

NLP Recent History

Text Features

- Natural Language Processing = Machine Learning with Text
- Step 1 – Identify text-based features
- Step 2 – Label data for a task
- Step 3 – Train a classifier

Text Features – Old School

- Lexical Features
 - Part-of-speech tags (Verb/Noun/Adj)
 - Stems / Lemmas
 - Word Frequency
 - Psycholinguistic Features
 - Terminologies
 - Morphology (affixes)

Text Features – Old School

- Sentential Features
 - Sentence Structure
 - Parse Tree Features
 - Sentence Dependencies
 - POS ratios
 - Punctuation frequency
 - Averages of lexical features

Text Features – Word Embeddings

- Represent each word as a vector of numbers

Text Features – Word Embeddings

- Represent each word as a vector of numbers

The cat sat on the mat

Text Features – Word Embeddings

- Represent each word as a vector of numbers

the →

0.23
0.12
0.65
...
...

The cat sat on the mat

Text Features – Word Embeddings

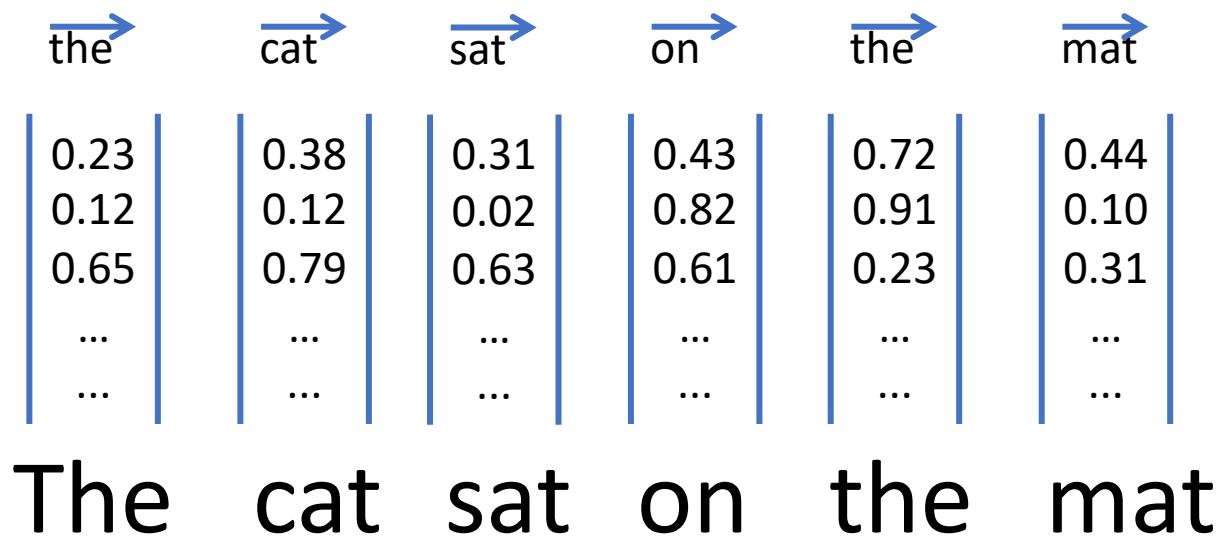
- Represent each word as a vector of numbers

the	cat
0.23	0.38
0.12	0.12
0.65	0.79
...	...
...	...

The cat sat on the mat

Text Features – Word Embeddings

- Represent each word as a vector of numbers



Text Features – Count-based Embeddings

- The cat sat on the mat
 - Dogs and cats are animals
 - Cats like milk
 - The dog chased the stick
 - Dogs like bones

Text Features – Count-based Embeddings

- The cat sat on the mat
- Dogs and cats are animals
- Cats like milk
- The dog chased the stick
- Dogs like bones

	cat	sat	mat	dog	chase	stick	animal	bone	milk
cat	3	1	1	1	0	0	1	0	1
sat	1	1	1	0	0	0	0	0	0
mat	1	1	1	0	0	0	0	0	0
dog	1	0	0	3	1	1	1	1	0
chase	0	0	0	1	1	1	0	0	0
stick	0	0	0	1	1	1	0	0	0
animal	1	0	0	1	0	0	1	0	0
bone	0	0	0	1	0	0	0	1	0
milk	1	0	0	1	0	0	0	0	1

Text Features – Count-based Embeddings

- The cat sat on the mat
- Dogs and cats are animals
- Cats like milk
- The dog chased the stick
- Dogs like bones

cat	sat	mat	dog	chase	stick	animal	bone	milk
3	1	1	1	0	0	1	0	1
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	0	0	3	1	1	1	1	0
0	0	0	1	1	1	0	0	0
0	0	0	1	1	1	0	0	0
1	0	0	1	0	0	1	0	0
0	0	0	1	0	0	0	1	0
1	0	0	1	0	0	0	0	1

Text Features – Count-based Embeddings

- The cat sat on the mat
 - Dogs and cats are animals
 - Cats like milk
 - Issue
 - Dimensionality = vocab size
 - Reduce with PCA..
- The dog chased the stick
 - Dogs like bones

cat	sat	mat	dog	chase	stick	animal	bone	milk
3	1	1	1	0	0	1	0	1
1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0
1	0	0	3	1	1	1	1	0
0	0	0	1	1	1	0	0	0
0	0	0	1	1	1	0	0	0
1	0	0	1	0	0	1	0	0
0	0	0	1	0	0	0	1	0
1	0	0	1	0	0	0	0	1

Text Features – Count-based Embeddings

- Which animal likes milk?

animal
1
0
0
1
0
0
1
0
0

+

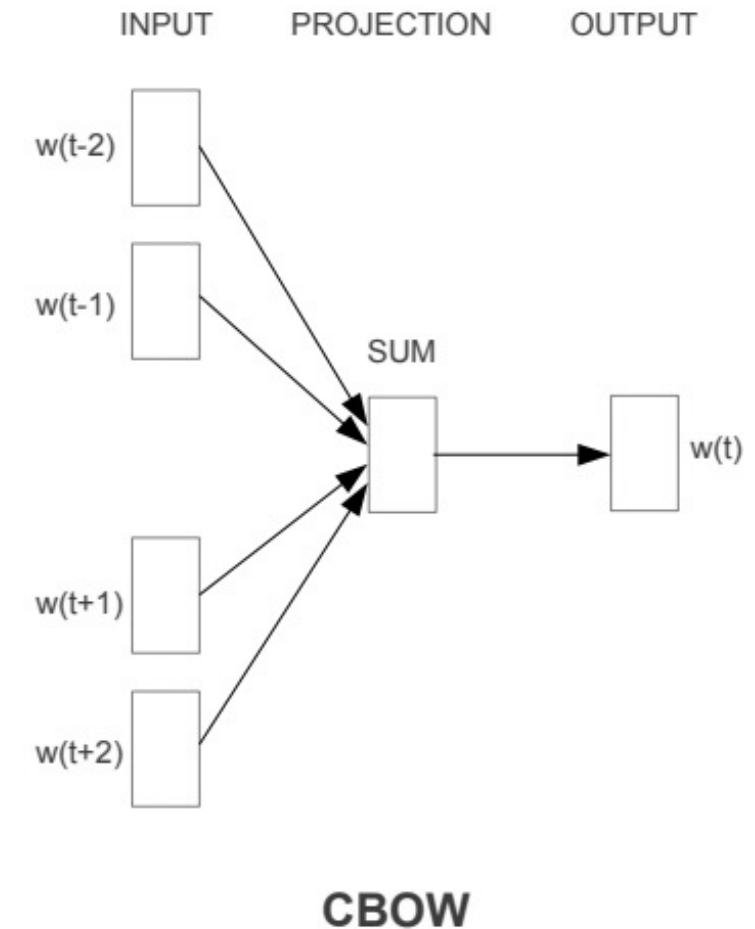
milk
1
0
0
0
0
0
0
0
1

?

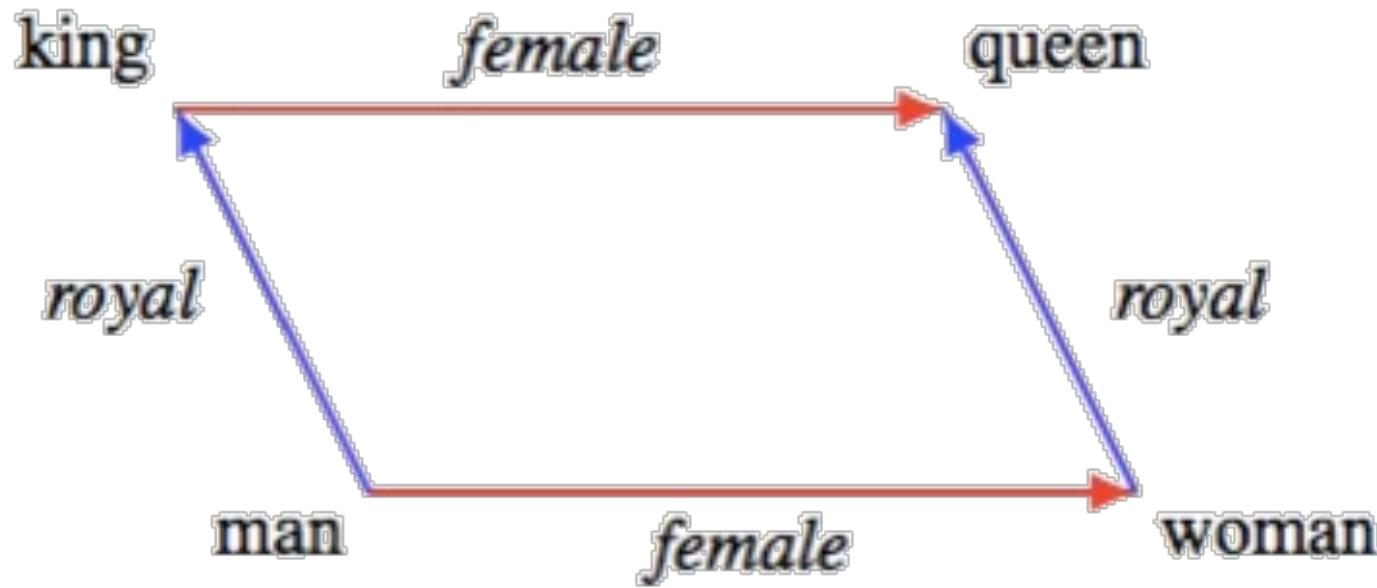
cat
3
1
1
1
0
0
1
0
1

Text Features – Learnt Embeddings (Word2Vec)

- Learn a set of word vectors that can be used to predict surrounding words
- Initialise random vectors and train neural network, updating vector weights with network
- End result is a set of vectors that perform like count-based, but without counting

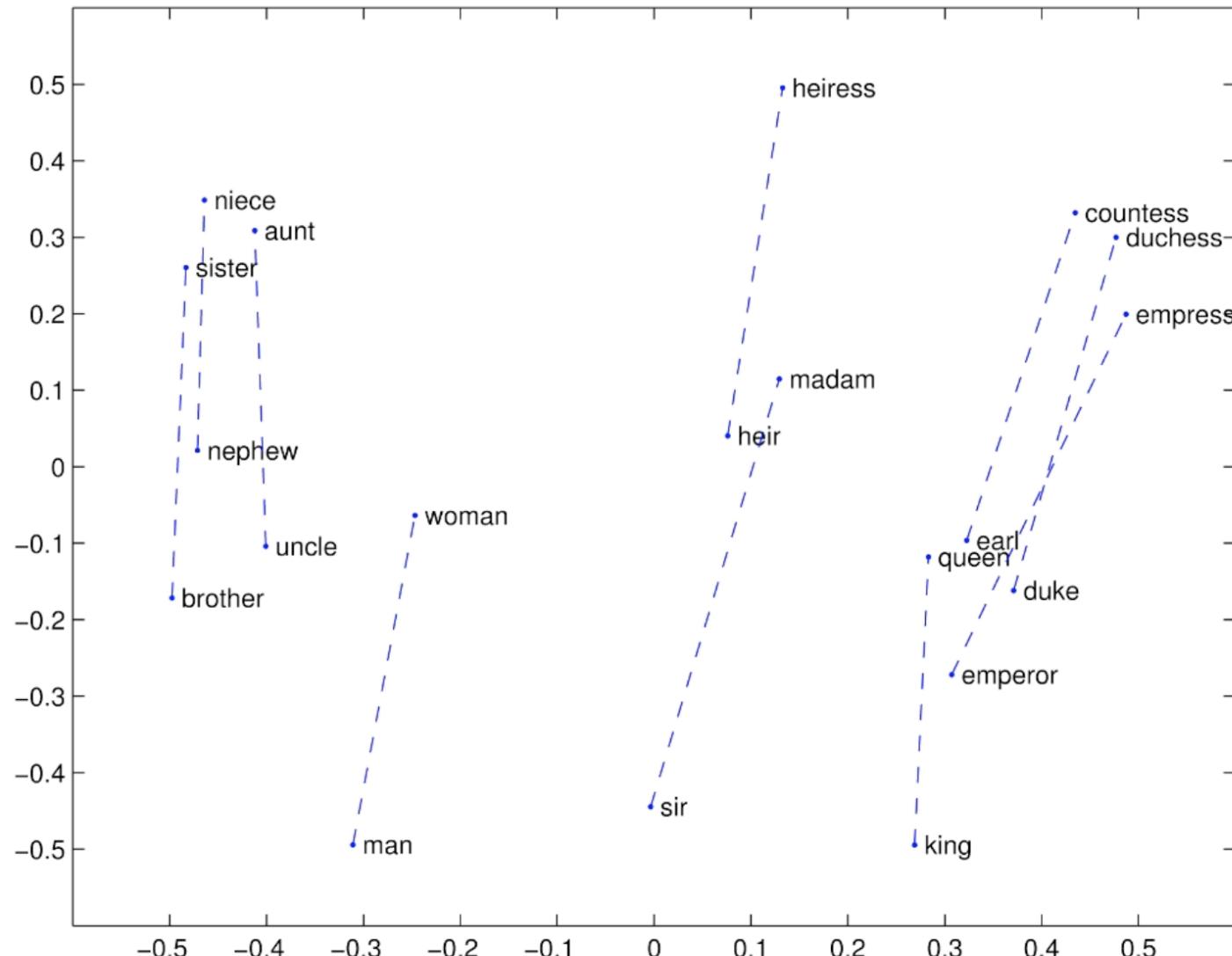


Text Features – Learnt Embeddings (Word2Vec)

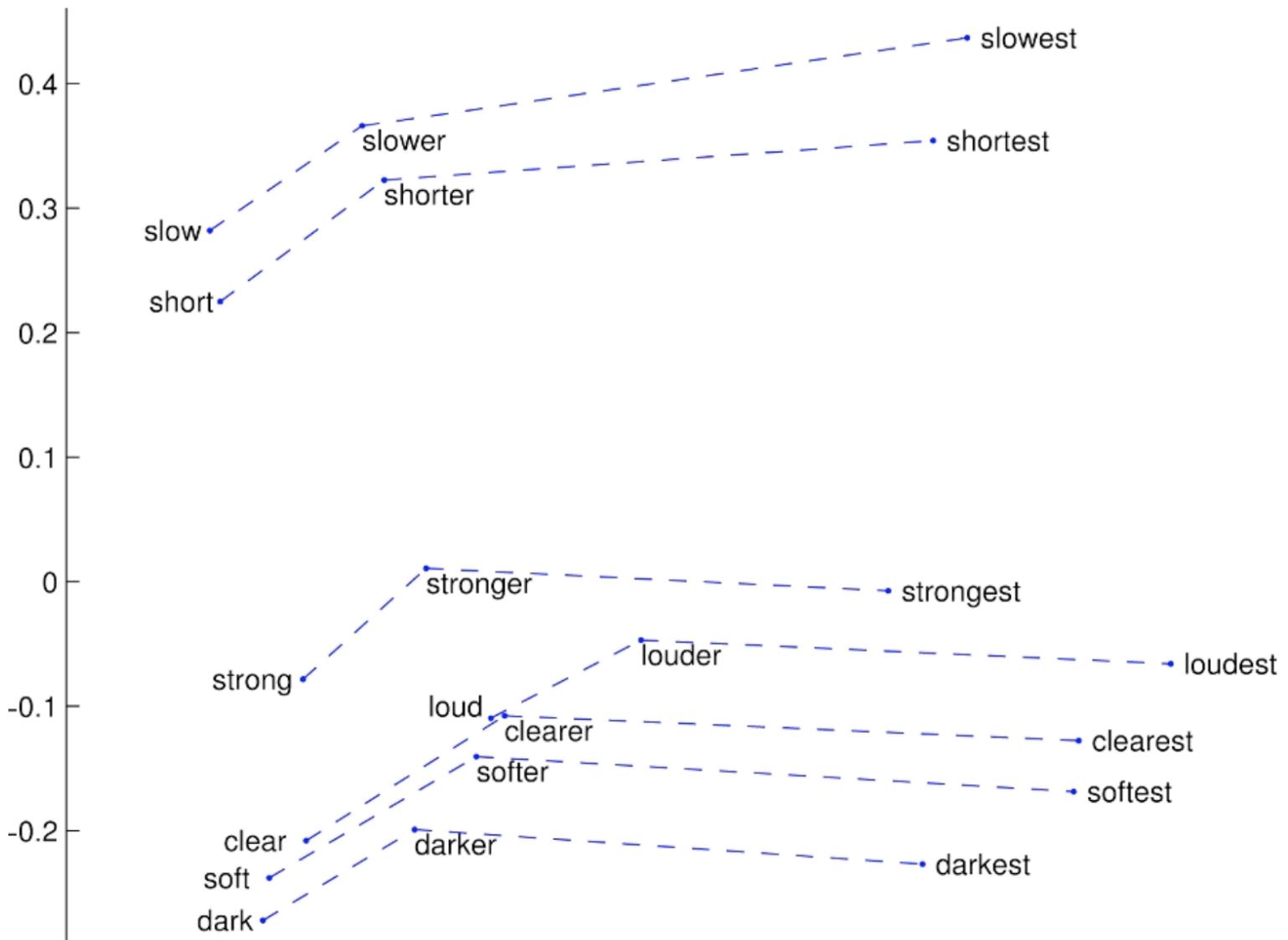


$$\vec{\text{king}} - \vec{\text{man}} + \vec{\text{woman}} \approx \vec{\text{queen}}$$

Text Features – Learnt Embeddings (Word2Vec)



Text Features – Learnt Embeddings (Word2Vec)



Example Tasks

- Classification
 - Sentiment Analysis
 - Readability
 - Categorisation
- Text Generation
 - Machine Translation
 - Summarisation
 - Data-to-text
- Natural Language Understanding
 - Inference / Reasoning
 - Chatbots / Intelligent Tutoring

<http://nlpprogress.com/>

Text Classification with Bert

Bert

- Bidirectional Encoder Representations from Transformers
 - Transformer Architecture
 - Masked Language Model

The cat sat <Mask> the mat.

Bert

- Bidirectional Encoder Representations from Transformers
 - Transformer Architecture
 - Masked Language Model

The cat sat on the <Mask>.

Bert

- Bidirectional Encoder Representations from Transformers
 - Transformer Architecture
 - Masked Language Model

The <Mask> sat on the mat.

Bert

- Bidirectional Encoder Representations from Transformers
 - Transformer Architecture
 - Masked Language Model
 - Next Sentence Prediction

The cat sat on the mat.

The cat was asleep.



Bert

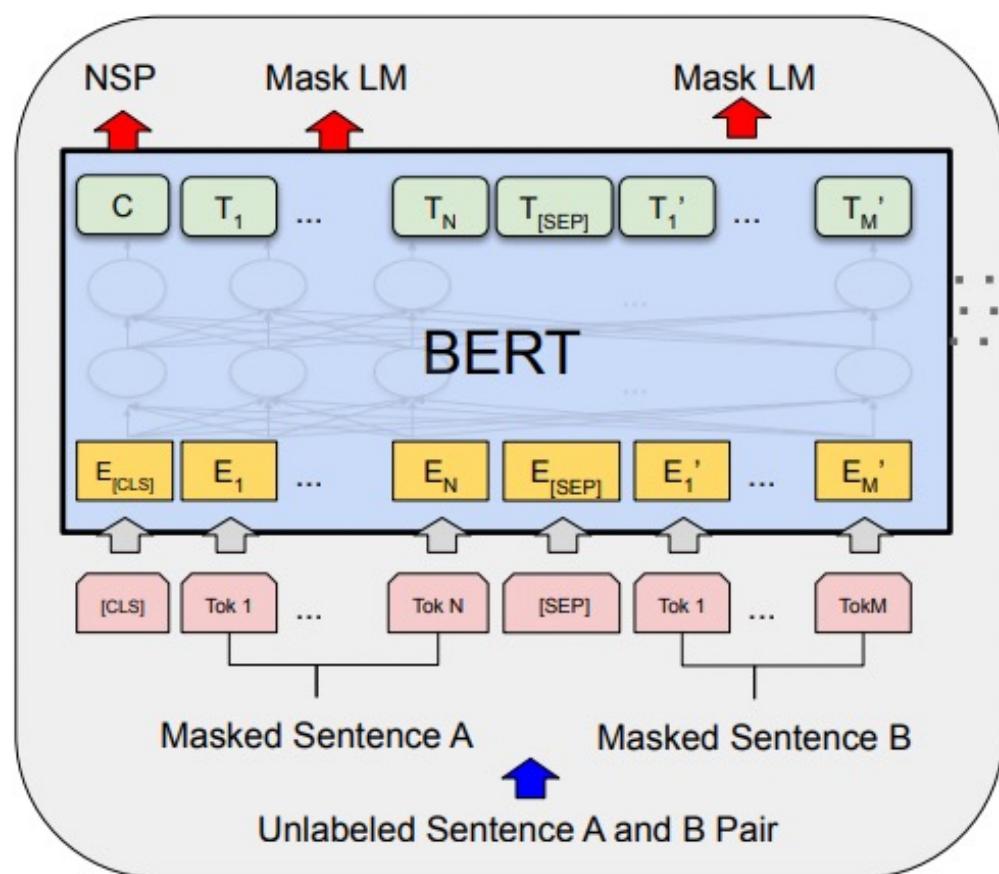
- Bidirectional Encoder Representations from Transformers
 - Transformer Architecture
 - Masked Language Model
 - Next Sentence Prediction

The cat sat on the mat.

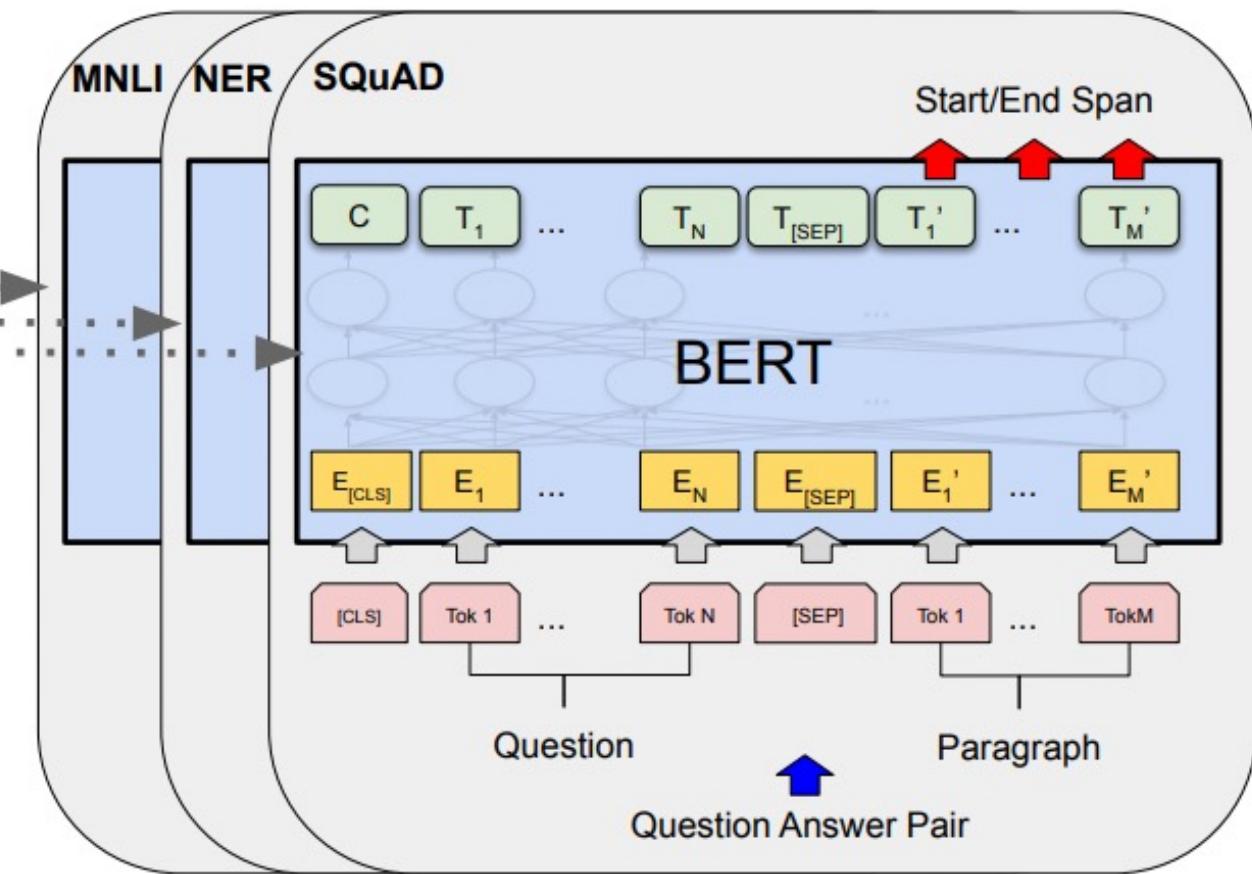
I like tomato soup.



Bert Architecture



Pre-training

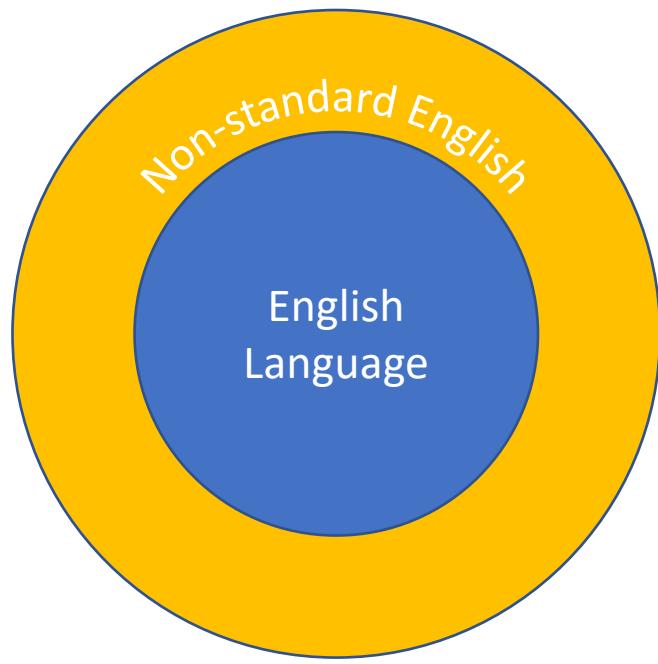


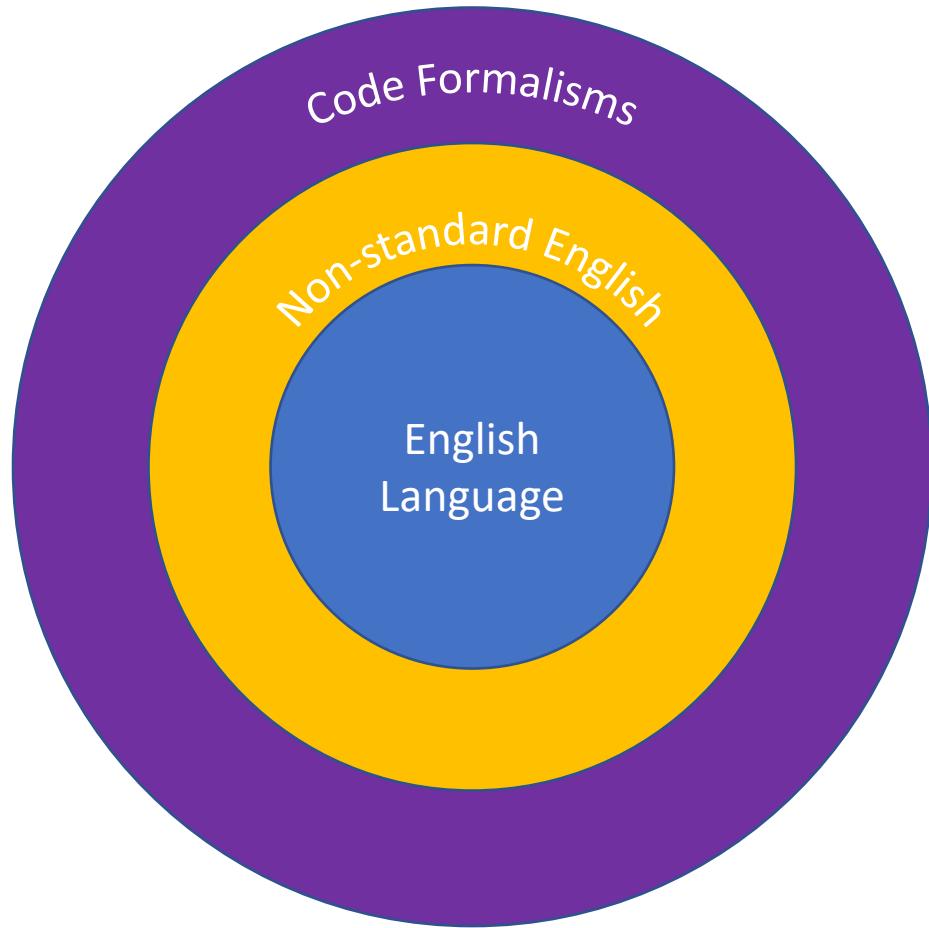
Fine-Tuning

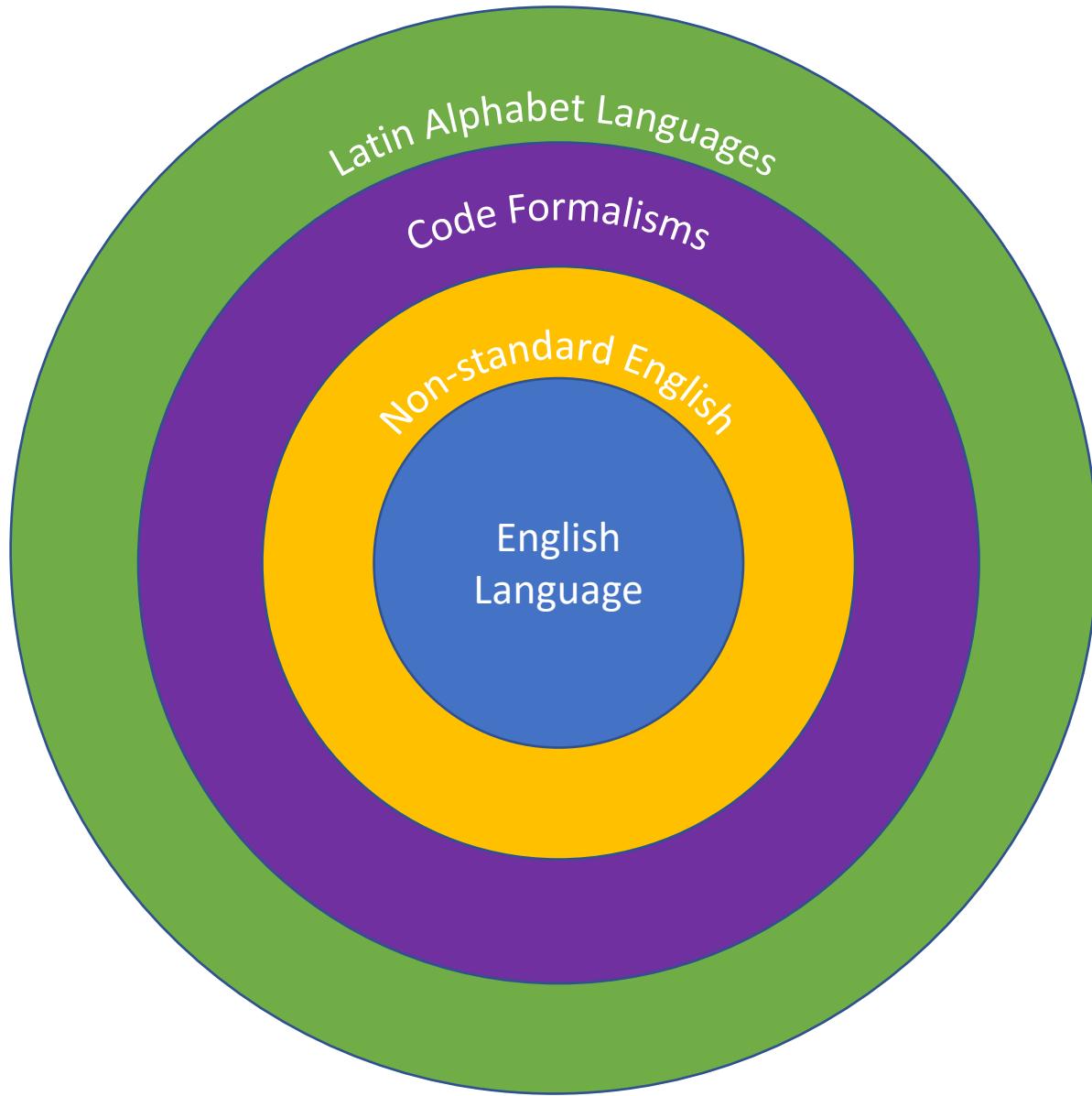
Tokenisation

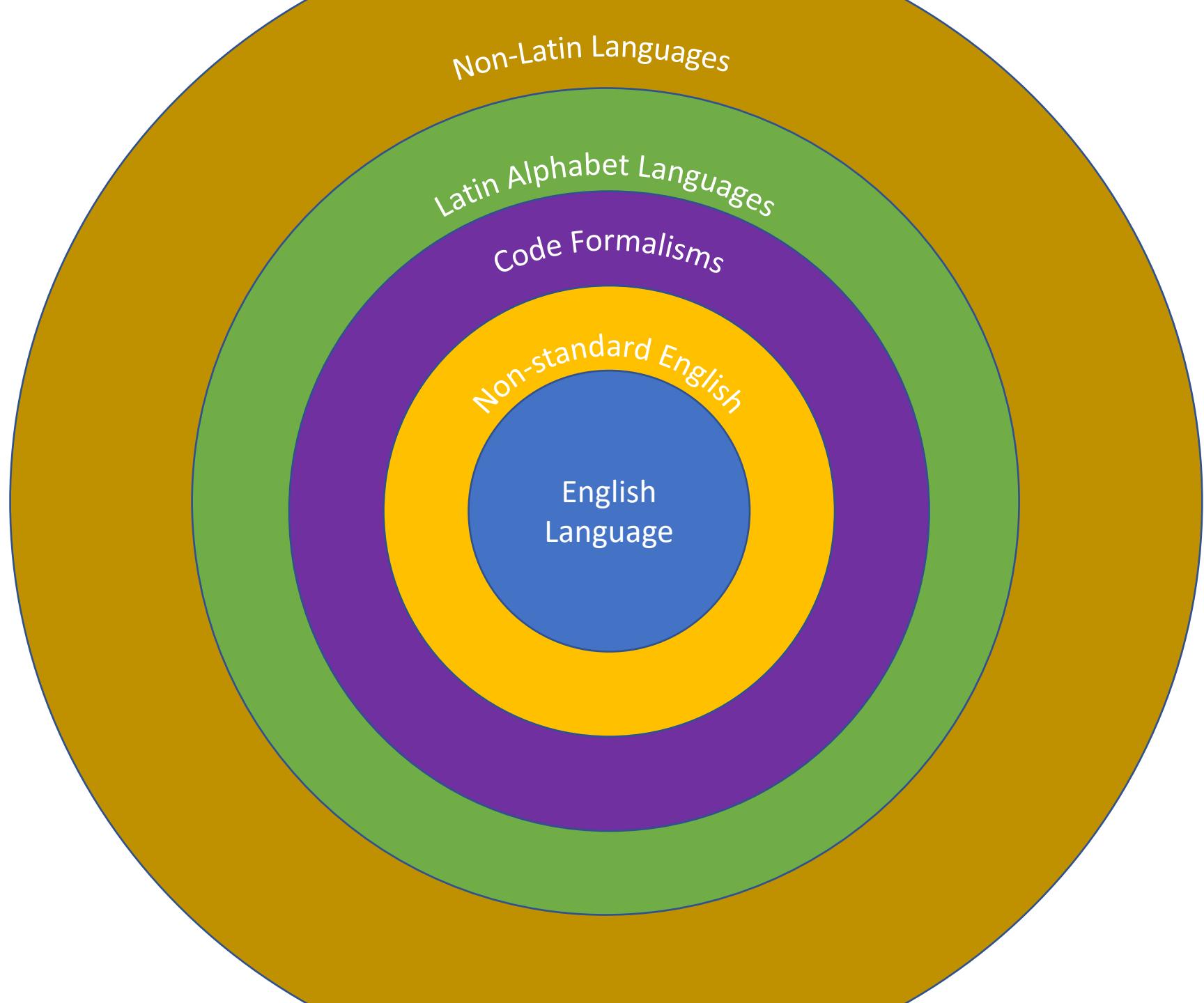


English
Language









Byte-Pair Encoding

- Used for data compression (1994)
- Replace most common byte-pairs with a single byte
- Retrieve original encoding with lookup table

bbcbbaabbaac

Byte-Pair Encoding

- Used for data compression (1994)
- Replace most common byte-pairs with a single byte
- Retrieve original encoding with lookup table

bbcbb**aabbaac**

Byte-Pair Encoding

- Used for data compression (1994)
- Replace most common byte-pairs with a single byte
- Retrieve original encoding with lookup table

XcXaaXaac

bb=X

Byte-Pair Encoding

- Used for data compression (1994)
- Replace most common byte-pairs with a single byte
- Retrieve original encoding with lookup table

Xc~~XYXY~~c

bb=X

aa=Y

Byte-Pair Encoding

- Used for data compression (1994)
- Replace most common byte-pairs with a single byte
- Retrieve original encoding with lookup table

Xc**Z**c

bb=X

aa=Y

XY=Z

Byte-Pair Encoding

- Used for tokenisation in same way
- Combine most frequent bytes together and store top-level tokens

plug plan hug tan rug man

Byte Pair Encoding

GPT-3 Codex

```
hello world
```

Clear

Show example

Tokens

2

Characters

11

```
hello world
```

Playground here: <https://platform.openai.com/tokenizer>

Byte Pair Encoding

GPT-3 Codex

banana spelled backwards is ananab

Clear

Show example

Tokens

8

Characters

34

banana spelled backwards is ananab

Playground here: <https://platform.openai.com/tokenizer>

Byte Pair Encoding

GPT-3

Codex

```
public class HelloWorld{
    public static void main(String [] args){
        System.out.println("Hello World!");
    }
}
```

Clear

Show example

Tokens

33

Characters

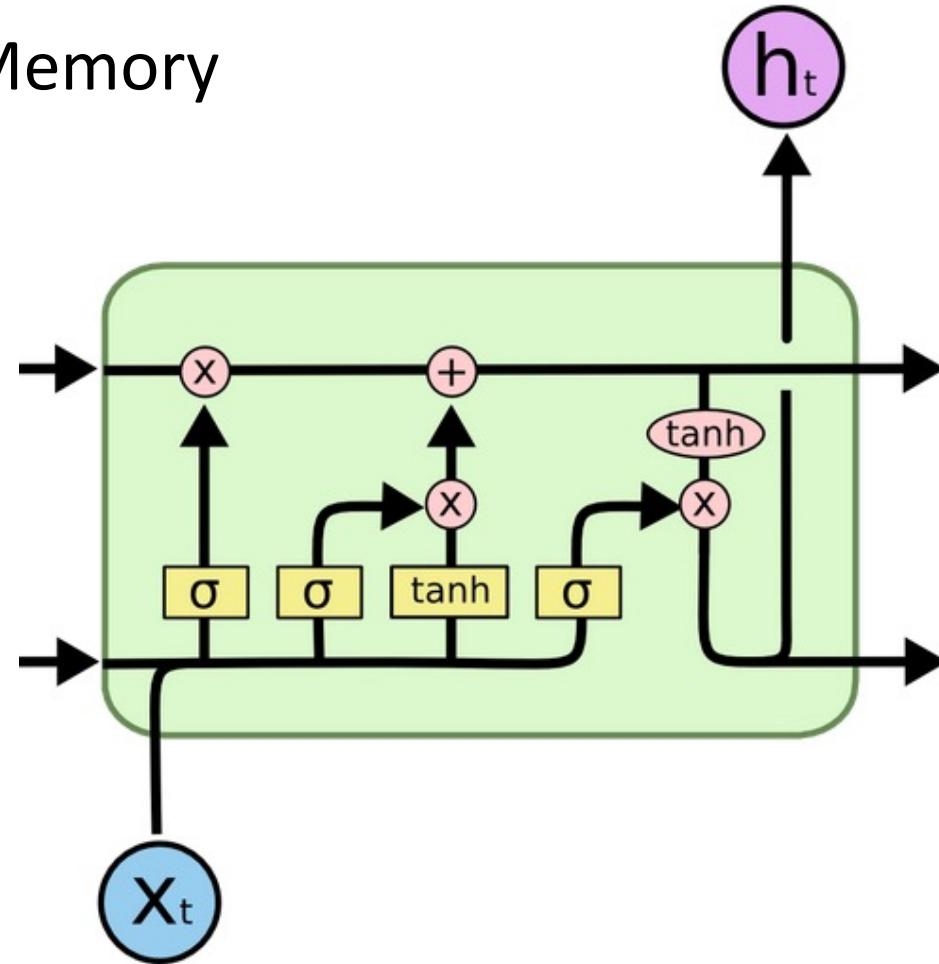
113

```
public class HelloWorld{
    public static void main(String [] args){
        System.out.println("Hello World!");
    }
}
```

Playgroun

Sequence classification

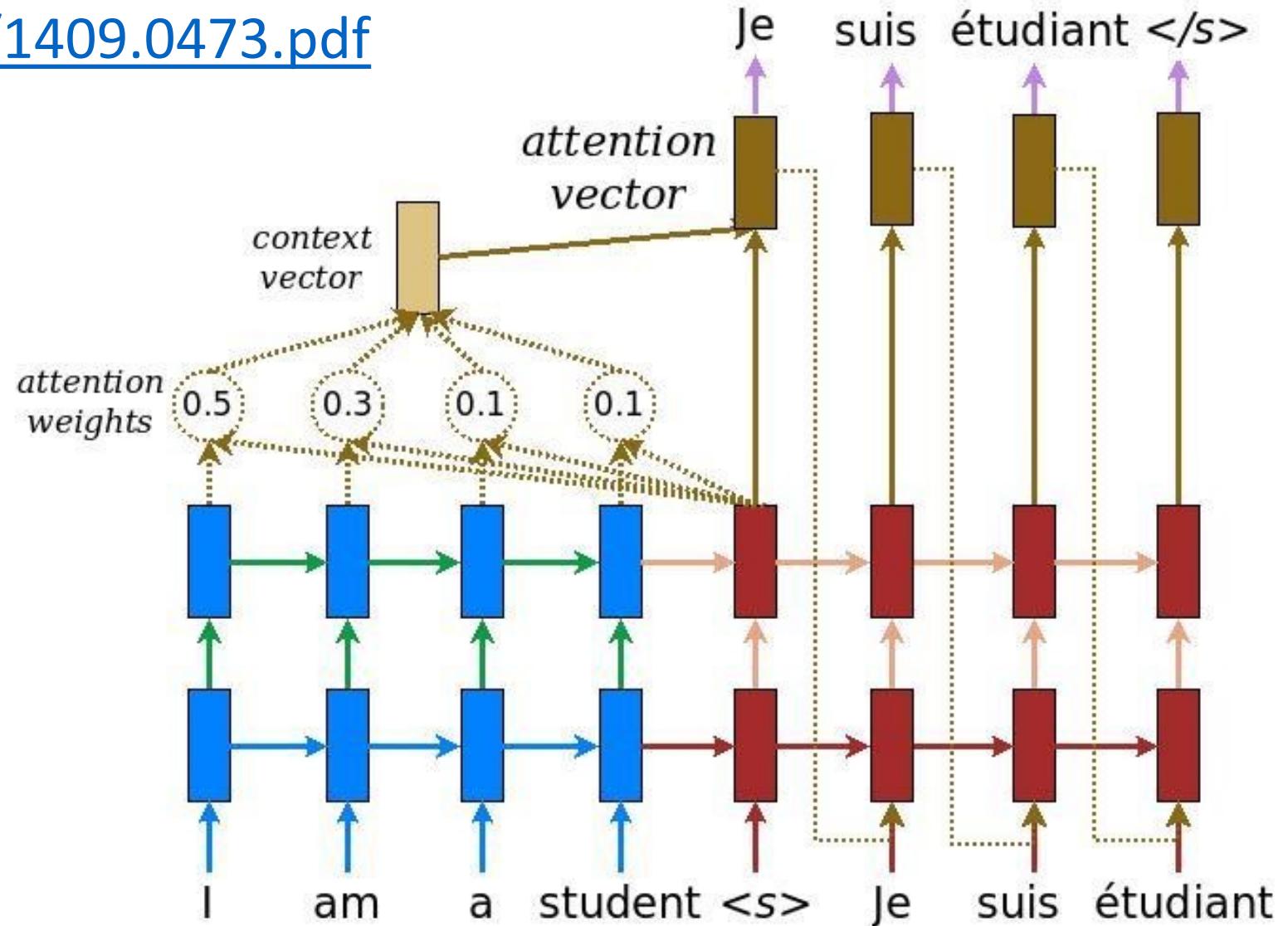
- Long Short Term Memory



Sequence classification

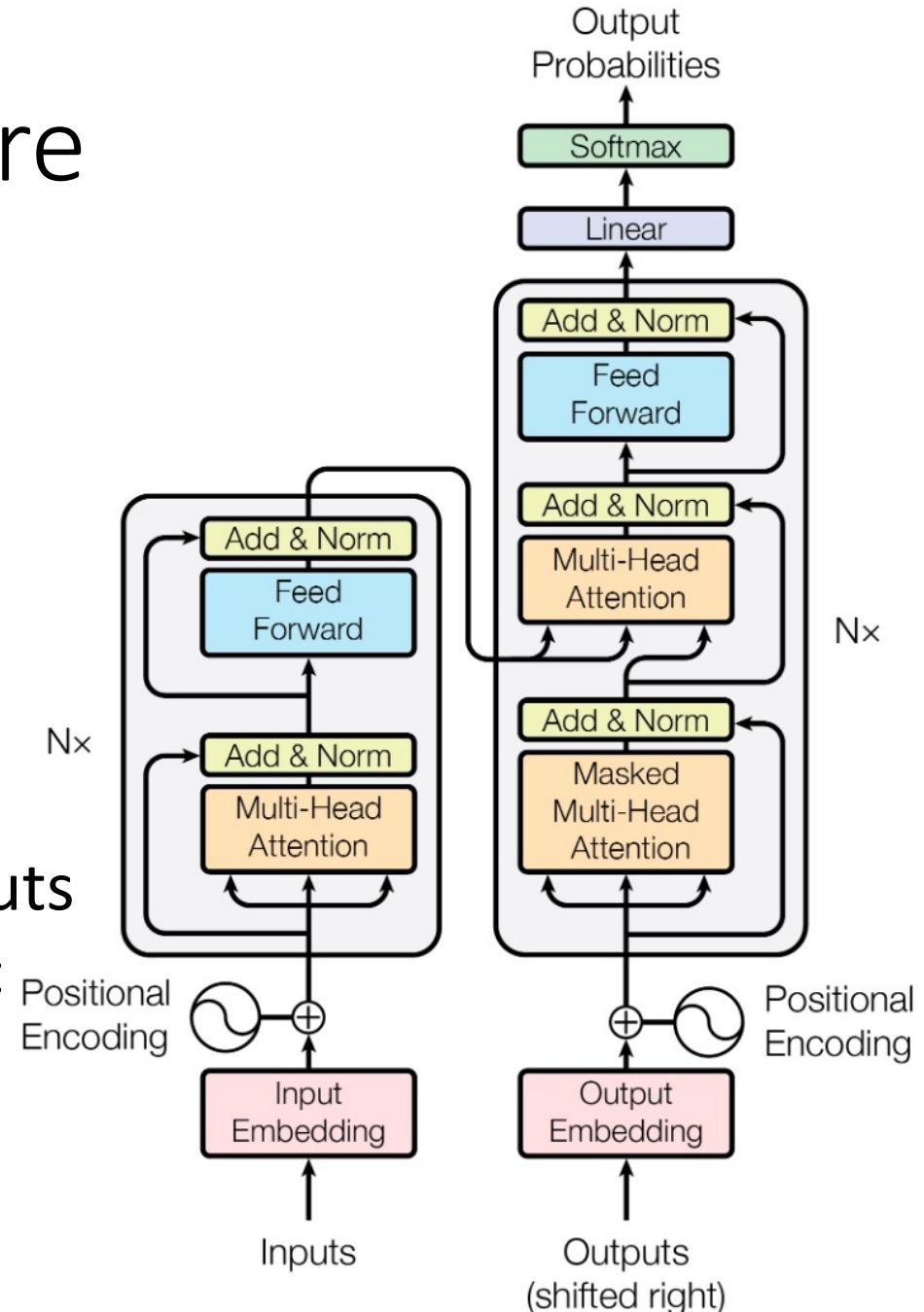
<https://arxiv.org/pdf/1409.0473.pdf>

- LSTMs + Attention



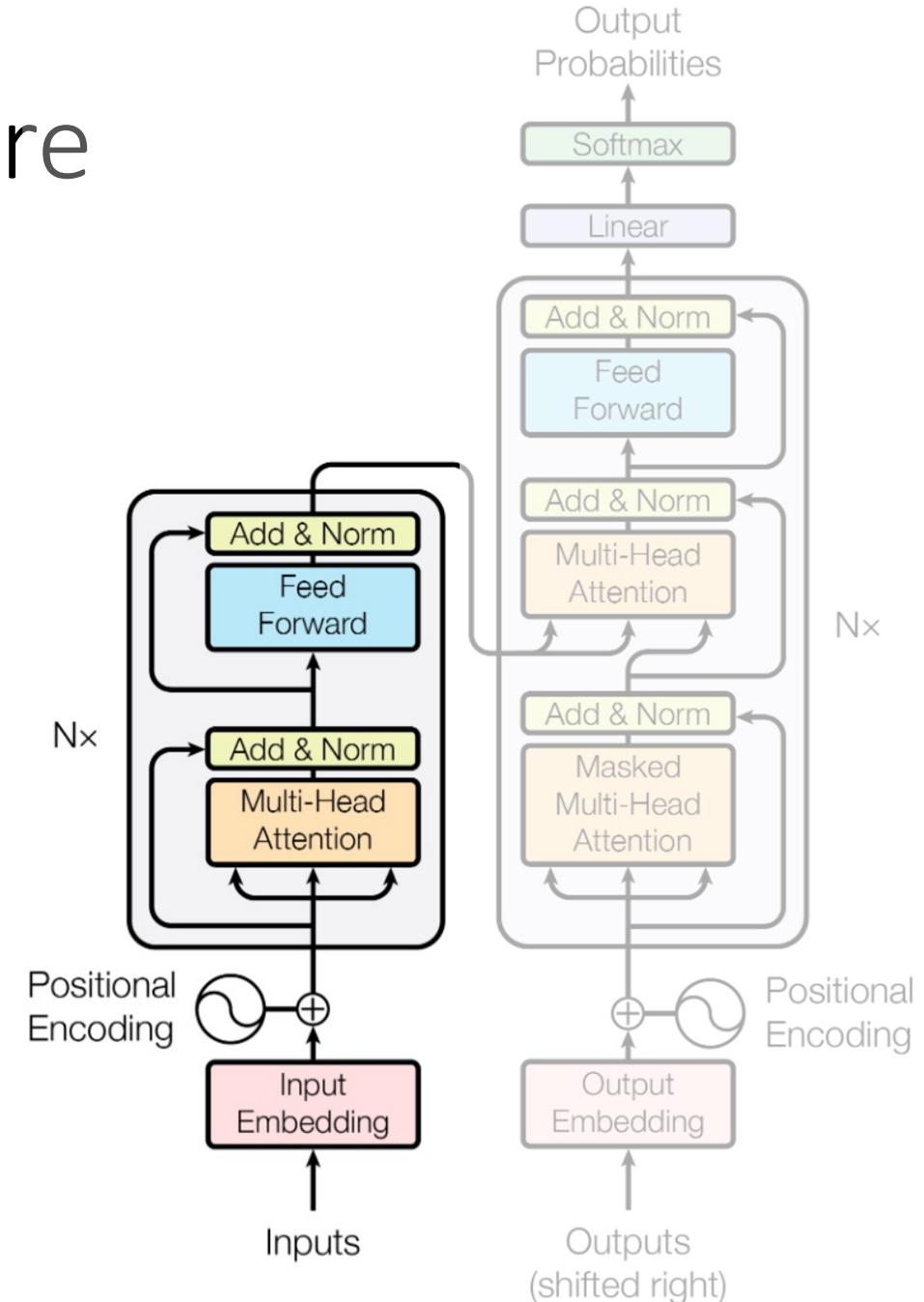
The Transformer - Architecture

- Encoder-Decoder architecture
 - Feature space acts as an interlingua
- Encoder
 - Inputs (words) are encoded using embeddings
 - Transforms input embeddings to feature space
 - 6 stacked layers attention and feed-forward
- Decoder – transform feature space to outputs
 - Converts output of last encoder layer into a set of probabilities which predict the next word



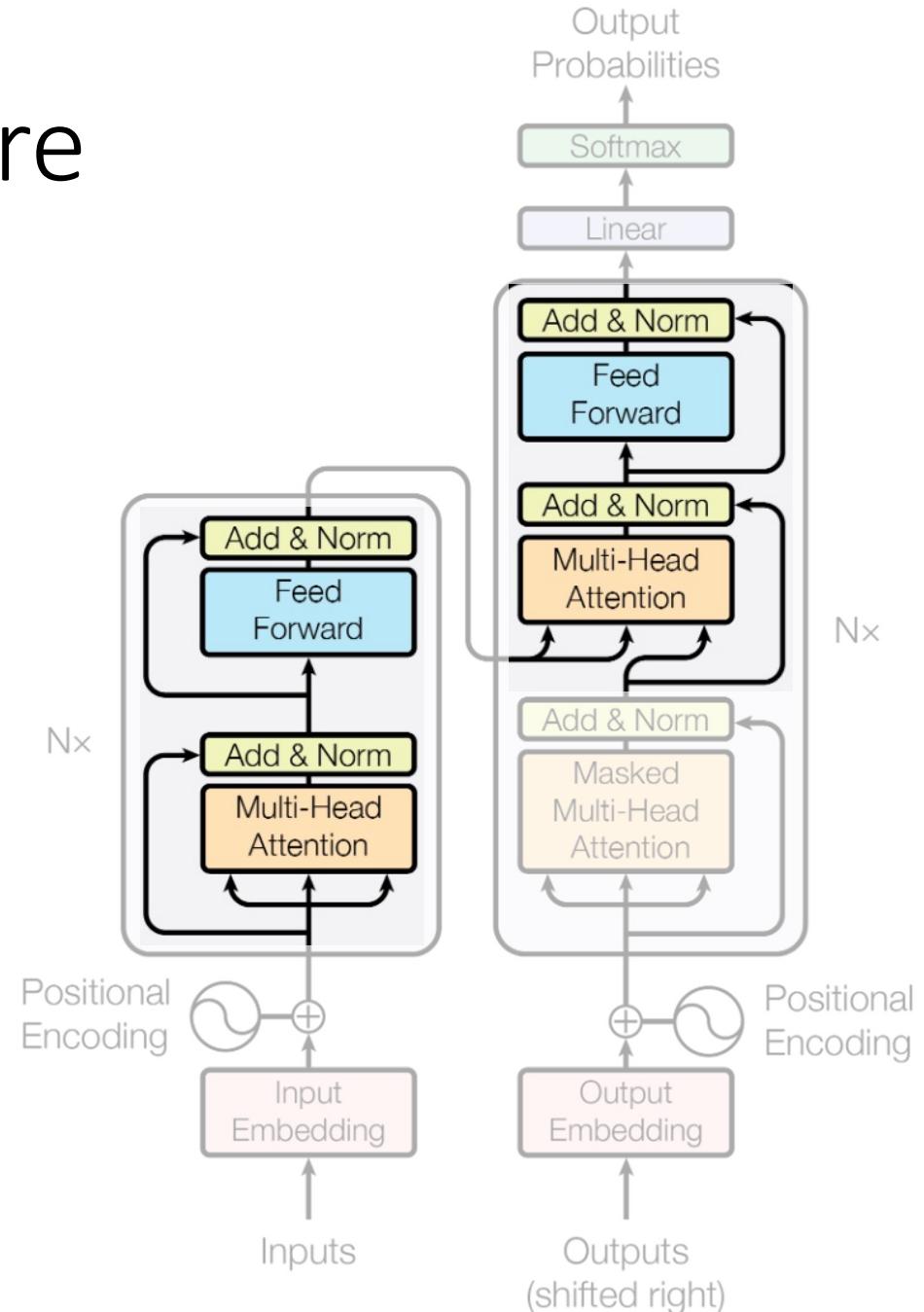
The Transformer - Architecture

- Encoder Layer
- Creates context vector for input sequence
- N=6 stacked layers



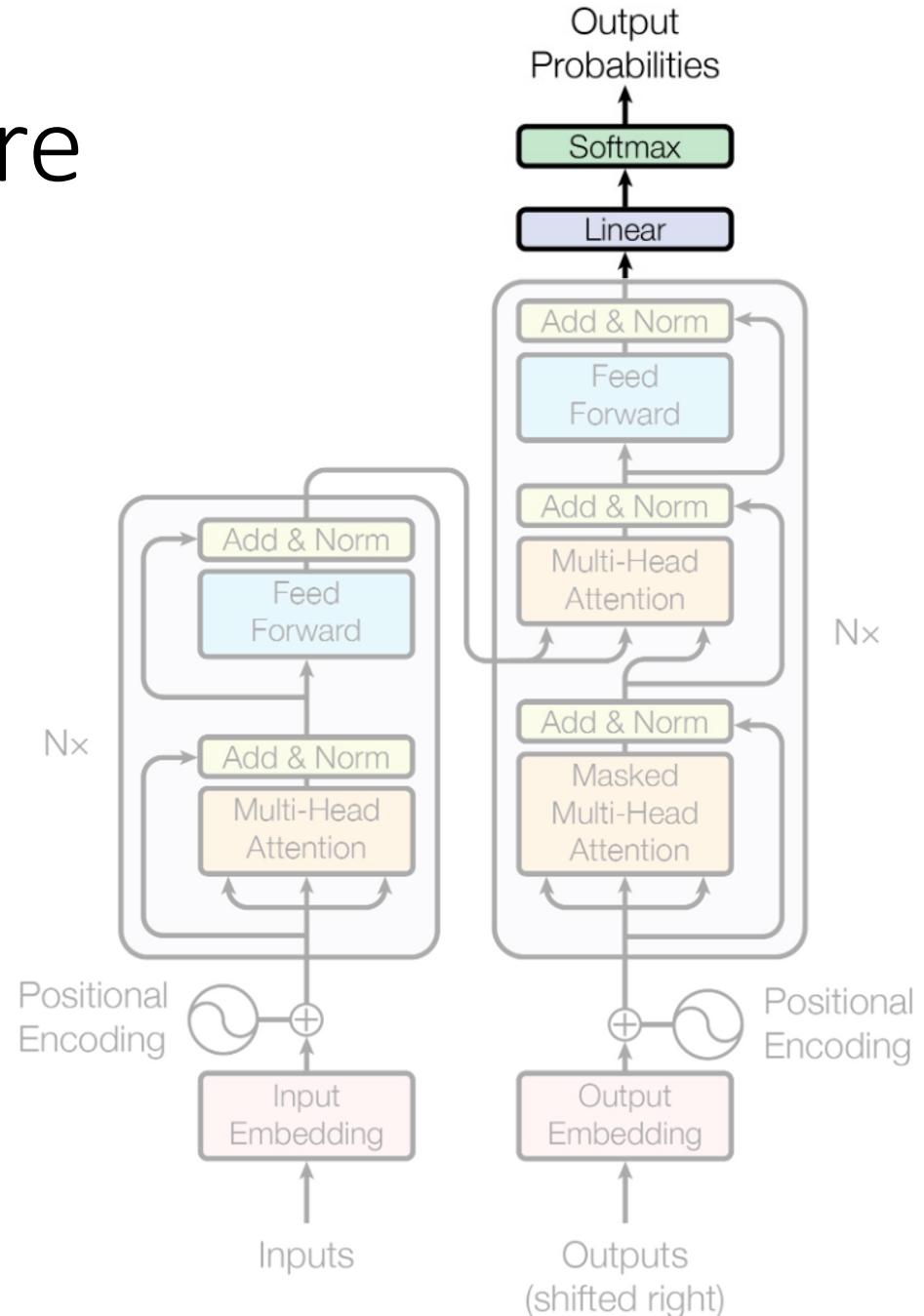
The Transformer - Architecture

- Decoder – Replicates encoder
- Multi-head self attention and feed forward network are carried over
 - Applied to the attention weights from encoder
 - Also applied to the last output from decoder



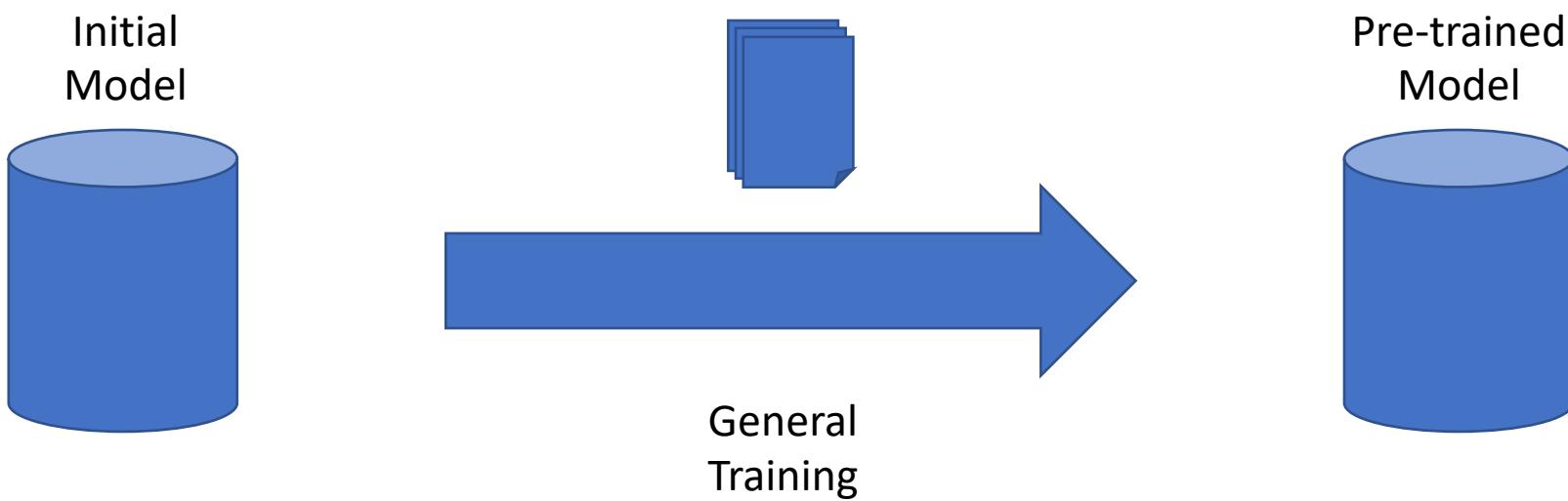
The Transformer - Architecture

- Output
- Linear layer
 - Fully connected NN
 - Converts k-dim vector to size of output vocab
- Softmax layer
 - Normalises probabilities in 0-1 range.



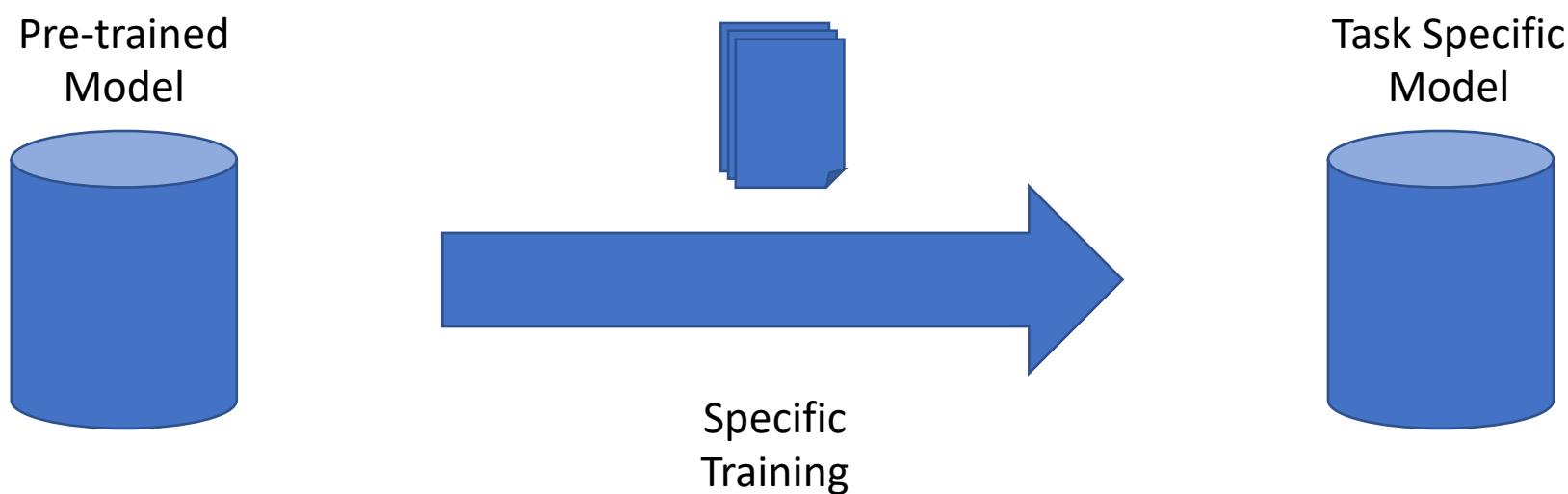
Pre-trained Models

- NLP would be very expensive if we had to retrain models every time
- Fortunately, pre-trained models are available that we can use

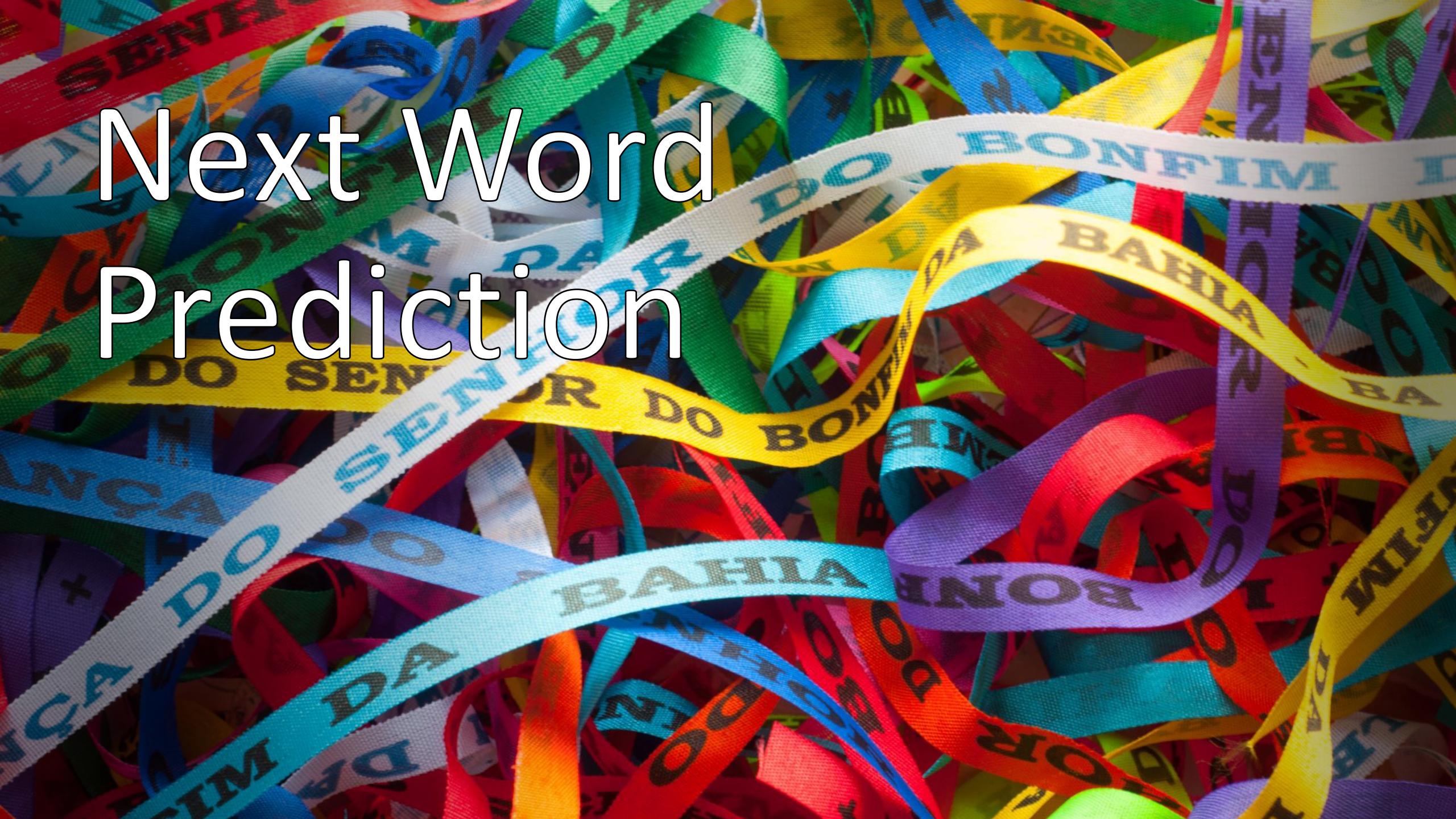


Fine-tuning

- Pre-trained models can be adapted to a specific task



Next Word Prediction



Language Model



The

cat



Language Model



The

cat

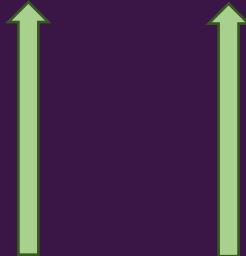


Language Model



The cat

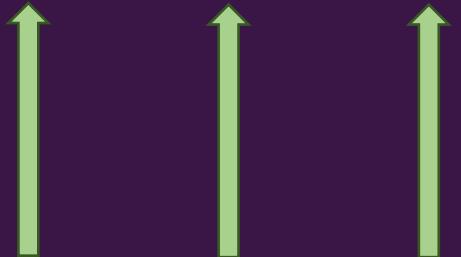
cat sat



Language Model

The cat

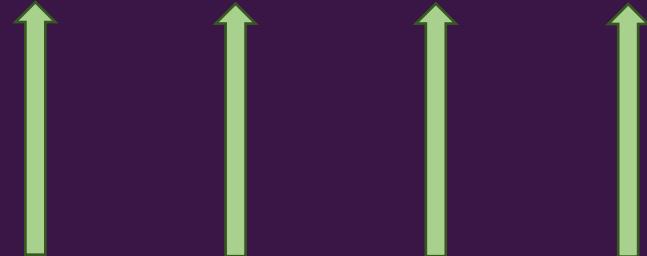
cat sat on



Language Model

The cat sat

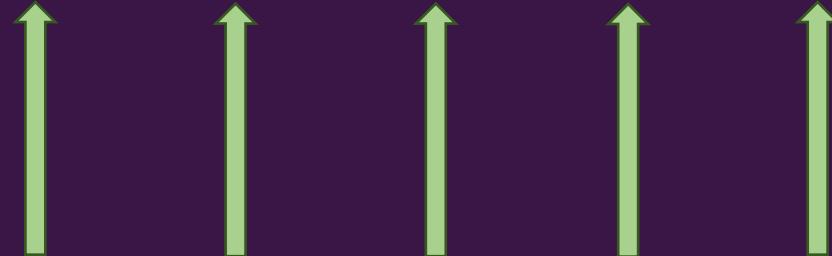
cat sat on the



Language Model

The cat sat on

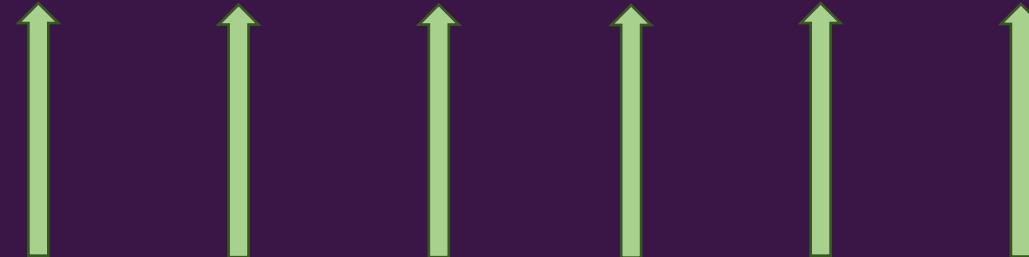
cat sat on the mat



Language Model

The cat sat on the

cat sat on the mat .



Language Model

The cat sat on the mat

cat sat on the mat . END

Language Model

The cat sat on the mat .

Decoding Strategy

Decoding Strategy

User: Who is the president of the USA?

LM: The President of the USA is...

Decoding Strategy

User: Who is the president of the USA?

LM: The President of the USA is...

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding Strategy 1 – Greedy Search

User: Who is the president of the USA?

LM: The President of the USA is...

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding Strategy 2 – Top-k Sampling

User: Who is the president of the USA?

LM: The President of the USA is...

k=3

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding Strategy 2 – Top-k Sampling

User: Who is the president of the USA?

LM: The President of the USA is...

k=5

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding Strategy 3 – Top-p Sampling

User: Who is the president of the USA?

LM: The President of the USA is...

p=0.9

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding Strategy 3 – Top-p Sampling

User: Who is the president of the USA?

LM: The President of the USA is...

p=0.8

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Decoding

- Reweighting of probability mass with softmax

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=0}^N e^{z_j}}$$

Decoding

- Reweighting of probability mass with softmax + temperature ($\theta = 0.2$)

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

$$\sigma(z_i) = \frac{e^{\frac{z_i}{\theta}}}{\sum_{j=0}^N e^{\frac{z_j}{\theta}}}$$

Decoding

- Reweighting of probability mass with softmax + temperature ($\theta = 0.8$)

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

$$\sigma(z_i) = \frac{e^{\frac{z_i}{\theta}}}{\sum_{j=0}^N e^{\frac{z_j}{\theta}}}$$

Hallucination

Hallucination

User: Who is the president of the USA?

LM: The President of the USA is...

Joe	0.57
Barack	0.20
Donald	0.12
George	0.03
Trump	0.02
Mr.	0.01
the	0.01
...	...

Problem: Only one of these is correct!

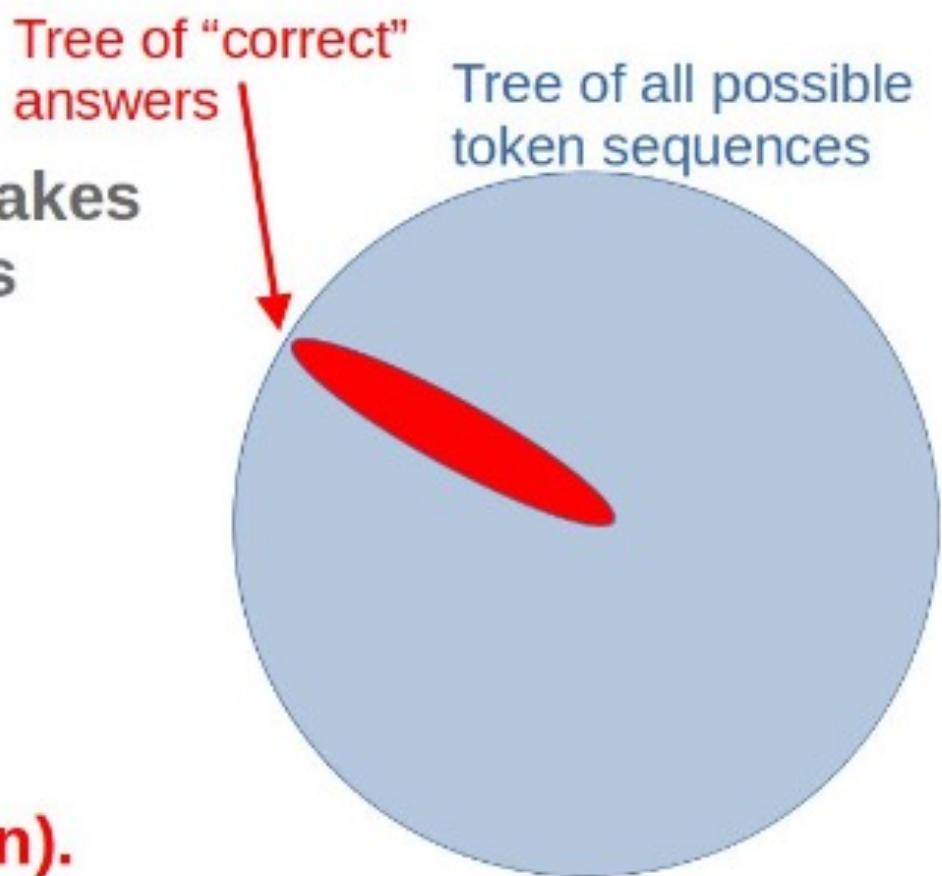
Unpopular Opinion about AR-LLMs



- ▶ Auto-Regressive LLMs are **doomed**.
- ▶ They cannot be made factual, non-toxic, etc.
- ▶ They are not controllable

- ▶ Probability e that any produced token takes us outside of the set of correct answers
- ▶ Probability that answer of length n is correct:
 - ▶ $P(\text{correct}) = (1-e)^n$

- ▶ **This diverges exponentially.**
- ▶ **It's not fixable (without a major redesign).**



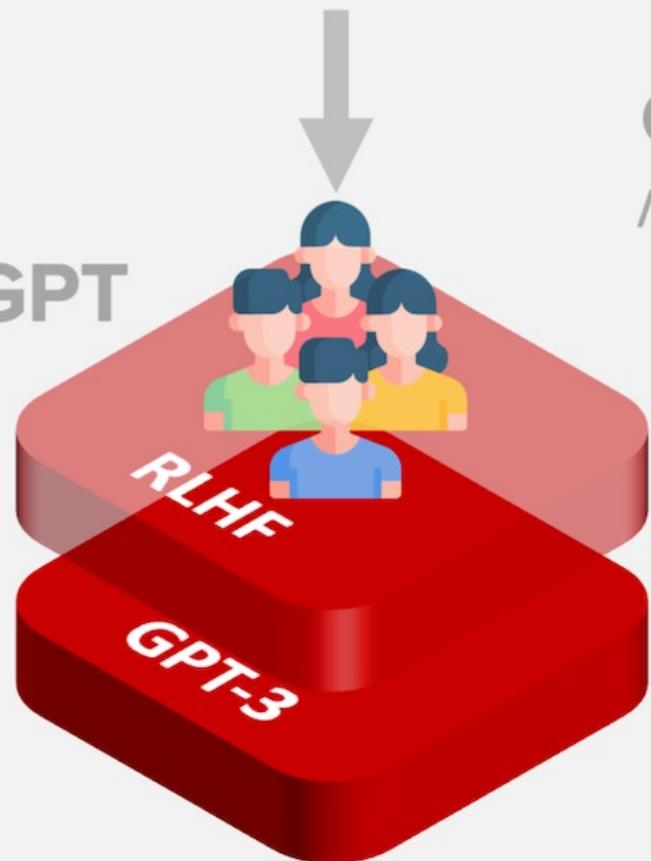


Conversational AI



OpenAI

InstructGPT
Jan/2022



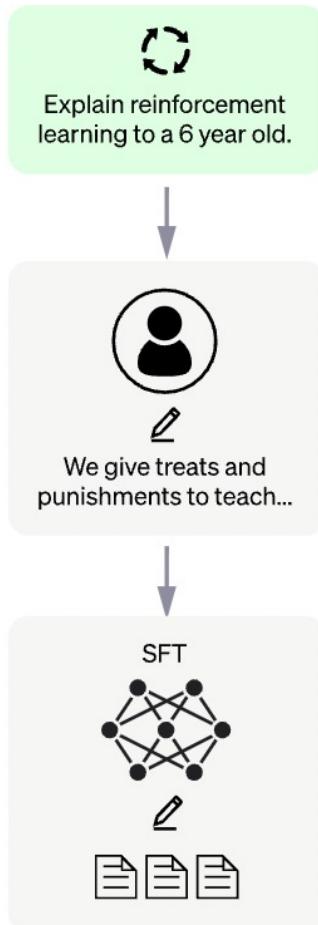
ChatGPT
Nov/2022



Step 1

Collect demonstration data and train a supervised policy.

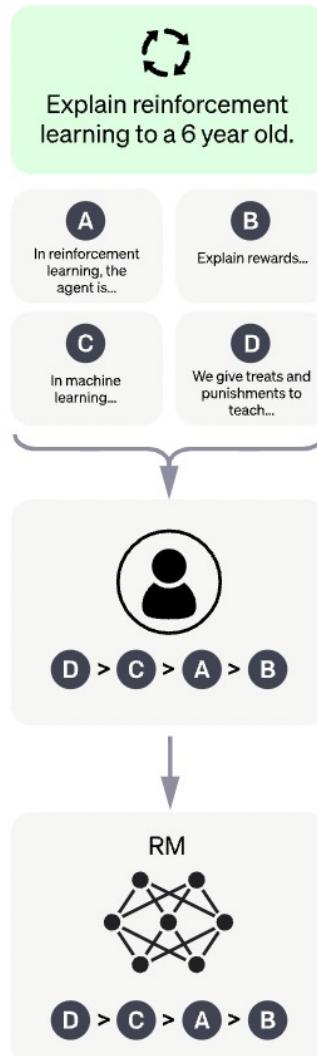
A prompt is sampled from our prompt dataset.



Step 2

Collect comparison data and train a reward model.

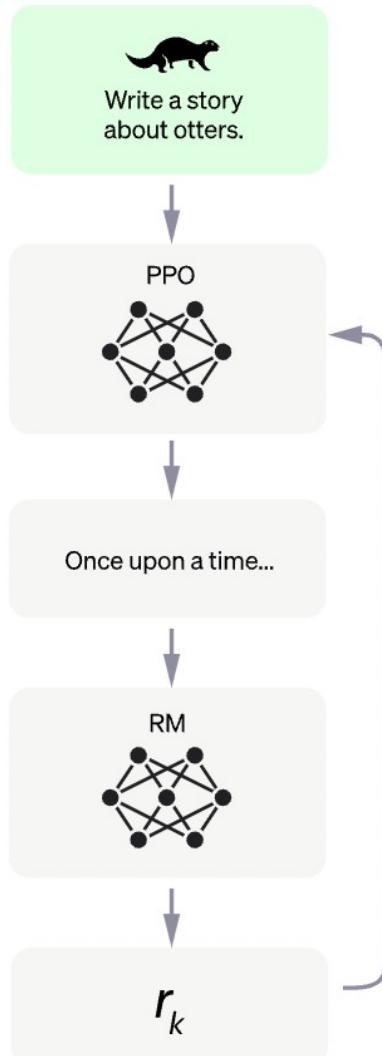
A prompt and several model outputs are sampled.

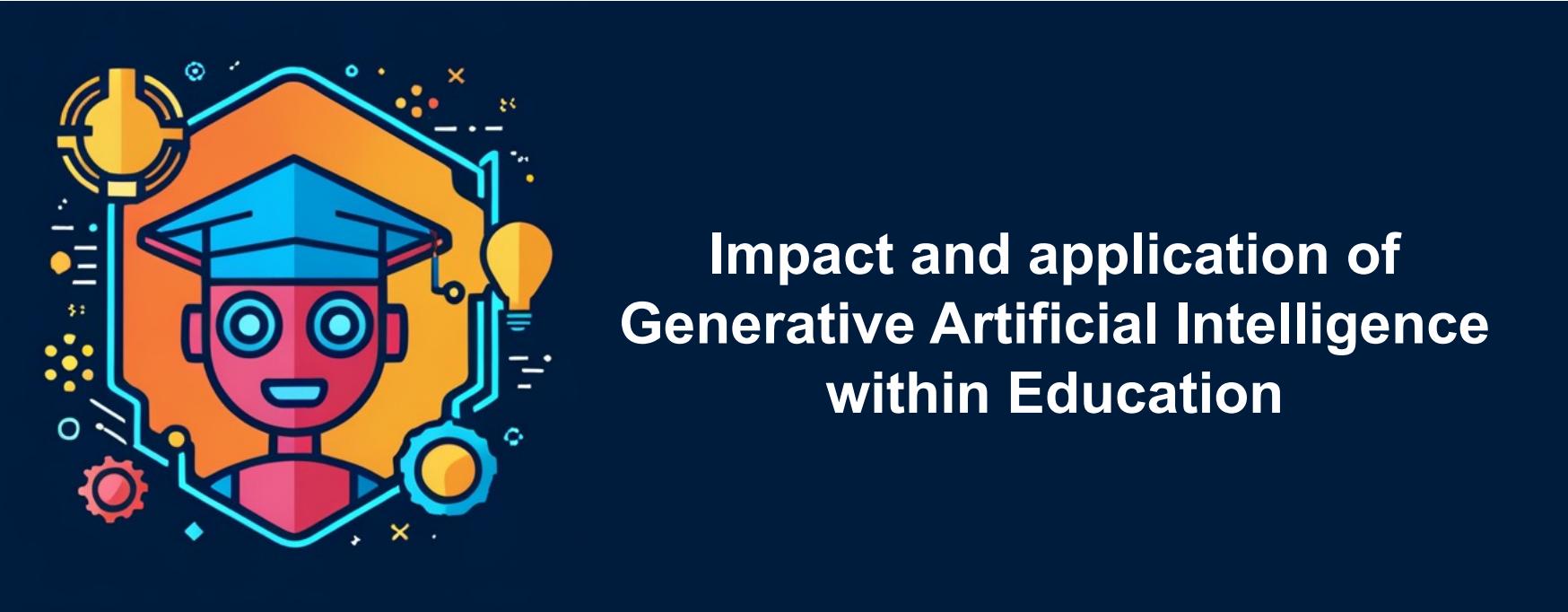


Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

A new prompt is sampled from the dataset.





Impact and application of Generative Artificial Intelligence within Education

What can ChatGPT Do?



Dr. Matthew Shardlow (m.shardlow@mmu.ac.uk)
Manchester Metropolitan University
Senior Lecturer



MA

Name two football teams that play in Manchester



The two football teams that play in Manchester are Manchester United and Manchester City. thumb up thumb down

MA

Which one plays at the Etihad stadium?



Manchester City plays at the Etihad Stadium. thumb up thumb down

MA

How many people can the Etihad stadium seat?



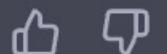
The Etihad Stadium, which is home to Manchester City football club, has a seating capacity of approximately 55,000 spectators. thumb up thumb down

MA

Write a 4 line poem about the perils of educating computer science undergraduates who would rather be playing video games



They sit in class, eyes glazed,
Thinking of the worlds they'd rather explore,
Their minds on pixels, not code,
Education lost to the lure of the digital war.





Even in highly-developed countries, as many as 15-30% of the population can only understand texts written using a basic vocabulary. Their understanding of everyday texts is limited, which prevents them from taking an active role in society and making informed decisions regarding healthcare, legal representation, or democratic choice. Lexical simplification is a natural language processing task that aims to make text understandable to everyone by replacing complex vocabulary and expressions with simpler ones, while preserving the original meaning. It has attracted considerable attention in the last 20 years, and fully automatic lexical simplification systems have been proposed for various languages. The main obstacle for the progress of the field is the absence of high-quality datasets for building and evaluating lexical simplification systems. We present a new benchmark dataset for lexical simplification in English, Spanish, and (Brazilian) Portuguese, and provide details about data selection and annotation procedures. This is the first dataset that offers a direct comparison of lexical simplification systems for three languages. To showcase the usability of the dataset, we adapt two state-of-the-art lexical simplification systems with differing architectures (neural vs.\ non-neural) to all three languages (English, Spanish, and Brazilian Portuguese) and evaluate their performances on our new dataset. For a fairer comparison, we use several evaluation measures which capture varied aspects of the systems' efficacy, and discuss their strengths and weaknesses. We find a state-of-the-art neural lexical simplification system outperforms a state-of-the-art ...

MA

explain the following abstract in simple terms that a non-expert would understand:



Even in highly-developed countries, as many as 15-30% of the population can only understand texts written using a basic vocabulary. Their understanding of everyday texts



"Lexical simplification" is a way to make text easier to understand for people who may have trouble with complex vocabulary or expressions. This is especially important for people in developed countries who may not have the reading skills to fully understand everyday texts. However, there is currently a lack of good data to build and test systems that can simplify text. This study presents a new dataset for testing lexical simplification in English, Spanish, and Portuguese, and uses it to compare the performance of different systems. Two different types of systems, one using neural networks and one not, were tested, and it was found that the neural network system performed better.



languages (English, Spanish, and Brazilian Portuguese) and evaluate their performances on our new dataset. For a fairer comparison, we use several evaluation measures which capture varied aspects of the systems' efficacy, and discuss their strengths and weaknesses. We find a state-of-the-art neural lexical simplification system outperforms a state-of-the-art ...

MA

can you write me a solution to fizzbuzz in python



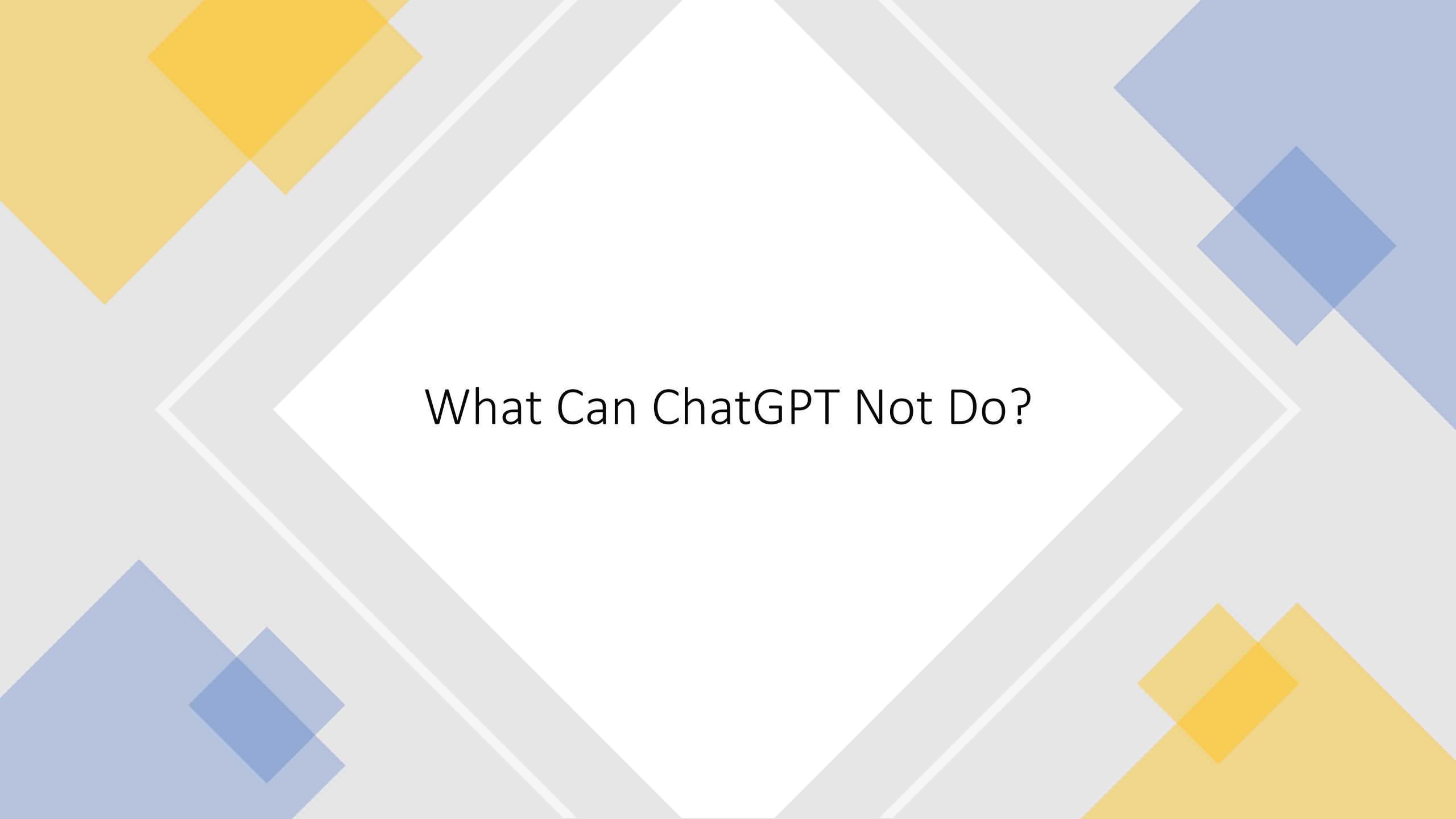
Sure! Here's a simple solution to FizzBuzz in Python:



Copy code

```
for i in range(1, 101):
    if i % 15 == 0:
        print("FizzBuzz")
    elif i % 3 == 0:
        print("Fizz")
    elif i % 5 == 0:
        print("Buzz")
    else:
        print(i)
```

This code uses a for loop to iterate from 1 to 100 (inclusive) and checks the remainder of each number when divided by 15, 3, and 5. If the remainder is 0, it prints "FizzBuzz", "Fizz", or "Buzz" respectively. If the remainder is not 0, it simply prints the number.



What Can ChatGPT Not Do?

MA

what is the latest on the war in Ukraine?



I'm sorry, I don't have access to current information as my knowledge cut-off date is 2021.  
The situation in Ukraine is complex and has been evolving rapidly, so I would recommend checking with a more recent source for the latest information.

MA

give me 3 references for support vector machines



1. "Support Vector Networks" by Cortes and Vapnik (1995)
2. "A Tutorial on Support Vector Machines for Pattern Recognition" by Cristianini and Shawe-Taylor (2000)
3. "Support Vector Machine Solvers" by Suykens and Vandewalle (2002)

MA

give me 3 references for support vector machines



1. "Support Vector Networks" by Cortes and Vapnik (1995)
2. "A Tutorial on Support Vector Machines for Pattern Recognition" by Cristianini and Shawe-Taylor (2000)
3. "Support Vector Machine Solvers" by Suykens and Vandewalle (2002)

Correct Reference

Doesn't exist

Wrong authors



Sister's Age Calculation



When I was 6 my sister was half my age. Now I'm 70 how old is my sister?



If you are 70 years old and your sister was half your age when you were 6, then your sister was $6 / 2 = <<6/2=3>>3$ years old when you were 6.

If your sister was 3 years old when you were 6 and you are now 70, then your sister is now $3 + 70 = <<3+70=73>>73$ years old.

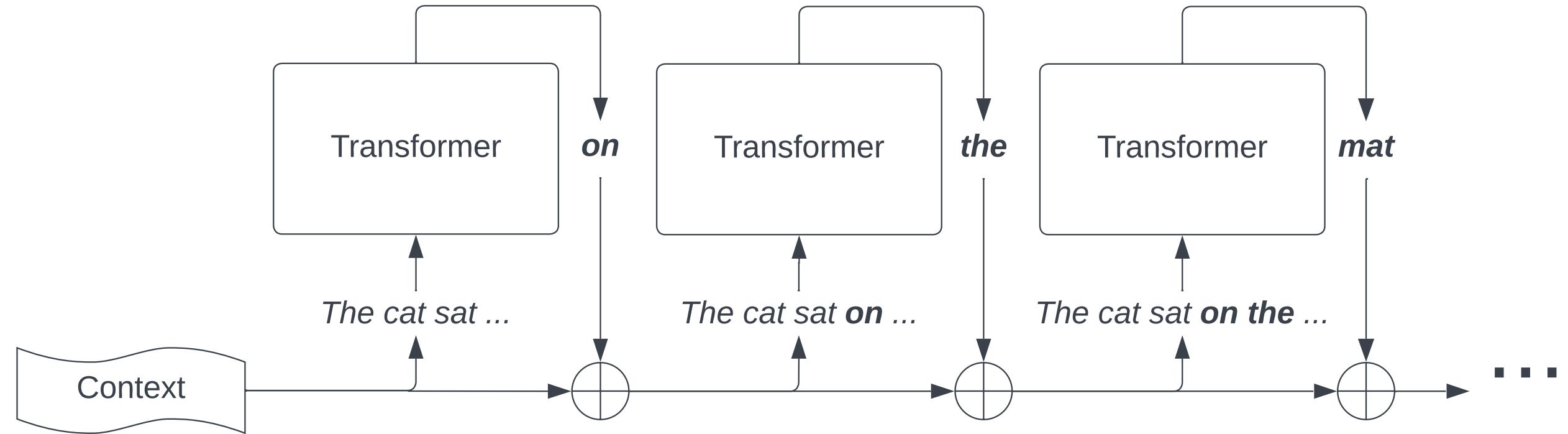
So your sister is 73 years old.

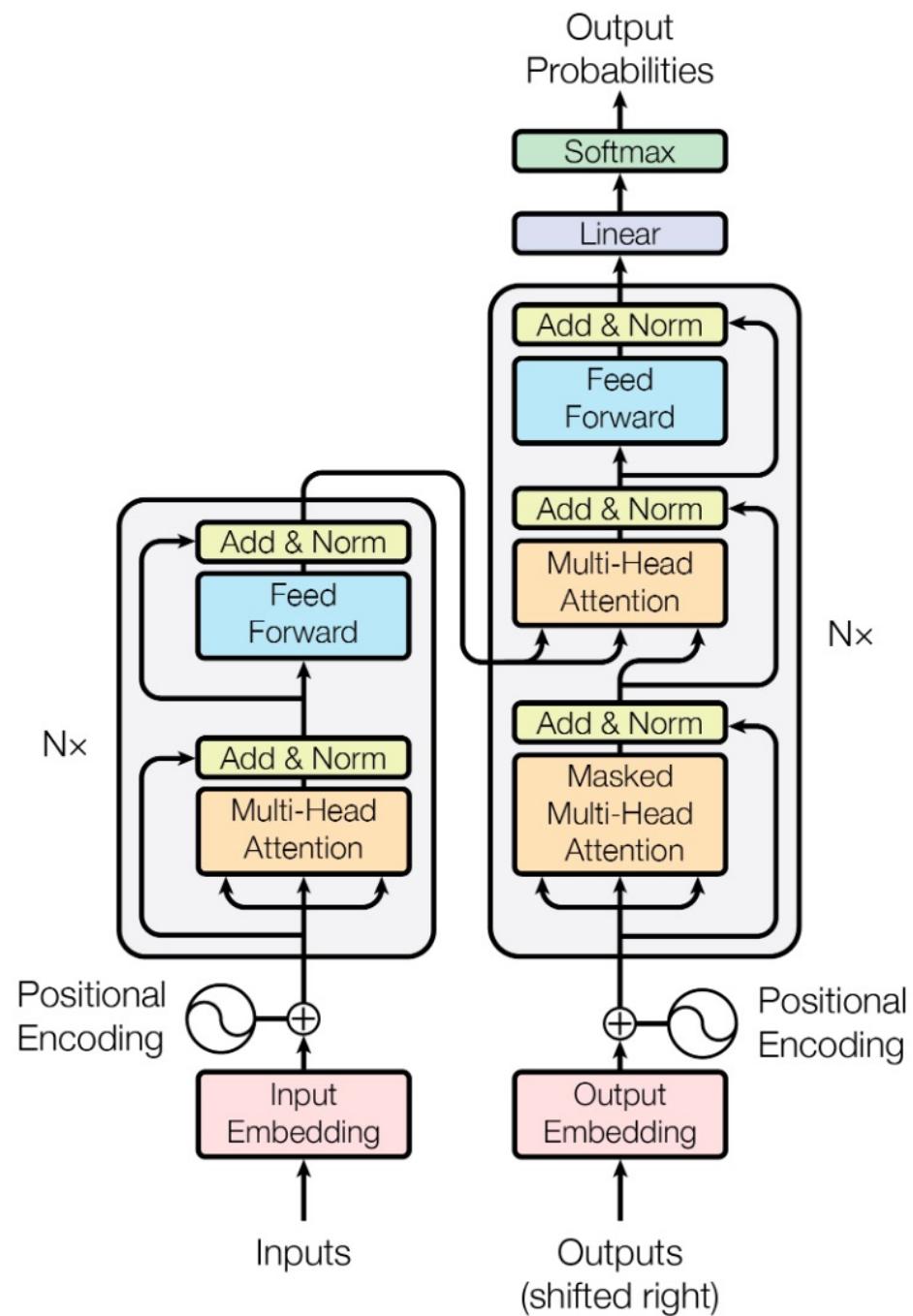


Text Generation with GPT

GPT

- Generative Pre-trained Transformer
- Transformer model that is trained to predict the next word in a sentence
- Trained over many NLP tasks using different model output layers
- GPT -> GPT-2 -> GPT-3 -> GPT-3.5





GPT-3

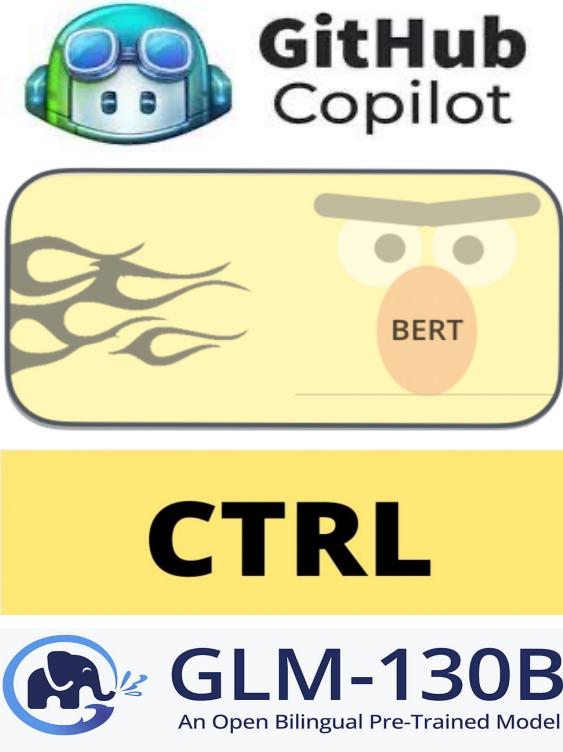
175 Billion Parameter Language Model

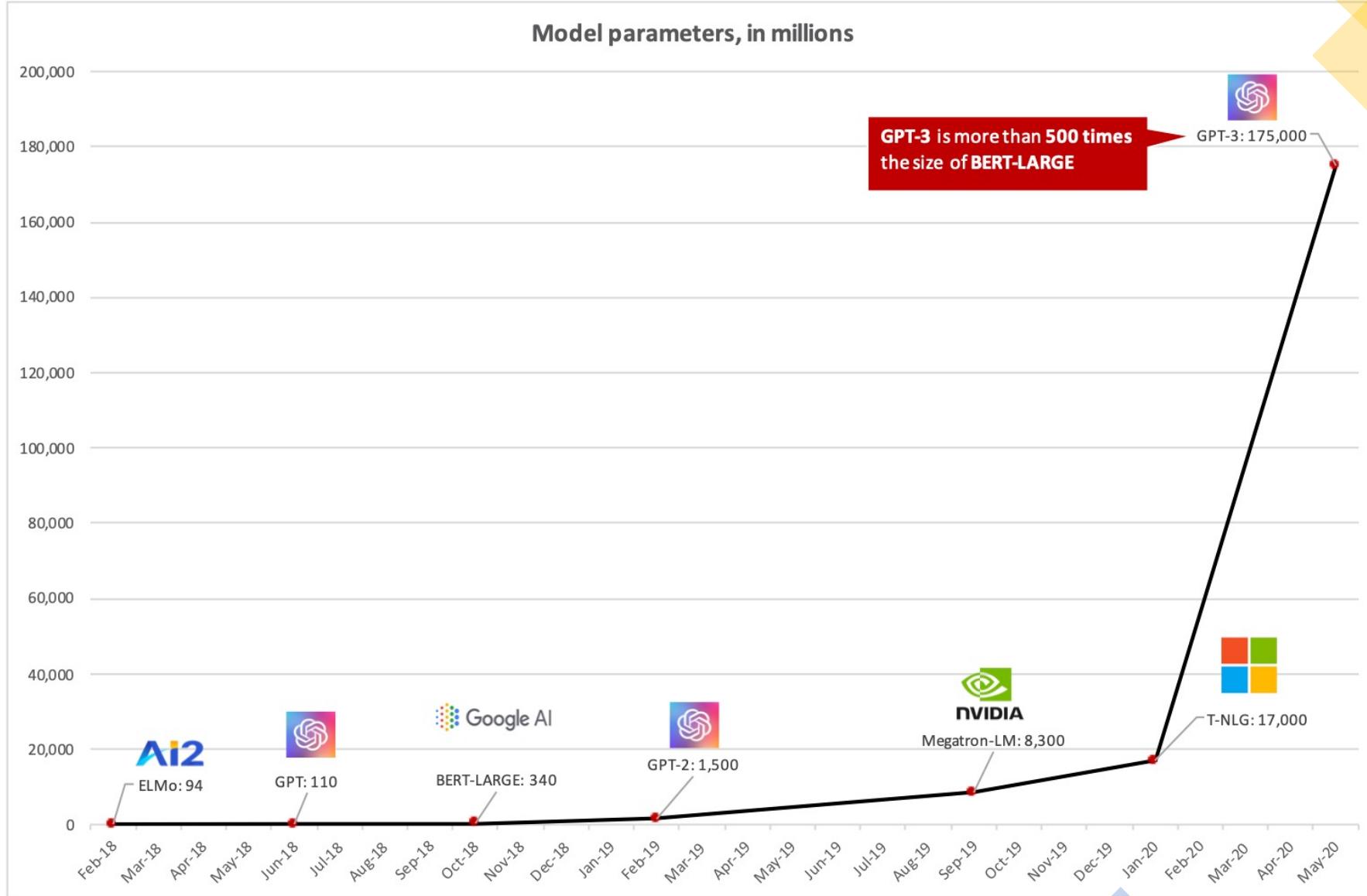
96 layers X 96 attention heads

Trained on a corpus of 300 Billion words

Predicts the next word in a sequence.

Transformer Family





Total Compute Used During Training

Training Petaflop/s-days

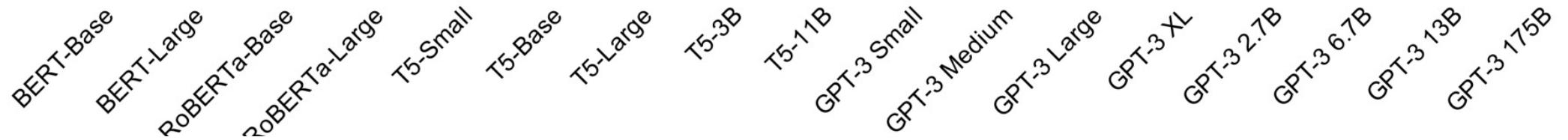
10000

1000

100

10

1



TriviaQA

