From Transformers to Self-Instructing networks

Roberto Basili
Machine Learning, Web Mining & Retrieval 2022/2023

Outline

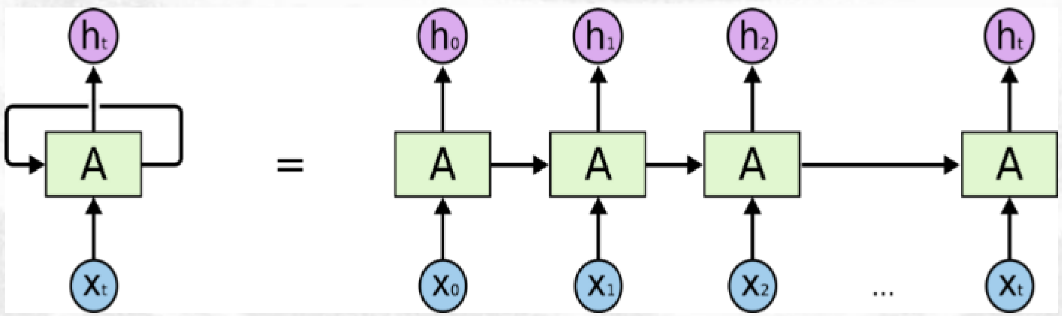
- Trasformers Recap
- Textual Inference and Other applications
- Attention Mechanisms in Encoder-Decoder architectures
- The zero or Few shot learning paradigm
- From Decoder-Only architectures to ChatGPT
  - Instructing LLMs
  - A reward model for Instructions
- Applications and Perspectives

Making Language Modeling the basis for Artificial Intelligence

- Complex NN architectures are Modular
  - Enconding architectures as BERT can be seen as the basis for complex NL Inference tasks
    - Paraphrase Detection
    - Textual Entailment
  - Stacking Dense Layer is a form of «compositional» mechanism (see Framenet in Logical approaches in NLU)
- Large Language Models capture
  - Morphologic
  - Syntactic
  - Semantic phenomena
- as a basis for consistent NLU, reasoning and generation
- Larger language models seem to exhibit stronger generalization capabilities

Machine learning paradigms underlying ChatGPT

RNNs
1986

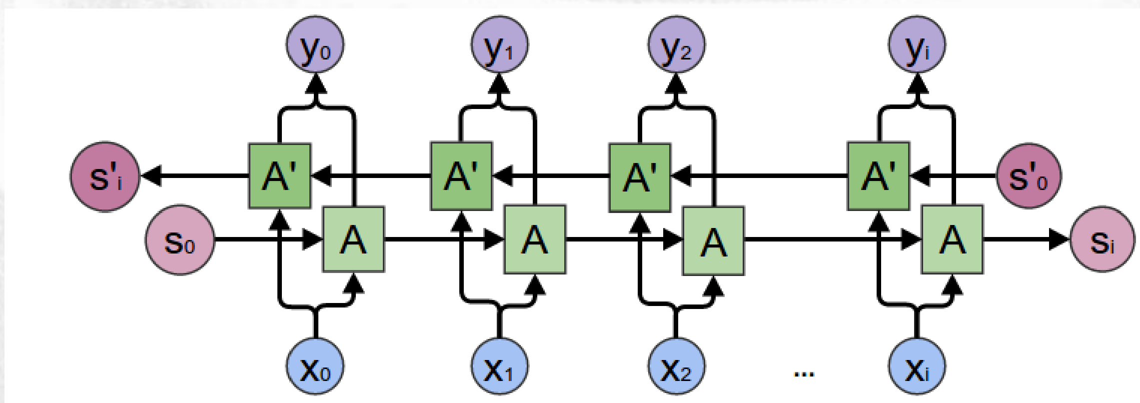


Williams, Ronald J.; Hinton, Geoffrey E.; Rumelhart, David E. (October 1986).

Machine learning paradigms underlying ChatGPT

RNNs
1986

Bidirectional
RNNs
1997



Schuster, Mike, and Kuldip K. Paliwal. 1997

Examples: Language understanding

https://github.com/Microsoft/CNTK/wiki/Hands-On-Labs-Language-Understanding

Task: Slot tagging with an LSTM

```
1 |# BOS            |# O
1 |# show           |# O
1 |# flights        |# O
1 |# from           |# O
1 |# burbank        |# B-fromloc.city_name
1 |# to             |# O
1 |# st.            |# B-toloc.city_name
1 |# louis          |# I-toloc.city_name
1 |# on             |# O
1 |# monday         |# B-depart_date.day_name
1 |# EOS            |# O
```

Examples: language understanding

https://github.com/Microsoft/CNTK/wiki/Hands-On-Labs-Language-Understanding

Task: Slot tagging with an LSTM

```
19   |x 178:1 |# BOS      |y 128:1 |# O
19   |x 770:1 |# show     |y 128:1 |# O
19   |x 429:1 |# flights  |y 128:1 |# O
19   |x 444:1 |# from     |y 128:1 |# O
19   |x 272:1 |# burbank  |y 48:1  |# B-fromloc.city_name
19   |x 851:1 |# to       |y 128:1 |# O
19   |x 789:1 |# st.      |y 78:1  |# B-toloc.city_name
19   |x 564:1 |# louis    |y 125:1 |# I-toloc.city_name
19   |x 654:1 |# on       |y 128:1 |# O
19   |x 601:1 |# monday   |y 26:1  |# B-depart_date.day_name
19   |x 179:1 |# EOS      |y 128:1 |# O
```

```
     ^
     |
 +-------+
 | Dense |
 +-------+
     ^
     |
 +------
>| LSTM |-
 +------+
     ^
     |
 +-------+
 | Embed |
 +-------+
     ^
```

Examples: language understanding

https://github.com/Microsoft/CNTK/wiki/Hands-On-Labs-Language-Understanding

Task: Slot tagging with an LSTM

```
19  |x 178:1 |# BOS       |y 128:1 |# O
19  |x 770:1 |# show      |y 128:1 |# O
19  |x 429:1 |# flights   |y 128:1 |# O
19  |x 444:1 |# from      |y 128:1 |# O
19  |x 272:1 |# burbank   |y 48:1  |# B-fromloc.city_name
19  |x 851:1 |# to        |y 128:1 |# O
19  |x 789:1 |# st.       |y 78:1  |# B-toloc.city_name
19  |x 564:1 |# louis     |y 125:1 |# I-toloc.city_name
19  |x 654:1 |# on        |y 128:1 |# O
19  |x 601:1 |# monday    |y 26:1  |# B-depart_date.day_name
19  |x 179:1 |# EOS       |y 128:1 |# O
```

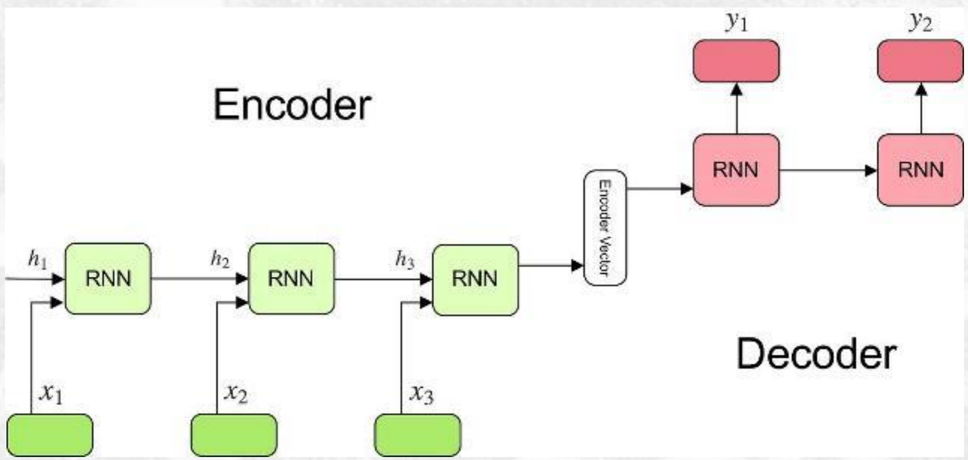
```
y        "O"        "O"        "O"        "O"   "B-fromloc.city_name"
          ^          ^          ^          ^          ^
          |          |          |          |          |
     +-------+  +-------+  +-------+  +-------+  +-------+
     | Dense |  | Dense |  | Dense |  | Dense |  | Dense |  ...
     +-------+  +-------+  +-------+  +-------+  +-------+
          ^          ^          ^          ^          ^
          |          |          |          |          |
     +------+   +------+   +------+   +------+   +------+
0 -->| LSTM |-->| LSTM |-->| LSTM |-->| LSTM |-->| LSTM |-->...
     +------+   +------+   +------+   +------+   +------+
          ^          ^          ^          ^          ^
          |          |          |          |          |
     +-------+  +-------+  +-------+  +-------+  +-------+
     | Embed |  | Embed |  | Embed |  | Embed |  | Embed |  ...
     +-------+  +-------+  +-------+  +-------+  +-------+
          ^          ^          ^          ^          ^
          |          |          |          |          |
x ------>+--------->+--------->+--------->+--------->+------...
         BOS      "show"    "flights"    "from"   "burbank"
```

Machine learning paradigms underlying ChatGPT

RNNs
1986

Bidirectional RNNs
1997

Encoder-Decoder RNNs
2014



I. Sutskever, O. Vinyals, & Q.V. Le, 2014

Machine learning paradigms underlying ChatGPT

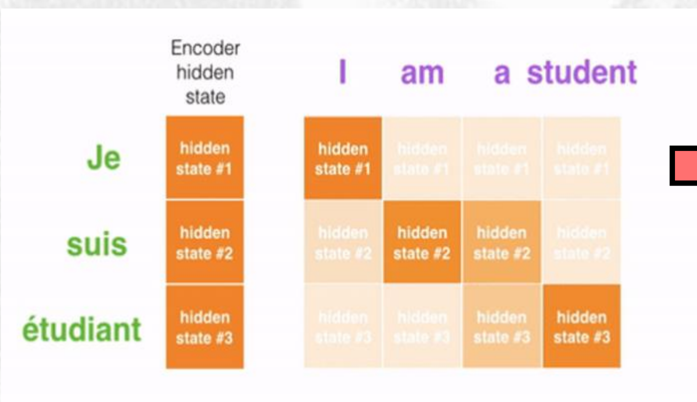
RNNs
1986

Bidirectional
RNNs
1997

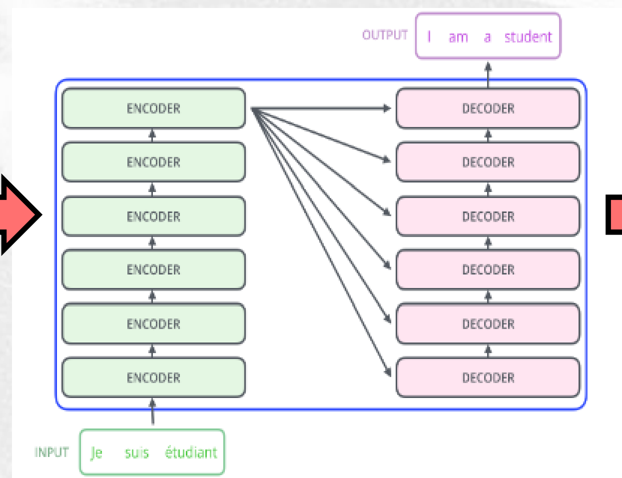
Encoder-Decoder
RNNs
2014

Transformers
2017

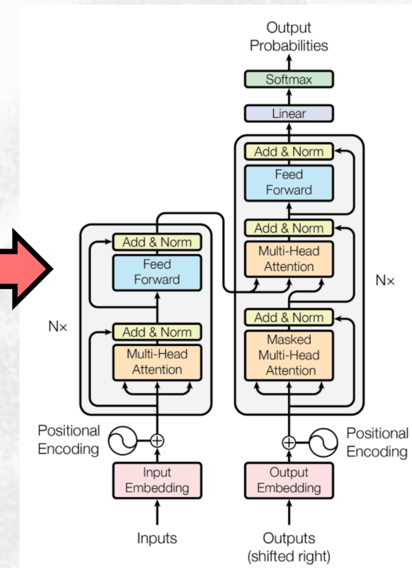
Attention
Mechanism



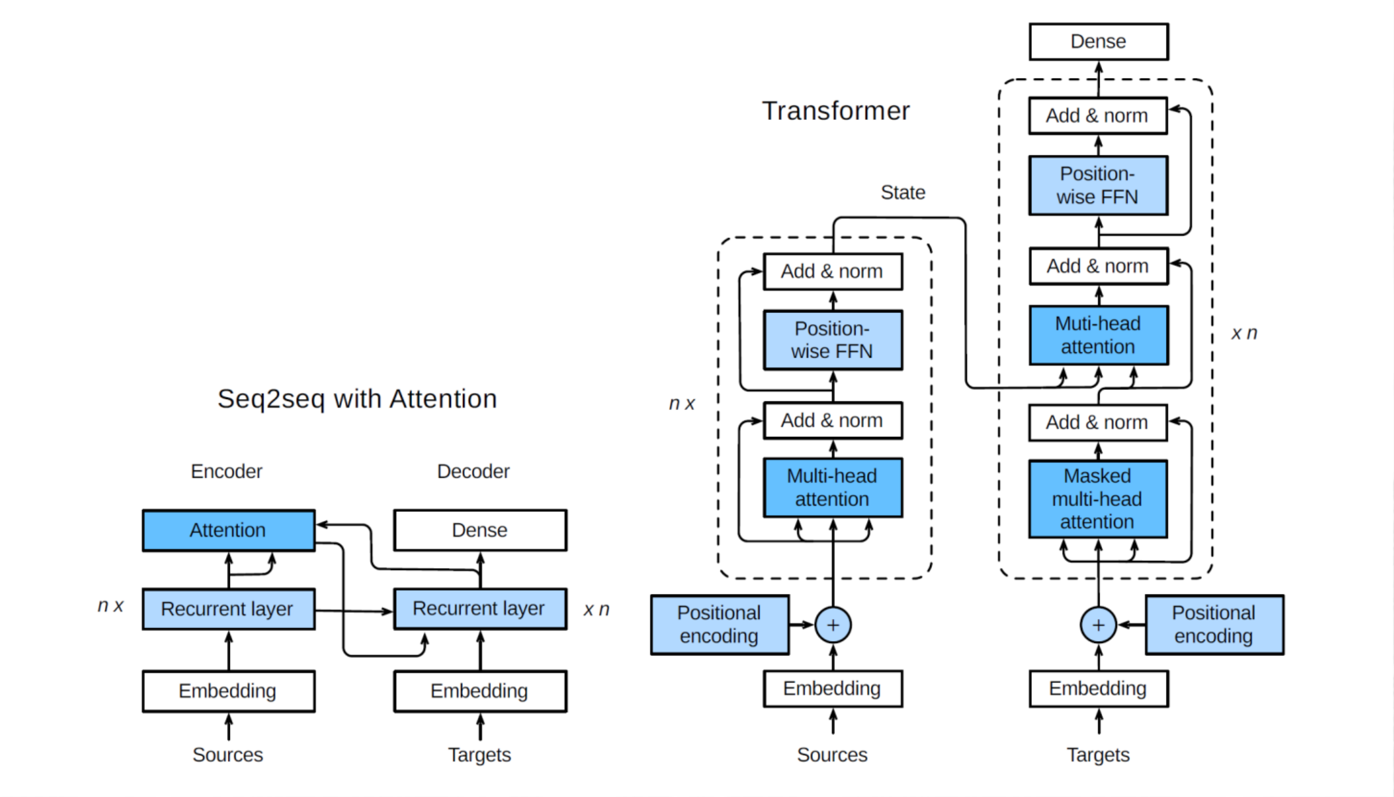
Stacking



Multihead



From attention to Transfomers



Machine learning paradigms underlying ChatGPT

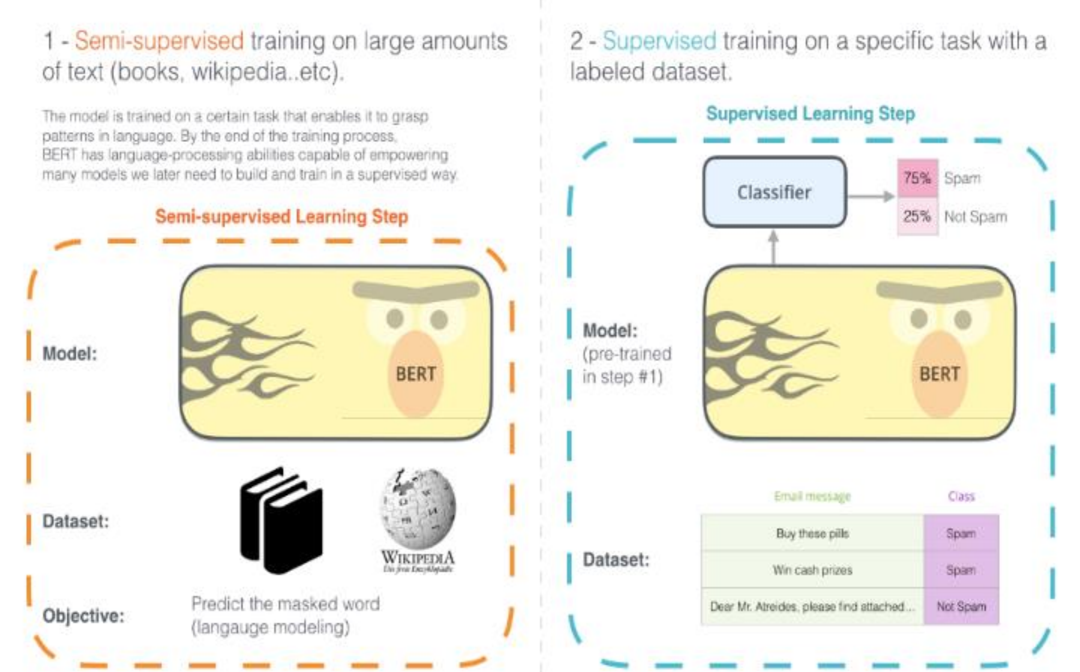
RNNs 1986

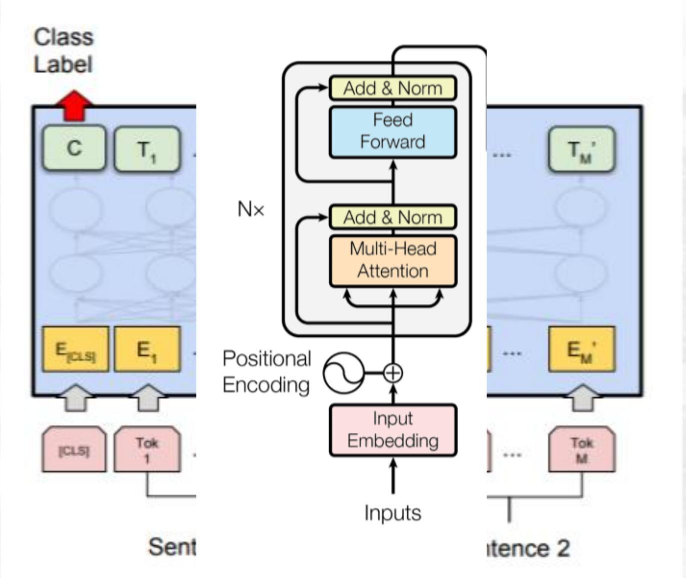
Bidirectional RNNs 1997

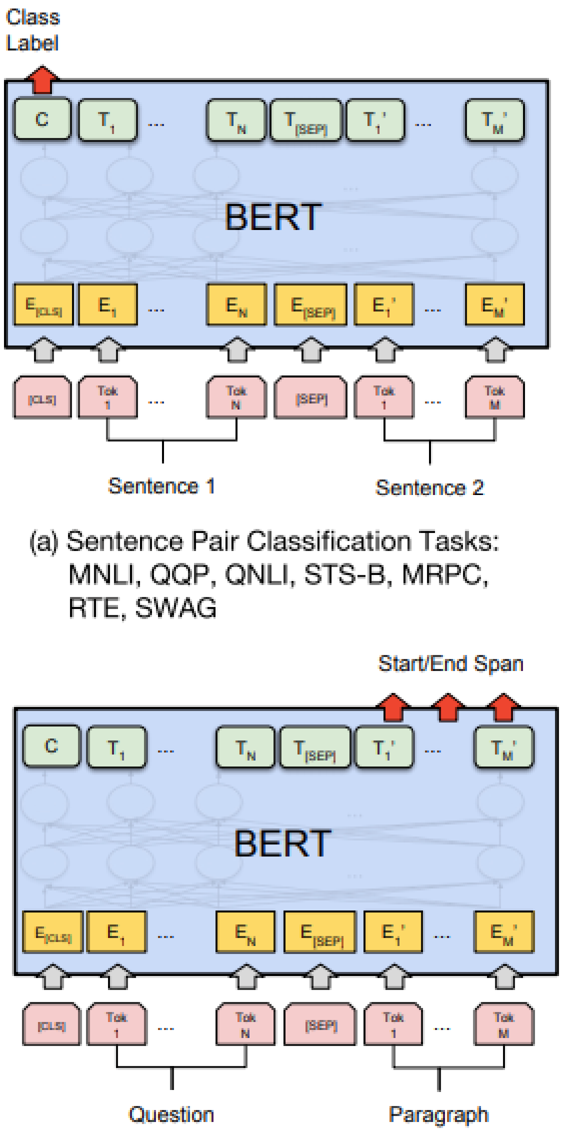
Encoder-Decoder RNNs 2014

Transformers 2017

BERT 2018







(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA

(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

Language Modeling and Reasoning

- Logical Entailment: the axiomatic «logical» view
- Training Automatic Entailment systems
  - From formal logic to NL
  - Recognizing Textual Entailment
- Applied RTE
  - Sentence Pairs
  - Pattern based and Prompting
- Applications

Entailment: the «logical» view

- Logical implication is used to express the entailment relationship between two subformulas

$$A \rightarrow B \qquad\qquad \forall x\, A(x) \rightarrow B(x)$$

- Logics helps in expressing logical reasoning schemata through normalized forms, e.g.,

$$A \rightarrow B \equiv \neg A \vee B \qquad \forall x\, A(x) \rightarrow B(x) \equiv \neg A(e) \vee B(e)$$ (after Skolemization)

- or equivalent variants

$$A \rightarrow B \equiv \neg(A \wedge \neg B) \qquad \forall x\, A(x) \rightarrow B(x) \equiv \forall x\, \neg(A(x) \wedge \neg B(x))$$

Entailment: semantics

- Logical implication is tightly related to semantics as it is the basis for an efficent approach to logical reasoning.

- Infact $\{A\} \vDash B$ iff $\{\} \vDash (A \rightarrow B)$

- B is semantically implied by A (only) if $(A \rightarrow B)$ is a tautology. This is used for the algorithms based on proof by contradiction, i.e.,

$\{A\} \vDash B$ iff $\{A, \neg B\} \vDash \bot$ or (with $\bot$ denoting the always false formula)

$\{\Delta, A\} \vDash B$ iff $\{\Delta, A, \neg B\} \vDash \bot$

Entailment & Transfomers

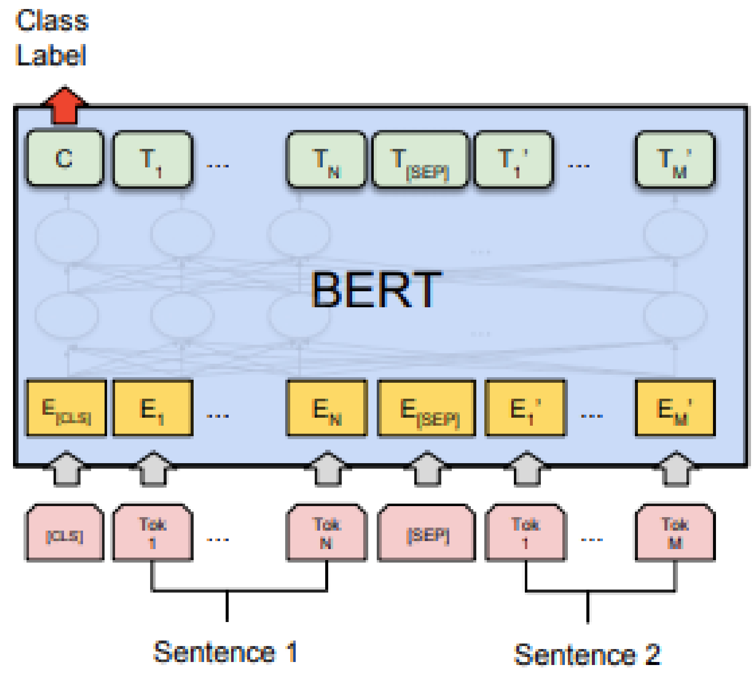
- Logical implication is usually managed through a chain of deductive steps (as in logic programming) from the input query (i.e. a theorm to be demonstrated) to its fully resolved facts, or through contadictions
- However, when uncertainty does not allow to design all needed facts (i.e. the axiomatic system Δ is not fully known a priori) deduction can be challenging and inconsistent.
- Neural Networks can be adopted to limit the impact of incompleteness or noise in the reference rules and minimze the rick of mistakes in entailment.

Entailment & Transfomers (2)

- A possible direction is
  - Map the axiomatic system into a training dataset
  - Map the input theorem into a natural language sentence
  - Solve the inference task of accepting or rejecting the entailment into a binary classification task
- In other words, given a training set of axioms such as
  - Δ: $\{A_1 \rightarrow B_1, \dots, A_n \rightarrow B_n\}$
  - Induc a function RTE such that for every future pair (A_i, B_j)
    - $h(A_i, B_j) = true$ iff $\{\Delta, A_i\} \vDash B_j$
    - or alternatively
    - $h(A_i \rightarrow B_j) = true$ iff $\{\Delta, A_i\} \vDash B_j$

The role of trasformers

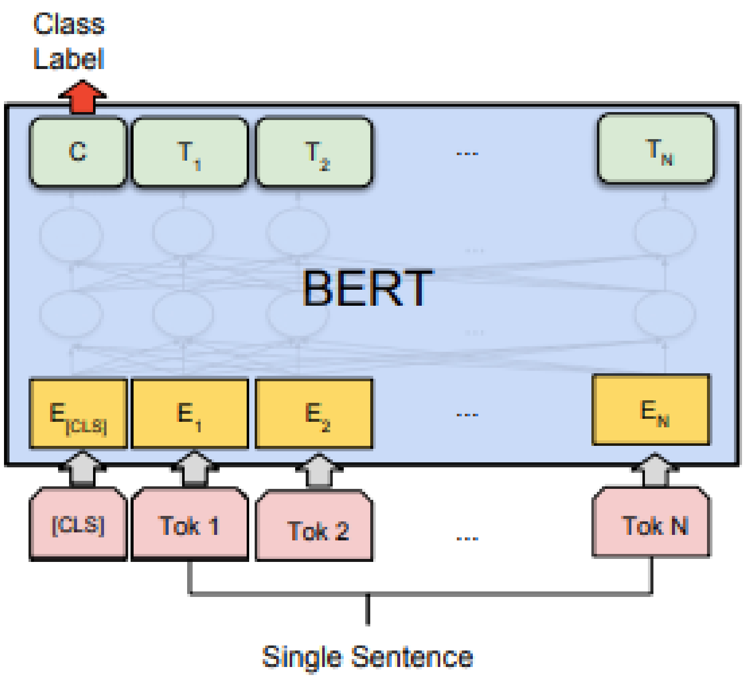
- First setting
  - $h(A_i, B_j) = true$ iff $\{\Delta, A_i\} \Vdash B_j$
  - Input given by 2 sentences
  - BERT used as the encoder
  - A stacked classifier is trained on labeled pairs
  - Type of Inference:
    - PARAPHRASING
    - TEXTUAL ENTAILMENT



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

The role of trasformers (2)

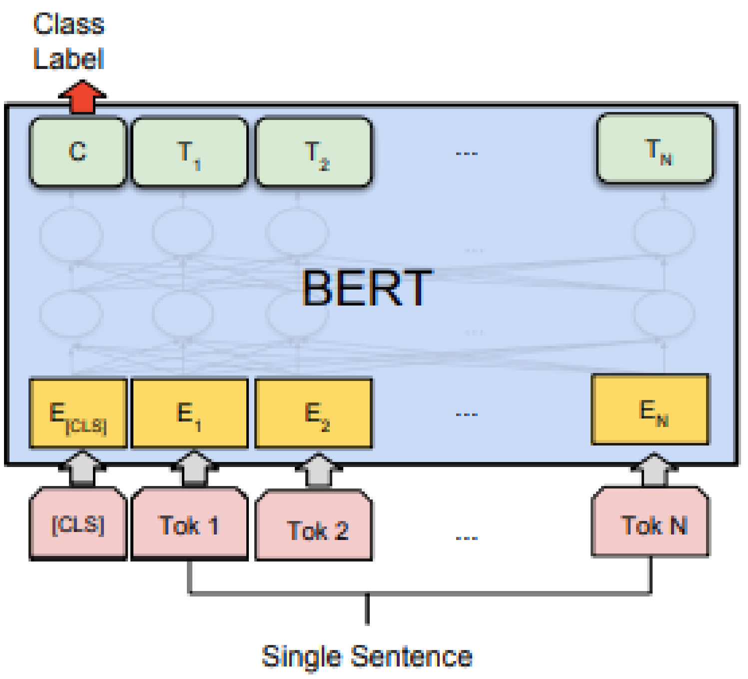
- Second setting
  - $h(A_i \rightarrow B_j) = true$ iff $\{\Delta, A_i\} \Vdash B_j$
  - Input given 1 sentence expressing the task over A_i and B_j
  - BERT used as the encoder
  - A stacked classifier is trained on labeled pairs
  - Example (PARAPHRASING):
  - «The sentence B_j has the same meaning of sentence A_i»
  - «Sentence A_i means the same as B_j»



(b) Single Sentence Classification Tasks: SST-2, CoLA

The role of trasformers (3)

- Second setting
  - $h(A_i \rightarrow B_j) = true$ iff $\{\Delta, A_i\} \Vdash B_j$
  - Input given 1 sentence expressing the task over A_i and B_j
  - BERT used as the encoder
  - A stacked classifier is trained on labeled pairs
  - Example (TEXTUAL ENTAILMENT):
  - «The sentence B_j is implied by sentence A_i»
  - «Sentence A_i guarantees the truth of B_j»



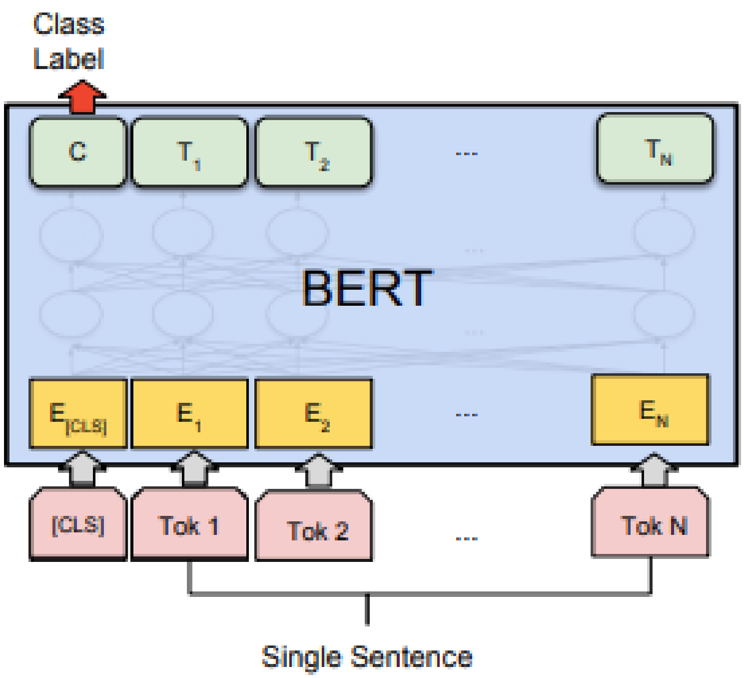
(b) Single Sentence Classification Tasks: SST-2, CoLA

Neural Entailment: applications

- The setting

$$h(A_i \rightarrow B_j) = true \ \text{ iff } \ \{\Delta, A_i\} \Vdash B_j$$

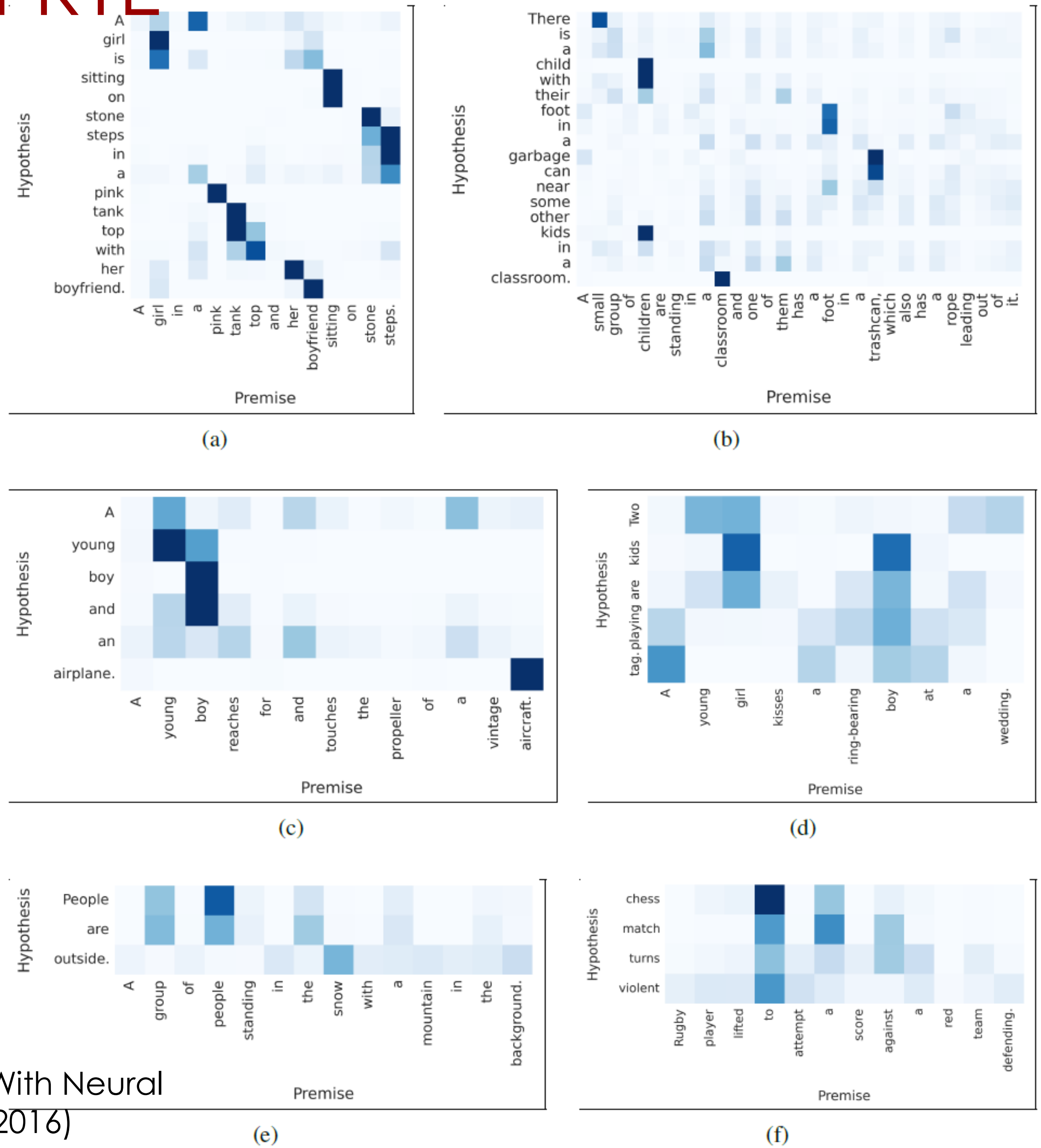
- correspond to sentences that depend on on complex interactions between A_i and B_j mapped into an individual sentences
  - BERT is always used as the encoder
  - The stacked classifier is an automatic entailment recognition tool
  - It can be preserved for future TEXTUAL ENTAILMENT tasks, e.g., :
  - Topical Classification
    - «The sentence B_j is classified by label A_i»
    - «Label A_i corresponds to the topic of B_j»
  - Sentiment Analysis:
    - «A_i implies the sentiment label B_j»
    - «A_i expresses sentiment B_j»



(b) Single Sentence Classification Tasks: SST-2, CoLA

Attention and RTE

- Word-by-word attention can easily detect simple reorderings of words in the premise (a).
- It is able to resolve synonyms (“airplane” and “aircraft”, (c) and capable of matching multi-word expressions to single words (“garbage can” to “trashcan”, 3b).
- Irrelevant parts of the premise, e.g., whole uninformative relative clauses, are correctly neglected for determining entailment (“which also has a rope leading out of it”, (b).
- Deeper semantics or common-sense knowledge (“snow” can be found “outside” and a “mother” is an “adult”, (e) and (g).
- The model seems able to resolve one-to-many relationships (“kids” to “boy” and “girl”, (d)
- Attention can fail, for example when the two sentences and their words are entirely unrelated (3f).

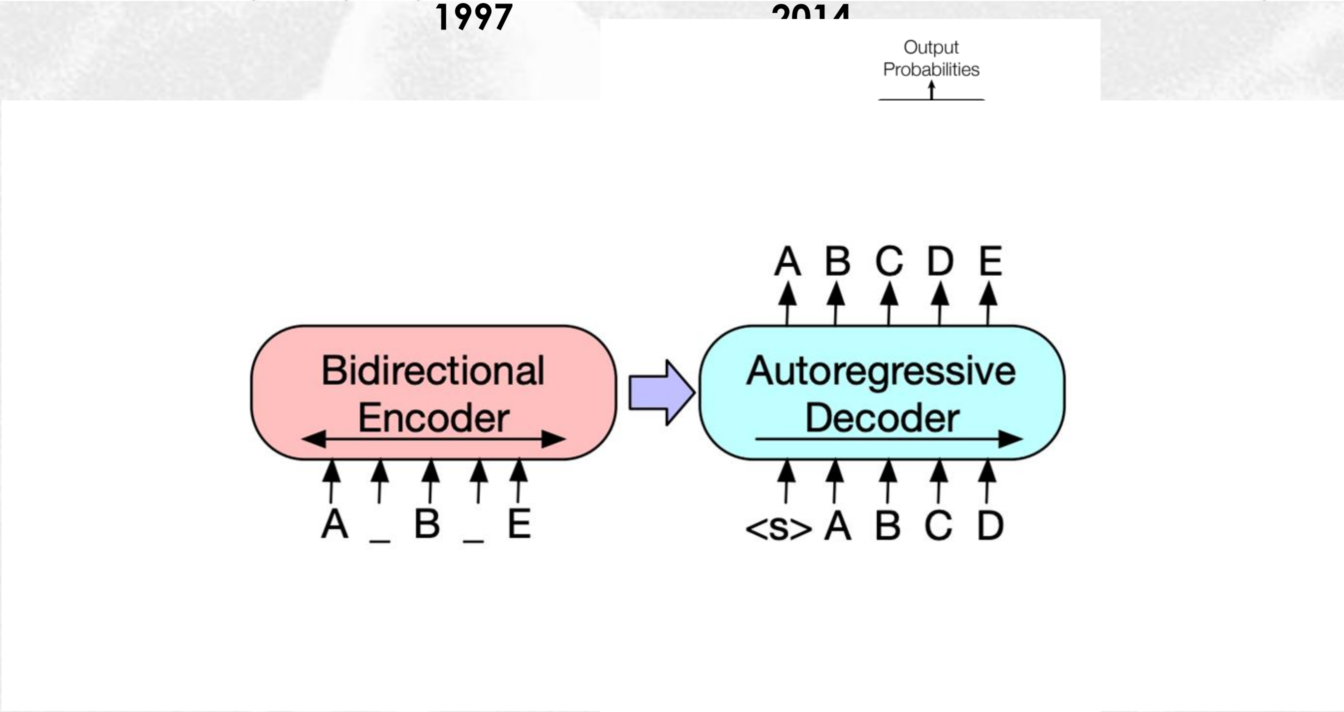


(a) (b) (c) (d) (e) (f)

from “Reasoning About Entailment With Neural Attention” (Rocktaschel et al., ICLR 2016)

Machine learning paradigms underlying ChatGPT

- RNNs 1986
- Bidirectional RNNs 1997
- Encoder-Decoder RNNs 2014
- Transformers 2017
- BERT 2018
- BART 2019



GPT-2: decoder only architectures (Radford et al., 2019)

- "We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText"
- GPT-2 is a large transformer-based language model with 1.5 billion parameters, trained on a dataset of 8 million web pages.
- GPT-2 is trained with a simple objective: predict the next word, given all of the previous words within some text.
- The diversity of the dataset causes this simple goal to contain naturally occurring demonstrations of many tasks across diverse domains.
- GPT-2 is a direct scale-up of GPT, with more than 10X the parameters and trained on more than 10X the amount of data

GPT-2: sources of insipiration

- Multitask QA Networks (MQAN) (McCann et al, 2018)

Examples

Question	Context	Answer
What is a major importance of Southern California in relation to California and the US?	...Southern California is a major economic center for the state of California and the US....	major economic center
What is the translation from English to German?	Most of the planet is ocean water.	Der Großteil der Erde ist Meerwasser
What is the summary?	Harry Potter star Daniel Radcliffe gains access to a reported £320 million fortune...	Harry Potter star Daniel Radcliffe gets £320M fortune...
Hypothesis: Product and geography are what make cream skimming work. Entailment, neutral, or contradiction?	Premise: Conceptually cream skimming has two basic dimensions – product and geography.	Entailment
Is this sentence positive or negative?	A stirring, funny and finally transporting re-imagining of Beauty and the Beast and 1930s horror film.	positive
What has something experienced?	Areas of the Baltic that have experienced eutrophication.	eutrophication
Who is the illustrator of Cycle of the Werewolf?	Cycle of the Werewolf is a short novel by Stephen King, featuring illustrations by comic book artist Bernie Wrightson.	Bernie Wrightson
What is the change in dialogue state?	Are there any Eritrean restaurants in town?	food: Eritrean
What is the translation from English to SQL?	The table has column names... Tell me what the notes are for South Australia	SELECT notes from table WHERE 'Current Slogan' = 'South Australia'
Who had given help? Susan or Joan?	Joan made sure to thank Susan for all the help she had given.	Susan

Figure 1: Overview of the decaNLP dataset with one example from each decaNLP task in the order presented in Section 2. They show how the datasets were pre-processed to become question answering problems. Answer words in red are generated by pointing to the context, in green from the question, and in blue if they are generated from a classifier over the output vocabulary.

- Our speculation is that a language model with sufficient capacity will begin to learn to infer and perform the tasks demonstrated in natural language sequences in order to better predict them, regardless of their method of procurement. If a language model is able to do this it will be, in effect, performing unsupervised multitask learning.

GPT-2: architecture

- Modifications:
  - Local attention: Sequence tokens are divided into blocks of similar length and attention is performed in each block independently. In our experiments, we choose to have blocks of 256 tokens.
  - Memory-compressed attention: After projecting the tokens into the query, key, and value embeddings, we reduce the number of keys and values by using a strided convolution. The number of queries remains unchanged.
- "They allow us in practice to process sequences 3x in length over the T-D model (Vaswani et al., 2017)."

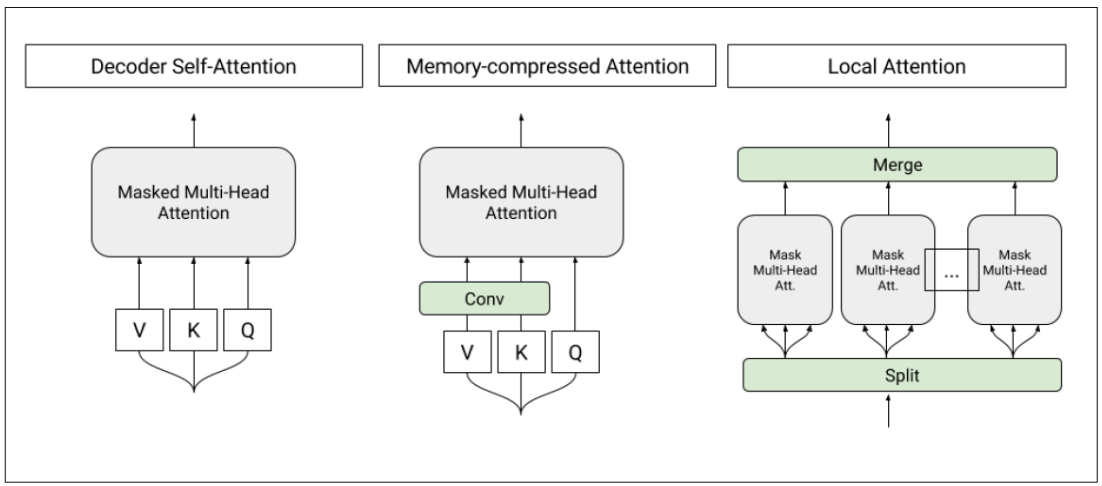


Figure 1: The architecture of the self-attention layers used in the T-DMCA model. Every attention layer takes a sequence of tokens as input and produces a sequence of similar length as the output. **Left:** Original self-attention as used in the transformer-decoder. **Middle:** Memory-compressed attention which reduce the number of keys/values. **Right:** Local attention which splits the sequence into individual smaller sub-sequences. The sub-sequences are then merged together to get the final output sequence.

GPT-2: architecture (2)

- From (Radford et al., 2017, GPT paper)

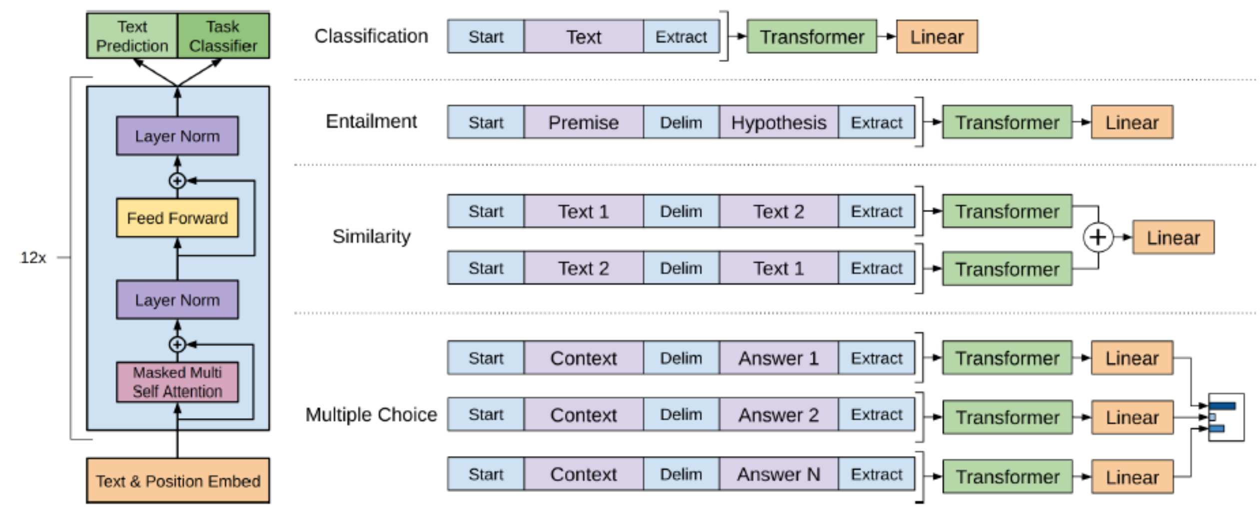


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

GPT-2: results

Language Models are Unsupervised Multitask Learners

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

- The LAMBADA dataset (Paperno et al., 2016)
  - It tests the ability of systems to model long-range dependencies in text.
  - The task is to predict the final word of sentences which require at least 50 tokens of context for a human to successfully predict.

GPT-2: results on Lambada

- The LAMBADA dataset (Paperno et al., 2016)
  - It tests the ability of systems to model long-range dependencies in text.
  - The task is to predict the final word of sentences which require at least 50 tokens of context for a human to successfully predict.

(1) *Context:* "Yes, I thought I was going to lose the baby." "I was scared too," he stated, sincerity flooding his eyes. "You were ?" "Yes, of course. Why do you even ask?" "This baby wasn't exactly planned for."
Target sentence: "Do you honestly think that I would want you to have a _____ ?"
Target word: miscarriage

(2) *Context:* "Why?" "I would have thought you'd find him rather dry," she said. "I don't know about that," said Gabriel. "He was a great craftsman," said Heather. "That he was," said Flannery.
Target sentence: "And Polish, to boot," said _____.
Target word: Gabriel

(3) *Context:* Preston had been the last person to wear those chains, and I knew what I'd see and feel if they were slipped onto my skin-the Reaper's unending hatred of me. I'd felt enough of that emotion already in the amphitheater. I didn't want to feel anymore. "Don't put those on me," I whispered. "Please."
Target sentence: Sergei looked at me, surprised by my low, raspy please, but he put down the _____.
Target word: chains

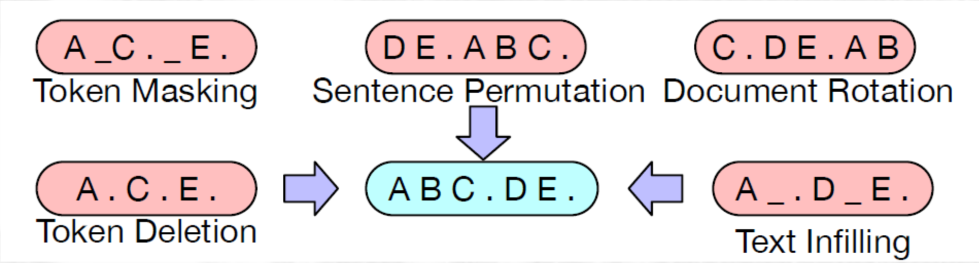
(4) *Context:* They tuned, discussed for a moment, then struck up a lively jig. Everyone joined in, turning the courtyard into an even more chaotic scene, people now dancing in circles, swinging and spinning in circles, everyone making up their own dance steps. I felt my feet tapping, my body wanting to move.
Target sentence: Aside from writing, I 've always loved _____.
Target word: dancing

- GPT-2 improves the state of the art from 99.8 (Grave et al., 2016) to 8.6 perplexity and increases the accuracy of LMs on this test from 19% (Dehghani et al., 2018) to 52.66%. Adding a stop-word filter as an approximation to this further increases accuracy to 63.24%.

- Investigating GPT-2's errors showed most predictions are valid continuations of the sentence, but are not valid final words

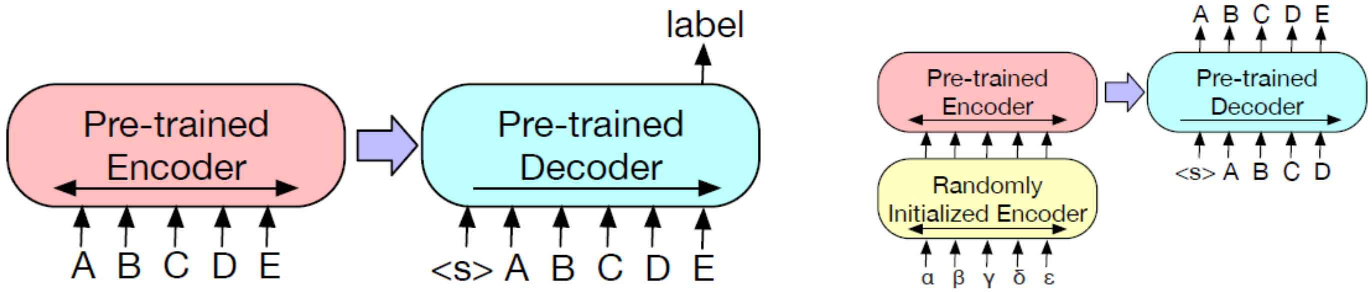
BART (Lewis et al., 2019) - Facebook

- Enconding decoding architecture based on Pretraining and fine tuned towards different tasks such as: RTE, SA, …
- Two stages of PRETRAINING
  - Text is first corrupted with an arbitrary noising function,
  - A sequence-to-sequence model is learned to reconstruct the original text.



- FINE TUNING:
  - **MNLI** (Williams et al., 2017), a bitext classification task to predict whether one sentence **entails** another. The fine-tuned model concatenates the two sentences with appended an EOS token, and passes them to both the BART encoder and decoder. In contrast to BERT, the representation of the EOS token is used to classify the sentences relations.
  - ELI5 (Fan et al., 2019), a long-form abstractive question answering dataset. Models generate answers conditioned on the concatenation of a question and supporting documents.

Applying BART



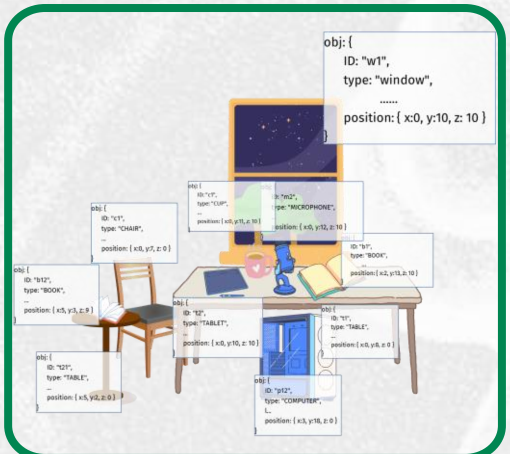
(a) To use BART for classification problems, the same input is fed into the encoder and decoder, and the representation from the final output is used.

(b) For machine translation, we learn a small additional encoder that replaces the word embeddings in BART. The new encoder can use a disjoint vocabulary.

Figure 3: Fine tuning BART for classification and translation.

GrUT: The Overall Flow

Command: *“Prendi il volume sul tavolo vicino la finestra"*



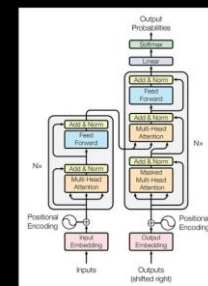
Entities Retrieval

Linguistic Extraction

MD: *b1, conosciuto anche come libro o volume, è un’istanza della classe BOOK, t1, conosciuto anche come tavolo o scrivania, è un’istanza della classe TABLE # b1 è vicino t1*

Input: *Command + MD*

GrUT-IT



Output:
TAKING(Theme(b1))

Hromei et al, 2022, "Embedding Contextual Information in Seq2seq Models for Grounded Semantic Role Labeling"

Experimental Evaluation

FP = Frame Prediction
AIC = Argument Identification and Classification
EM = Exact Match
HM = Head Match

Model	Learning Rate	FP	AIC-Exact Match	AIC-Head Match
LU4R	-	*95.32%*	*77.67%*	*86.35%*
GrUT-IT	$5 \cdot 10^{-5}$	**96.86%**	**82.30%**	**85.19%**

Results here are reported as F1 values on 10-fold cross-validation schema with 80/10/10 data split.
Performance for LU4R is reported in *italic* as it is not entirely comparable with.

LU4R: TAKING(Theme("libro"))
GrUT-IT: TAKING(Theme(b1))

Machine learning paradigms underlying ChatGPT

Transformers 2017

GPT-3 2020

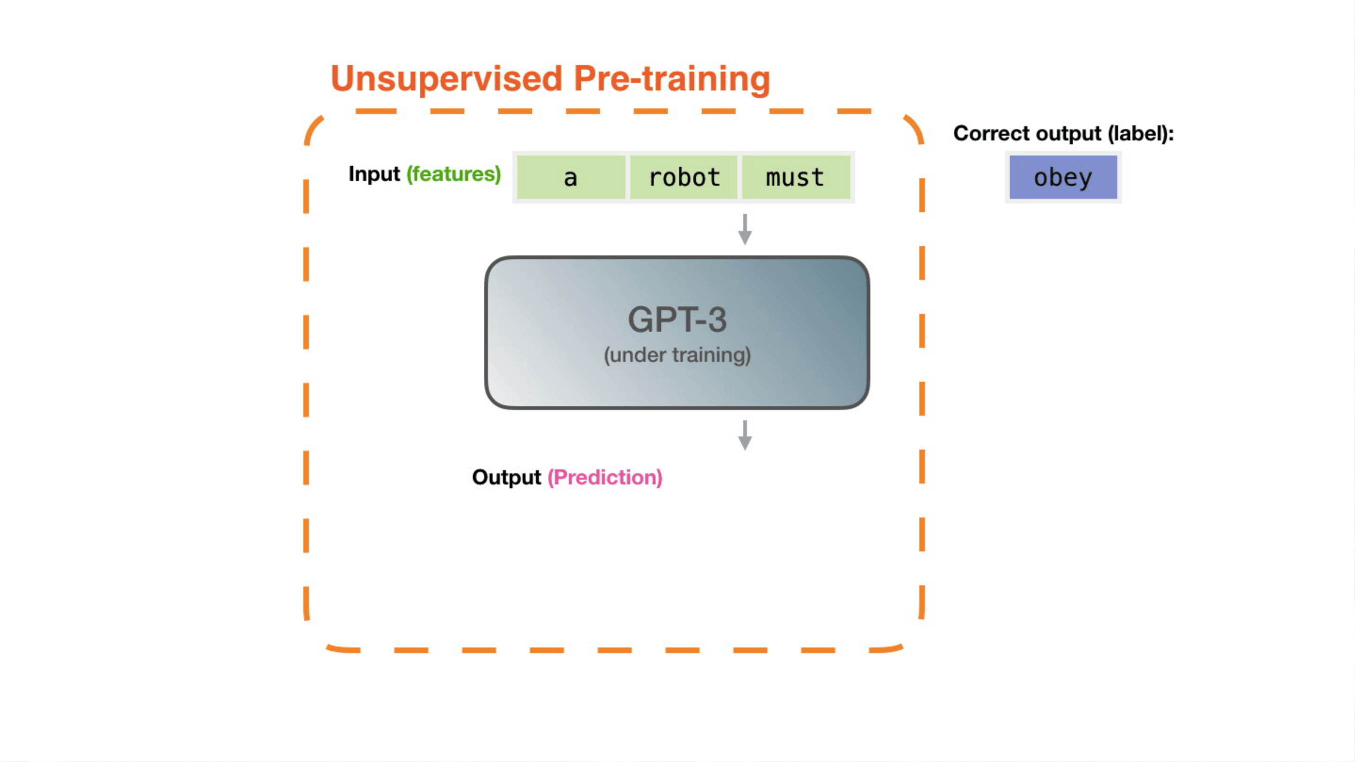
RNNs 1986

Bidirectional RNNs 1997

Encoder-Decoder RNNs 2014

BERT 2018

BART 2019



GPT3: novelty

- «Language Models are Few-Shot Learners" (Brown et al., 2020)

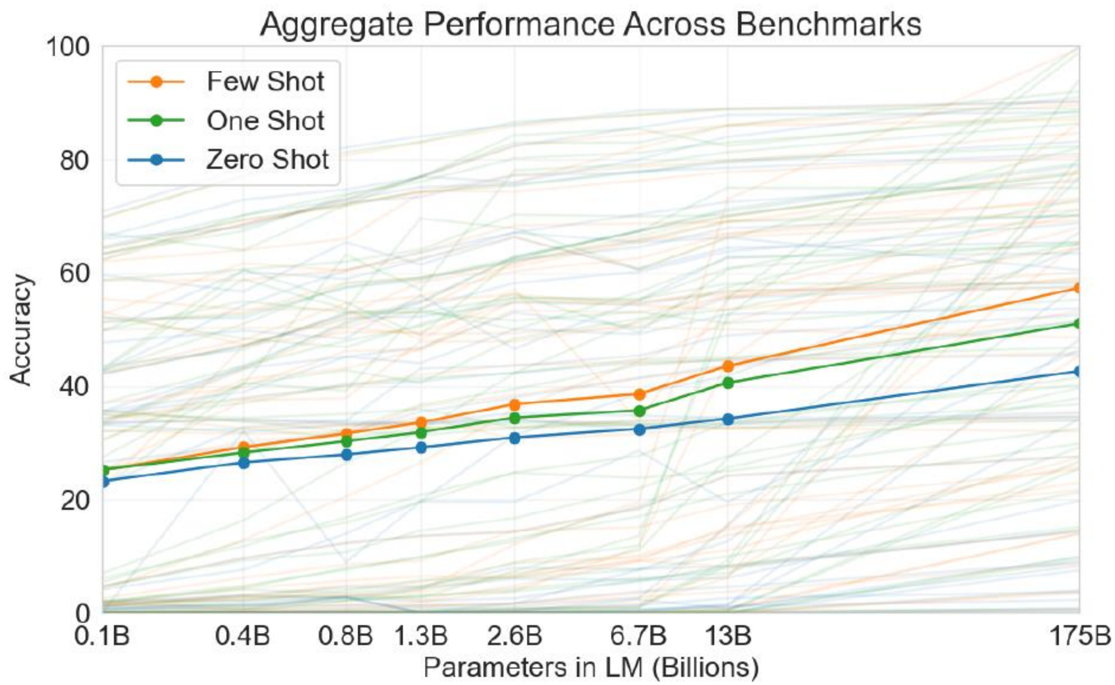
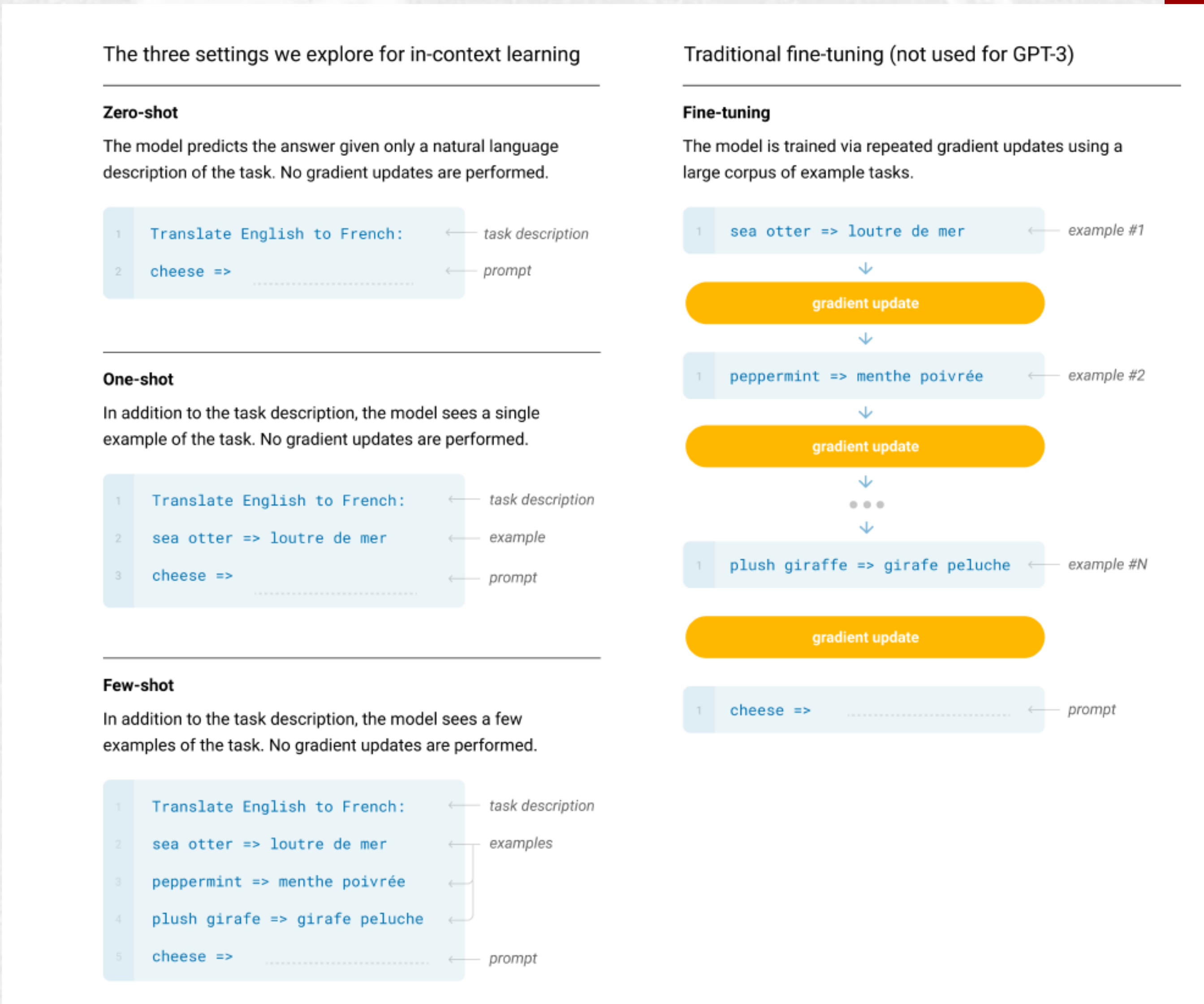


Figure 1.3: Aggregate performance for all 42 accuracy-denominated benchmarks While zero-shot performance improves steadily with model size, few-shot performance increases more rapidly, demonstrating that larger models are more proficient at in-context learning. See Figure 3.8 for a more detailed analysis on SuperGLUE, a standard NLP benchmark suite.


The three settings we explore for in-context learning
Zero-shot
The model predicts the answer given only a natural language description of the task. No gradient updates are performed.
Translate English to French:
task description
cheese =>
prompt
One-shot
In addition to the task description, the model sees a single example of the task. No gradient updates are performed.
Translate English to French:
task description
sea otter => loutre de mer
example
cheese =>
prompt
Few-shot
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.
Translate English to French:
task description
sea otter => loutre de mer
examples
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
prompt
Traditional fine-tuning (not used for GPT-3)
Fine-tuning
The model is trained via repeated gradient updates using a large corpus of example tasks.
sea otter => loutre de mer
example #1
gradient update
peppermint => menthe poivrée
example #2
gradient update
plush giraffe => girafe peluche
example #N
gradient update
cheese =>
prompt

GPT-3: size

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

- Here n_{params} is the total number of trainable parameters, n_{layers} is the total number of layers, d_{model} is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer, d_{ff}=4xd_{model}), and d_{head} is the dimension of each attention head.
- All models use a context window of n_{ctx} = 2048 tokens

Machine learning paradigms underlying ChatGPT

Transformers
2017

GPT-3
2020

RNNs
1986

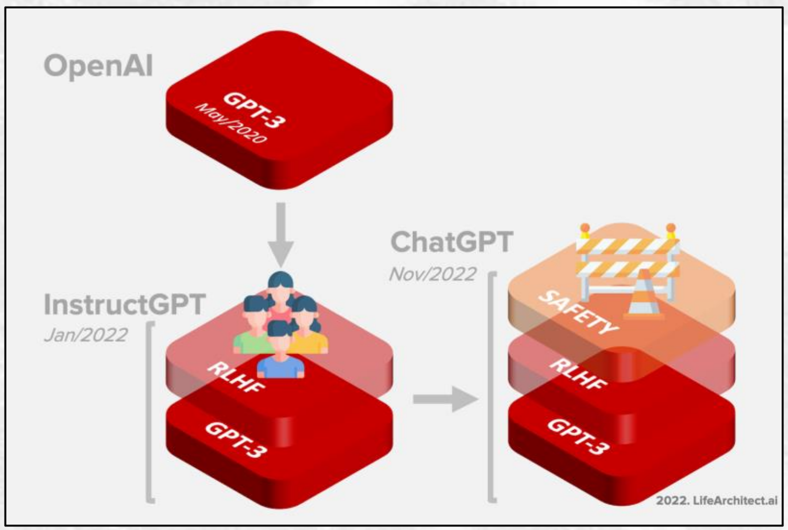
Bidirectional
RNNs
1997

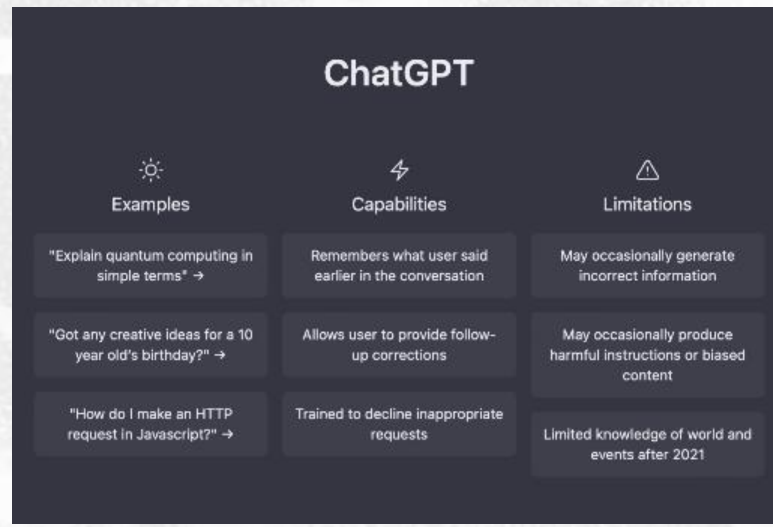
Encoder-Decoder
RNNs
2014

BER
T
201
8

BAR
T
201
9

<u>ChatGPT</u>
<u>2022</u>





Limitations of GPT-3

- Large language models often express unintended behaviors such as making up facts, generating biased or toxic text, or simply not following user instructions. This is because the language modeling objective is misaligned.

- The idea: aligning language models by training them to act in accordance with the user's intention (Leike et al., 2018).
  - explicit intentions such as following instructions
  - implicit intentions such as staying truthful, and not being biased, toxic, or otherwise harmful.

- Overall Objective: language models should be helpful (they should help the user solve their task), honest (they shouldn't fabricate information or mislead the user), and harmless (they should not cause physical, psychological, or social harm to people or the environment).

InstructGPT

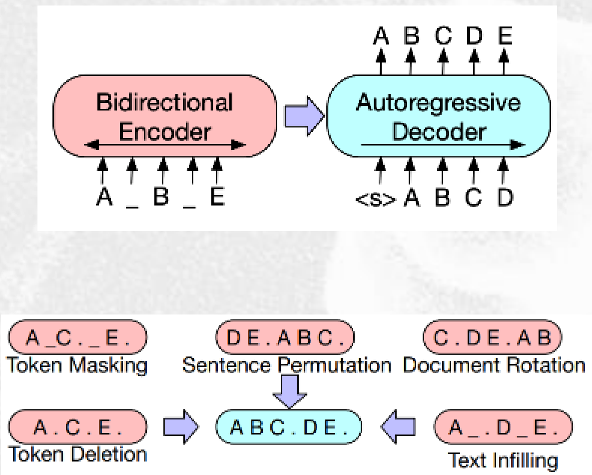
- **Step 1**: Collect demonstration data, and train a supervised policy. Labelers provide demonstrations of the desired behavior on the input prompt distribution. Then, fine-tuning of a pretrained GPT-3 model on this data using supervised learning is carried out.

- **Step 2**: Collect comparison data, and train a reward model. A dataset of comparisons between model outputs is collected: labelers indicate which output they prefer for a given input. A reward model to predict the human-preferred output is then trained.

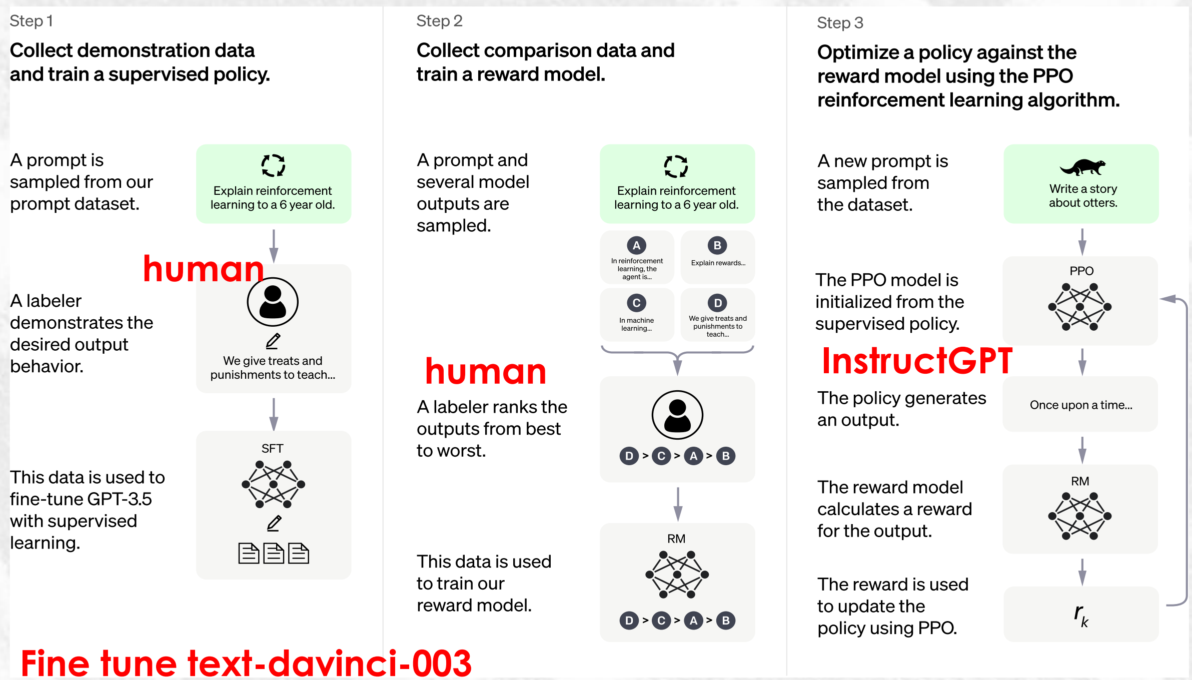
- **Step 3:** Optimize a policy against the reward model using PPO. We use the output of the RM as a scalar reward. We fine-tune the supervised policy to optimize this reward using the proximal policy optimization (PPO) algorithm (Schulman et al., 2017).

At the heart of ChatGPT (from BART to ChatGPT)

BART Training-steps



ChatGPT Training-steps

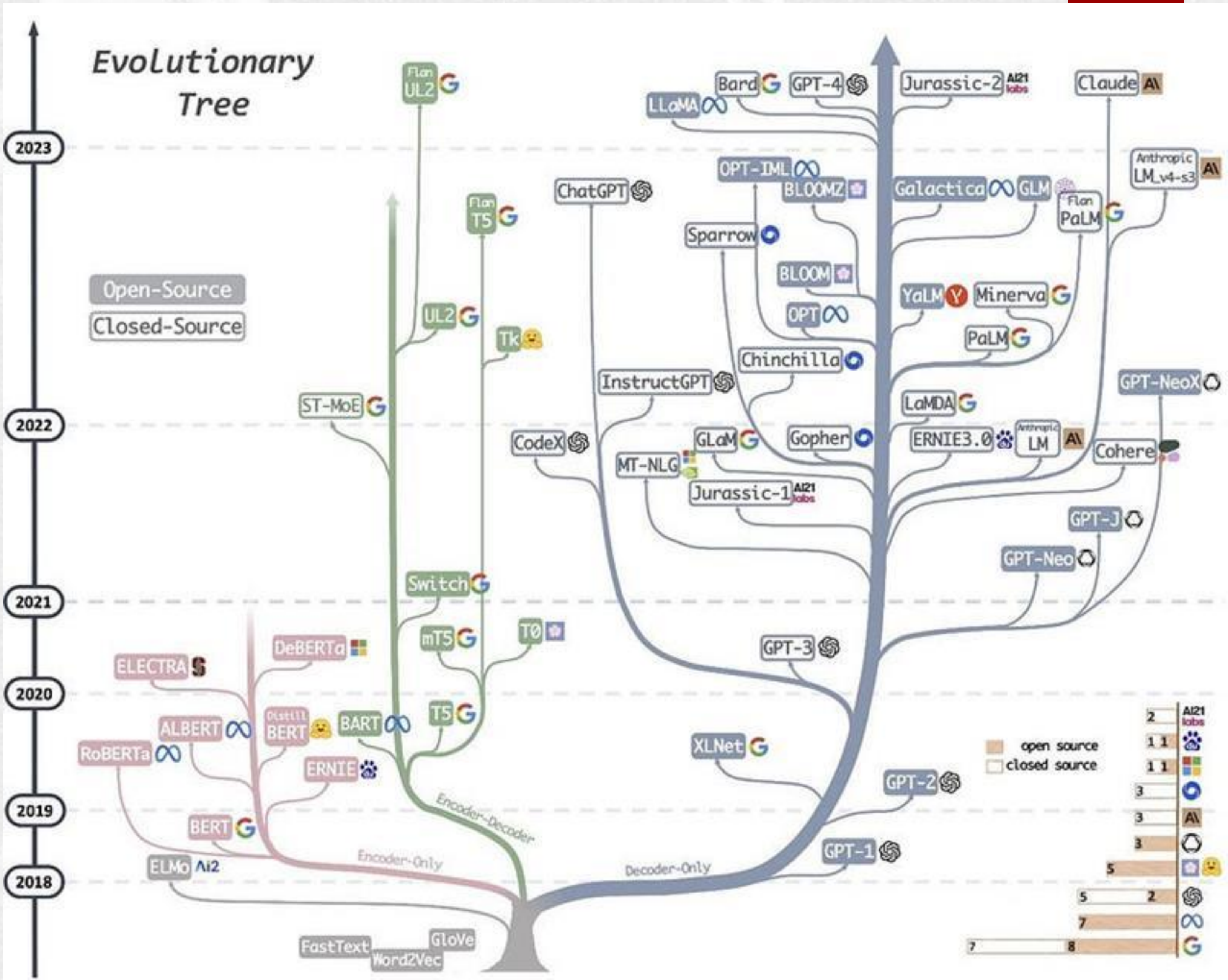


Fine tune text-davinci-003 to get InstructGPT

The Environment

from Ouyang, L., Wu, J., Jiang, et al. (2022). *Training language models to follow instructions with human feedback*

Foundational Models



Neural writing

write a biblical verse in the style of the king james bible explaining how to remove a peanut butter sandwich from a VCR

And it came to pass that a man was troubled by a peanut butter sandwich, for it had been placed within his VCR, and he knew not how to remove it.

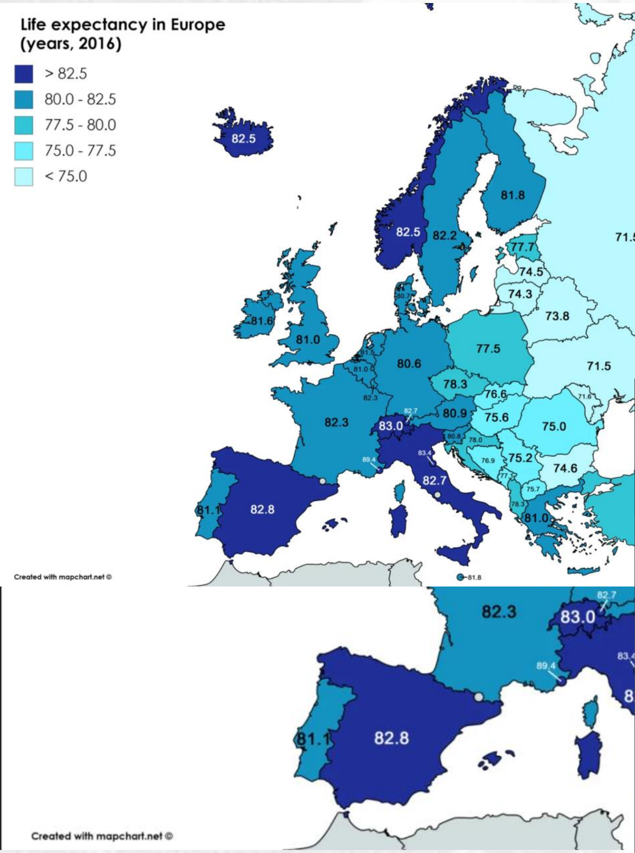
And he cried out to the Lord, saying, "Oh, Lord, how can I remove this sandwich from my VCR, for it is stuck fast and will not budge?"

And the Lord spoke unto him, saying, "Fear not, my child, for I shall guide thy hand and show thee the way. Take thy butter knife, and carefully insert it between the sandwich and the VCR, and gently pry them apart. And with patience and perseverance, the sandwich shall be removed, and thy VCR shall be saved."

And the man did as the Lord commanded, and lo and behold, the sandwich was removed from the VCR, and the man was saved.

And the Lord said, "Verily I say unto thee, seek not to put thy peanut butter sandwiches in thy VCR, for it is not a suitable place for such things. Rather, keep thy sandwiches in thy refrigerator or on thy plate, where they belong."

And the man heeded the Lord's words, and from that day forth, he kept his sandwiches in their proper place, and was saved from trouble and woe. Amen.



MA

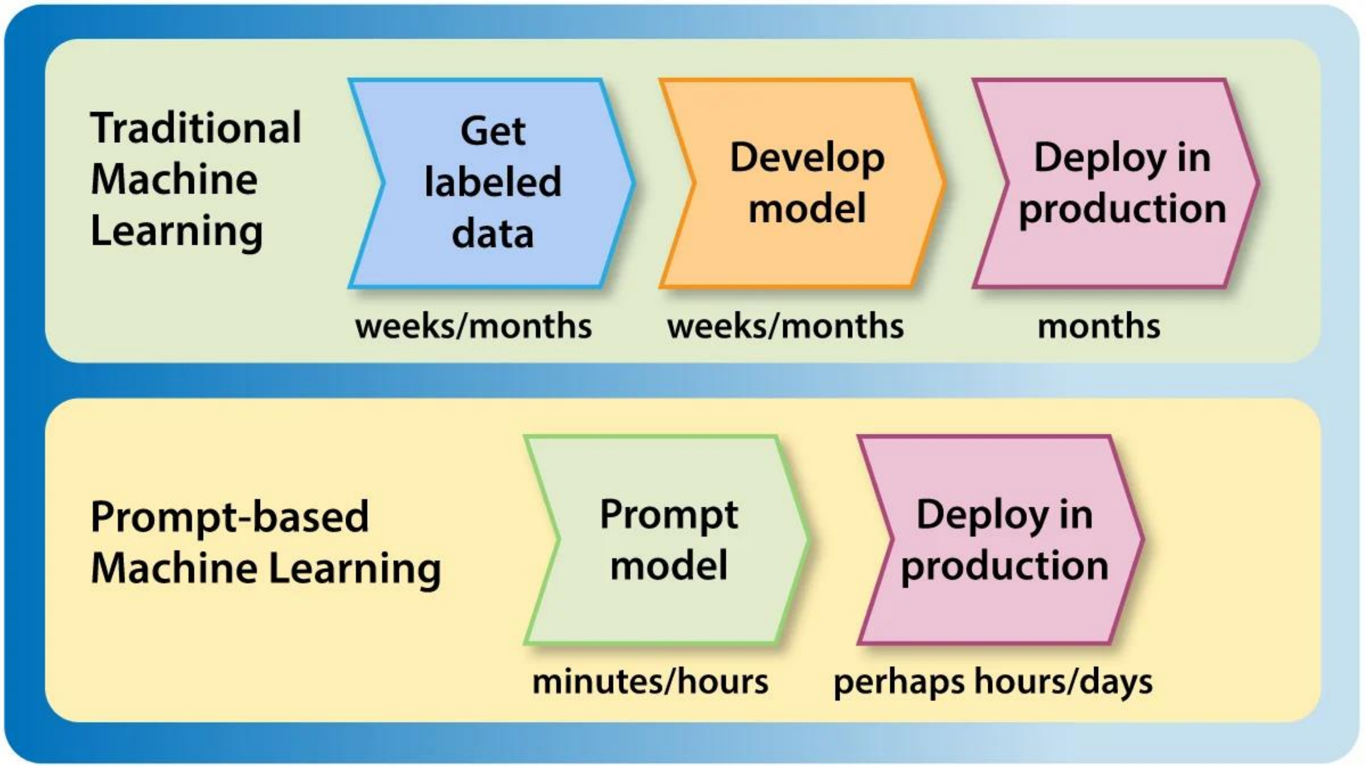
Describe the image for me: https://en.wikipedia.org/wiki/List_of_European_countries_by_life_expectancy#/media/File:Life_expectancy_in_Europe.png

The image depicts a color-coded map of Europe, with different shades of green and yellow representing the life expectancy of each country. The map is titled "Life expectancy in Europe" and shows a gradient from lighter to darker shades of green, representing higher life expectancies.

The countries with the highest life expectancy, such as Switzerland, Spain, Italy, and France, are shown in the darkest shades of green, indicating life expectancies of over 80 years. The countries with lower life expectancies, such as Russia, Belarus, and Ukraine, are depicted in yellow and lighter shades of green, indicating life expectancies of under 75 years.

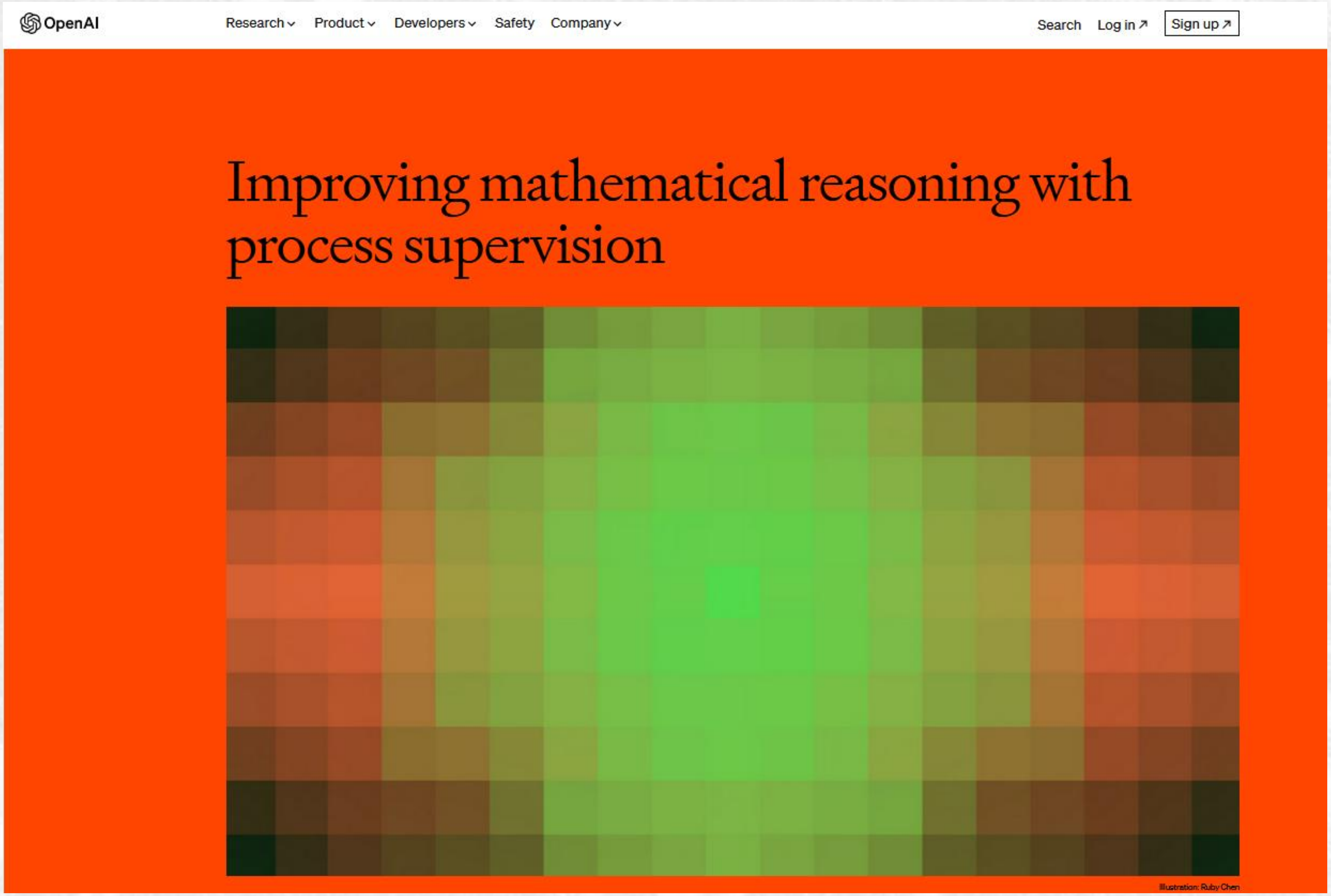
The map also includes a legend that provides a color-coded scale and corresponding life expectancy ranges, allowing viewers to easily interpret the data. Overall, the image provides a visual representation of the variation in life expectancy across different European countries, highlighting the disparities between countries and regions.

Trends …



Traditional Machine Learning
Get labeled data
weeks/months
Develop model
weeks/months
Deploy in production
months
Prompt-based Machine Learning
Prompt model
minutes/hours
Deploy in production
perhaps hours/days

Future directions



Beyond Transformer
bibliography

- (Vaswani 2017), Attention is all you need, https://arxiv.org/abs/1706.03762
- (Devlin et al 2018), BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, https://arxiv.org/abs/1810.04805
- Rocktaschel et al., “Reasoning About Entailment With Neural Attention” (ICLR 2016)
- T5: (Wolf et al, 2019) Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. ArXiv, abs/1910.03771.
- BART Encoding-Decoding: Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.https://arxiv.org/abs/1910.13461
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, "Improving Language Understanding by Generative Pre-Training", 2019
- 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: Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901. https://arxiv.org/abs/2005.14165, NeurIPS 2020.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, Paul F. Christiano: Learning to summarize with human feedback. NeurIPS 2022