Friedrich-Alexander-Universität
Technische Fakultät

Machine Learning
Data Analytics

Machine Learning for Time Series

Dr. Emmanuelle Salin
Machine Learning and Data Analytics (MaD) Lab
Friedrich-Alexander-Universität Erlangen-Nürnberg
16.01.2025

Topics Overview

1. Time series fundamentals and definitions (Part 1)
2. Time series fundamentals and definitions (Part 2)
3. Bayesian Inference and Gaussian Processes
4. State space models (Kalman Filters)
5. State space models (Particle Filters)
6. Autoregressive models
7. Data mining on time series
8. Deep Learning (DL) for Time Series (Introduction to DL)
9. DL – Convolutional models (CNNs)
10. DL – Recurrent models (RNNs and LSTMs)
11. DL – Attention-based models (Transformers)
12. DL – From BERT to ChatGPT
13. DL – New Trends in Time Series processing
14. Time series in the real world

Topics Overview

1. Time series fundamentals and definitions (Part 1)
2. Time series fundamentals and definitions (Part 2)
3. Bayesian Inference and Gaussian Processes
4. State space models (Kalman Filters)
5. State space models (Particle Filters)
6. Autoregressive models
7. Data mining on time series
8. Deep Learning (DL) for Time Series (Introduction to DL)
9. DL – Convolutional models (CNNs)
10. DL – Recurrent models (RNNs and LSTMs)
11. DL – Attention-based models (Transformers)
12. **DL – From BERT to ChatGPT**
13. DL – New Trends in Time Series processing
14. Time series in the real world

Motivation

The Transformer Model

User Input

The log file can be sent secretly with email or FTP to a specified receiver

Transformer Output

Die Protokoll datei kann heimlich per E - Mail oder FTP an einen bestimmten Empfänger gesendet werden .

Large Language Models (LLMs)

User Input

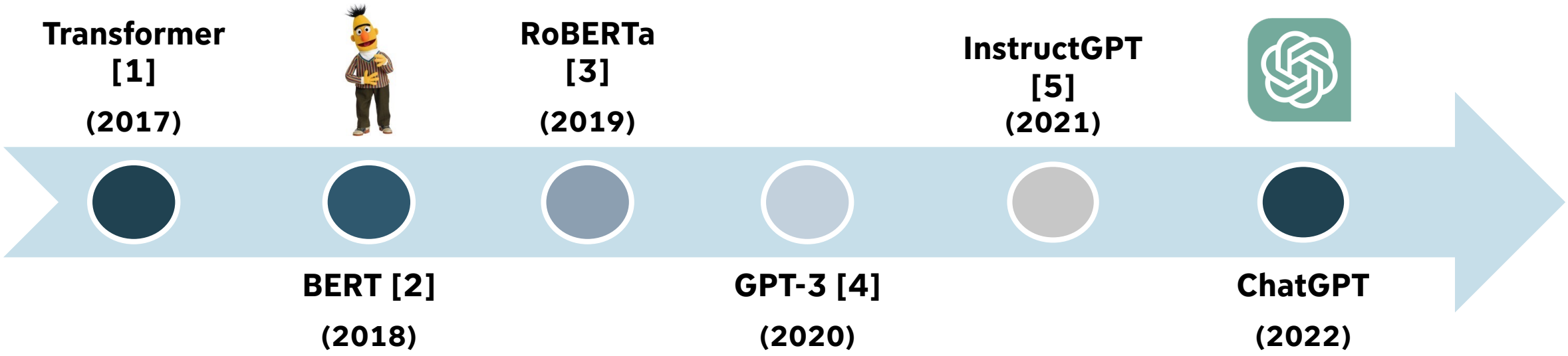
Write a 4 line poem in German

ChatGPT Output

Im Mondenschein, so still und sacht,
Träumt die Welt in tiefer Nacht.
Sterne flüstern leis im Wind,
Dass wir alle Träumer sind.

https://nlp.seas.harvard.edu/2018/04/03/attention.html

Timeline: From BERT to ChatGPT and Beyond



[1] “Attention is all you need”, Vaswani, et al.
[2] "Bert: Pre-training of deep bidirectional transformers for language understanding.” Devlin, et al.
[3] "Roberta: A robustly optimized bert pretraining approach." Liu, et al.
[4] "Language models are few-shot learners." Brown, et al.
[5] "Training language models to follow instructions with human feedback." Ouyang, et al.

In This Lecture...

- **Transformer-based Language Models**
- **Large Language Models**
- **From Large Language Model to Chatbot**
- **Multimodal Large Language Models**
- **Limitations of Large Language Models**

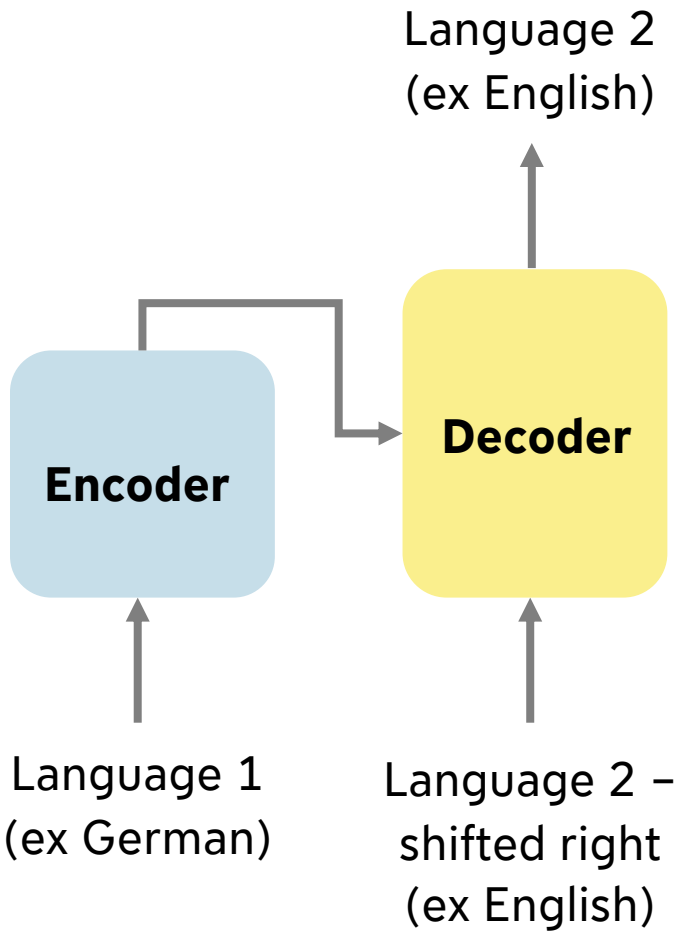
Deep Learning for Time Series – From BERT to ChatGPT and Beyond

Transformer-based Language Models

RECAP: Transformer model

The Transformer [1] model was introduced for translation tasks.

- **Encoder/decoder** architecture
- Built on the **attention** mechanisms (no recurrent architecture)
- Computation can be done in **parallel**
- Better understanding of **long term dependencies**



The Transformer for machine translation

[1] “Attention is all you need”, Vaswani, et al.

Language Models

Natural Language Processing research moved from "one model for one task" to the use of **language models** to encode the language, which can then be **adapted to downstream tasks**.

Language models encode **syntactic** and **semantic** information from words.

This information can then be used in different applications (information extraction, machine translation ...)

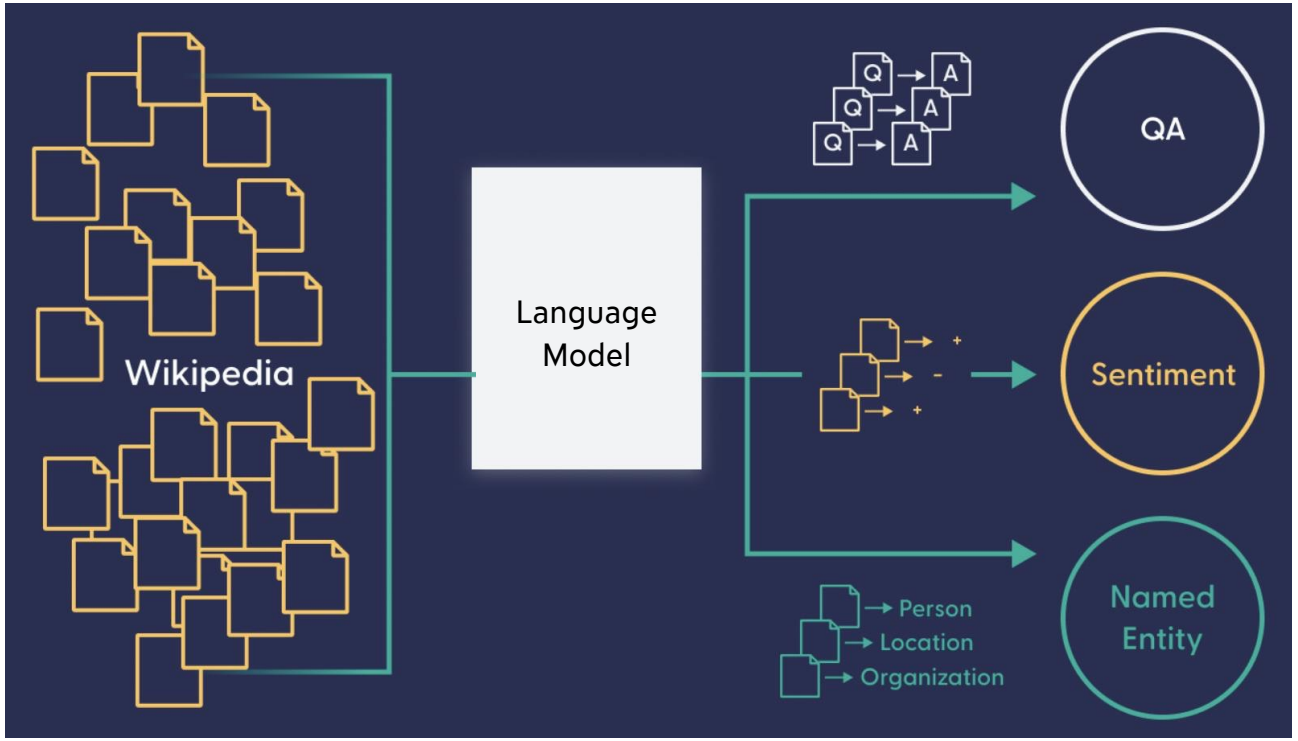


Image from https://www.deepset.ai/blog/what-is-a-language-model

Transformer-based Language Models

The development of the Transformer architecture led to **transformer-based language models**.

They are not trained using a supervised task (e.g. machine translation), but pre-trained on a **self-supervised** task to learn to encode the structure of language (syntax) and the meaning of words (semantic).

Self-supervision commonly means a task whose goal is **to predict any part of the input using any other part.**

Self-supervised tasks

- Predict the future from the past.
- Predict the future from the recent past.
- Predict the past from the present.
- Predict the top from the bottom.
- Predict the occluded from the visible
- Pretend there is a part of the input you don't know and predict that.

From LeCun

Pre-training and Fine-tuning

The goal of **pre-training** is to learn **general representations** that can be **fine-tuned** on **downstream supervised** tasks. In Natural Language Processing, those representations aim to encode linguistic information.

Pre-training is primarily used when **large-scale unlabeled datasets** are at disposal, but less labeled data.

Pre-training

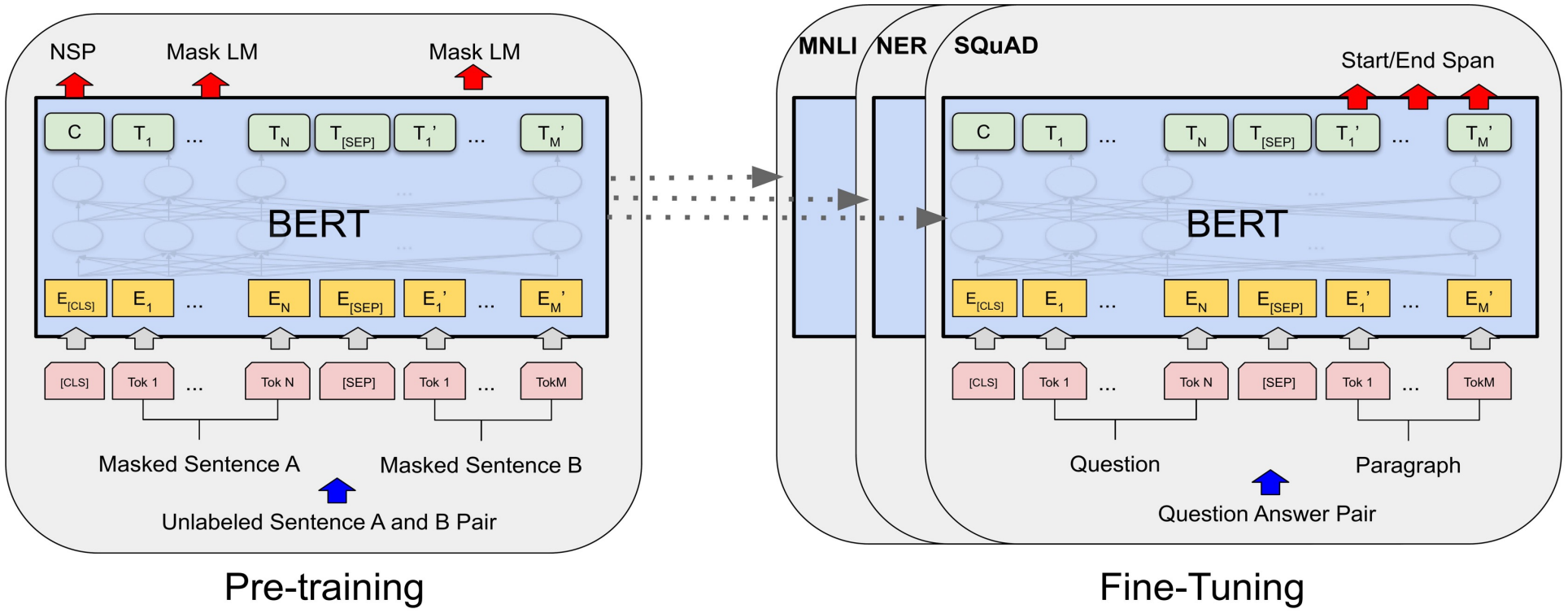
- Large unlabeled datasets
- Self-supervised
- Long training time

Fine-tuning

- Smaller supervised datasets
- Shorter training time

BERT: a Transformer-based Language Model

BERT [1], introduced in 2018, is a **transformer-based** model. It became **state-of-the-art** in many Natural Language Processing tasks.

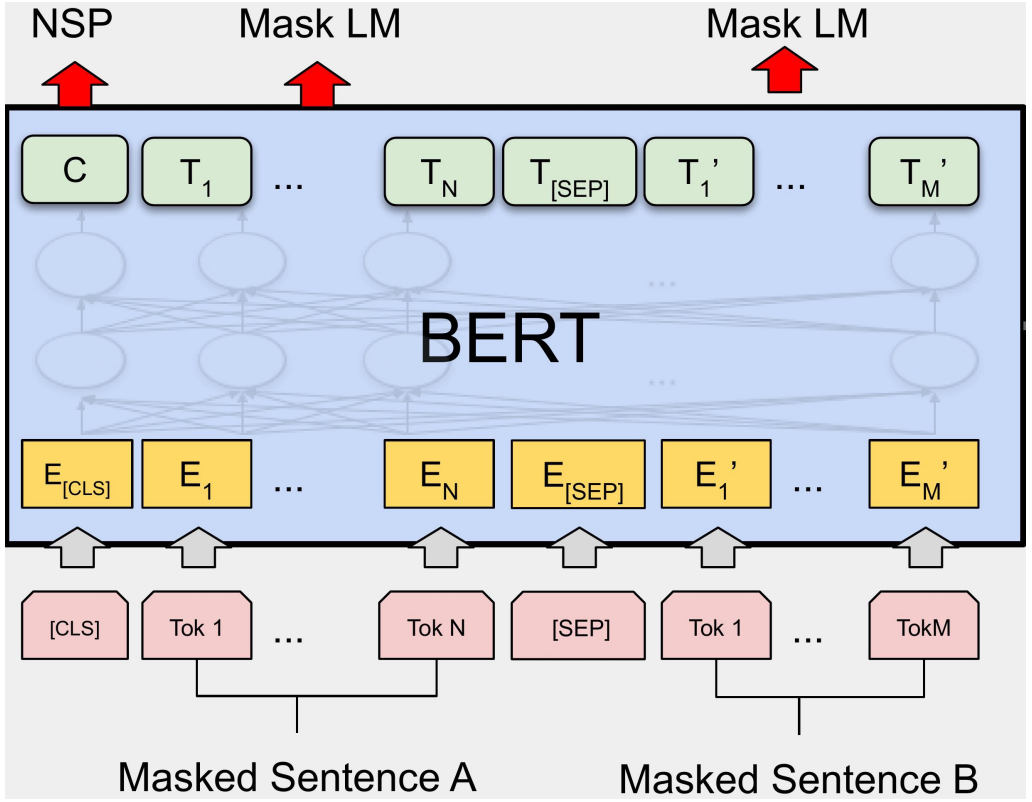


[1] "Bert: Pre-training of deep bidirectional transformers for language understanding." Devlin, et al.

Language Models Pre-training Tasks

Two pre-training tasks are used in BERT:

- **Masked language modeling**
  - 15% of tokens are **masked** and predicted
  - Enables use of **bidirectional** context to predict tokens
- **Next-sentence prediction**
  - A classification token is added to predict if sentence B follows sentence A



[1] "Bert: Pre-training of deep bidirectional transformers for language understanding." Devlin, et al.

Language Models Pre-training Tasks

Language Modeling, also called **auto-regressive** pre-training, is a well known type of self-supervised pre-training.

It is used by **GPT** models.

The goal of this task is to predict the next token in a sequence, using only previous tokens as input.

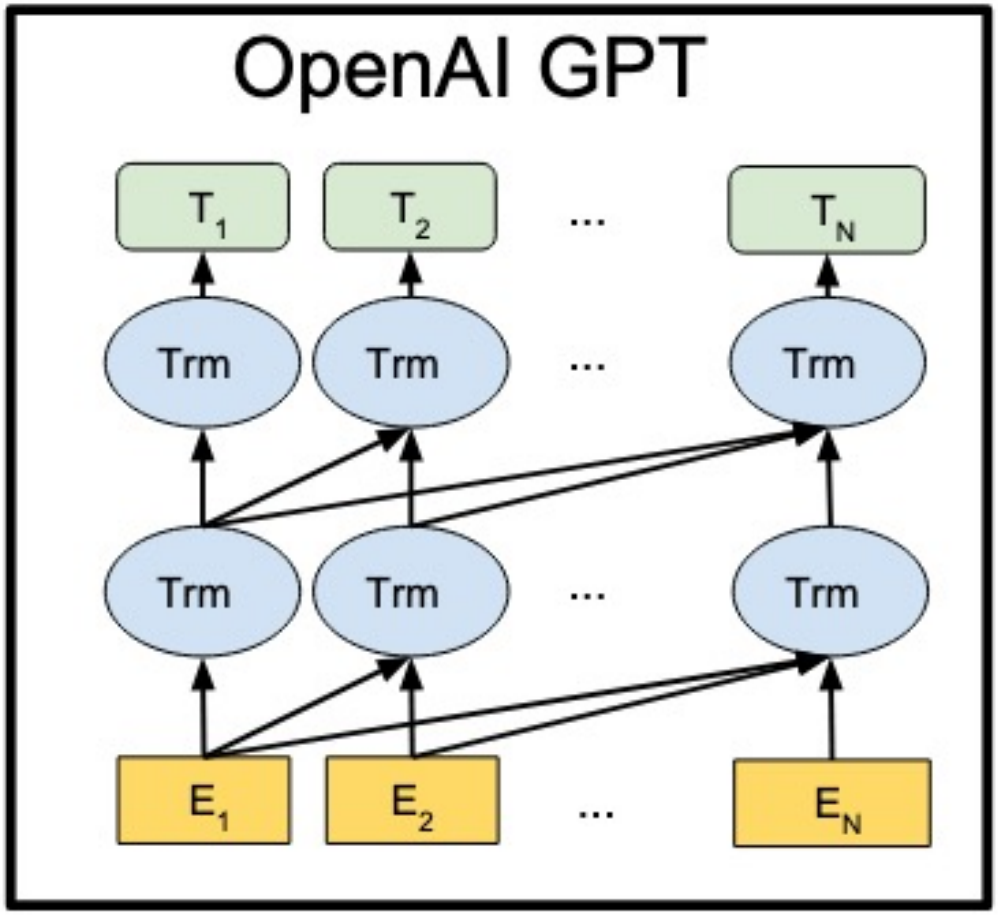


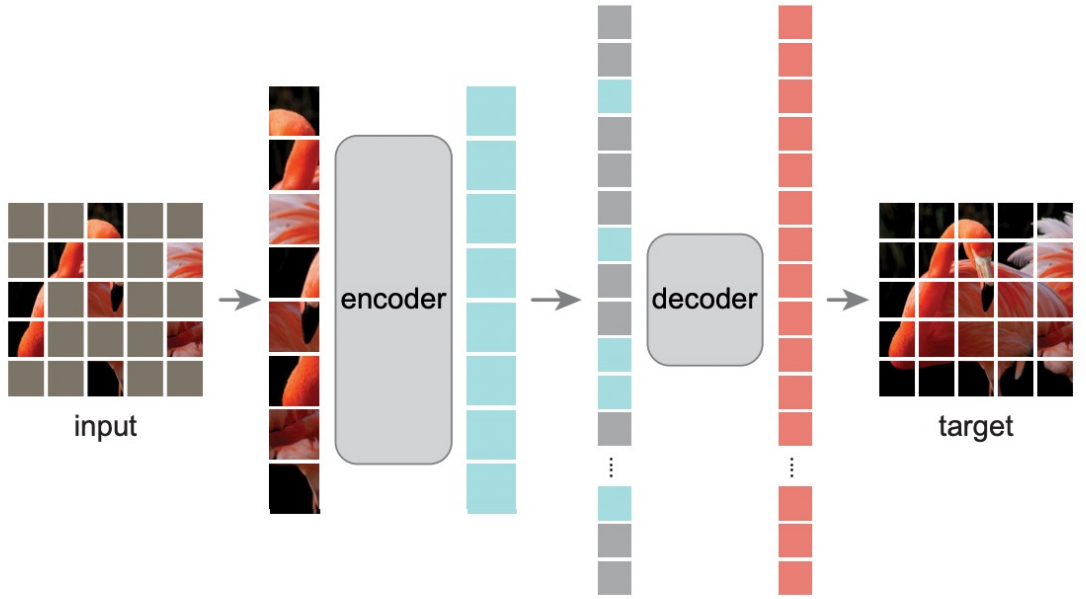
Image from "Bert: Pre-training of deep bidirectional transformers for language understanding." Devlin, et al.

Pre-training beyond Natural Language Processing

With the development of Transformer-based models in other domains, pre-training tasks have been adapted to learn to encode different types of information.

- **Computer Vision**:
  - Masked image modeling
  - Contrastive learning

Example of self-supervision in computer vision



He, Kaiming, et al. "Masked autoencoders are scalable vision learners."

Pre-training beyond Natural Language Processing

With the development of Transformer-based models in other domains, pre-training tasks have been adapted to learn to encode different types of information.

- **Computer Vision**:
  - Masked image modeling
  - Contrastive learning
- **Time Series**:
  - Masked time series modeling
  - Next time step prediction

Example of self-supervision in time series

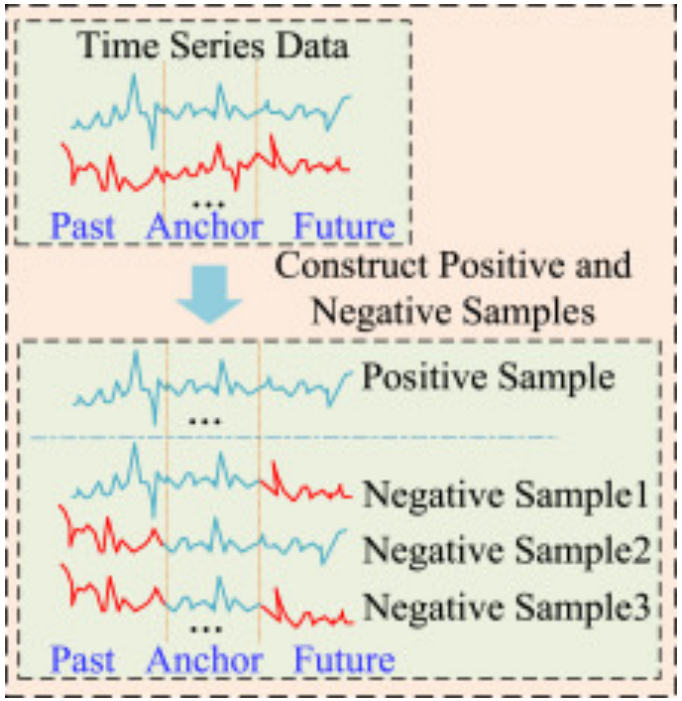
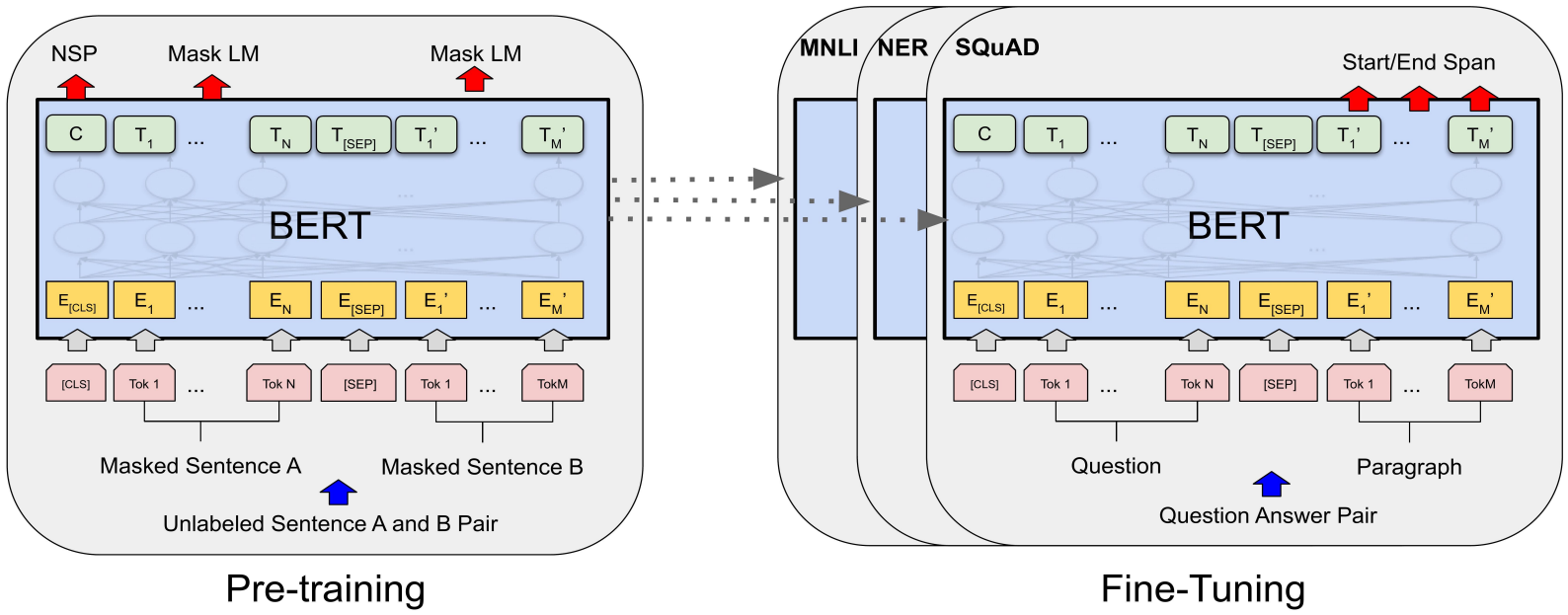


Image from Xi, Liang, et al. "Semi-supervised time series classification model with self-supervised learning."

Fine-tuning Transformer Models

Fine-tuning consists in adapting a **pre-trained** model to a supervised task. Then, one usually requires less **supervised data** and task-specific **training time**.

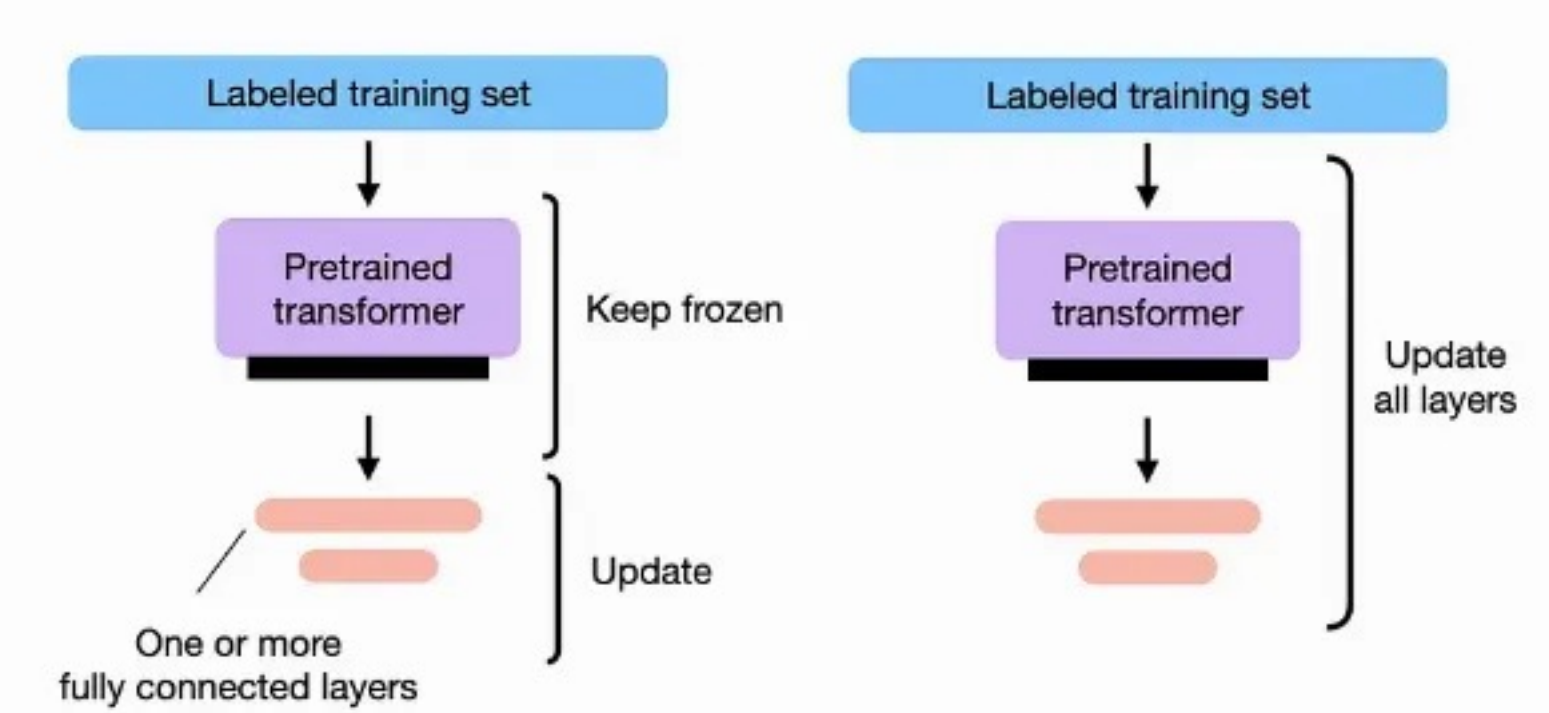


Supervised tasks:

- Natural Language Inference
- Named Entity Recognition
- Question Answering

Fine-tuning in Practice

During fine-tuning, a new **task-specific** loss is optimized on a **labeled** dataset. **Part (or all) of the weights** of the pre-trained model are updated, and **task-specific** layers can be added.



https://www.analyticsvidhya.com/blog/2023/08/fine-tuning-large-language-models/

Evaluating Language Models: Fine-tuning Benchmarks

With the development of pre-trained models, **fine-tuning benchmarks** such as **GLUE** (General Language Understanding Evaluation) have been developed. Their goal is to create a set of **diverse** dataset and tasks to evaluate how a language model performs across all of them.

However, they lose their meaning when nearing **human performance**.

Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP	MN
1	Microsoft Alexander v-team	Turing ULR v6		91.3	73.3	97.5	94.2/92.3	93.5/93.1	76.4/90.9	
2	JDExplore d-team	Vega v1		91.3	73.8	97.9	94.5/92.6	93.5/93.1	76.7/91.1	
3	Microsoft Alexander v-team	Turing NLR v5		91.2	72.6	97.6	93.8/91.7	93.7/93.3	76.4/91.1	

Wang, Alex. "Glue: A multi-task benchmark and analysis platform for natural language understanding." *arXiv preprint arXiv:1804.07461* (2018).

Time Series Downstream Tasks

The goal of pre-training is to encode general information available from the input. In the time series domain, this can include a various range of possible data domains and associated downstream tasks.

- **Biosignals:** ECG, EEG, EMG classification (e.g. arrythmia, seizures)
- **Electronic health record:** Patient outcome classification
- **Stock market :** Stock price forecasting
- **Financial transactions:** Classification, anomaly detection
- **Sensor data:** Anomaly detection, activity recognition

Transformer-based Language Models: Conclusion

Transformer-based language models follow a pre-training/fine-tuning process:

- **Self-supervised pre-training** on a large unlabeled dataset using tasks such as language modelling or masked language modelling.
- **Finetuning** on a smaller labeled dataset (i.e. downstream task)

They can be evaluated and compared to one another through a combination of downstream tasks.

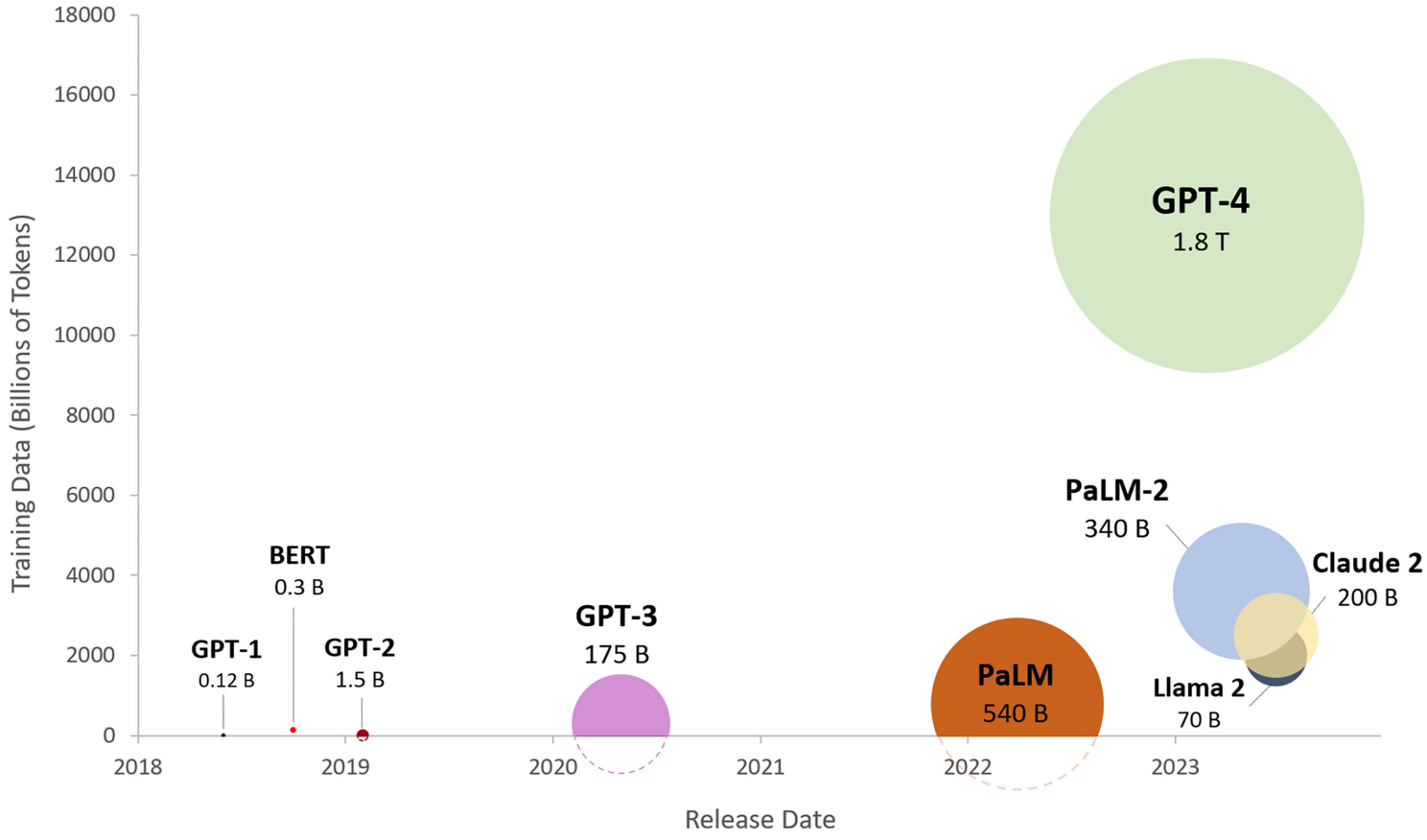
Deep Learning for Time Series – From BERT to ChatGPT and Beyond

Large Language Models

From BERT to GPT-4: a Question of Scale

The size of models and training data has considerably increased from 2018 to 2025.

- BERT was trained on **3.3 billion** words, compared with more than **10 000 billion** tokens for GPT-4
- The GPT-4 model has **6000 times** more parameters than BERT



Bhayana, R. 2024. "Chatbots and Large Language Models in Radiology: A Practical Primer for Clinical and Research Applications." Radiology, RSNA, vol. 310, no. 1, January 16. Accessed 2024-05-25.

Emergent Abilities

Capabilities can arise in models when they scale beyond a certain threshold, called **emergent abilities**. These are not observed in smaller models.

Standard Prompting

Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ✗

Chain of Thought Prompting

Input

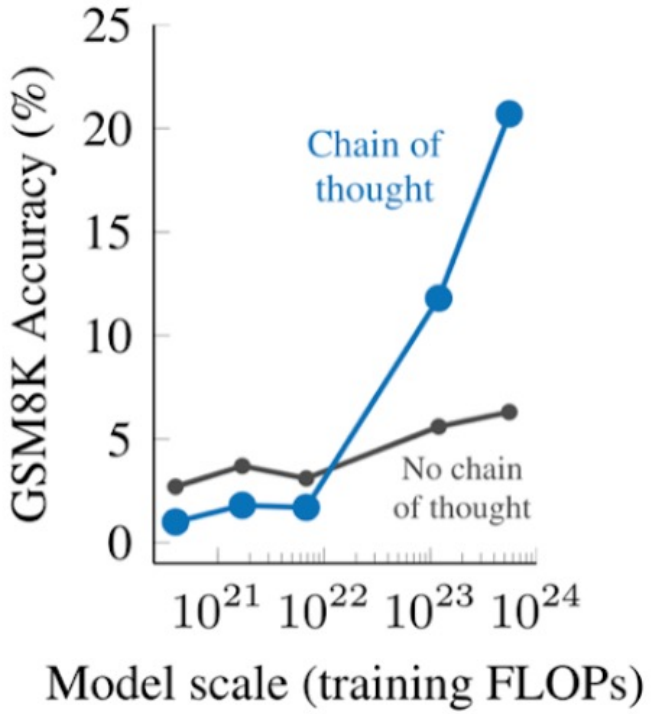
Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓



Example from https://hai.stanford.edu/news/examining-emergent-abilities-large-language-models

Large Language Model Engineering

Recent research has studied how different **training decisions** can affect the **performance** of a transformer model. The main choices are related to:

Architecture

Encoder/decoder,
Model size,
Attention type...

Pre-training Tasks

Auto-regressive loss,
Masked language
modeling...

Training Protocol

Hyperparameters
Optimizers,
Training time...

Data

Dataset quality,
Dataset size,
Data preprocessing...

Impact of Data and Model Size

It has been empirically shown that bigger models reach **better performances** than smaller models with same quantity of data. They also require **longer training time.**

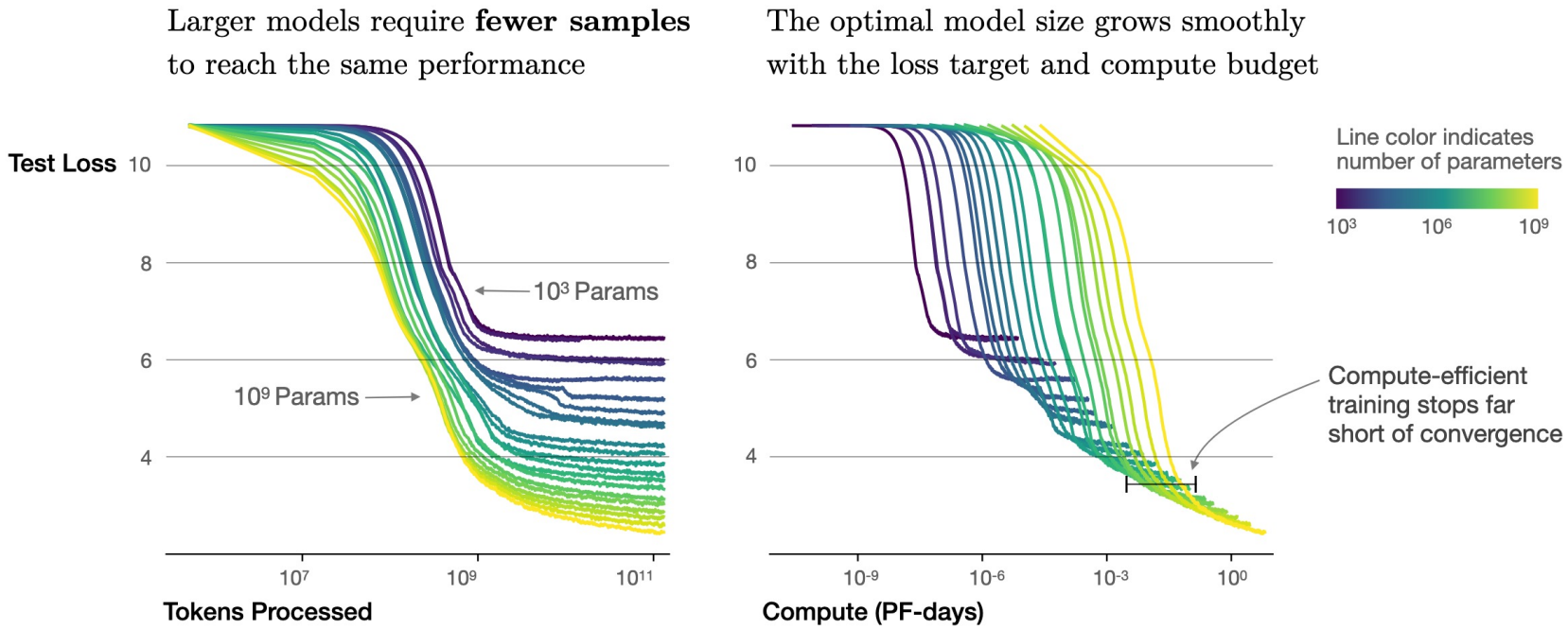


Figure 2 We show a series of language model training runs, with models ranging in size from 10^3 to 10^9 parameters (excluding embeddings).

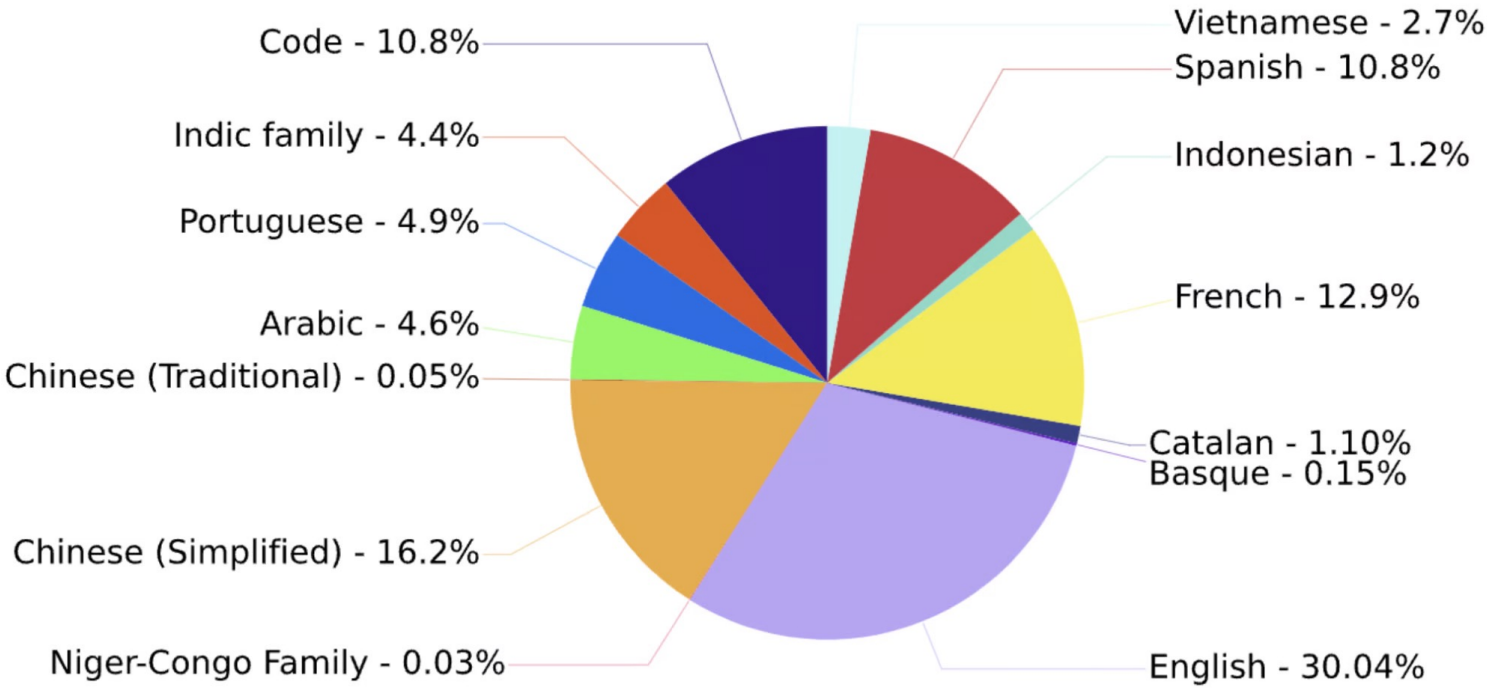
Kaplan, Jared, et al. "Scaling laws for neural language models." *arXiv preprint arXiv:2001.08361* (2020).

Use of Multilingual Datasets

Multilinguality: The ability to process, understand, and generate text in multiple languages. Large Language Models have been trained on multilingual corpora to be applied to multiple languages and cross lingual tasks (e.g. machine translation).

There are multiple challenges:

- Data imbalance
- Tokenization
- Low-resource languages…

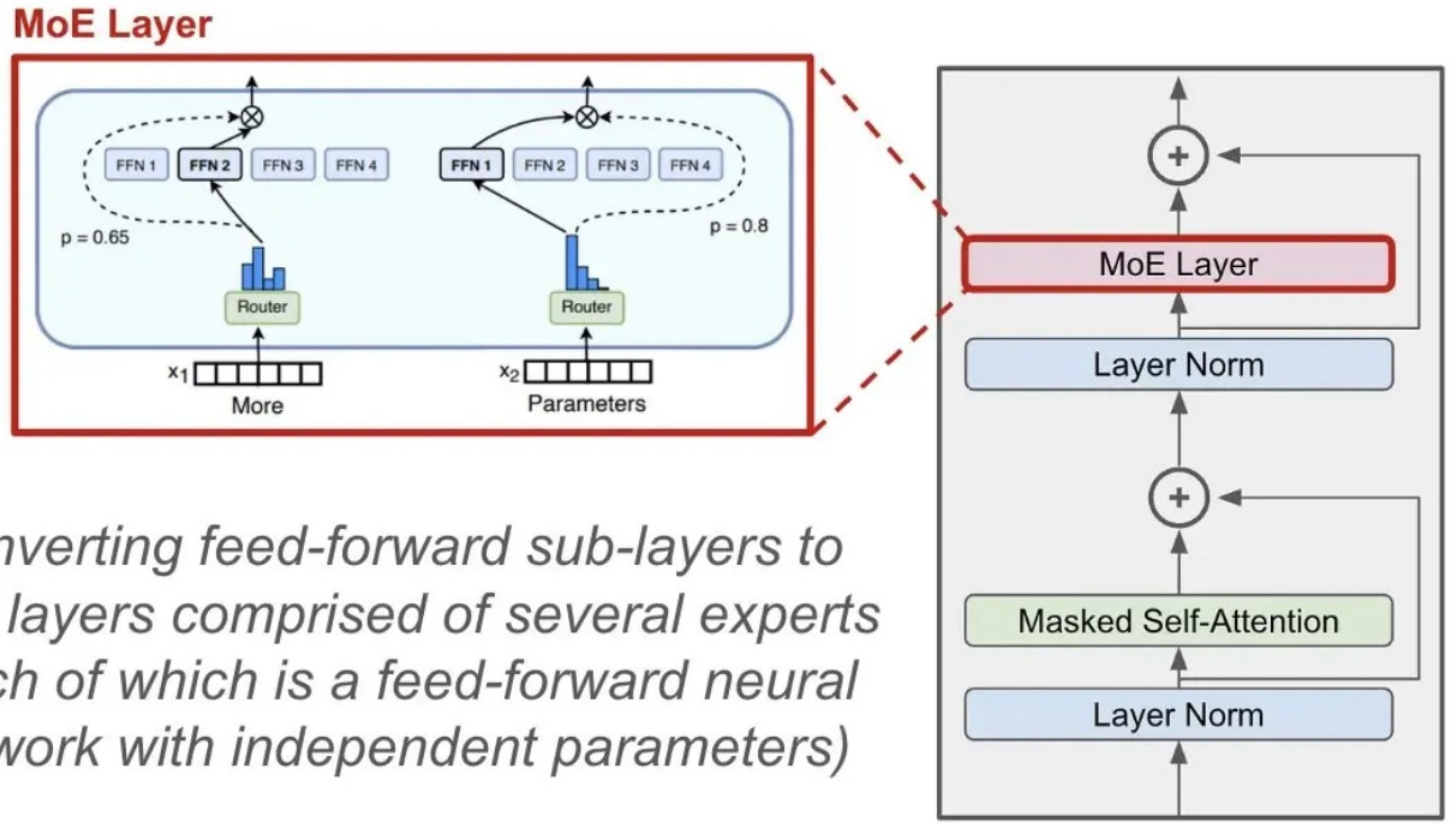


https://huggingface.co/bigscience/bloom

Architectural Innovations: The Example of Mixture of Experts

Mixture of Expert transformer models include two main elements: **sparse Mixture of Expert layers** and a **gate network** or **router**, that determines which tokens are sent to which expert.

They are used to improve the parameter-to-performance ratio of LLMs.



Converting feed-forward sub-layers to MoE layers comprised of several experts (each of which is a feed-forward neural network with independent parameters)

Antoine, Elie, Frédéric Béchet, and Philippe Langlais. "Part-Of-Speech Sensitivity of Routers in Mixture of Experts Models.

Architectural Innovations: The Example of Mixture of Experts

Mixture of Expert transformer models include two main elements: **sparse Mixture**

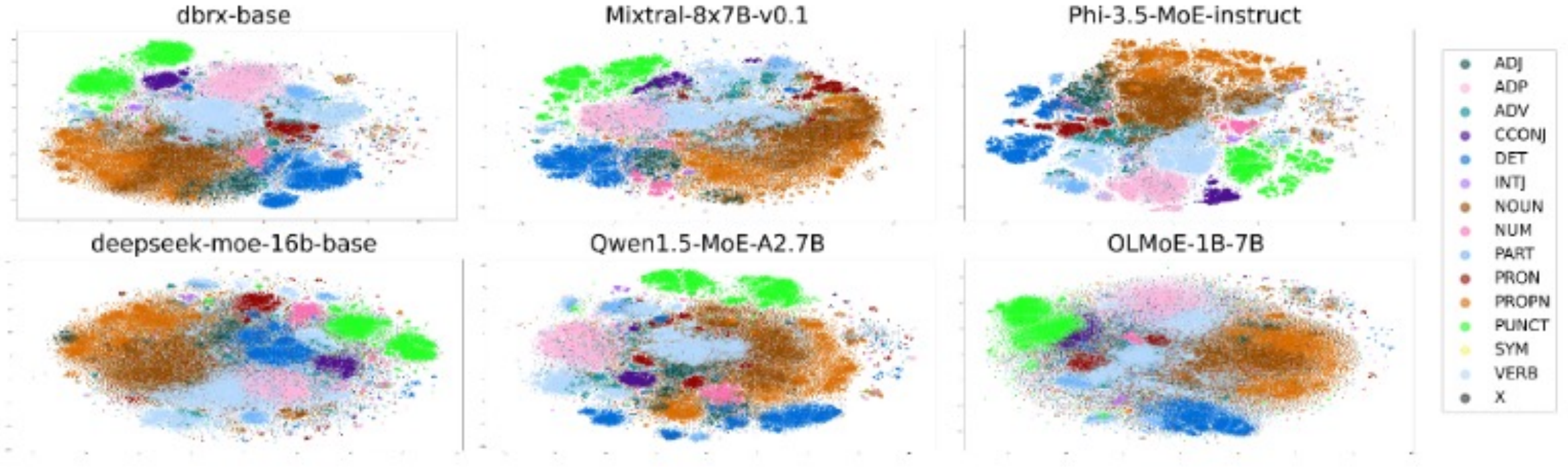


Figure 2: 2D-TSNE projection of token path

Antoine, Elie, Frédéric Béchet, and Philippe Langlais. "Part-Of-Speech Sensitivity of Routers in Mixture of Experts Models.

Efficient Fine-Tuning Methods

However, the inequality of access to compute has lead to techniques such as:

- **Few-shot fine-tuning:** Using only a few examples to fine-tune the model on a specific task
- **Reduced precision training:** using less bits to perform training (e.g., FP16 vs FP32)
- **Parameter efficient fine-tuning** (e.g. LoRa): Freezing most model weights and training low-rank weight matrices called 'adapters'

LoRa fine-tuning

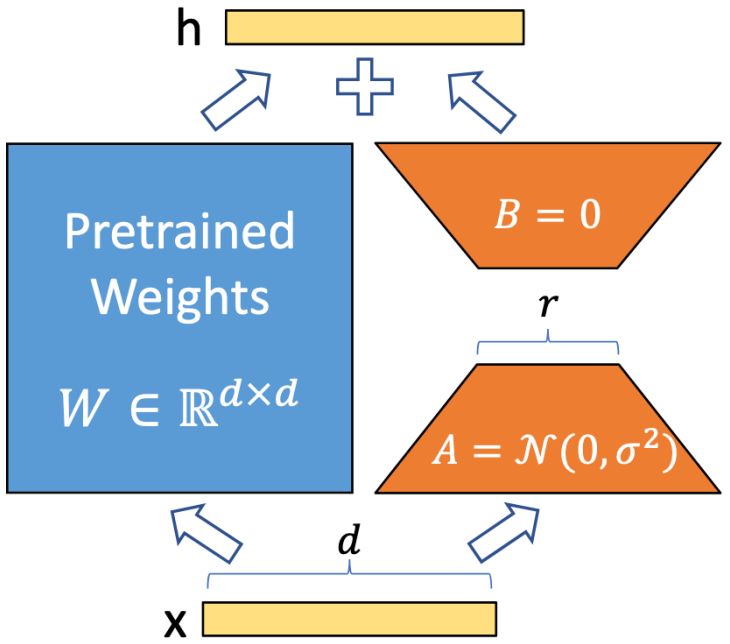


Figure 1: Our reparametrization. We only train A and B.

Hu, Edward J., et al. "Lora: Low-rank adaptation of large language models

Large Language Models: Conclusion

Multiple innovations have led to the improvement of the large language models:

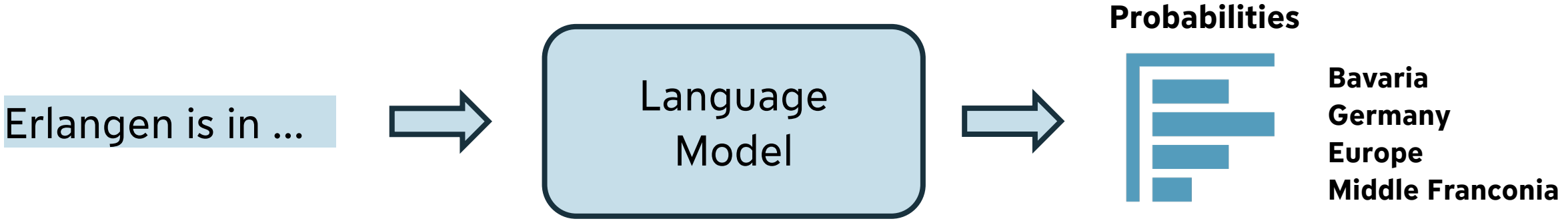
- **Architectural** innovations (e.g. mixture of experts, large models)
- Larger **datasets**, with more diversity (e.g. multilingual datasets)
- Longer **training time** made easier through hardware innovations
- Innovations in the **training protocol** (e.g. parameter efficient fine-tuning)

Deep Learning for Time Series – From BERT to ChatGPT and Beyond

From Large Language Model to Chatbot

Language Models as Text Generation Models

Language Models such as GPT-4 are pre-trained to generate “likely” text.



In text generation, common tokens (e.g. *the, a, an, is*...) are usually overpredicted.

To overcome that, **sampling strategies** have been developed (e.g. greedy sampling, top-k sampling...)

The use of **temperature** can also control the variability of a generated text:

Language Models as Text Generation Models

The use of **temperature** can also control the variability of a generated text:

$$P_i = \frac{e^{\frac{y_i}{T}}}{\sum_{k=1}^{n} e^{\frac{y_k}{T}}}$$

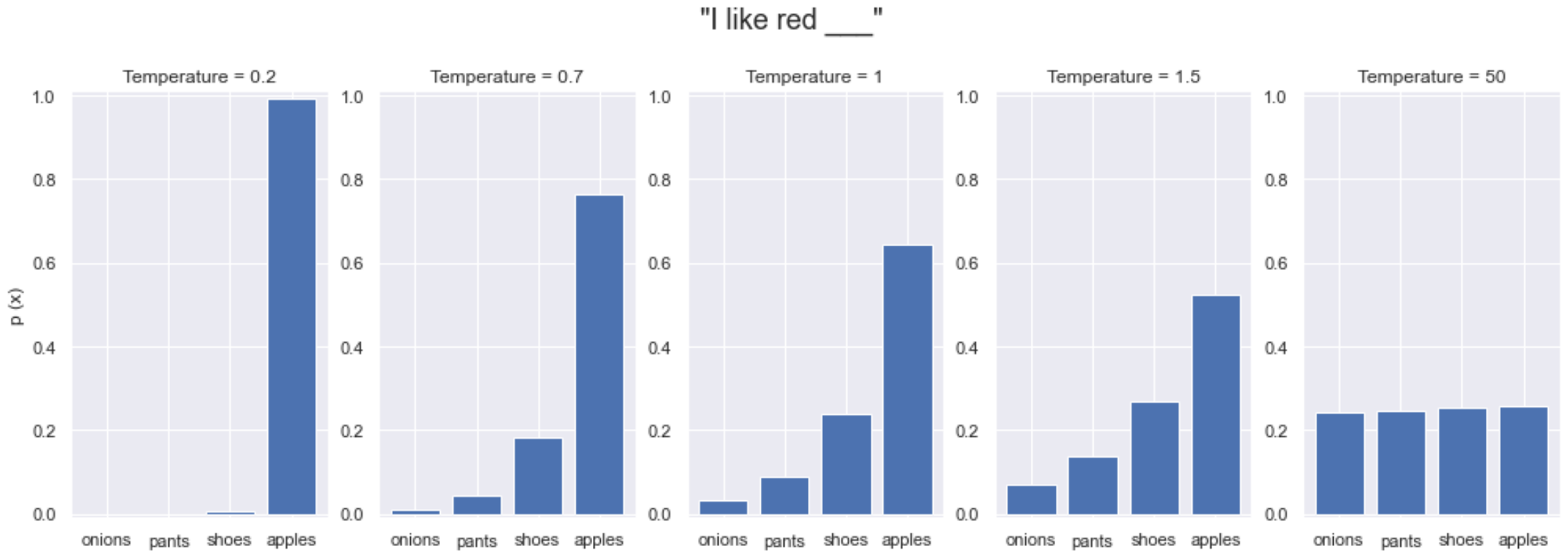
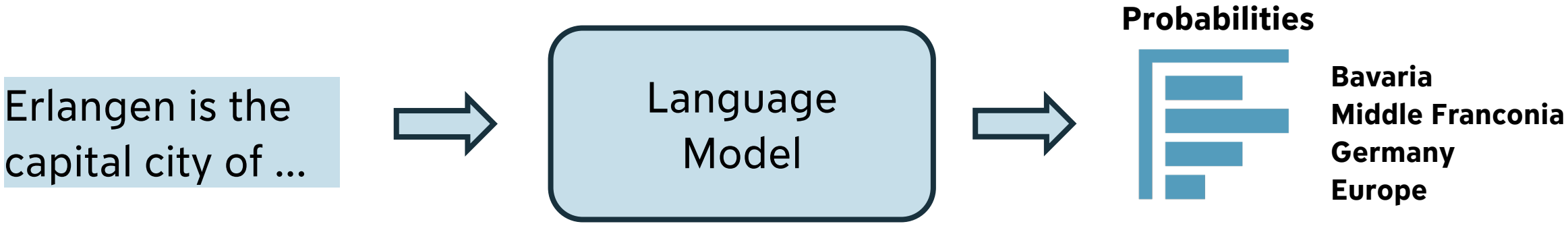


Image from https://medium.com/@imisri1/how-to-set-sampling-temperature-for-gpt-models-762887df1fac

Limits of Language Models as Chatbots

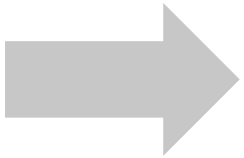
Language Models are pre-trained on **language modelling** types of tasks, which has several drawbacks in the use case of chatbots.



Limits of Language Models as Chatbots

A chatbot should:

- Obey instructions (creators or users)
- Be truthful
- Limit bias
- Limit toxicity
- Generalize to unseen situations



All the more complicated because those properties can be contradictory

Reinforcement Learning with Human Feedback

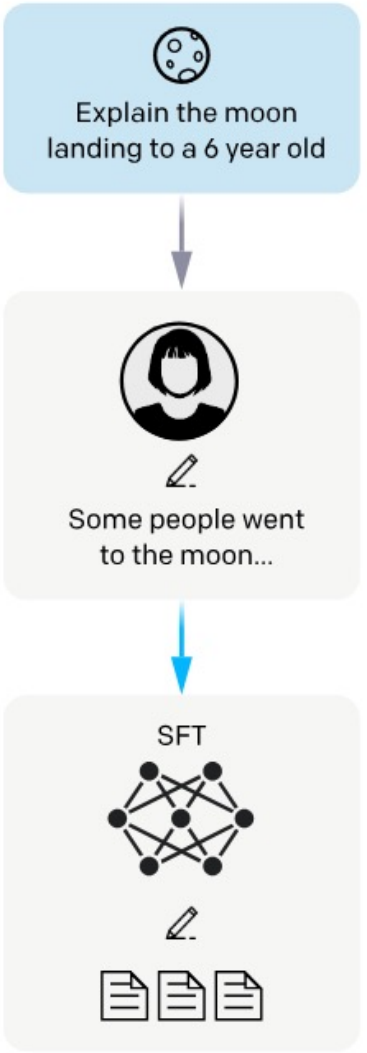
In order improve the **quality** of generated text (i.e. align with human preferences), OpenAI introduced InstructGPT, or **reinforcement learning with human feedback**.

1. Collect output data (i.e., answers) and train supervised model

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



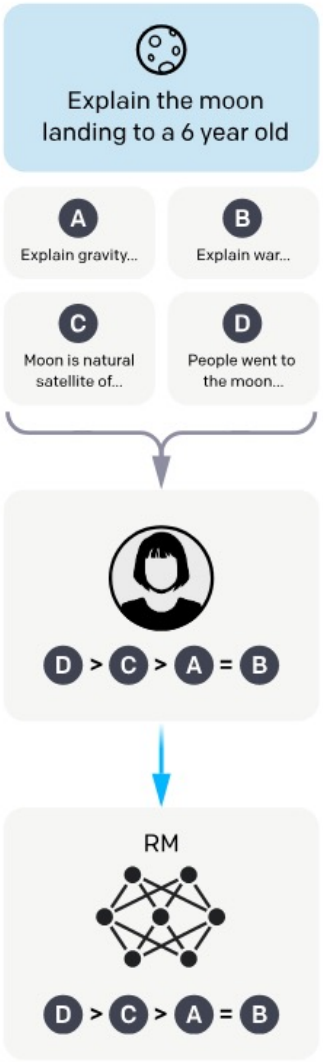
Images from Ouyang, Long, et al. "Training language models to follow instructions with human feedback." *Advances in neural information processing systems* 35 (2022): 27730-27744.

Reinforcement Learning with Human Feedback

In order improve the **quality** of generated text (i.e. align with human preferences), OpenAI introduced InstructGPT, or **reinforcement learning with human feedback**.

1. Collect output data (i.e., answers) and train supervised model
2. Collect ranking data, and train reward model

A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.

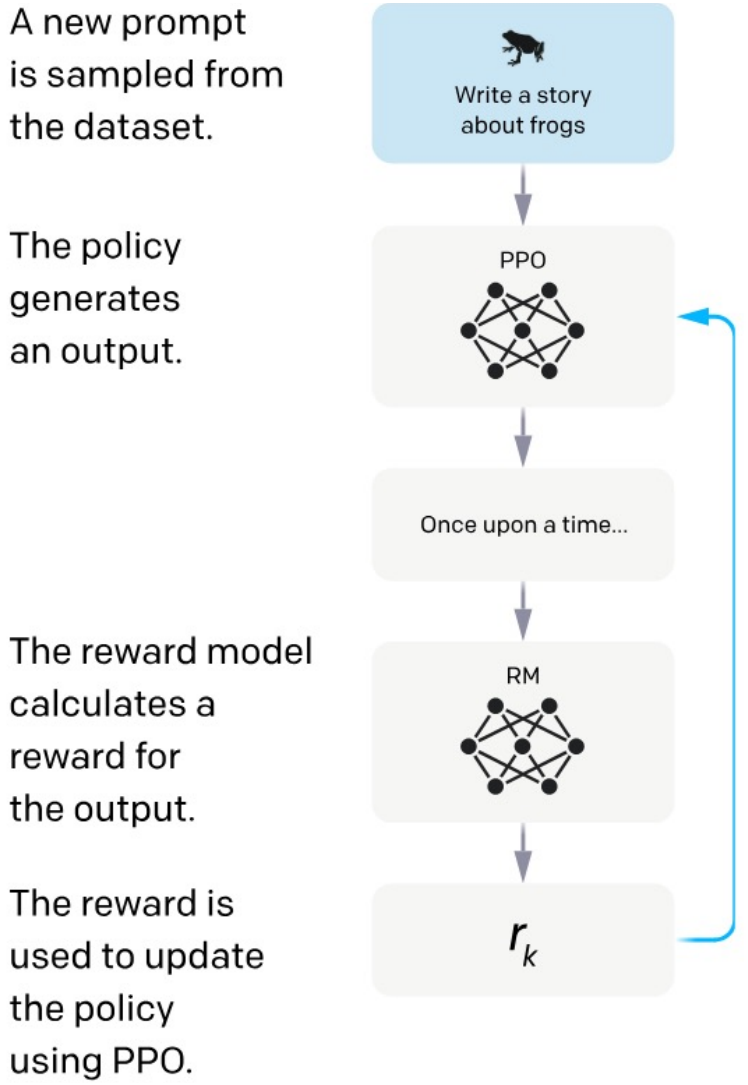
This data is used to train our reward model.

Images from Ouyang, Long, et al. "Training language models to follow instructions with human feedback." *Advances in neural information processing systems* 35 (2022): 27730-27744.

Reinforcement Learning with Human Feedback

In order improve the **quality** of generated text (i.e. align with human preferences), OpenAI introduced InstructGPT, or **reinforcement learning with human feedback**.

1. Collect output data (i.e., answers) and train supervised model
2. Collect ranking data, and train reward model
3. Optimize policy against reward model using reinforcement learning



Images from Ouyang, Long, et al. "Training language models to follow instructions with human feedback." *Advances in neural information processing systems* 35 (2022): 27730-27744.

ChatGPT

Since ChatGPT is **not** an open-source model, we cannot be sure what is its precise prediction process. However, based on similar open-source models, we can hypothesize that it integrates:

- A "**moderation**" layer, which filters harmful content: in the input before prediction, and in the output before delivering it.
- **Bias** detection and mitigation techniques
- A **mixture of experts** model
- Integration of **dialogue** and other **context** processing methods

Evaluating Generated Text

Automated metrics have been developed to evaluate text generation (BLEU, ROUGE). They measure overlap of n-grams between **generated and reference text**.

However, those metrics do not consider **semantic similarity** of words, which is a drawback in many text generation tasks (translation, summarization ...).

In Natural Language Processing, the most accurate way to evaluate text generation models remains **human evaluation**.

Newer metrics were developed based on language models, that measure the **semantic similarity** by computing contextual embeddings from pre-trained language models. **BERTscore** is one of them.

Eval

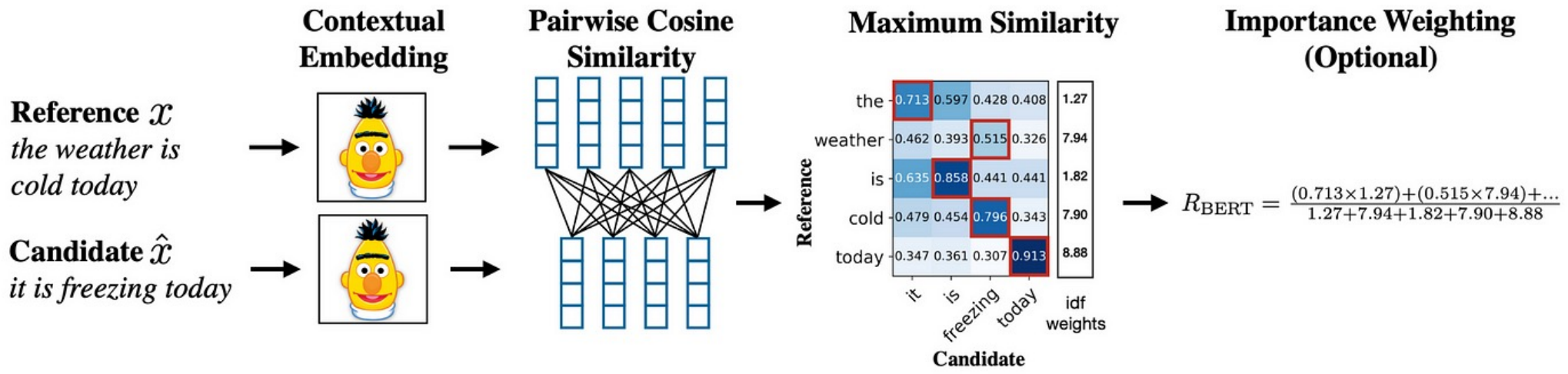
Auto
ROU

How
draw

In Na
mod

New
sema
langu

Introducing **BERTScore**



Source: Bertscore: Evaluating text generation with bert

Code for Bertscore is available at https://github.com/Tiiiger/bert score

Deep Learning for Time Series – From BERT to ChatGPT and beyond
Multimodal Large Language Models

Multimodality

Modality: the way in which an information is captured

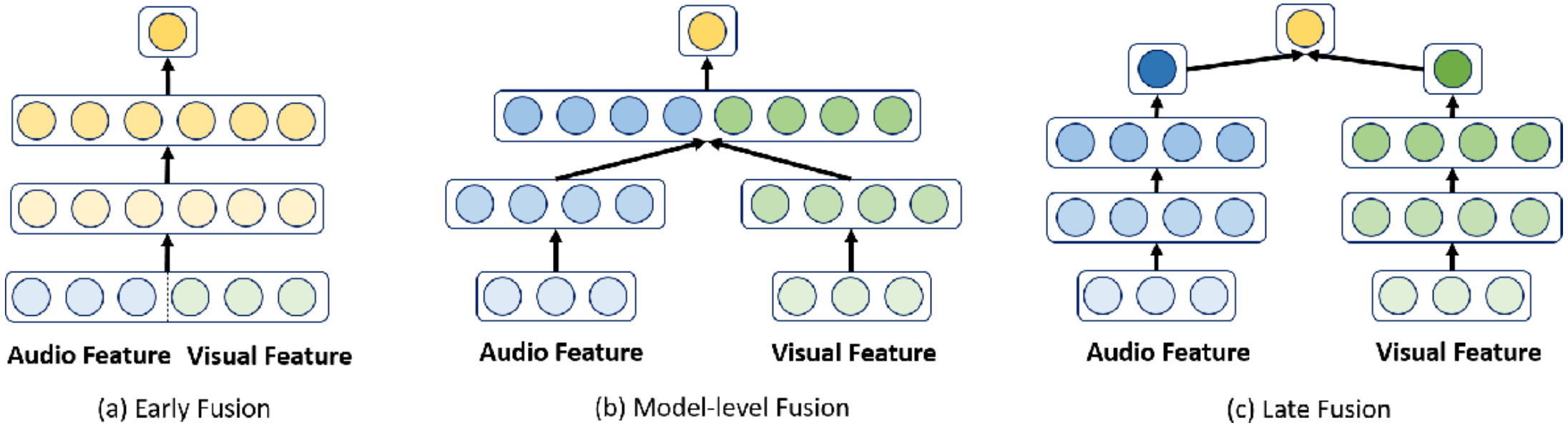
Text-only language models are not sufficient to represent **human language**, as language is inherently **multimodal** (human senses, other sensors).

This has led to the development of **multimodal** transformer models, designed to process encode **multiple modalities.**

Multimodal Transformers

There are various types of multimodal models (differences in architecture, data, training process...). One main difference is how they merge multiple modalities:

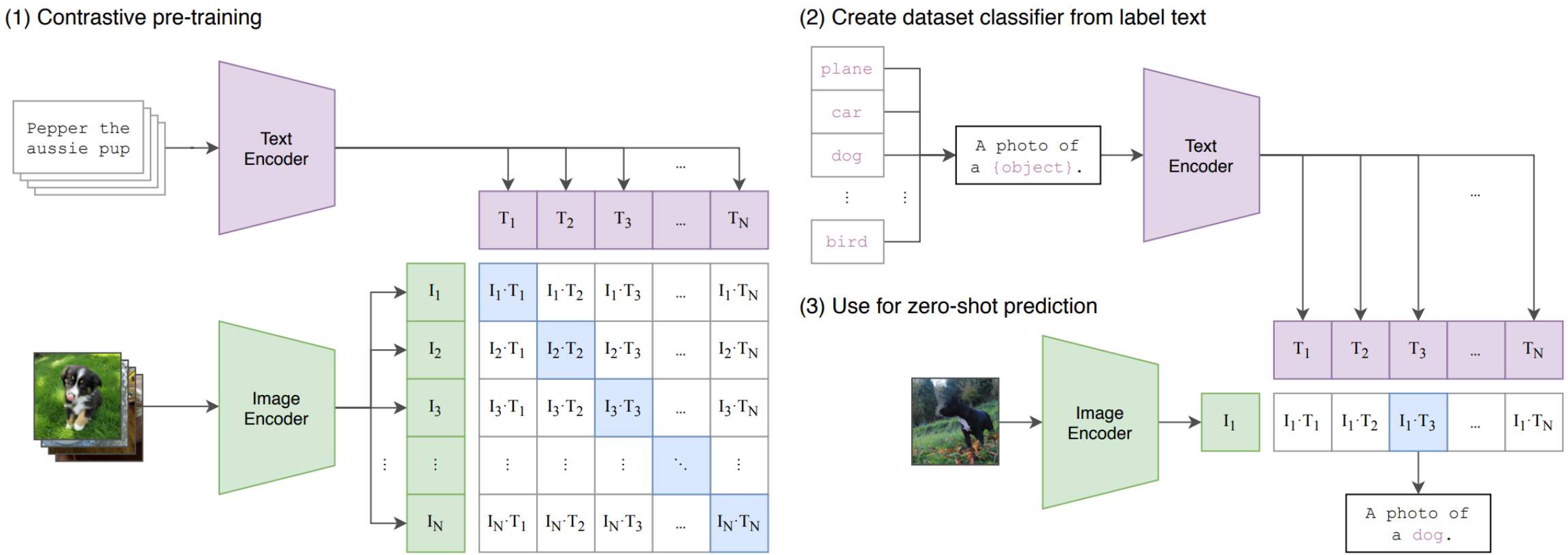
- **Late fusion**: Each modality is processed in its own model.
- **Early fusion:** the same model includes both image and text tokens as input.



Chen, Shizhe, and Qin Jin. "Multi-modal conditional attention fusion for dimensional emotion prediction."

Vision-Language Model: the Case of CLIP

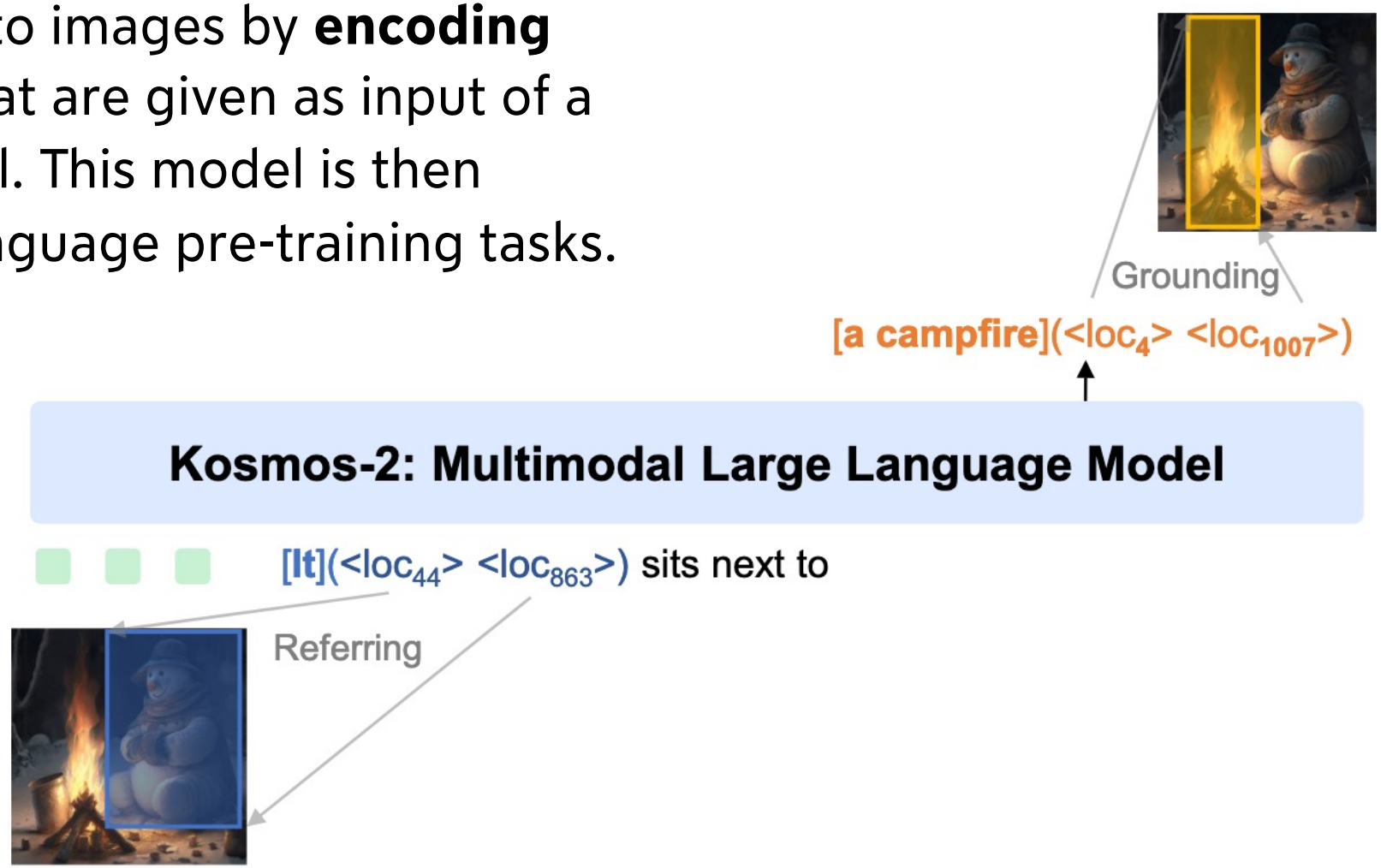
CLIP model processes each modality in a separate model, and aligns image and text representations in a **shared embedding space**.



Radford, Alec, et al. "Learning transferable visual models from natural language supervision." *International conference on machine learning*. PMLR, 2021.

Vision-Language Model: the Case of Kosmos

Kosmos grounds text to images by **encoding** images into vectors that are given as input of a Large Language Model. This model is then finetuned on vision-language pre-training tasks.



Peng, Zhiliang, et al. "Kosmos-2: Grounding multimodal large language models to the world." *arXiv preprint arXiv:2306.14824* (2023).

Training Multimodal Models

There are various types of multimodal LLMs, but they require two main components:

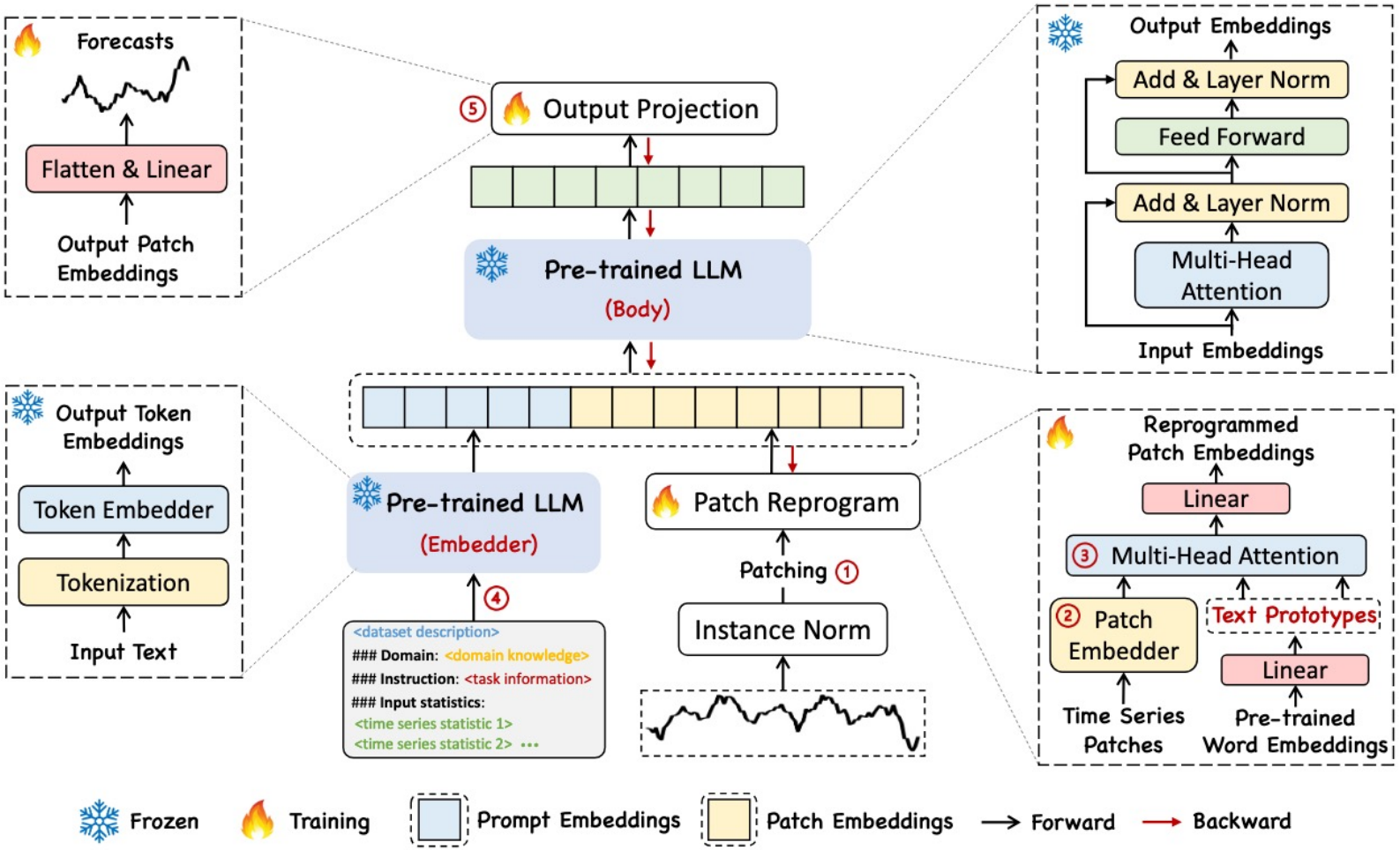
- **Large multimodal paired dataset:** This can consist in image and captions, or images embedded in web pages. The quality of the model will depend on how fine-grained the dependencies between text and images are.

- **Multimodal pre-training tasks:** A pre-training objective consisting on predicting elements of one modality based on another, or classifying between matching or non matching pairs.

Multimodal LLMs for Time Series

Building LLMs for time series can follow similar approaches. Time-LLM is an **early fusion** model which:

- Encodes the time series into patch embeddings
- Uses forecasting as training task



Jin, Ming, et al. "Time-llm: Time series forecasting by reprogramming large language models."

Deep Learning for Time Series – From BERT to ChatGPT and Beyond

Limitations of Large Language Models

Environmental Impacts

Large Language Models, because of their high training and inference costs, also have high levels of **water** and **energy consumption**.

This can lead to significant **environmental impacts.**

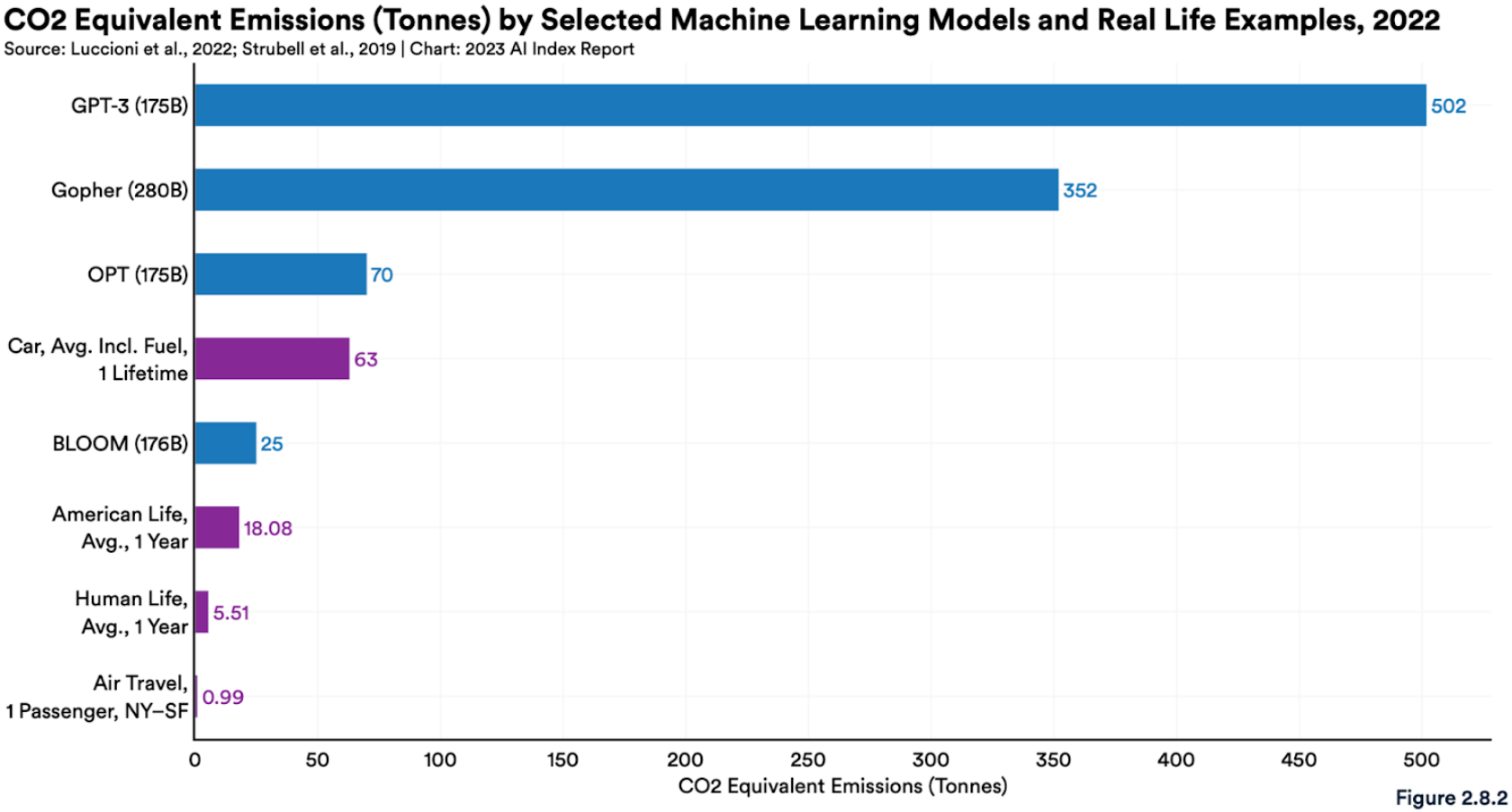


Figure 2.8.2

ChatGPT is much less **energy efficient** than a search engine.

Transformers and Information Retrieval

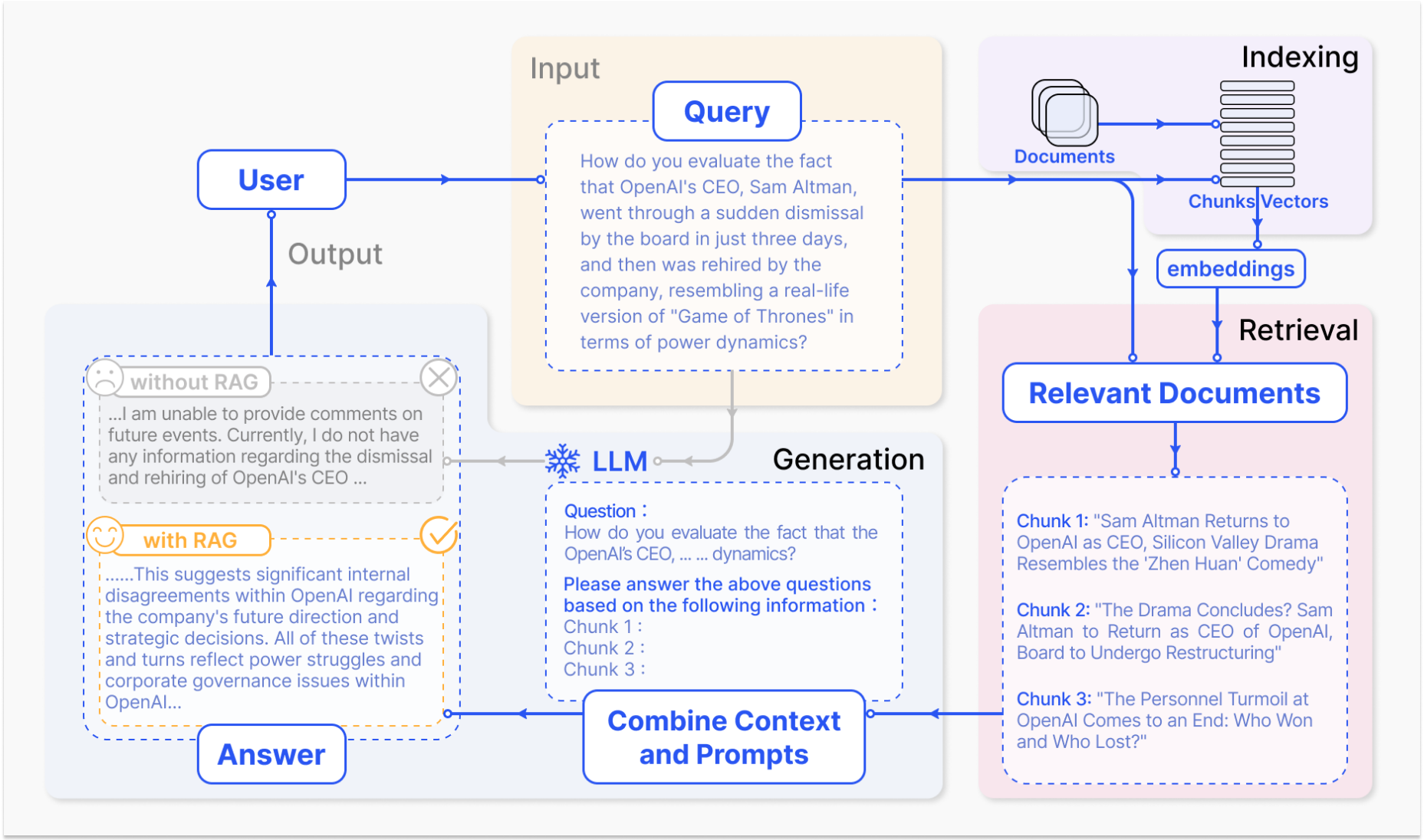
Transformer models are not appropriate for all text related tasks. Indeed, they are inherently not well suited for **information retrieval** (i.e. search engine), as they tend to generate the most **probable** outputs. They:

- Are not necessarily **truthful**
- Will not show a **diversity** of answers

However, transformer models can be adapted to information retrieval through the task of **retrieval augmented generation.**

Transformers and Information Retrieval

An example of **retrieval augmented generation**.

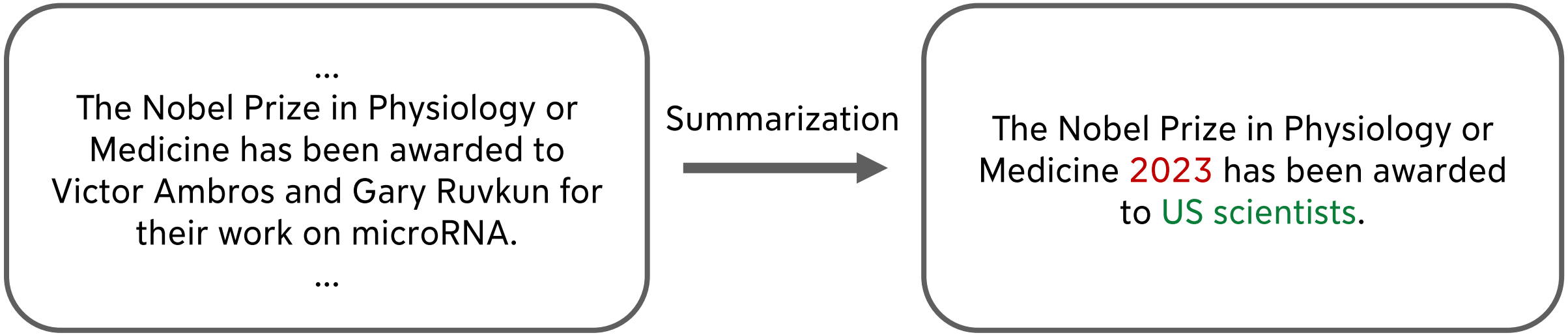


Gao, Yunfan, et al. "Retrieval-augmented generation for large language models: A survey." *arXiv preprint arXiv:2312.10997* (2023).

Hallucinations

When generating text, transformers predict a probable output token. However, in the case of tasks such as **summarization**, where output is highly constrained to input, they exhibit **hallucinations.**

In the case hallucinations in summarization tasks, transformer models generate text based on information (true or false) that is **not present in the input**.



Bias in Large Language Models

Transformer models trained on will reproduce **bias** found in their training data. This requires careful evaluation of fairness and bias to limit representational harm.

For instance, CLIP finds this sentence:

> This is a portrait of smiling housewife in an orange jumpsuit with the American flag

More similar to the image than this one:

> This is a portrait of an astronaut with the American flag

Bias **mitigation** techniques can be used to limit bias during data **pre-processing**, **training** and **post-processing**.

Image from Birhane, Abeba, Vinay Uday Prabhu, and Emmanuel Kahembwe. "Multimodal datasets: misogyny, pornography, and malignant stereotypes."

Data Diversity: the Example of Translation

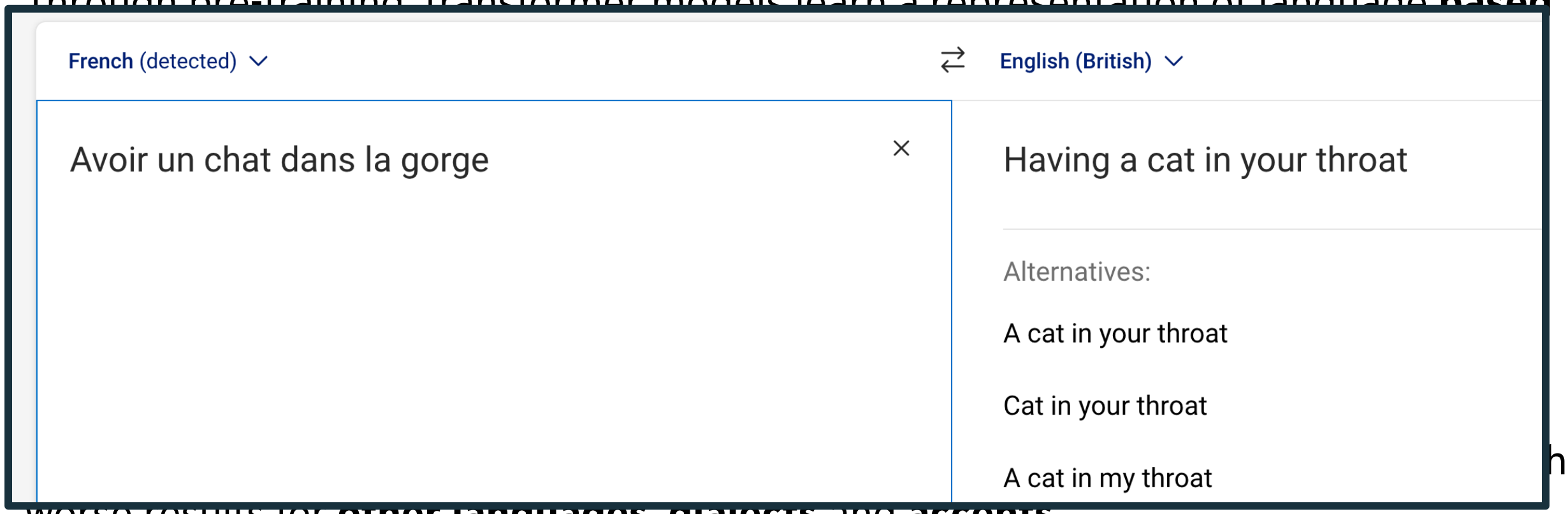
Through pre-training, transformer models learn a representation of language **based on their input data** (usually, English-speaking internet), which can bias the performances, in particular in terms of:

- **Resource scarce languages**
- Underrepresentation of other **perspectives**

Indeed, while English text is available in great quantities on the internet, this is not the case of all languages. As a result**, machine translation** models may exhibit much worse results for **other languages**, **dialects** and **accents.**

Data Diversity: the Example of Translation

Through pre-training, transformer models learn a representation of language **based**



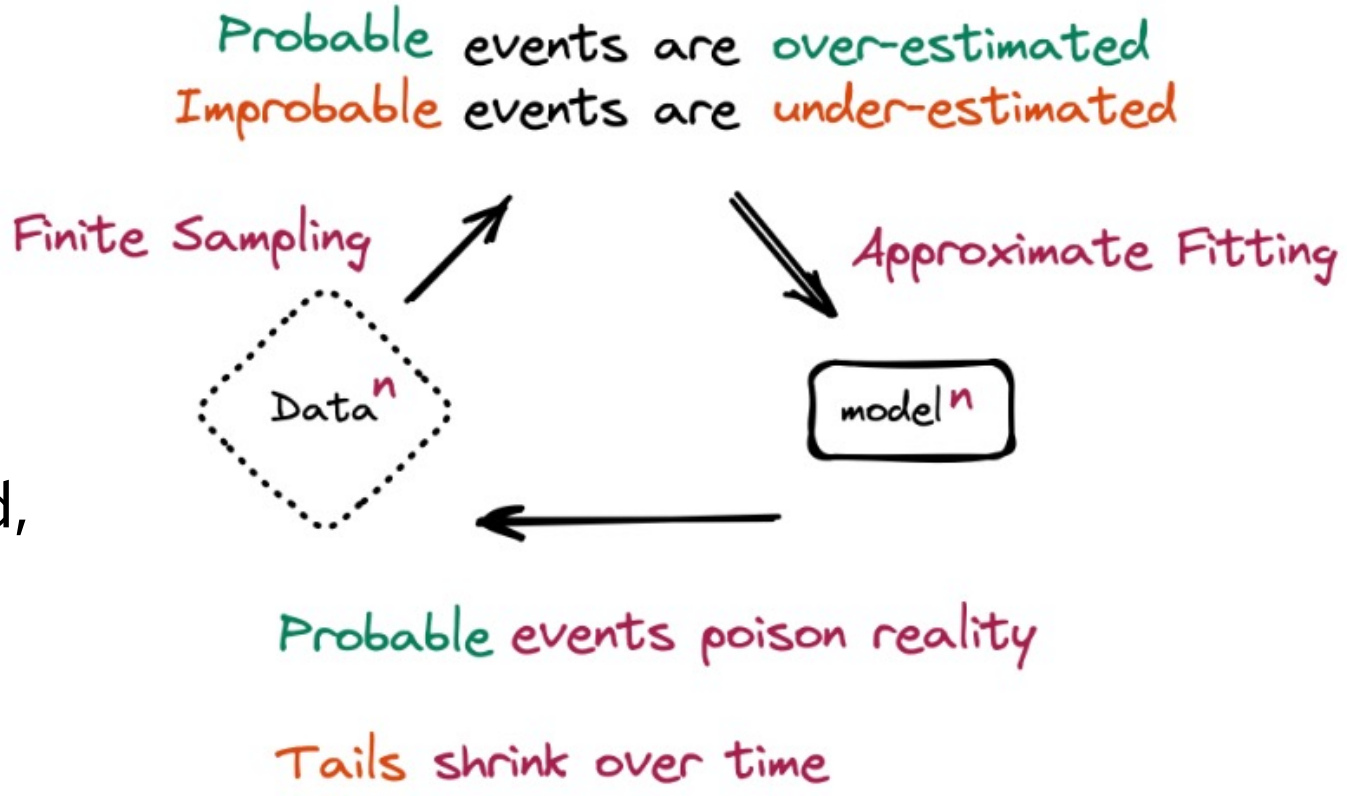
worse results for **other languages**, **dialects** and **accents.**

From Deepl

Data Diversity: Training on Generated Data

One problem that has become apparent is the fact that a lot of the data published post-ChatGPT on the internet has been **automatically** generated.

This raises concerns of data quality, especially in terms of diversity. Indeed, probable texts become **increasingly more** probable, erasing less probable ones form the input data.



Shumailov, Ilia, et al. "The curse of recursion: Training on generated data makes models forget." *arXiv preprint arXiv:2305.17493* (2023).

Limitations of Large Language Models: Conclusion

Despite their groundbreaking performances in many domains, LLMs have many limitations:

- Very **resource consuming** in training and during their use
- Lack of ability to **verify facts** (e.g. for information retrieval, hallucinations)
- **Biased** due to the training process and data (lack of diversity in the data)

In addition, the size of those models make it very difficult to explain the processes of those models (e.g. emergent abilities)

Deep Learning for Time Series – From BERT to ChatGPT and Beyond

Recap

In This Lecture

- **Transformer-based Language Models**
- **Large Language Models**
- **From Large Language Model to Chatbot**
- **Multimodal Large Language Models**
- **Limitations of Large Language Models**