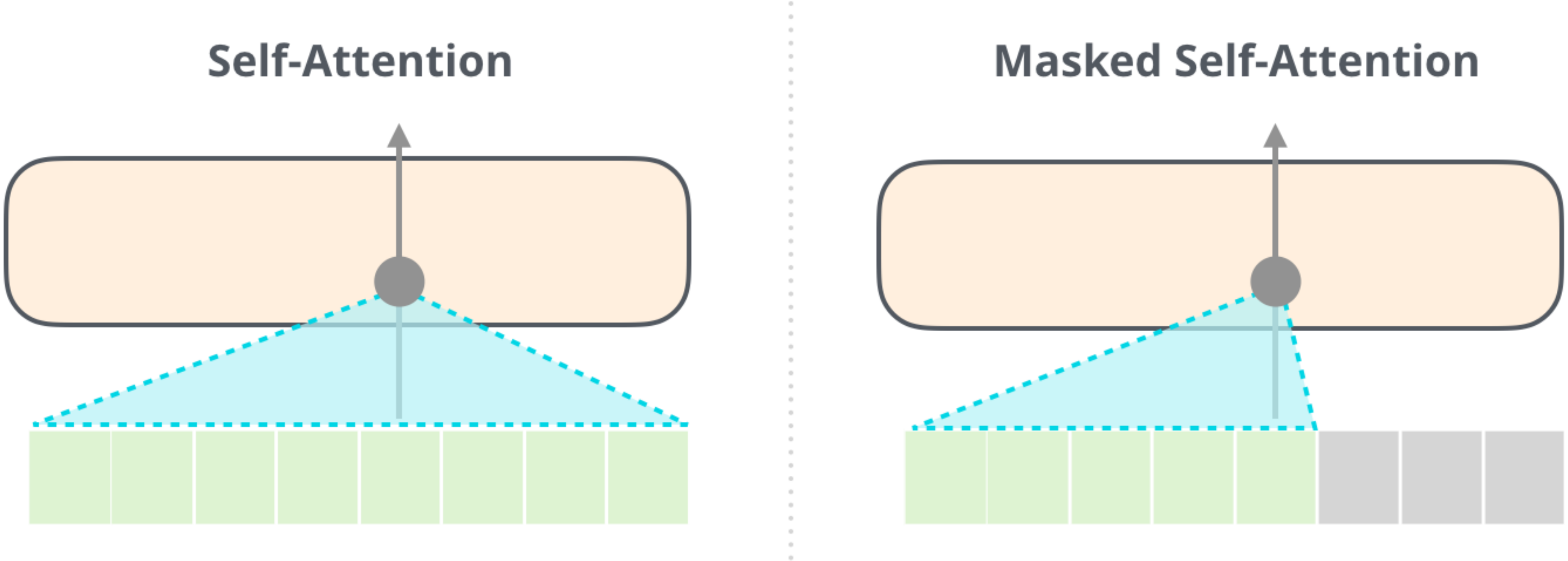
https://www.idenfy.com/blog/chat-gpt-4/

GPT fundamentals



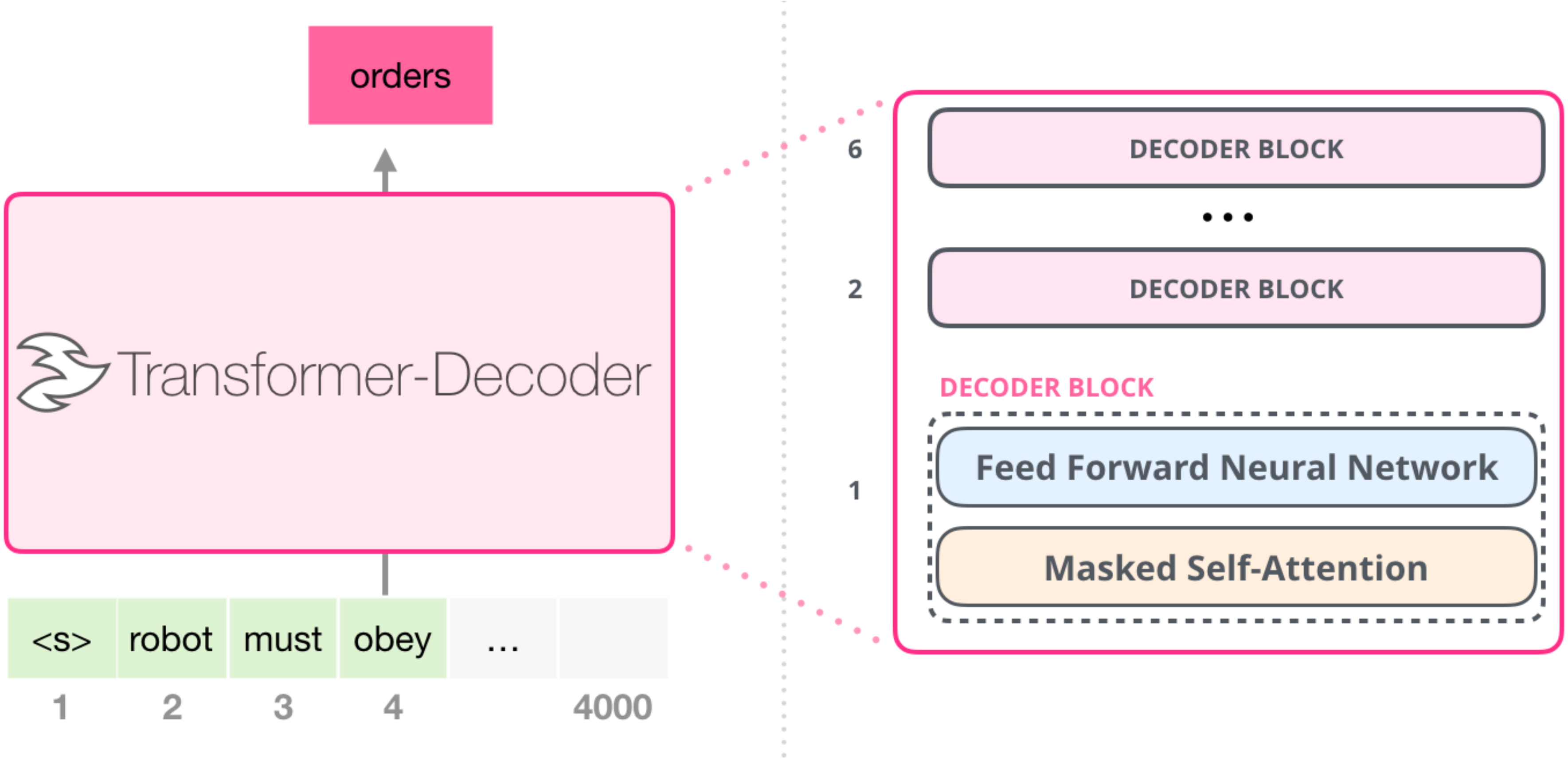
Jay Alammar, The illustrated GPT-2, https://jalammar.github.io/illustrated-gpt2/ (latest access: April 2021)

Masked self-attention



GPT fundamentals

- Reuse previous computations
  - At every step look for the result of <q,k,v> relative to the new output only
  - Linear complexity

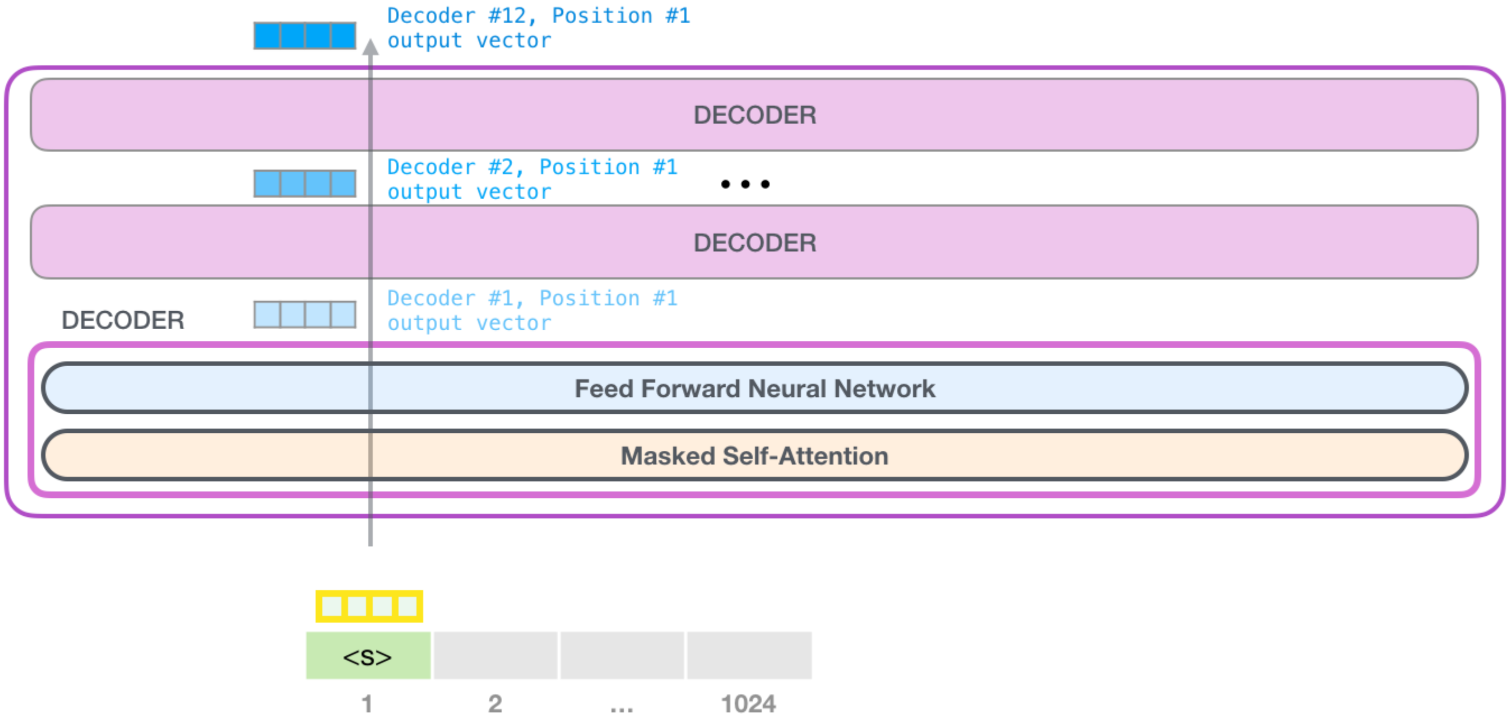


GPT fundamentals

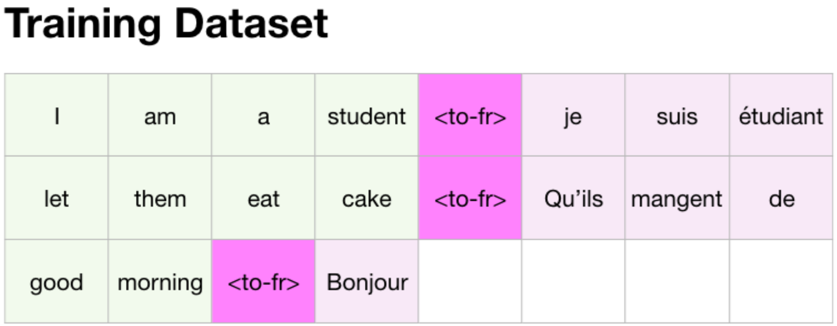
- Byte Pair Encoding
  - Break the words into pieces like –er, -est
  - Embed frequent fragments of words
- Example
  - Given *old, older, oldest*
  - Infer *smart, smarter, smartest*

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

GPT-2 decoder architecture

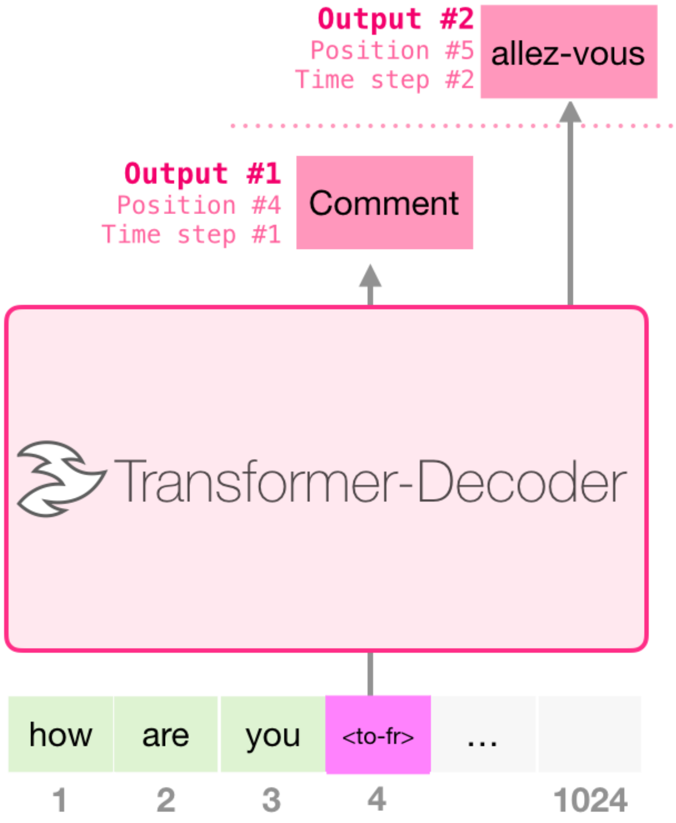


GPT-2 for Machine Translation



Training Dataset

I	am	a	student	<to-fr>	je	suis	étudiant
let	them	eat	cake	<to-fr>	Qu'ils	mangent	de
good	morning	<to-fr>	Bonjour				



Additional reading on GPT-2 (ADDITIONAL MATERIAL)

- Language Models are Unsupervised Multitask Learners. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever
- Download and read the paper: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf

Additional reading on GPT-3 (ADDITIONAL MATERIAL)

- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, Dario Amodei. Language Models are Few-Shot Learners. CoRR abs/2005.14165 (2020)
- Download and read the paper: https://arxiv.org/pdf/2005.14165.pdf

Bidirectional Encoder Representation from Transformers

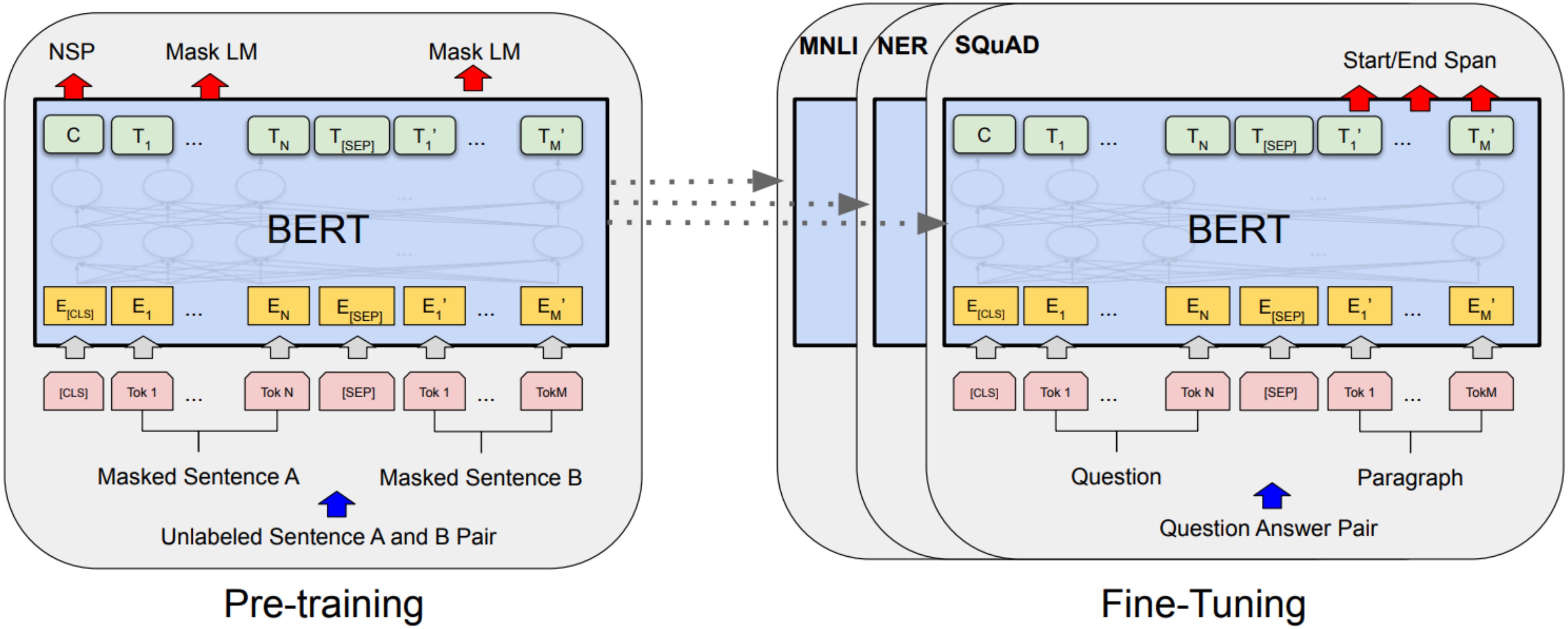
- Key idea
  - Use Transformers for sentence encoding
  - Unsupervised pretraining of bidirectional language models
    - Joint conditioning on both left and right context in all layers

- Key properties
  - State of the art for most NLP tasks
  - Fast to train
  - Easy to tune for specific tasks

- GitHub project (by Google AI Language)
  - https://github.com/google-research/bert

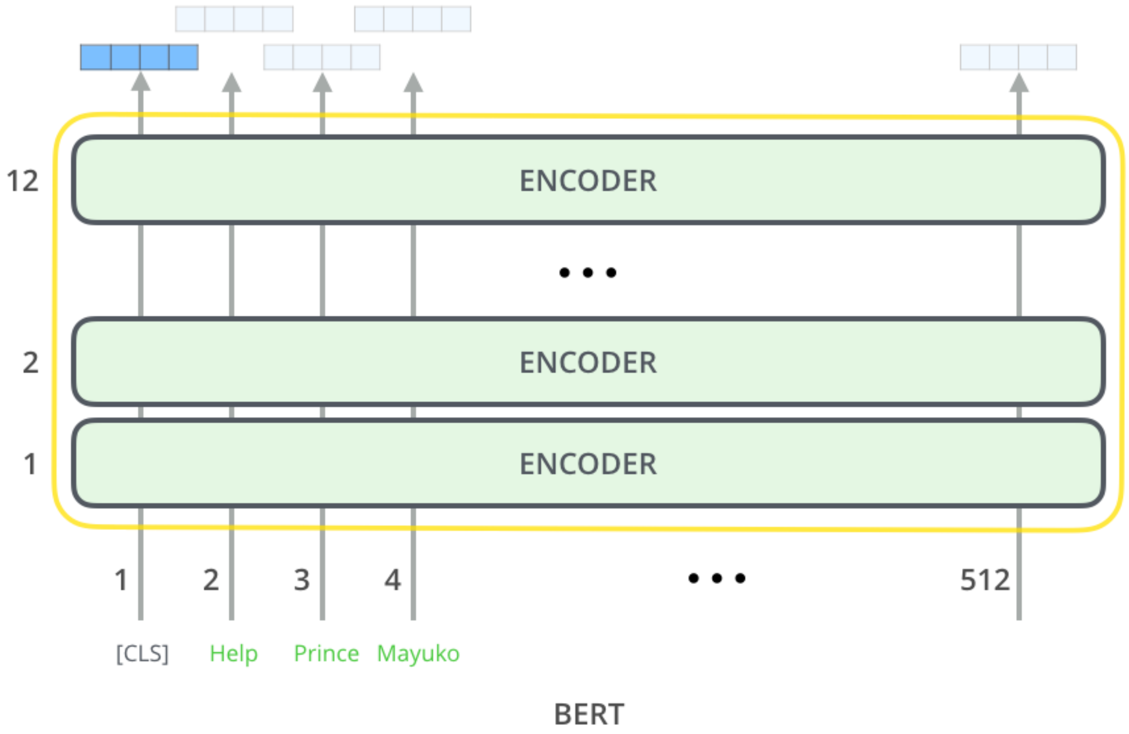
BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Bidirectional Encoder Representation from Transformers



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

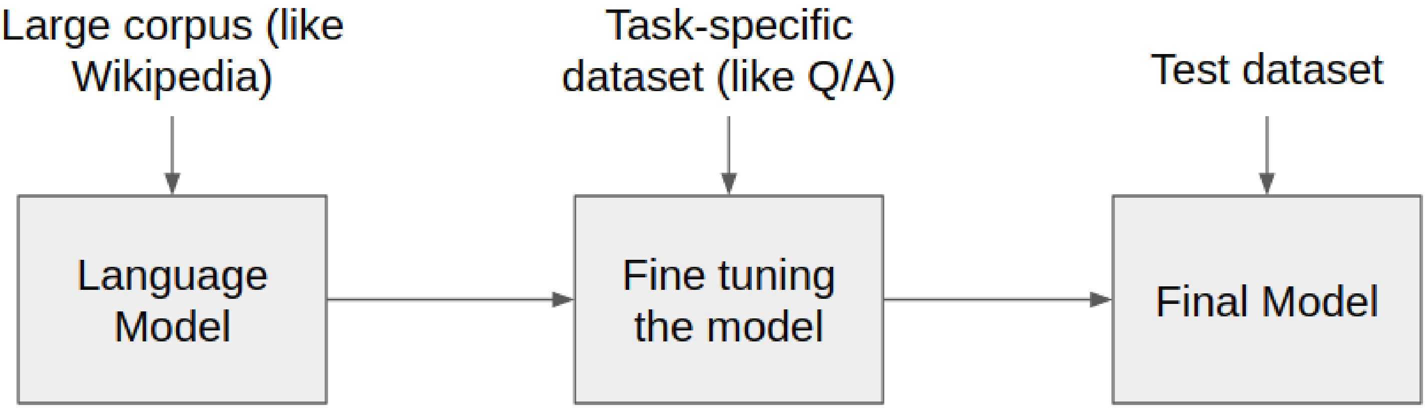
BERT encoder stack



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Bidirectional Encoder Representation from Transformers

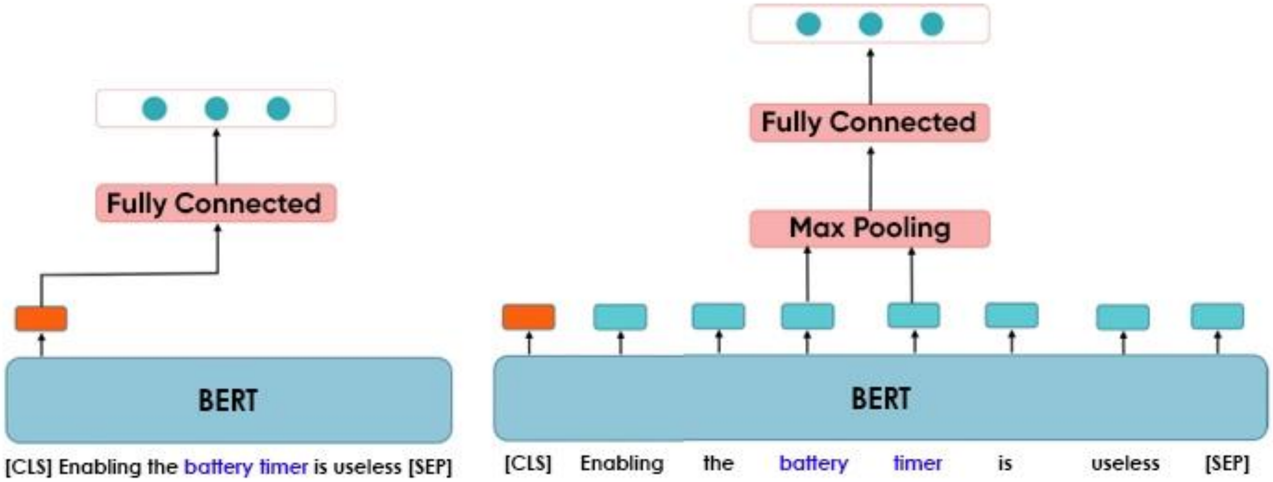
- How many examples you need?
  - Pre-training: 2.5 billions sentences
  - Fine-tuning: ~1K sentences (depending on the task)


Large corpus (like Wikipedia)
Task-specific dataset (like Q/A)
Test dataset
Language Model
Fine tuning the model
Final Model

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Bidirectional Encoder Representation from Transformers

- Vector representations of sentences
  - Obtained using the representation of the first “special” token [CLS]
  - Word-level vectors can be aggregated by using a pooling layer



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

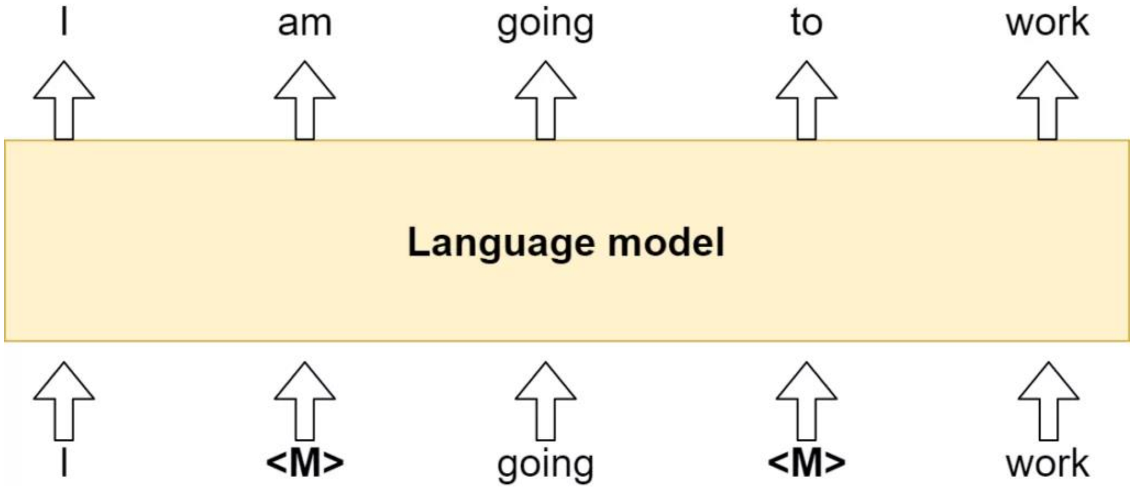
Model pretraining strategies

- Masked LM
  - predict the missing words
- Next Sentence Prediction
  - Predict the semantic relationships among sentences
    - IsNextSentence
    - NotNextSentence
- Joint use of the aforesaid strategies
  - Minimization of the combined loss function

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

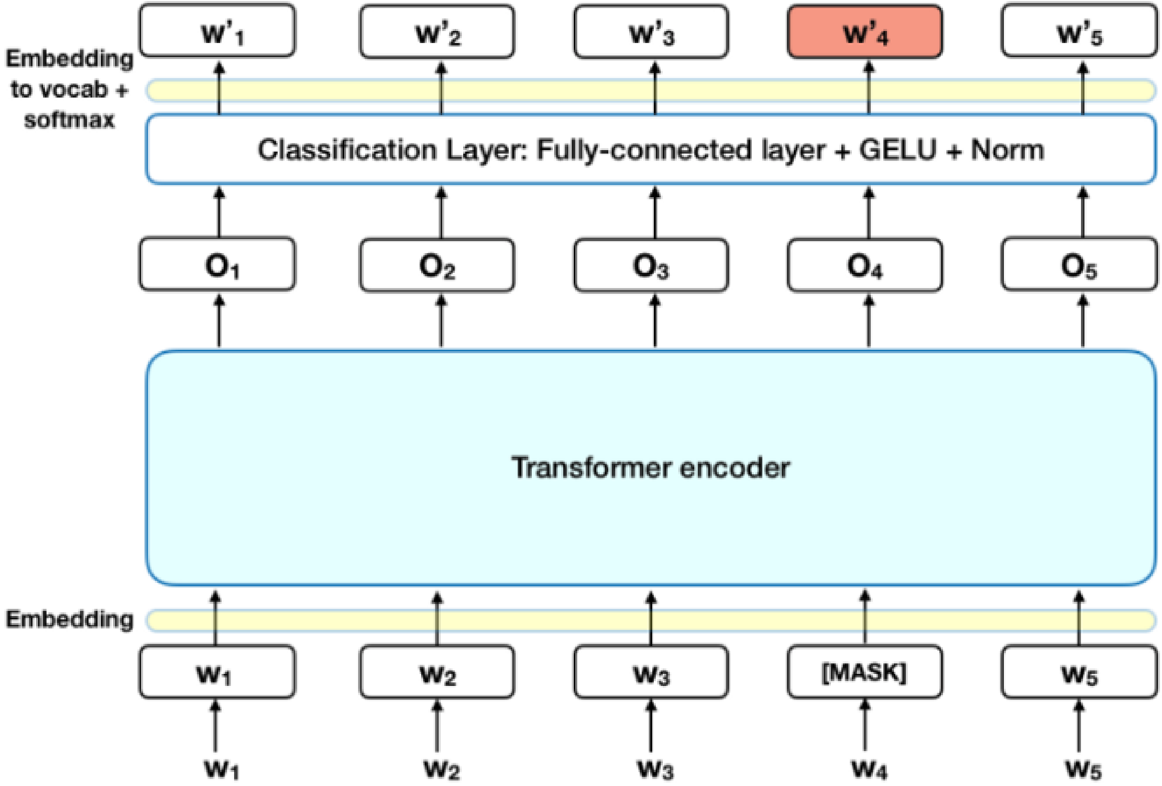
Masked LM

- Self-supervised training
  - 15% of the words in each sequence are replaced with a [MASK] token
- The model attempts to predict the original value of the masked words, based on the context provided by the other, non-masked, words in the sequence



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

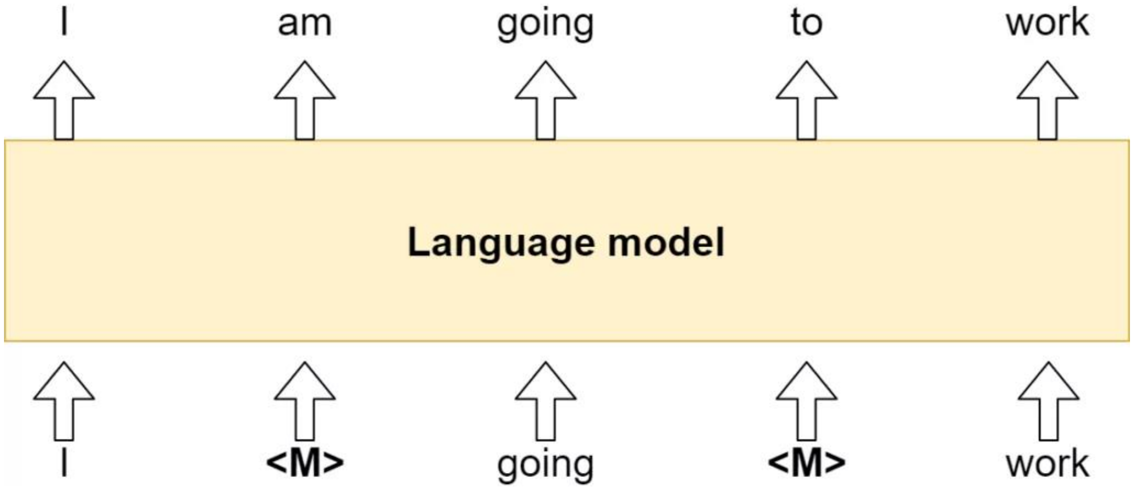
Masked LM


W'1
W'2
W'3
W'4
W'5
Embedding to vocab + softmax
Classification Layer: Fully-connected layer + GELU + Norm
O1
O2
O3
O4
O5
Transformer encoder
Embedding
W1
W2
W3
[MASK]
W5
W1
W2
W3
W4
W5

BERT Explained: State of the art language model for NLP. Rani Horev. Nov 2018.

Masked LM

- A classification layer is added on top of the encoder output
- The output vectors are multiplied by the embedding matrix, transforming them into the vocabulary dimension
- The probability of each word in the vocabulary is computed using softmax

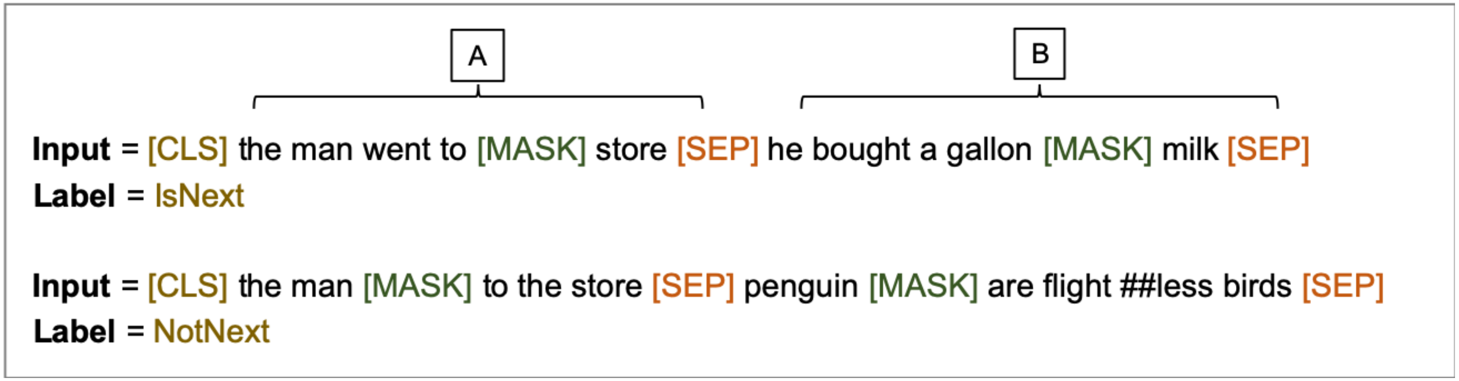


BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Next Sentence Prediction

- Input
  - Sentence pairs

- Task
  - Predict if the second sentence in the pair is the subsequent sentence in the original document



Input = [CLS] the man went to [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]
Label = IsNext

Input = [CLS] the man [MASK] to the store [SEP] penguin [MASK] are flight ##less birds [SEP]
Label = NotNext

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

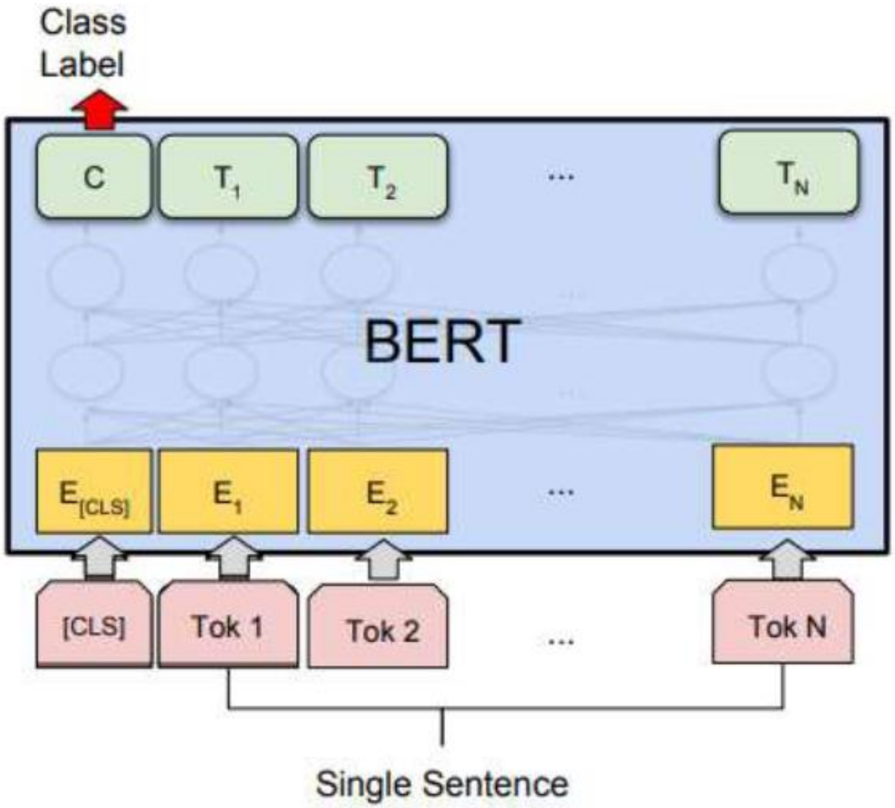
Next Sentence Prediction

- Self-supervised training
  - 50% of the inputs are sentence pairs in which the second sentence is the subsequent sentence in the original document
  - In the other 50% a random sentence from the corpus is chosen as the second sentence
- Assumption
  - the random sentence will be disconnected from the first sentence

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Model fine-tuning

- Task-dependent



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

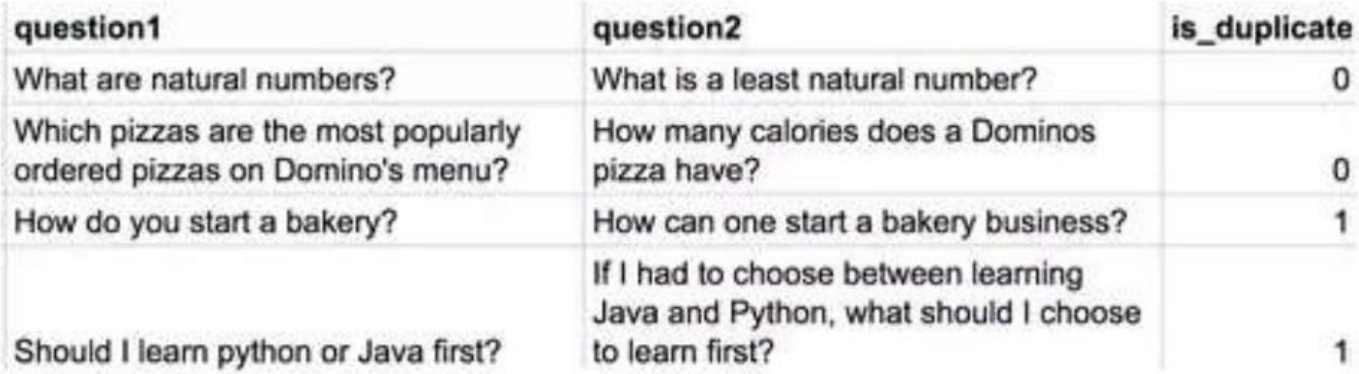
Model fine-tuning

- Tasks
  - Sentence pair classification
  - Single sentence classification/regression
  - Question Answering
  - Sentence tagging
  - …

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Model fine-tuning

- Sentence pair classification
  - Determine whether two text snippets are semantically similar with each other



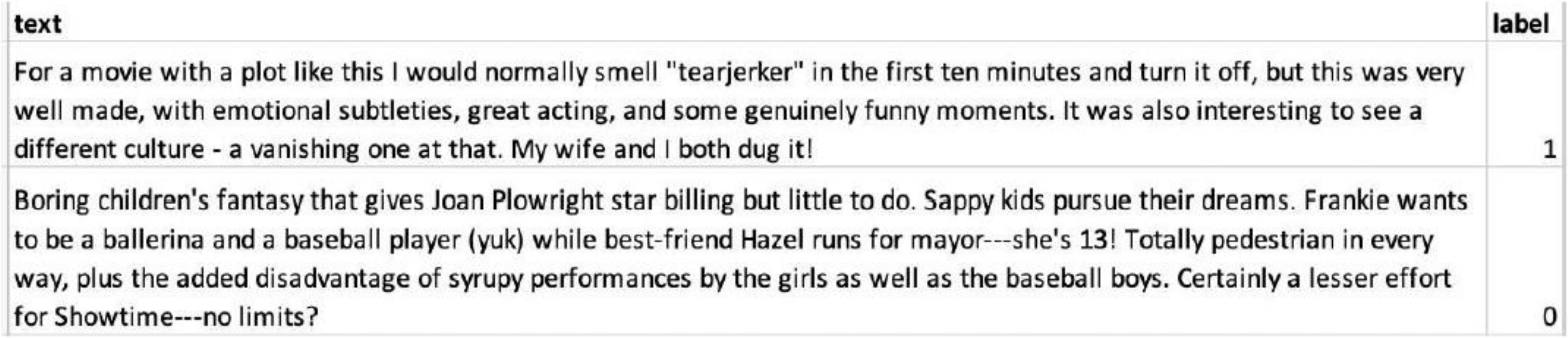
question1	question2	is_duplicate
What are natural numbers?	What is a least natural number?	0
Which pizzas are the most popularly ordered pizzas on Domino's menu?	How many calories does a Dominos pizza have?	0
How do you start a bakery?	How can one start a bakery business?	1
Should I learn python or Java first?	If I had to choose between learning Java and Python, what should I choose to learn first?	1

https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Model fine-tuning

- Single sentence classification
  - Assign a label to a given sentence
    - E.g., for sentiment analysis



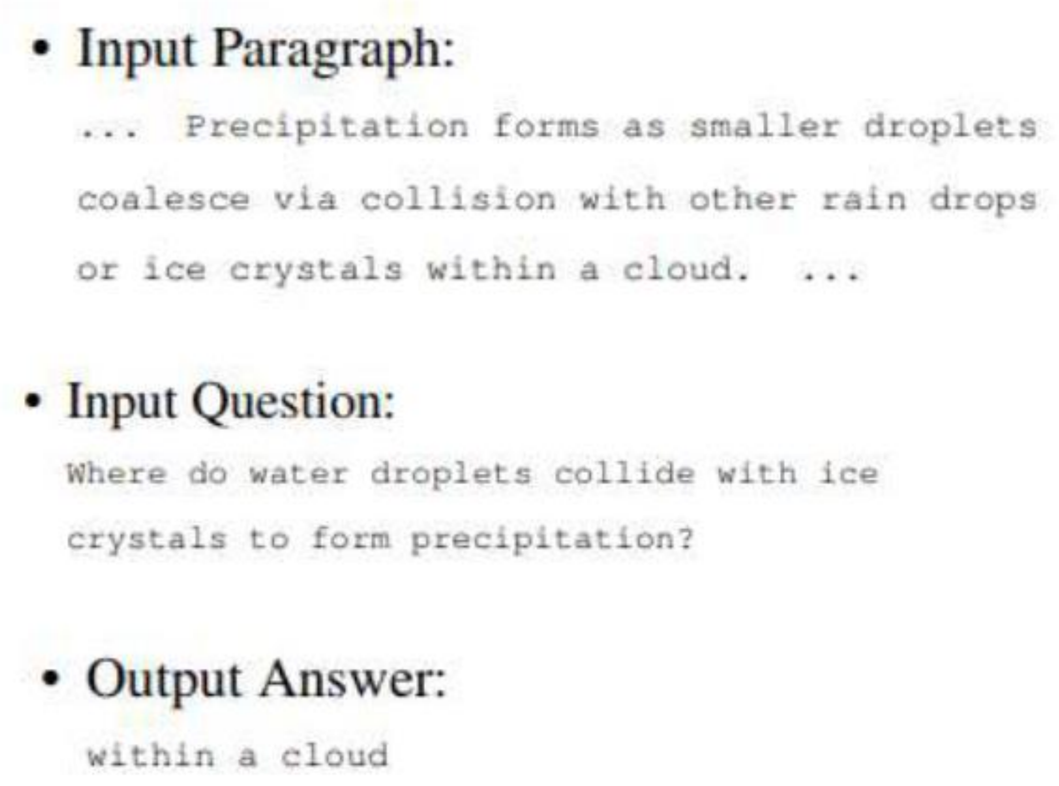
text	label
For a movie with a plot like this I would normally smell "tearjerker" in the first ten minutes and turn it off, but this was very well made, with emotional subtleties, great acting, and some genuinely funny moments. It was also interesting to see a different culture - a vanishing one at that. My wife and I both dug it!	1
Boring children's fantasy that gives Joan Plowright star billing but little to do. Sappy kids pursue their dreams. Frankie wants to be a ballerina and a baseball player (yuk) while best-friend Hazel runs for mayor---she's 13! Totally pedestrian in every way, plus the added disadvantage of syrupy performances by the girls as well as the baseball boys. Certainly a lesser effort for Showtime---no limits?	0

Large Movie Review Dataset

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Model fine-tuning

- Question answering
  - Answer to a specific question



- Input Paragraph:

  ... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. ...

- Input Question:

  Where do water droplets collide with ice crystals to form precipitation?

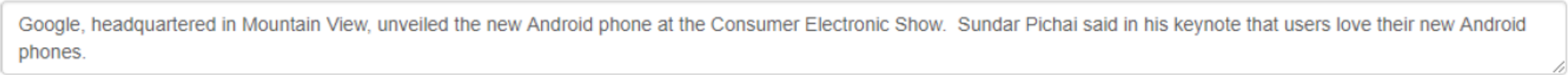
- Output Answer:

  within a cloud

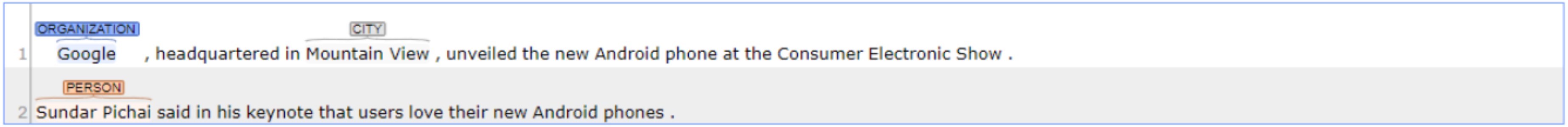
BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

Model fine-tuning

- Sentence tagging
  - Enrich sentences with metadata
    - E.g., Named Entity Recognition



Google, headquartered in Mountain View, unveiled the new Android phone at the Consumer Electronic Show. Sundar Pichai said in his keynote that users love their new Android phones.



Google , headquartered in Mountain View , unveiled the new Android phone at the Consumer Electronic Show .

Sundar Pichai said in his keynote that users love their new Android phones .

http://corenlp.run/

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.

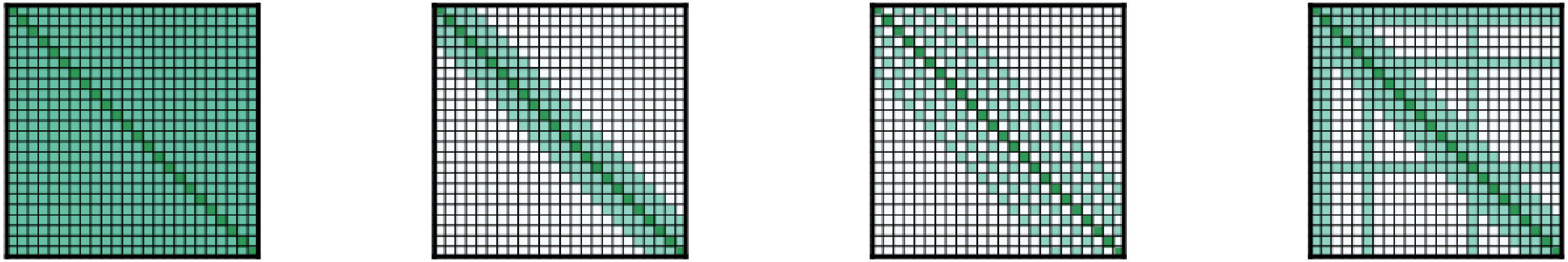
Long-Document Transformer

- BERT encodes sequences including at most 512 tokens using a full self-attention mechanism
    - Quadratic complexity with the sequence length
- LongFormer allows encoding longer sequences (up to 32K)
    - Its attention mechanism can act as a drop-in replacement for the self-attention mechanism
    - Linear complexity with the sequence length
- LED is the encoder-decoder architecture based on LongFormer as encoder
    - https://huggingface.co/docs/transformers/model_doc/led

Iz Beltagy, Matthew E. Peters, Arman Cohan: Longformer: The Long-Document Transformer. CoRR abs/2004.05150 (2020)

Long-Document Transformer

- Key idea
  - sparsify the full self-attention matrix according to an “attention pattern” specifying pairs of input locations attending to one another



(a) Full n^2 attention

(b) Sliding window attention

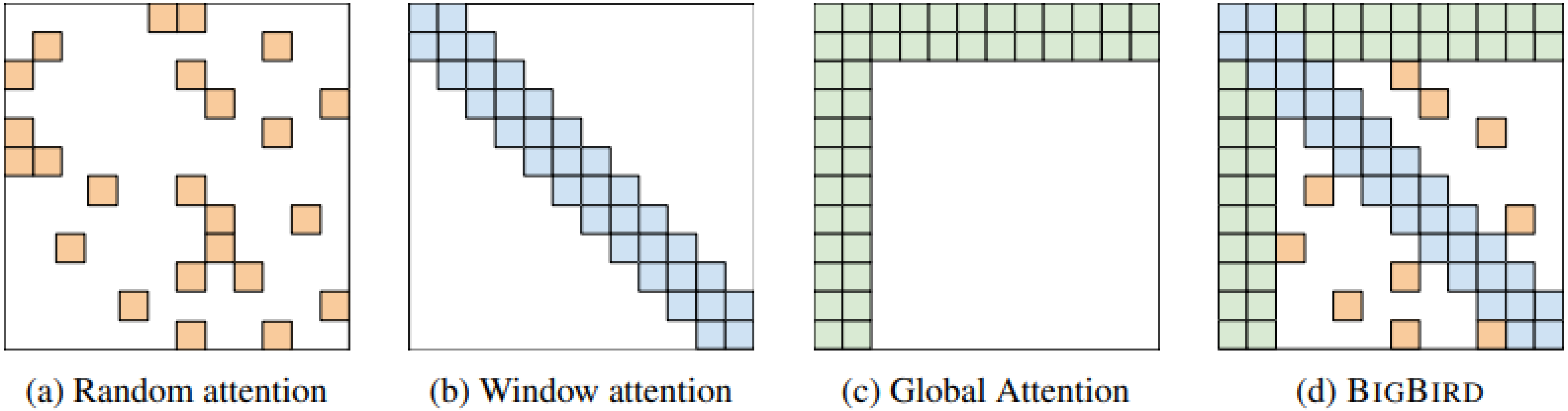
(c) Dilated sliding window

(d) Global+sliding window

Iz Beltagy, Matthew E. Peters, Arman Cohan: Longformer: The Long-Document Transformer. CoRR abs/2004.05150 (2020)

BigBird

- Transformer-based encoder that extends LongFormer using a sparse attention mechanism
  - Linear with the number of sequence tokens
- It considers
  - A set of g global tokens attending on all parts of the sequence.
  - All tokens attending to a set of w local neighboring tokens.
  - All tokens attending to a set of r random tokens.



(a) Random attention (b) Window attention (c) Global Attention (d) BIGBIRD

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, Amr Ahmed: Big Bird: Transformers for Longer Sequences. NeurIPS 2020

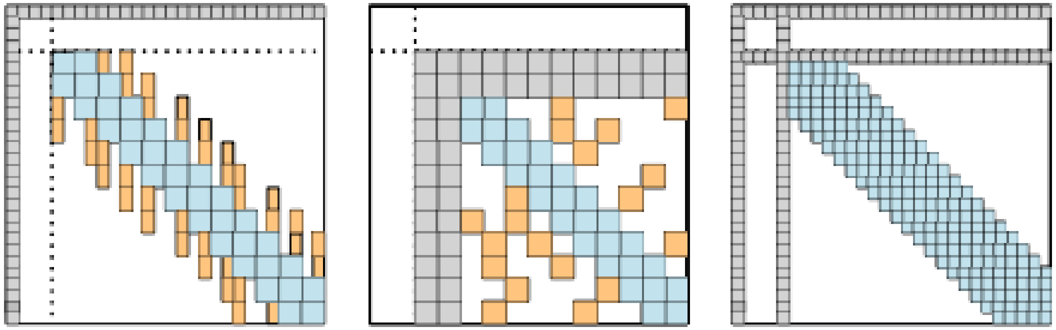
LSG

- Efficient approach to transformer-based encoding
- Key ideas
  - Locally, a token needs to capture low level information
    - dense attention is preferred.
  - As the context grows, higher level information is sufficient
    - A limited number of tokens need to be attended to by following specific selection and computation rules

Charles Condevaux, Sébastien Harispe: LSG Attention: Extrapolation of Pretrained Transformers to Long Sequences. PAKDD (1) 2023: 443-454

LSG

- **Local attention**: capture local context using a fixed length sliding window
  - Adopted also by LongFormer together with global attention
- **Sparse connections**: capture extended context by selecting an additional set of tokens following a set of rules
  - Not using random attention as in BigBird
- **Global attention**: it attends to every tokens across the sequence and all tokens attend to them
  - Like BERT and LongFormer



LSG

Big Bird

Longformer

Charles Condevaux, Sébastien Harispe: LSG Attention: Extrapolation of Pretrained Transformers to Long Sequences. PAKDD (1) 2023: 443-454

Additional reading on BERT

- BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova. Proceedings of NAACL-HLT 2019.
- Download and read the paper: https://arxiv.org/pdf/1810.04805.pdf

Additional reading on LongFormer

- Iz Beltagy, Matthew E. Peters, Arman Cohan: Longformer: The Long-Document Transformer. CoRR abs/2004.05150 (2020)
- Download and read the paper: https://arxiv.org/abs/2004.05150

Additional reading on BigBird

- Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, Amr Ahmed: Big Bird: Transformers for Longer Sequences. NeurIPS 2020
- Download and read the paper: https://arxiv.org/pdf/2007.14062.pdf

Additional reading on LSG

- Charles Condevaux, Sébastien Harispe: LSG Attention: Extrapolation of Pretrained Transformers to Long Sequences. PAKDD (1) 2023: 443-454
- Download and read the paper: https://arxiv.org/pdf/2210.15497.pdf

Acknowledgements and copyright license

- Copyright licence
  - Attribution + Noncommercial + NoDerivatives
- Acknowledgements
  - I would like to thank Dr. Moreno La Quatra, who collaborated to the writing and revision of the teaching content
- Affiliation
  - The author and his staff are currently members of the Database and Data Mining Group at Dipartimento di Automatica e Informatica (Politecnico di Torino) and of the SmartData interdepartmental centre
    - https://dbdmg.polito.it
    - https://smartdata.polito.it

Thank you!