



# Natural Language Processing and LLMs

NYU Shanghai AI Summer Program

Instructor: Chen Zhao

# Outline

---

- Part 1: NLP Background
- Part 2: Background and Transformer
- Part 3: Morden LLMs and ChatGPT
- Part 4: Large Reasoning Models and Deepseek R1

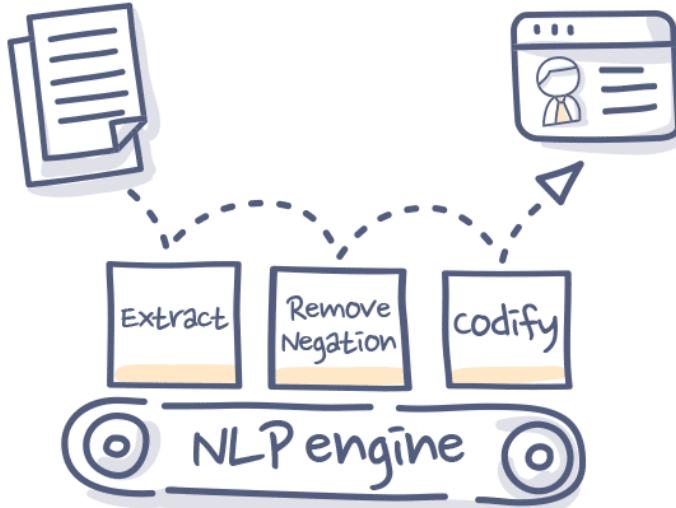
# Outline

---

- Part 1: NLP Background
- Part 2: Background and Transformer
- Part 3: Morden LLMs and ChatGPT
- Part 4: Large Reasoning Models and Deepseek R1

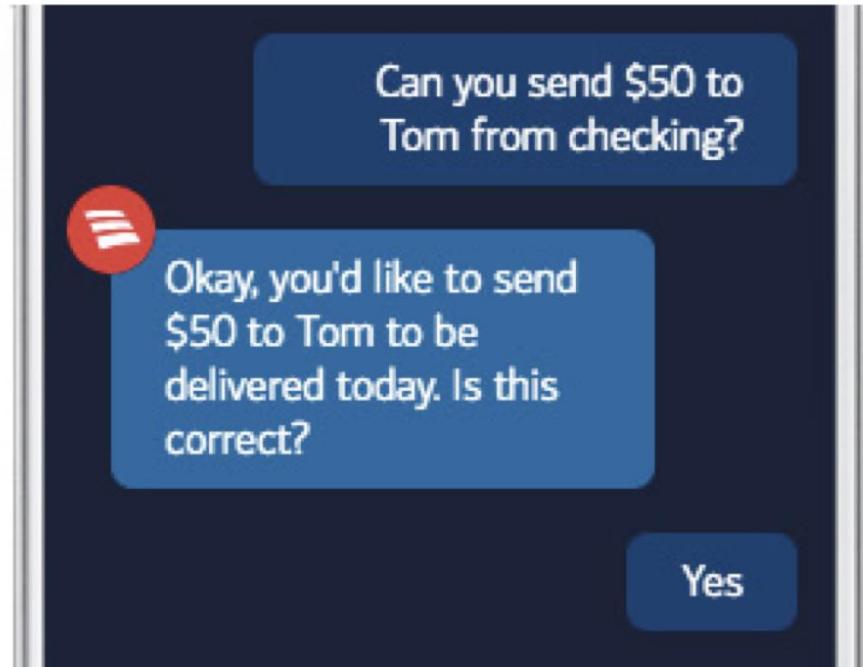
# What is NLP?

- **Natural Language Processing:**
  - build program to automatically analyse, understand and generate **human language** in text
  - Important branch of Artificial Intelligence
- NLP is an interdisciplinary field
  - Healthcare, Law, Finance, etc

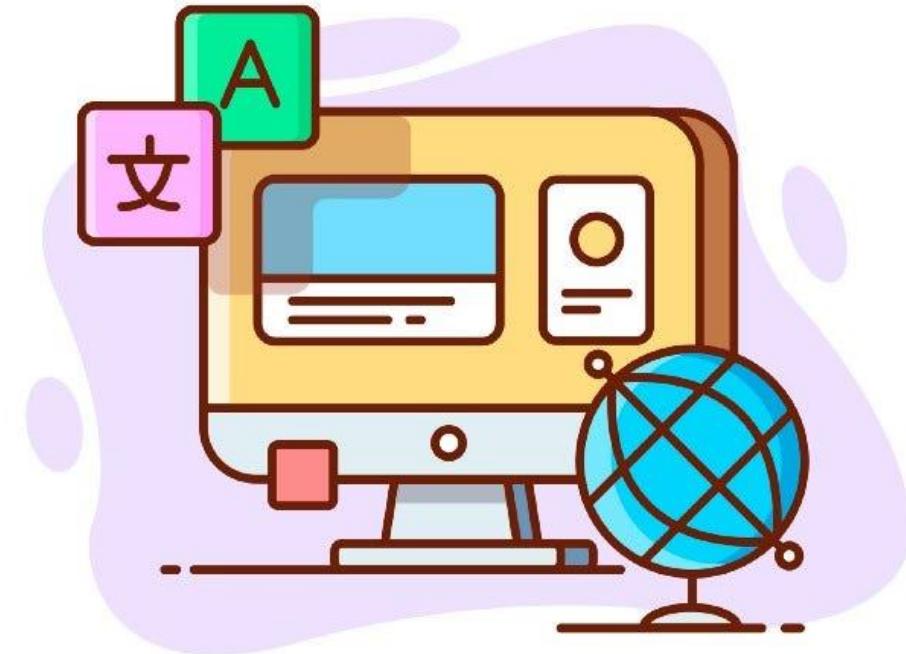


Legal Entity Recognition	Legal Entity Linking	Assertion Status	Relation Extraction
<p>This Loan Agreement dated as of November 17, 2014 (this «Agreement»), is made by and among <b>Auxilium Pharmaceuticals Inc.</b>, a corporation incorporated under the laws of the State of Delaware («U.S. Borrower»), <b>Auxilium UK LTD</b>, a private company limited by shares registered in England and Wales («UK Borrower») and, collectively with the <b>U.S. Borrower</b>, the «Borrowers») and <b>Endo Pharmaceuticals Inc.</b>, a corporation incorporated under the laws of the State of Delaware («Lender»).</p>	<p><b>Sauer Christopher</b></p> <p>Designates to sign the form of 731 and other documentation: → TRUE</p> <p>Neither...nor...is subject to a denial: → FALSE</p> <p>...may require approval to... → POSSIBLE</p>	<p><b>Sauer Christopher</b></p> <p>designates Michael Lin</p> <p>works for Spornon International</p> <p>has_power Sign 731 form</p>	<p><b>Legal Embeddings</b></p> <p><b>Zero-shot Relation Extraction</b></p> <p><b>Document Splitting</b></p> <p><b>Clause Extraction</b></p> <p><b>Knowledge Graphs</b></p> <p><b>Pattern Matching and Text Mining</b></p> <p><b>Long Span Extraction with Question Answering</b></p> <p><b>Deidentification</b></p>
<p><b>Sector &amp; Industry</b> Finance (4800) Major Banks (4805)</p> <p>Fiscal Year End December</p> <p>Exchange/ISIN SIX Swiss/CH0244767585</p> <p>SEDOL B01176</p> <p>Investor Relations Contact Martin A. Oevinga</p> <p>LEI 5493005239585GKANSI</p>	<p><b>Trainable &amp; Tunable</b></p> <p><b>Scalable to a Cluster</b></p> <p><b>Transformers</b></p> <p><b>Fast Inference</b></p> <p><b>Hardware Optimized</b></p>	<p><b>Apache Spark ML Pipelines</b></p>	<p><b>LightPipeline</b></p> <p><b>intel</b> <b>NVIDIA</b></p>

# What is NLP?



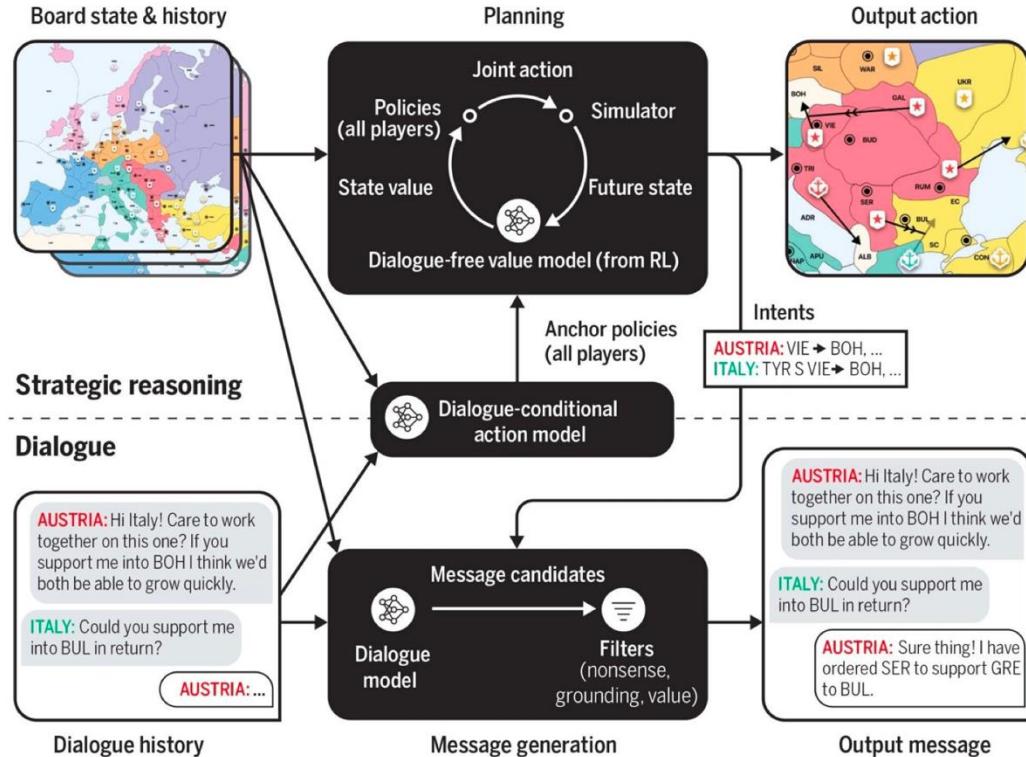
Banking assistant



# What is NLP?

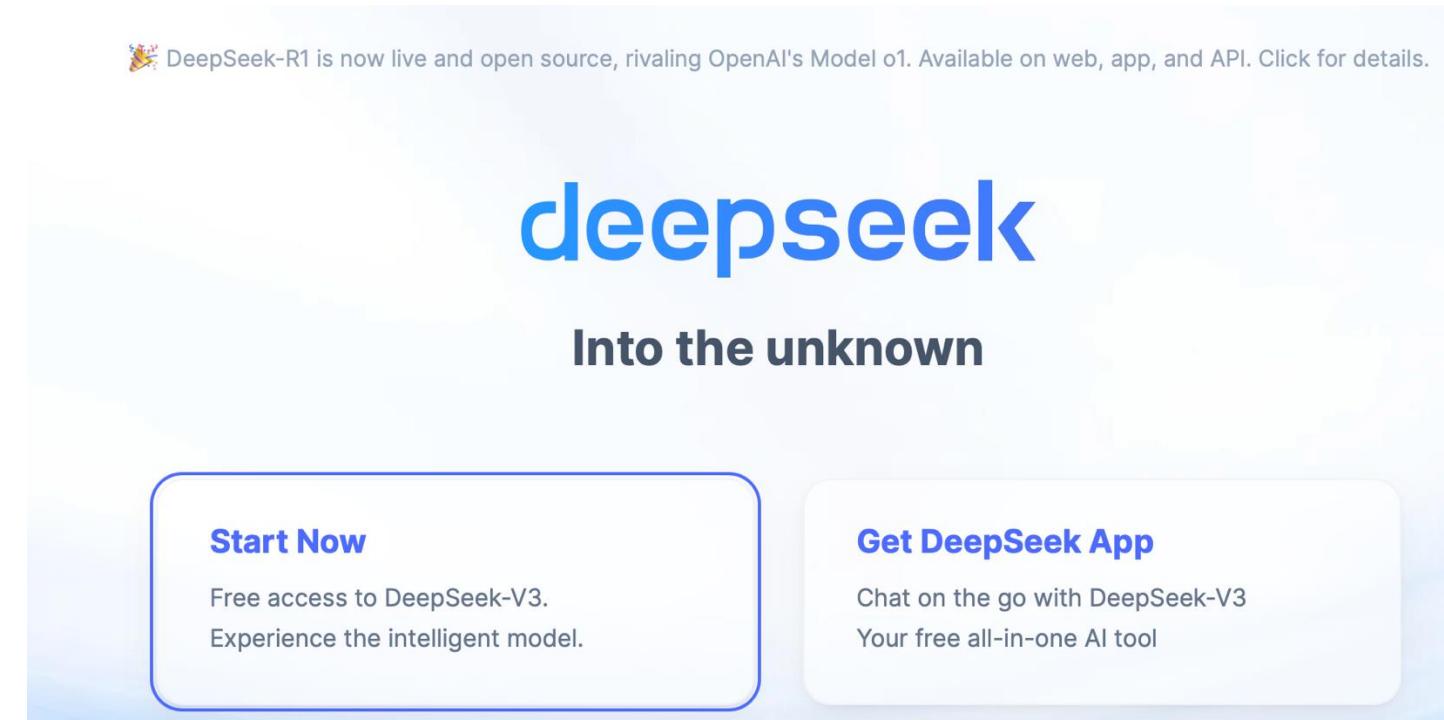
## Meta's New AI Ranked in the Top 10% at the Game 'Diplomacy'—and Human Players Were None the Wiser

By Edd Gent > November 28, 2022



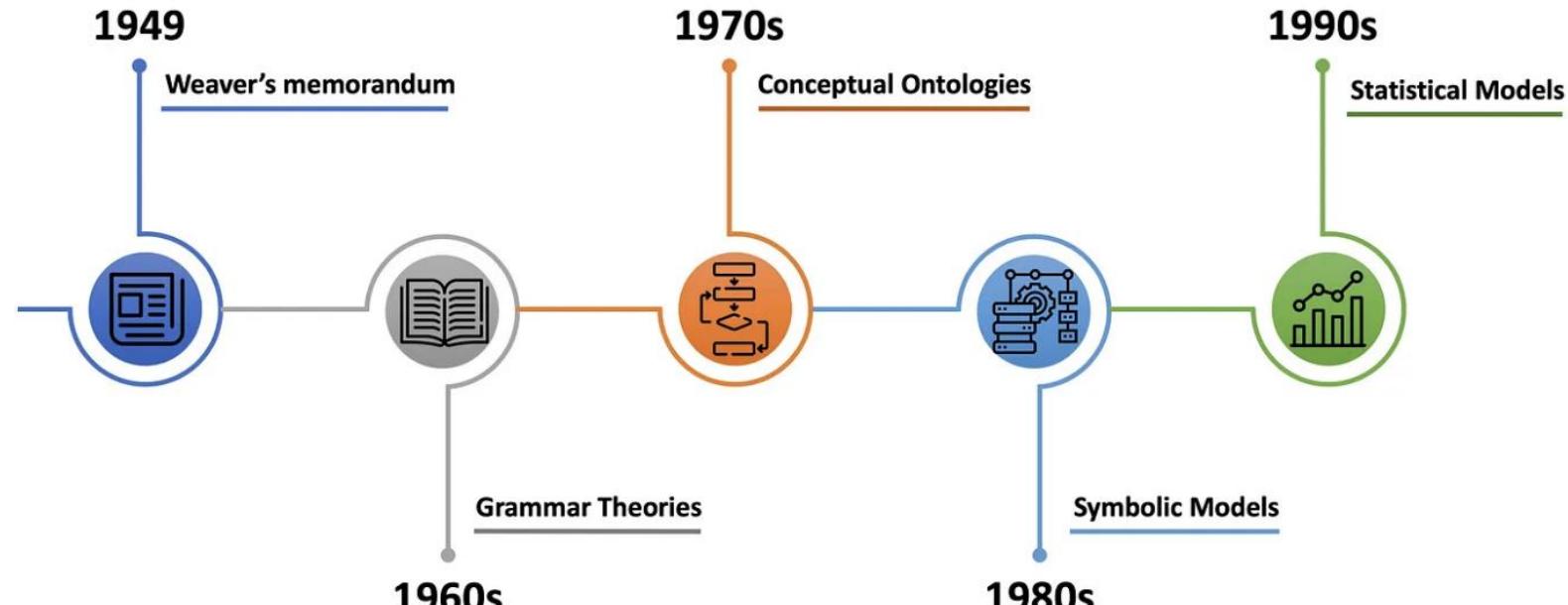
- Play Diplomacy game with Human players!

# What is NLP?



The landing page for DeepSeek-R1. At the top, there is a message: "DeepSeek-R1 is now live and open source, rivaling OpenAI's Model o1. Available on web, app, and API. Click for details." Below this is the DeepSeek logo, consisting of the word "deepseek" in a large blue sans-serif font with a small yellow flower icon above the letter "d". Underneath the logo is the tagline "Into the unknown". Two main call-to-action buttons are present: "Start Now" on the left and "Get DeepSeek App" on the right. Both buttons have descriptive text below them: "Free access to DeepSeek-V3. Experience the intelligent model." for "Start Now" and "Chat on the go with DeepSeek-V3. Your free all-in-one AI tool" for "Get DeepSeek App".

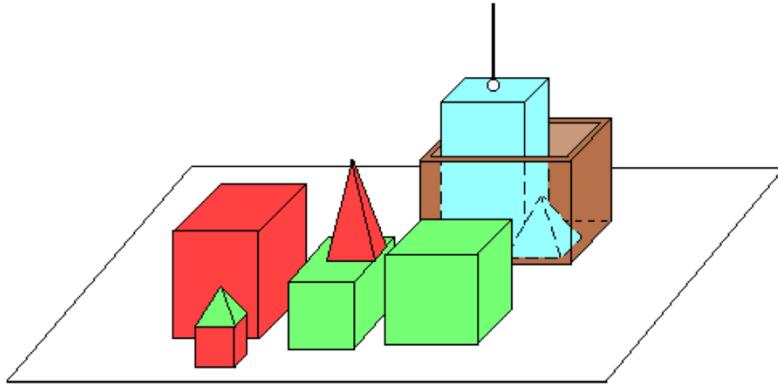
# NLP History 1



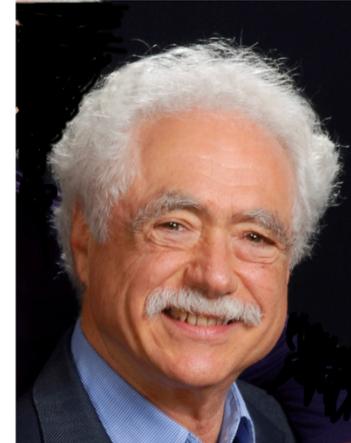
The big stages of NLP before the deep learning era.

<https://medium.com/@antoine.louis/a-brief-history-of-natural-language-processing-part-1-ffbcb937ebce>

# Rule Based NLP



SHRDLU,  
1968

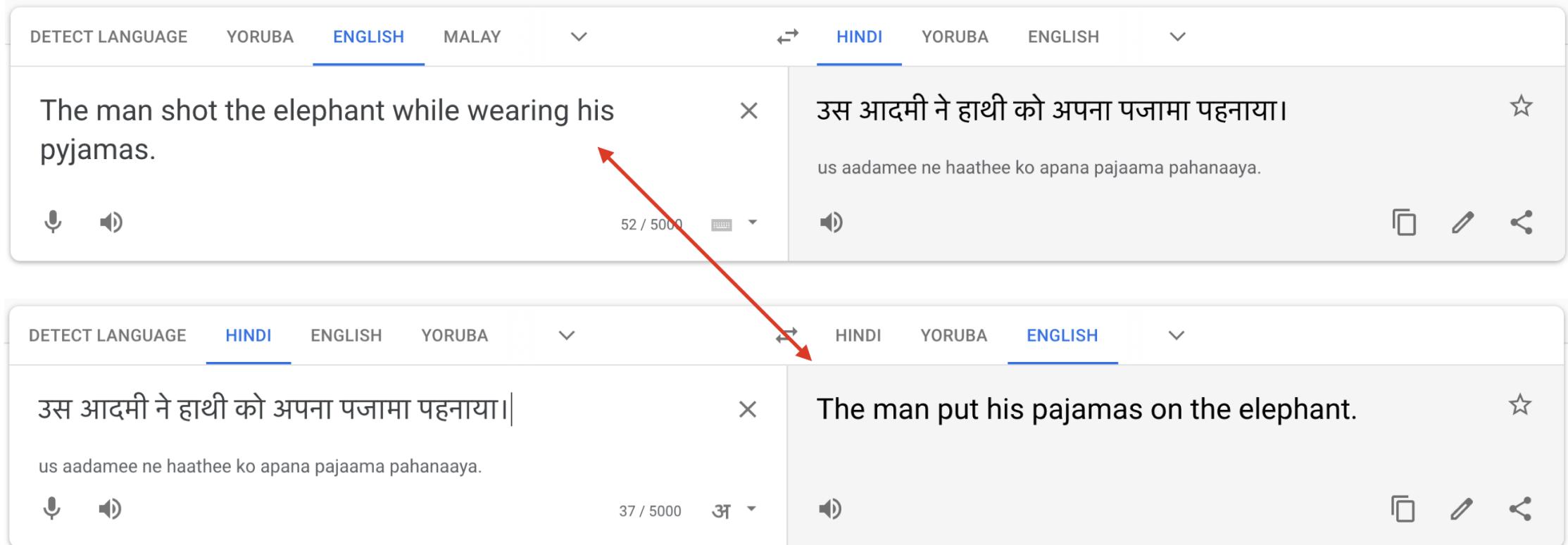


> How many red  
blocks are there?  
- THREE OF THEM

> Pick up the red  
block on top of a  
green one  
OK.

- Rule based system, require careful programming
- Limited Domains

# Statistical NLP



The screenshot shows two Google Translate interface windows side-by-side, illustrating a machine learning model's attempt at translating between English and Hindi.

**Top Window (English to Hindi):**

- Source: "The man shot the elephant while wearing his pyjamas."
- Target: "उस आदमी ने हाथी को अपना पजामा पहनाया।"
- Below the target text is the phonetic transcription: "us aadamee ne haathee ko apana pajaama pahanaaya."
- Bottom left: Microphone and speaker icons.
- Bottom right: Copy, edit, and share icons.

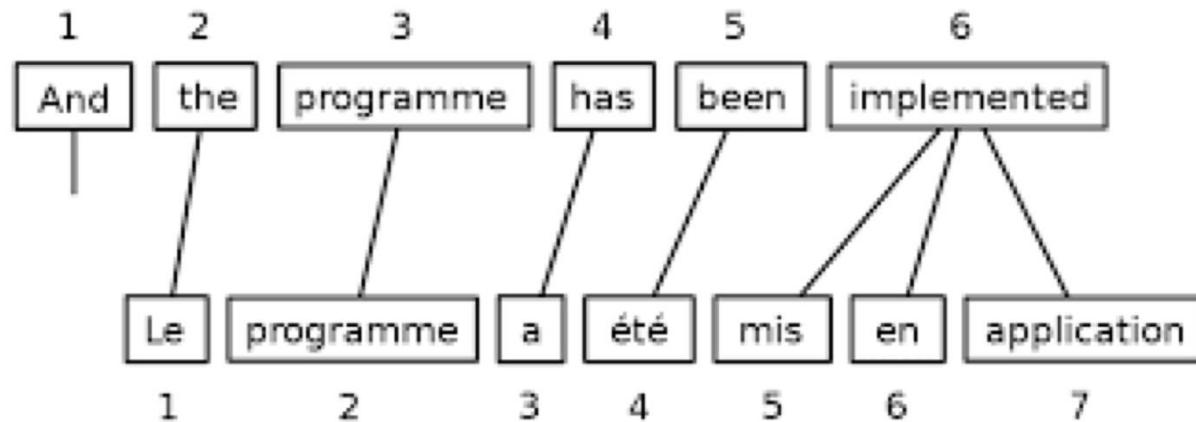
**Bottom Window (Hindi to English):**

- Source: "उस आदमी ने हाथी को अपना पजामा पहनाया।"
- Target: "The man put his pajamas on the elephant."
- Bottom left: Microphone and speaker icons.
- Bottom right: Copy, edit, and share icons.

A red arrow points from the English input of the top window to the Hindi output of the bottom window, highlighting a significant error in the generated translation.

# Statistical NLP

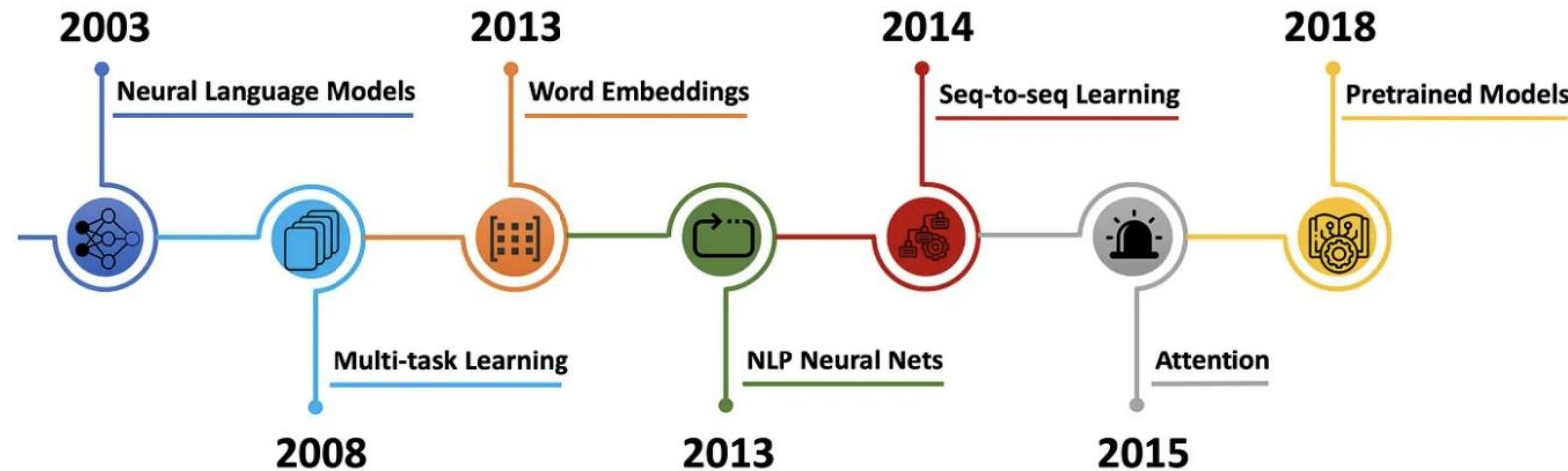
## IBM translation models



- Use machine learning approaches for NLP
- Statistical Machine Translation

# NLP History 2

## Part 2 — NLP during the Deep Learning Era



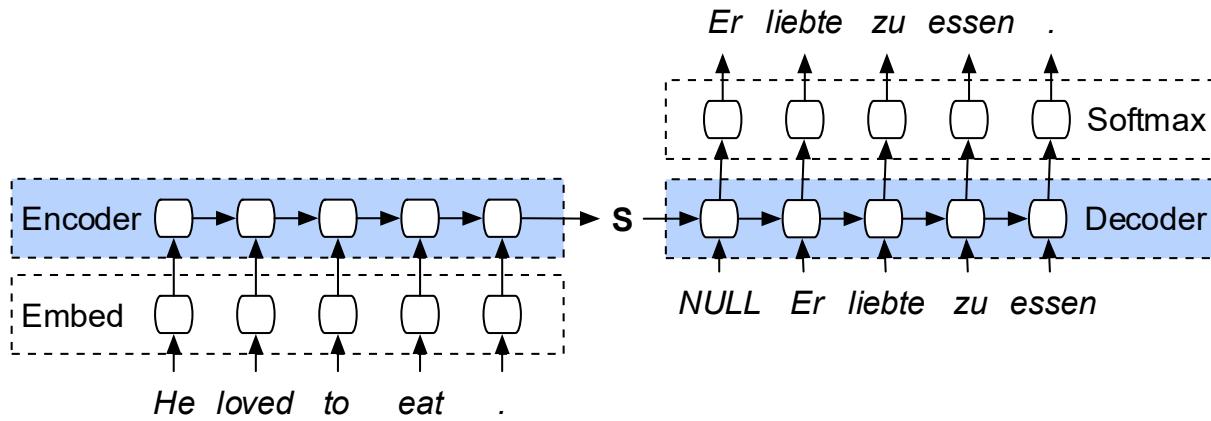
The big stages of NLP in the deep learning era.

<https://medium.com/@antoine.louis/a-brief-history-of-natural-language-processing-part-1-ffbcb937ebce>

# NLP with Deep Learning

- Significant Progress in NLP
- Lots of Compute resources, large corpora
- Little Feature Engineering

# NLP with Deep Learning – Neural Machine Translation



36M sentence pairs

Russian: Машинный перевод - это круто!

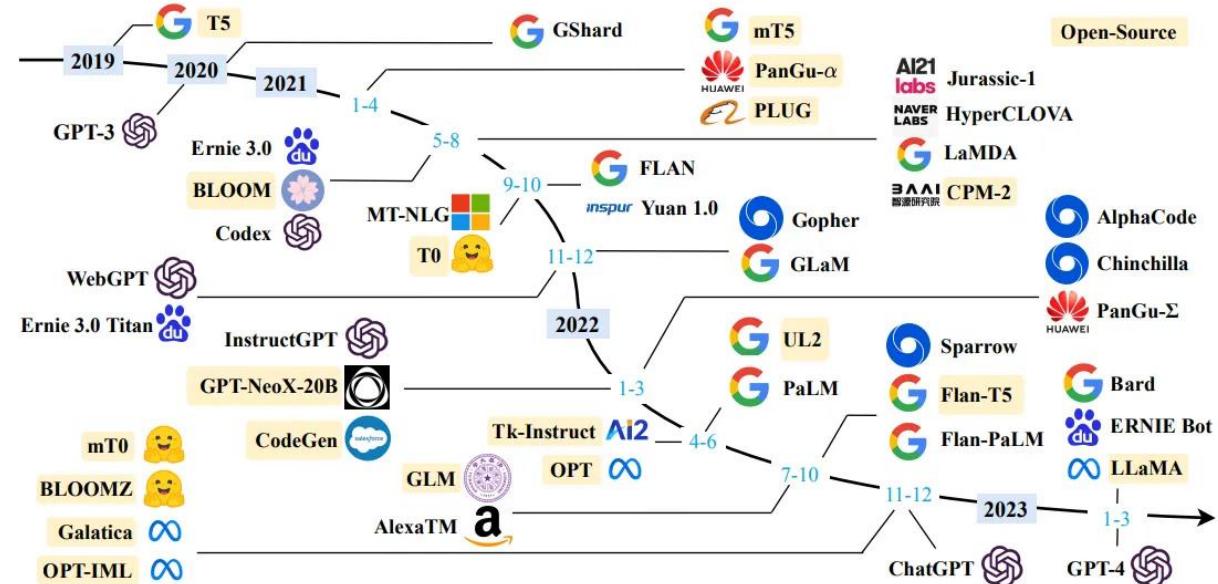


English: Machine translation is cool!

# NLP with LLMs

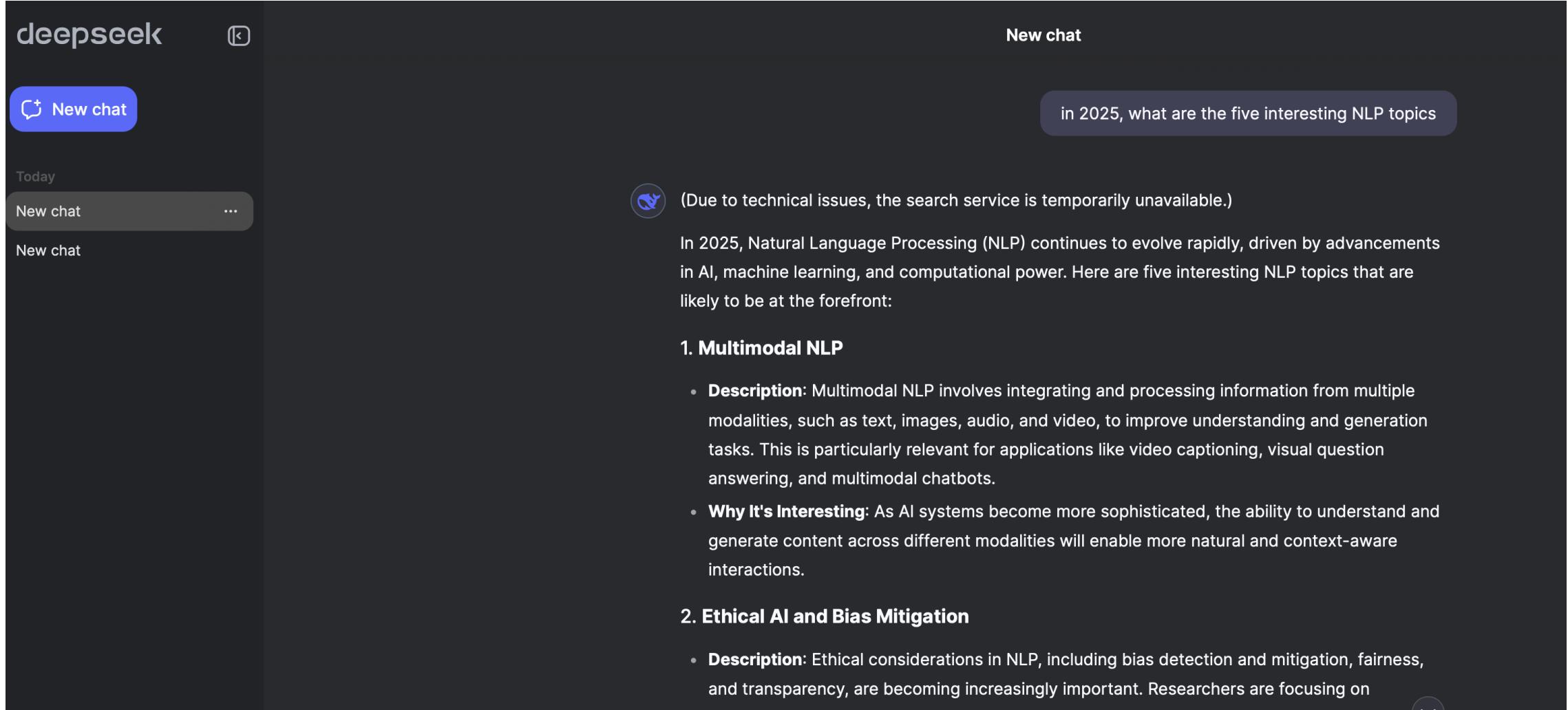


BERT, ELMo, ERNIE...



- Pre-trained on large corpus
- Fine-tuning / prompting on tasks
- One model for many tasks

# State-of-the-art LLMs



The screenshot shows a dark-themed AI interface named "deepseek". On the left, there's a sidebar with a "New chat" button and a "Today" section containing a "New chat" button and an ellipsis "...". The main area has a "New chat" button at the top right. A message from the AI says: "(Due to technical issues, the search service is temporarily unavailable.)". Below this, the AI provides a response: "In 2025, Natural Language Processing (NLP) continues to evolve rapidly, driven by advancements in AI, machine learning, and computational power. Here are five interesting NLP topics that are likely to be at the forefront:"

**1. Multimodal NLP**

- **Description:** Multimodal NLP involves integrating and processing information from multiple modalities, such as text, images, audio, and video, to improve understanding and generation tasks. This is particularly relevant for applications like video captioning, visual question answering, and multimodal chatbots.
- **Why It's Interesting:** As AI systems become more sophisticated, the ability to understand and generate content across different modalities will enable more natural and context-aware interactions.

**2. Ethical AI and Bias Mitigation**

- **Description:** Ethical considerations in NLP, including bias detection and mitigation, fairness, and transparency, are becoming increasingly important. Researchers are focusing on developing methods to identify and address biases in AI models, ensuring they are fair and responsible.

# Language Model Hallucinates



How many 'm's are in the word 'Weather'?



There is one 'm' in the word 'Weather'.



Are you sure?



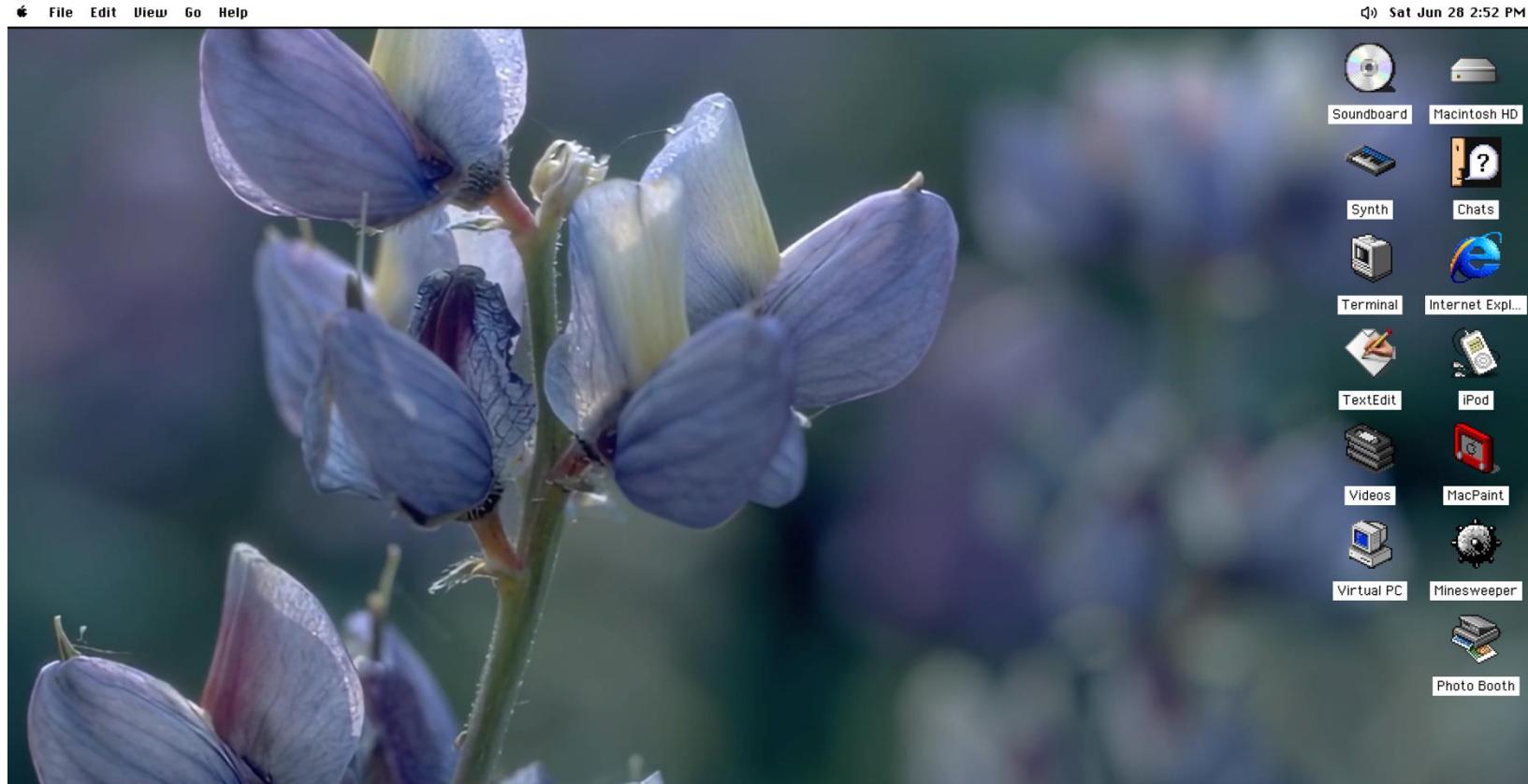
Apologies for the oversight. You're right; there are no 'm's in the word 'Weather'. Thank you for pointing it out.

# LLM in 2025: AI + Coding



- Code editor + LLM

# LLM in 2025: AI + Coding



- Operating system written by Cursor Only!!
- Do we still need software engineer ???

# Outline

---

- Part 1: NLP Background
- Part 2: Background and Transformer
- Part 3: Morden LLMs and ChatGPT
- Part 4: Large Reasoning Models and Deepseek R1

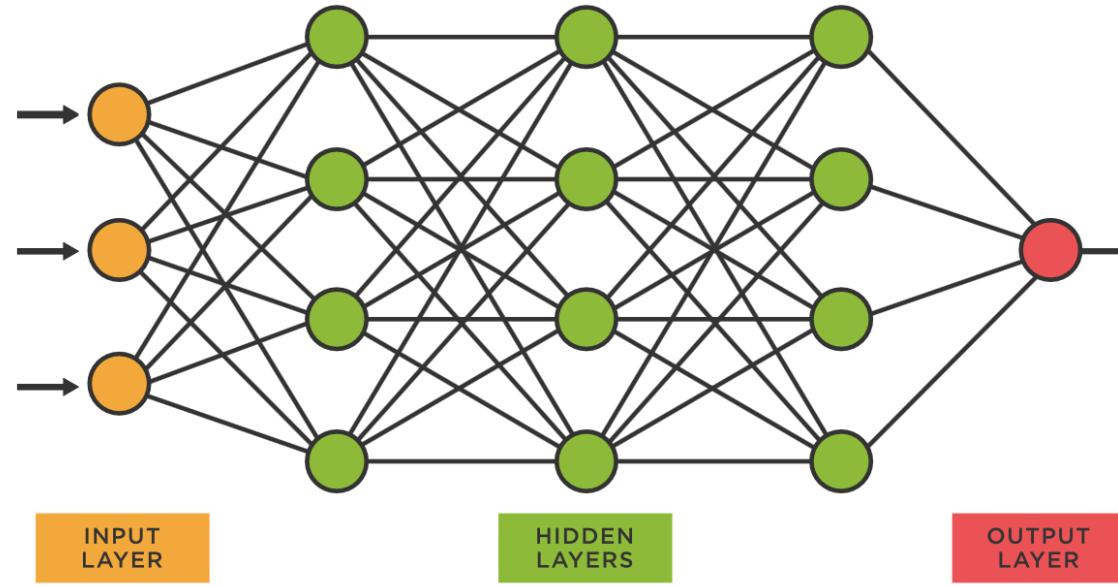
# Basics: Word Embeddings

- **Learned** dense vectors from text (i.e. corpora) for representing words
- Input:
  - A pre-defined vocabulary  $\mathbf{V}$
  - Dimension of word vectors  $\mathbf{d}$  (e.g., 300)
  - Text corpora (e.g., Wikipedia, Twitter, Common Crawl)
- Output:  $f : V \rightarrow \mathbb{R}^d$ 
  - Each word is represented by a dense vector
  - Note: each dimension does not have a specific meaning

$$v_{\text{cat}} = \begin{pmatrix} -0.224 \\ 0.130 \\ -0.290 \\ 0.276 \end{pmatrix} \quad v_{\text{dog}} = \begin{pmatrix} -0.124 \\ 0.430 \\ -0.200 \\ 0.329 \end{pmatrix}$$

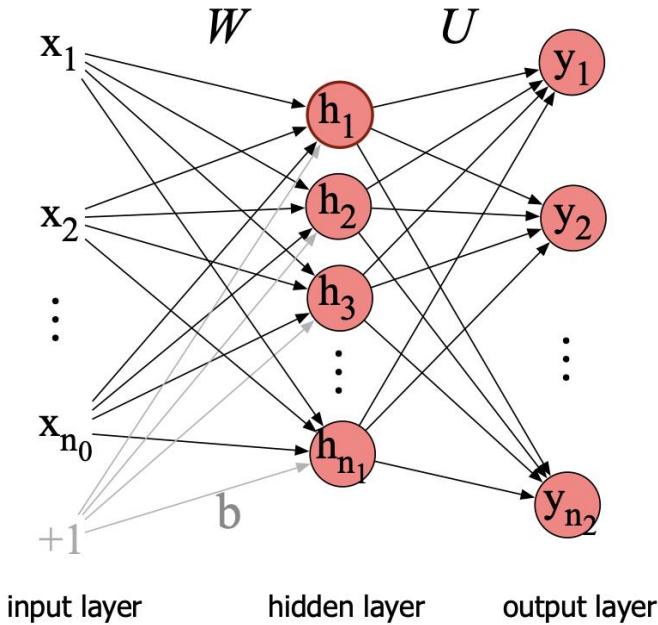
$$v_{\text{the}} = \begin{pmatrix} 0.234 \\ 0.266 \\ 0.239 \\ -0.199 \end{pmatrix} \quad v_{\text{language}} = \begin{pmatrix} 0.290 \\ -0.441 \\ 0.762 \\ 0.982 \end{pmatrix}$$

# Basics: Neural Network



- A network of small computing units
- **Deep learning:** Modern neural network (have many layers)
- Possible to learn any function

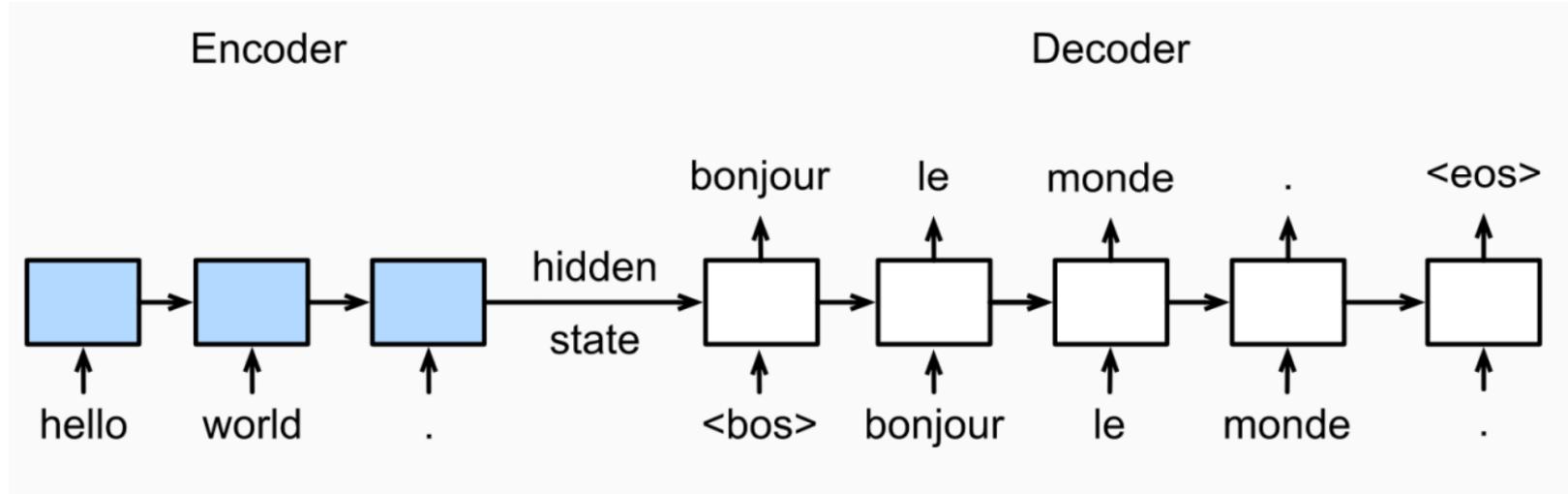
# Basics: Feedforward Neural Networks



- Sometimes called **multi-layer perceptron (MLPs)**
- Input units, hidden units, output units
- Fully-connected: each unit in each layer takes input from all units in the previous layer

$$\mathbf{h} = \sigma(\mathbf{Wx} + \mathbf{b})$$

# Neural Sequence Modeling



- Encoder-decoder Structure

# Transformers

---

## Attention Is All You Need

---

**Ashish Vaswani\***

Google Brain

avaswani@google.com

**Noam Shazeer\***

Google Brain

noam@google.com

**Niki Parmar\***

Google Research

nikip@google.com

**Jakob Uszkoreit\***

Google Research

usz@google.com

**Llion Jones\***

Google Research

llion@google.com

**Aidan N. Gomez\*** †

University of Toronto

aidan@cs.toronto.edu

**Łukasz Kaiser\***

Google Brain

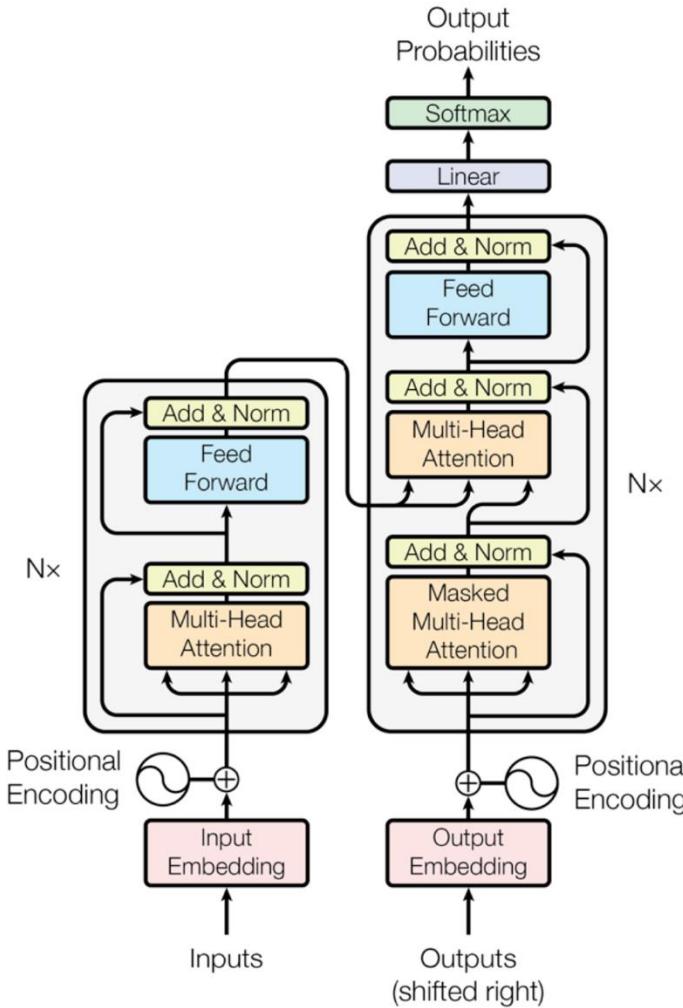
lukaszkaiser@google.com

**Illia Polosukhin\*** ‡

illia.polosukhin@gmail.com

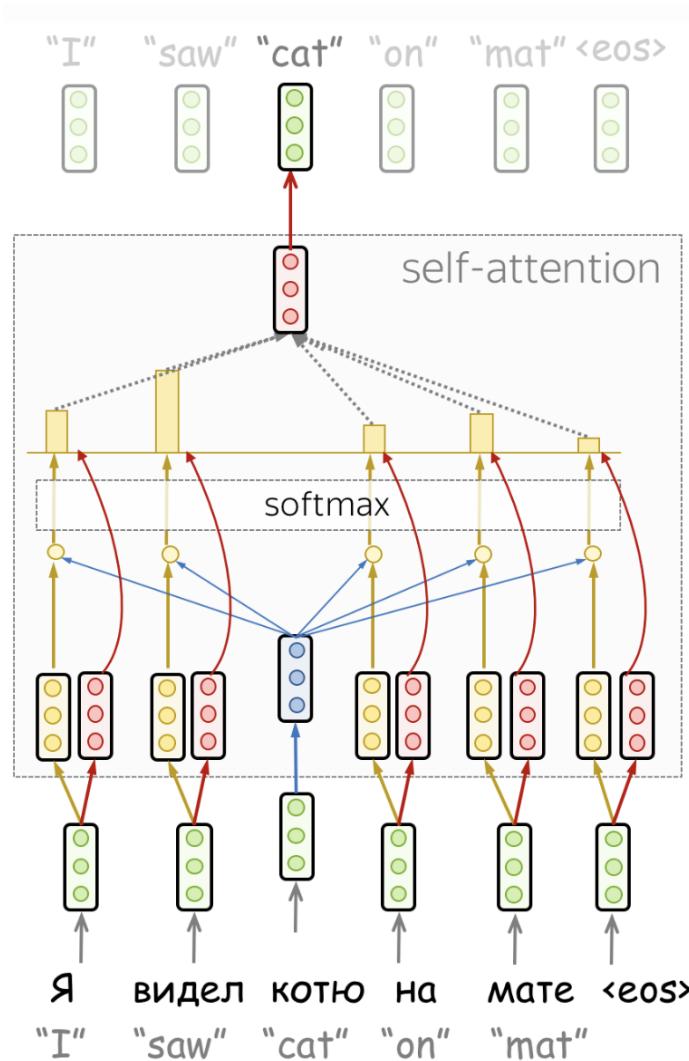
(Vaswani et al., 2017)

# Transformers



- Transformer Encoder + Decoder
- Replacement of Seq2seq
- No recurrent structures!
- **Key: Multi-head; Self-Attention**

# Self-Attention



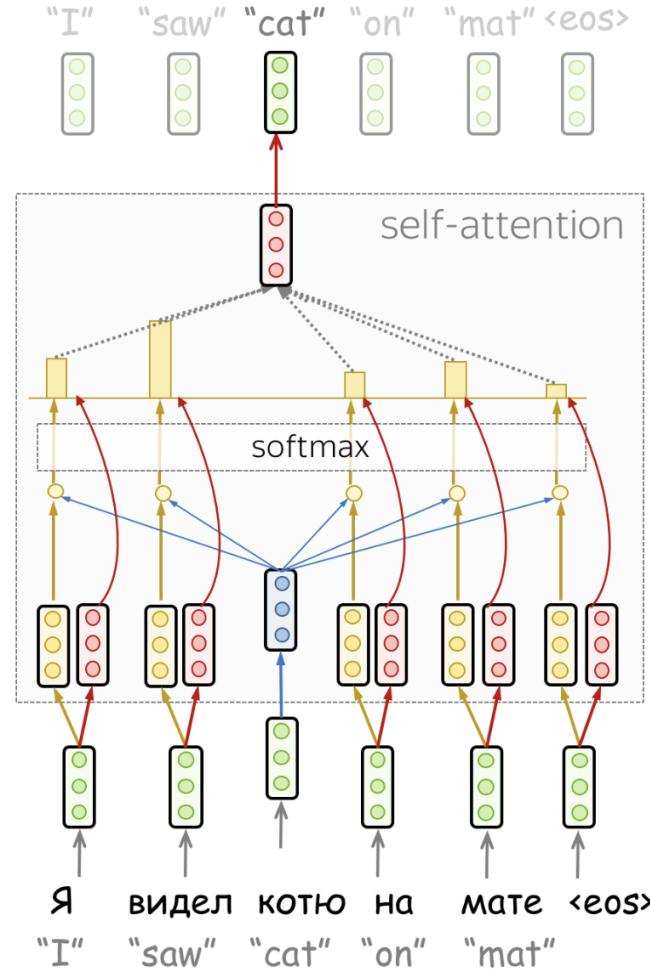
$$\text{Attention}(q, k, v) = \underbrace{\text{softmax}\left(\frac{qk^T}{\sqrt{d_k}}\right)v}_{\text{Attention weights}}$$

from                  to

vector dimensionality of K, V

- **From:** each state (i.e. input token)
- **To:** All other tokens in the sequence

# Query, Key and Value in Self-Attention



Each vector receives three representations (“roles”)

$$[W_Q] \times \begin{array}{|c|c|c|}\hline & \text{green} \\ \hline & \text{green} \\ \hline & \text{green} \\ \hline\end{array} = \begin{array}{|c|c|c|}\hline & \text{blue} \\ \hline & \text{blue} \\ \hline & \text{blue} \\ \hline\end{array}$$

**Query:** vector from which the attention is looking

“Hey there, do you have this information?”

$$[W_K] \times \begin{array}{|c|c|c|}\hline & \text{green} \\ \hline & \text{green} \\ \hline & \text{green} \\ \hline\end{array} = \begin{array}{|c|c|c|}\hline & \text{yellow} \\ \hline & \text{yellow} \\ \hline & \text{yellow} \\ \hline\end{array}$$

**Key:** vector at which the query looks to compute weights

“Hi, I have this information – give me a large weight!”

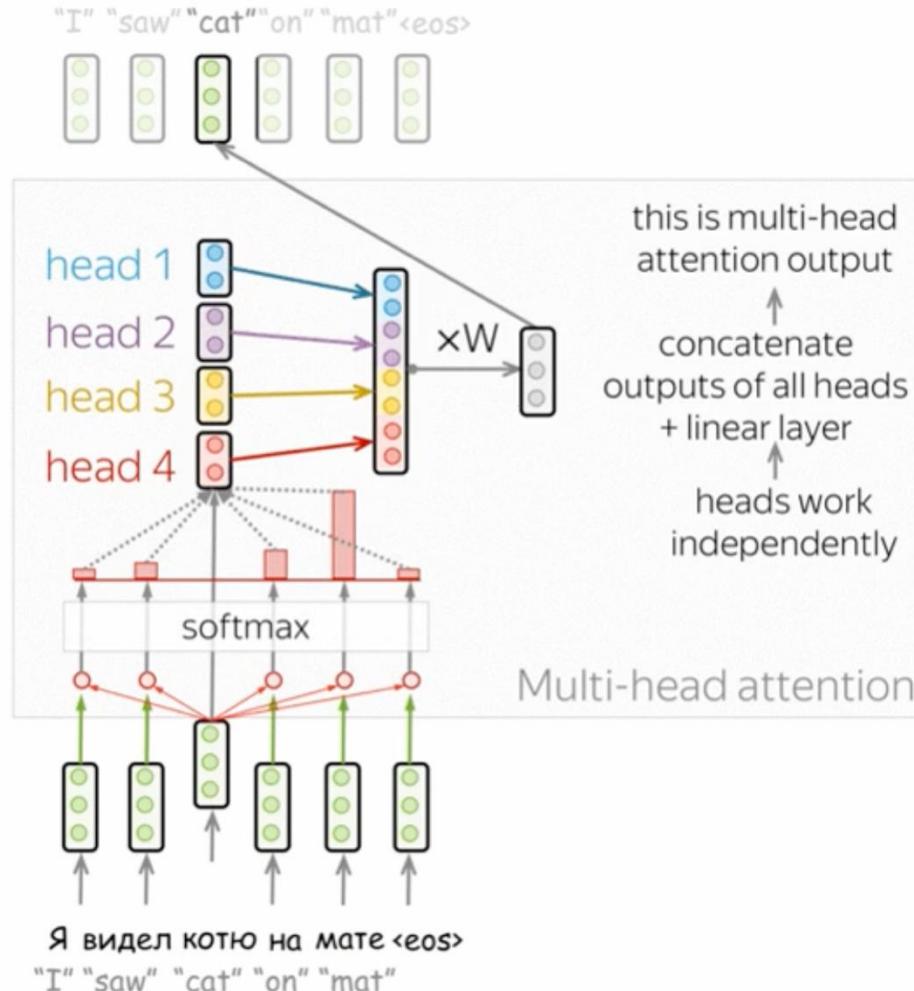
$$[W_V] \times \begin{array}{|c|c|c|}\hline & \text{green} \\ \hline & \text{green} \\ \hline & \text{green} \\ \hline\end{array} = \begin{array}{|c|c|c|}\hline & \text{pink} \\ \hline & \text{pink} \\ \hline & \text{pink} \\ \hline\end{array}$$

**Value:** their weighted sum is attention output

“Here’s the information I have!”

- **Query:** asking for information
- **Key:** saying it has some information
- **Value:** giving the information

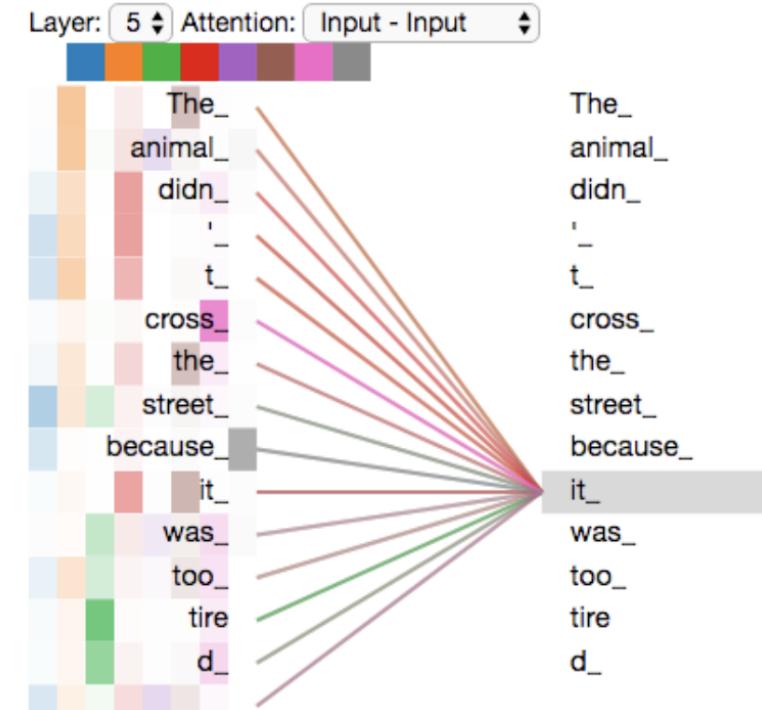
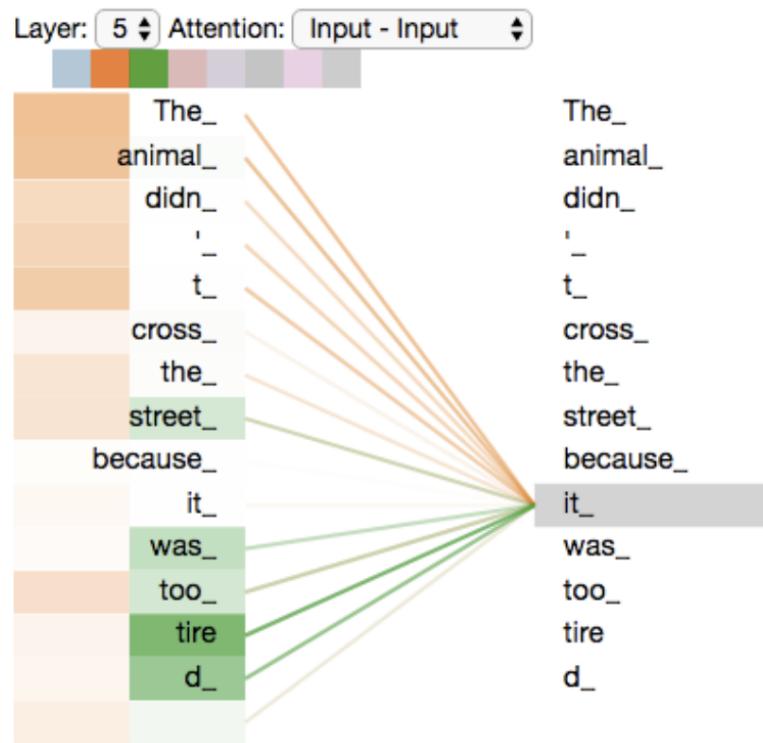
# Multi-Head Attention



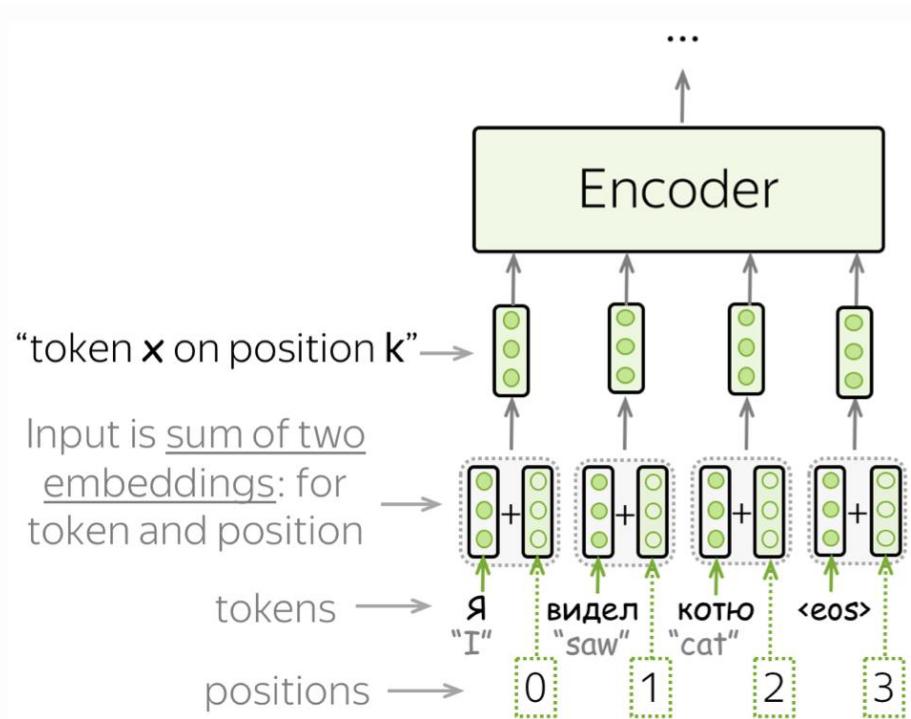
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_n)W_o,$$

$$\text{head}_i = \text{Attention}(QW_Q^i, KW_K^i, VW_V^i)$$

# Multi-Head Attention

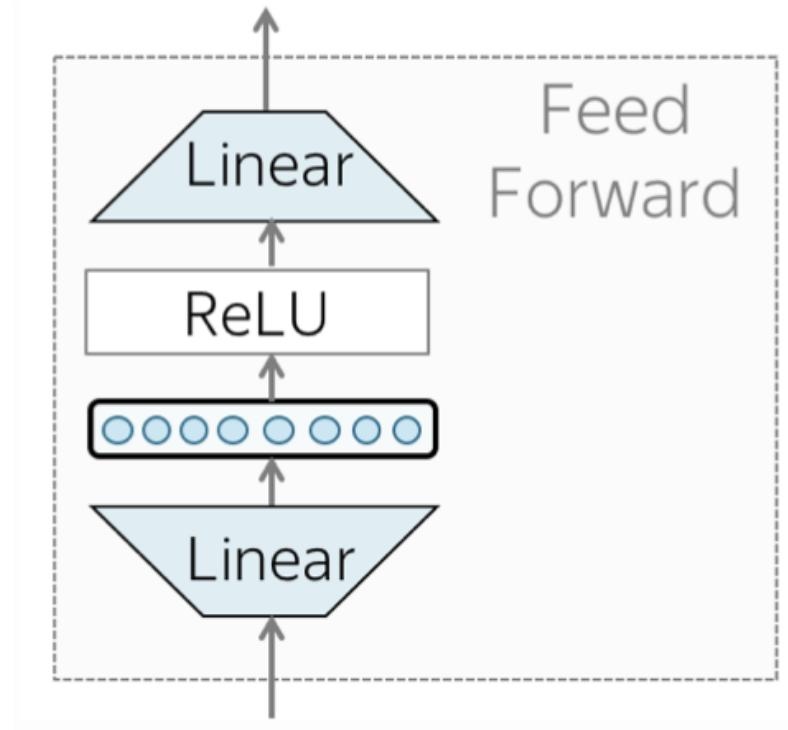


# Positional Encoding



- Transformer does not have recurrence
- Include order of tokens!
- People just use a learnable embedding for every unique position

# Feed-forward Blocks

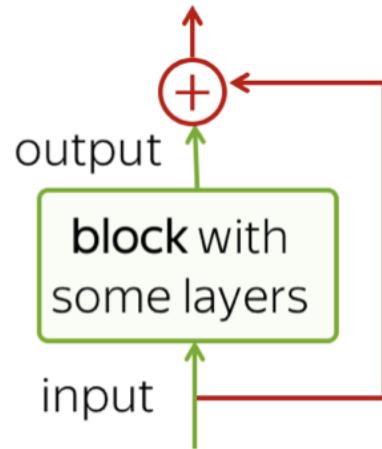


- There is no elementwise nonlinearities in self-attention; stacking more self-attention just re-average value vectors

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2.$$

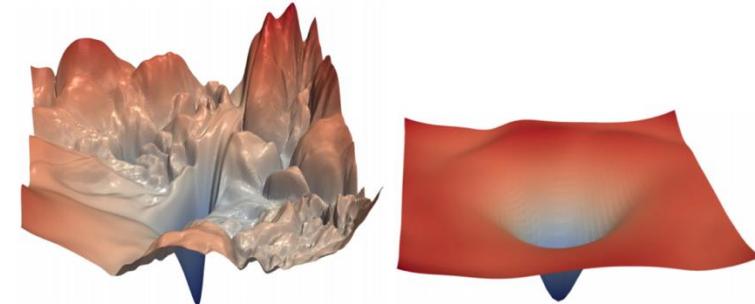
- **Attention:** Gather information from other tokens
- **FFN:** Process this information

# Residual Connections



**Residual connection:**  
add a block's input to  
its output

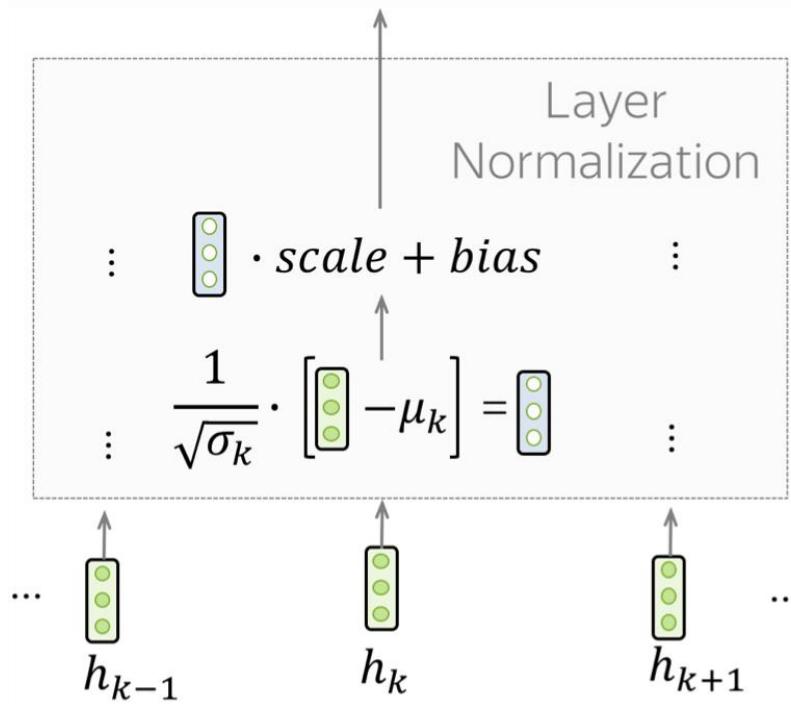
- Allow stacking multiple layers



[no residuals]

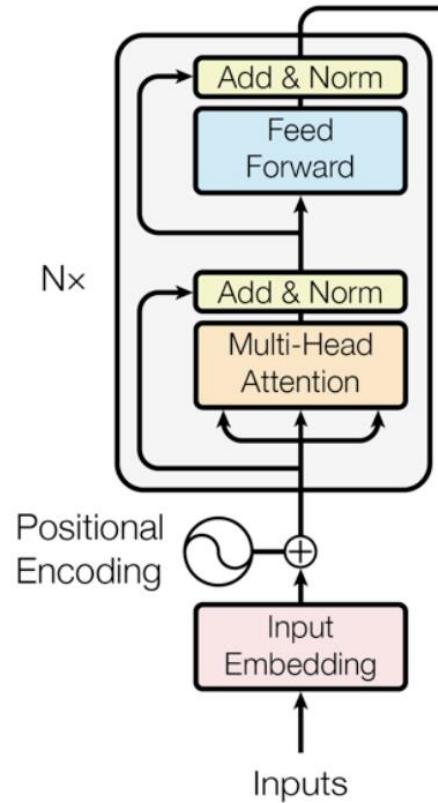
[Loss landscape visualization,  
[Li et al., 2018](#), on a ResNet]

# Layer Norm



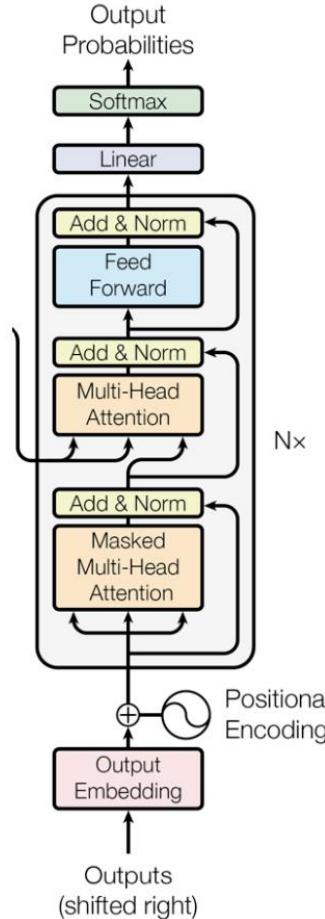
- A trick to help models train faster
- Normalize vector representation in batch
- Idea: cut down on uninformative variation in hidden vector values

# Transformers encoder



- Each encoder layer has two sub-layers:
  - A multi-head self-attention layer
  - A feedforward layer
- Residual connection
- Layer normalization

# Transformers decoder



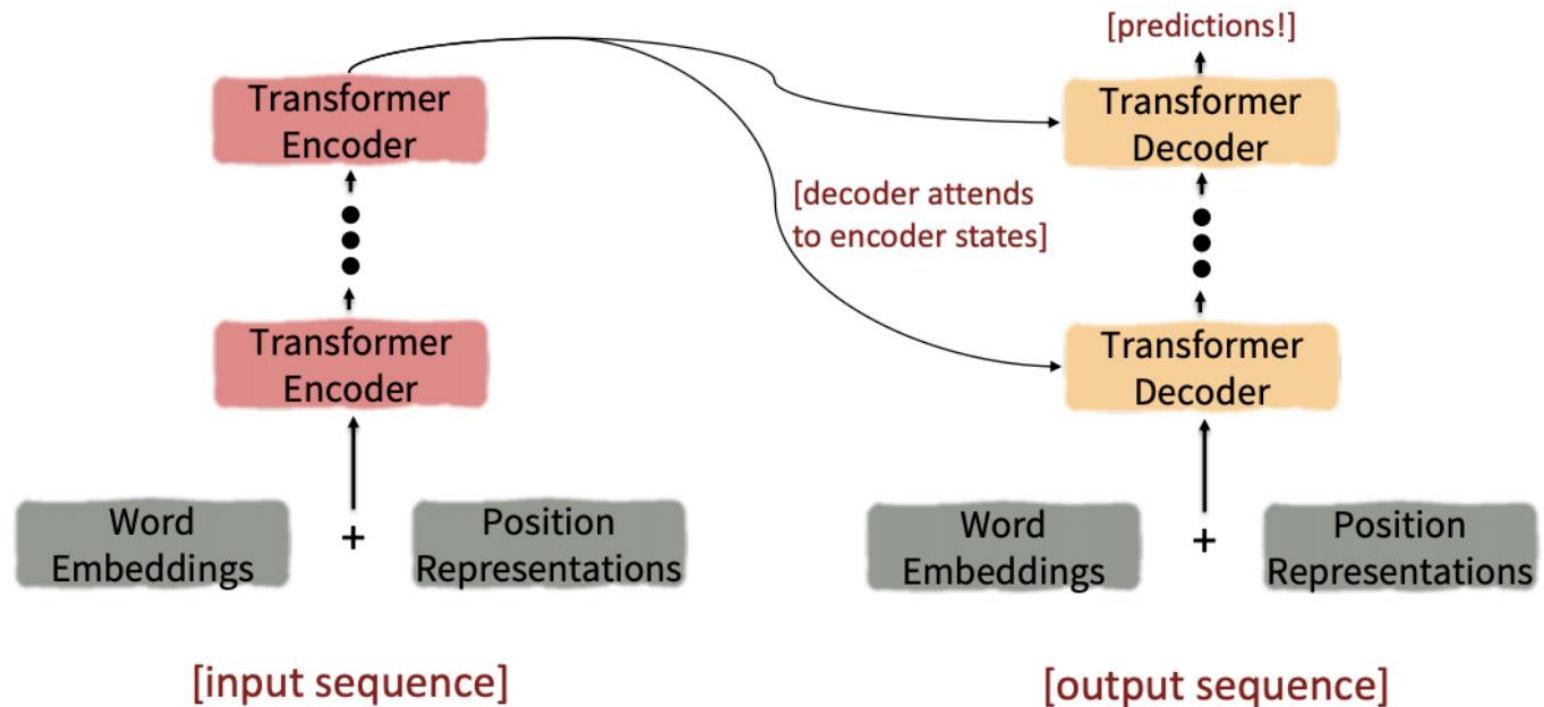
- Each decoder layer has three sub-layers:
  - A **masked** multi-head self-attention layer
  - A multi-head **cross-attention** layer
  - A feedforward layer
- Residual connection
- Layer normalization

# Training Transformer

- Training data: Parallel Corpus
- Loss: Cross Entropy
- Back-propagate gradients through both encoder and decode

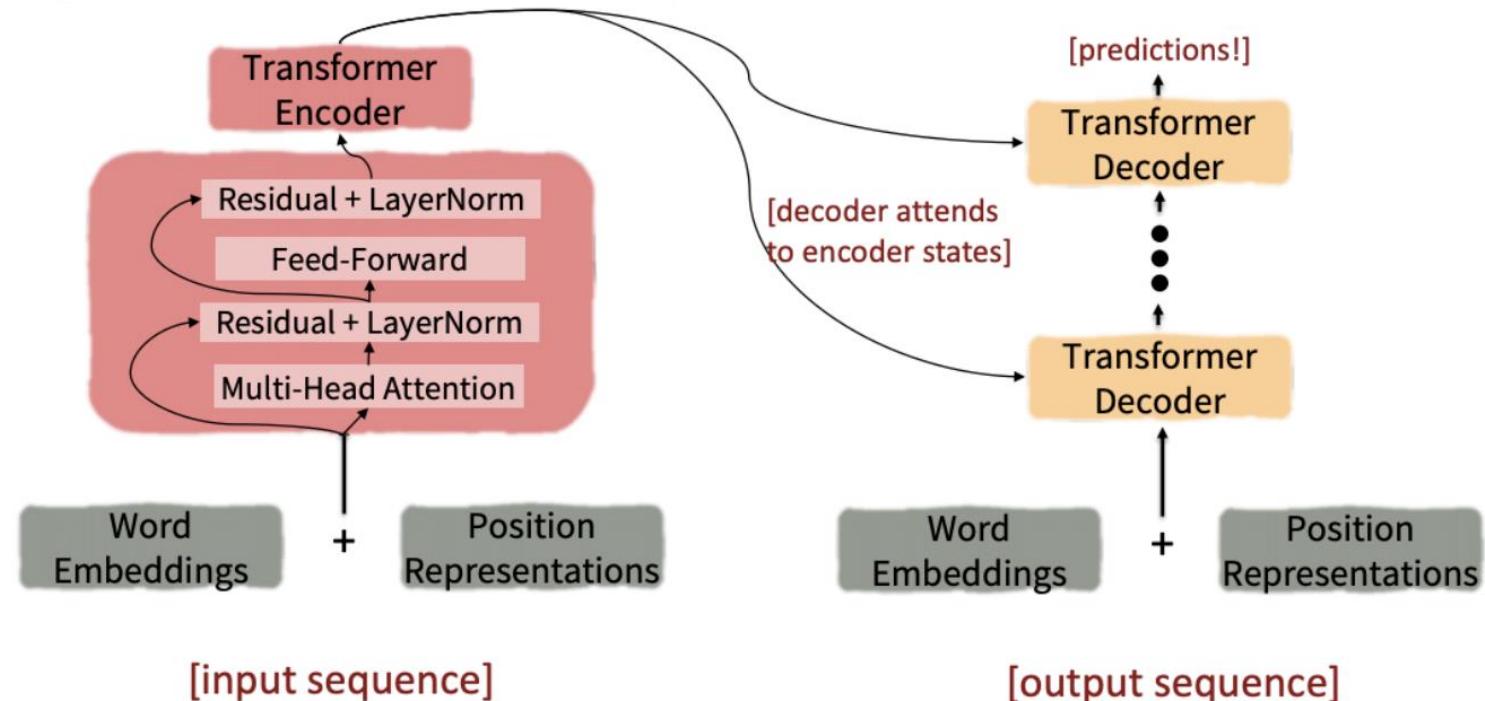


# Summary: Transformer



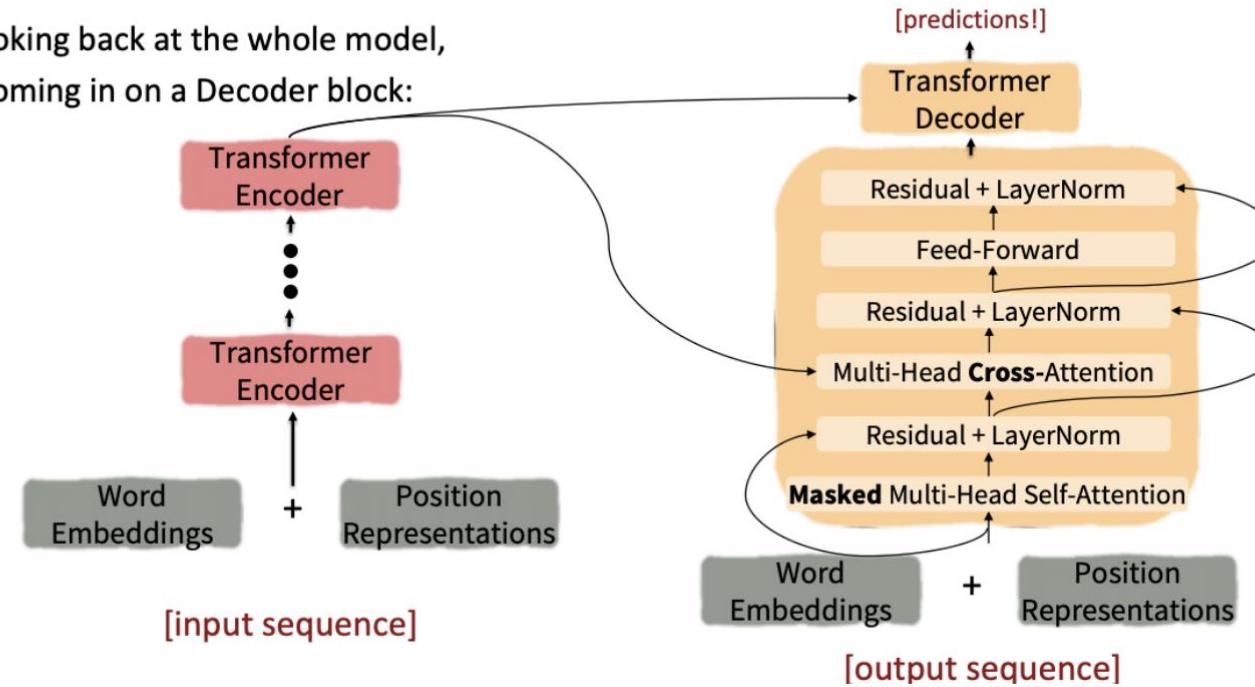
# Summary: Transformer

Looking back at the whole model, zooming in on an Encoder block:



# Summary: Transformer

Looking back at the whole model,  
zooming in on a Decoder block:

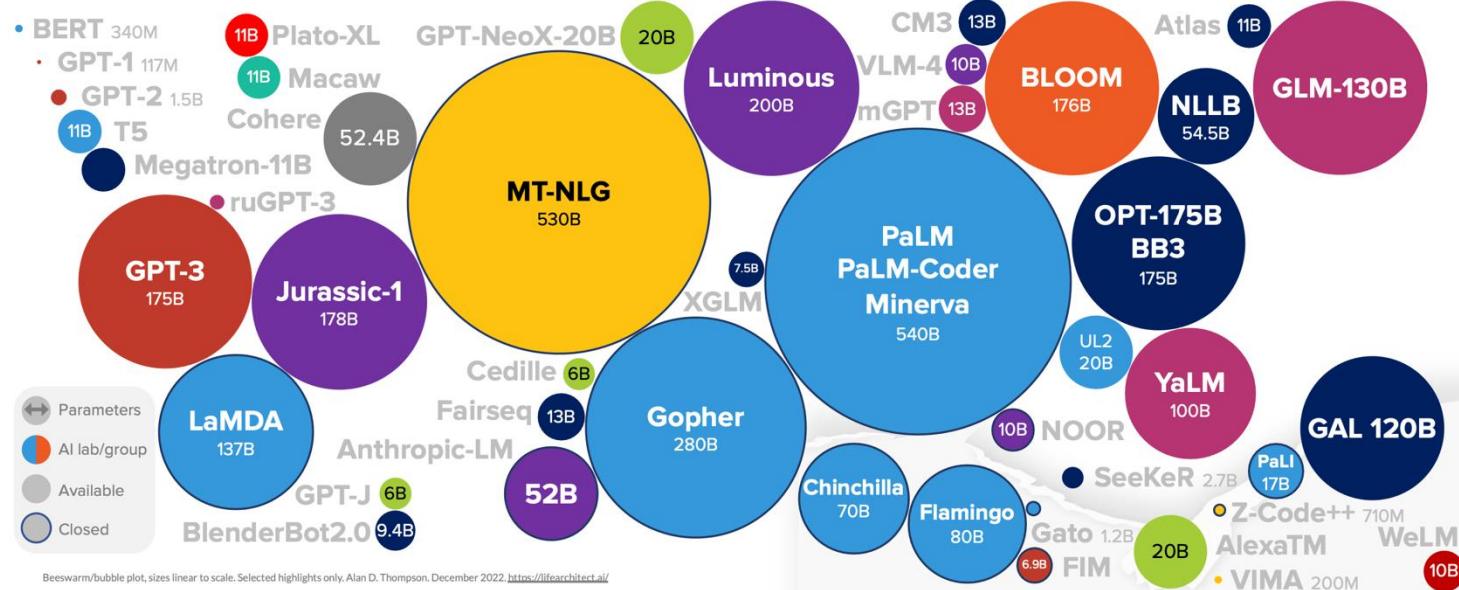


# Outline

---

- Part 1: NLP Background
- Part 2: Background and Transformer
- Part 3: Modern LLMs and ChatGPT
- Part 4: Large Reasoning Models and Deepseek R1

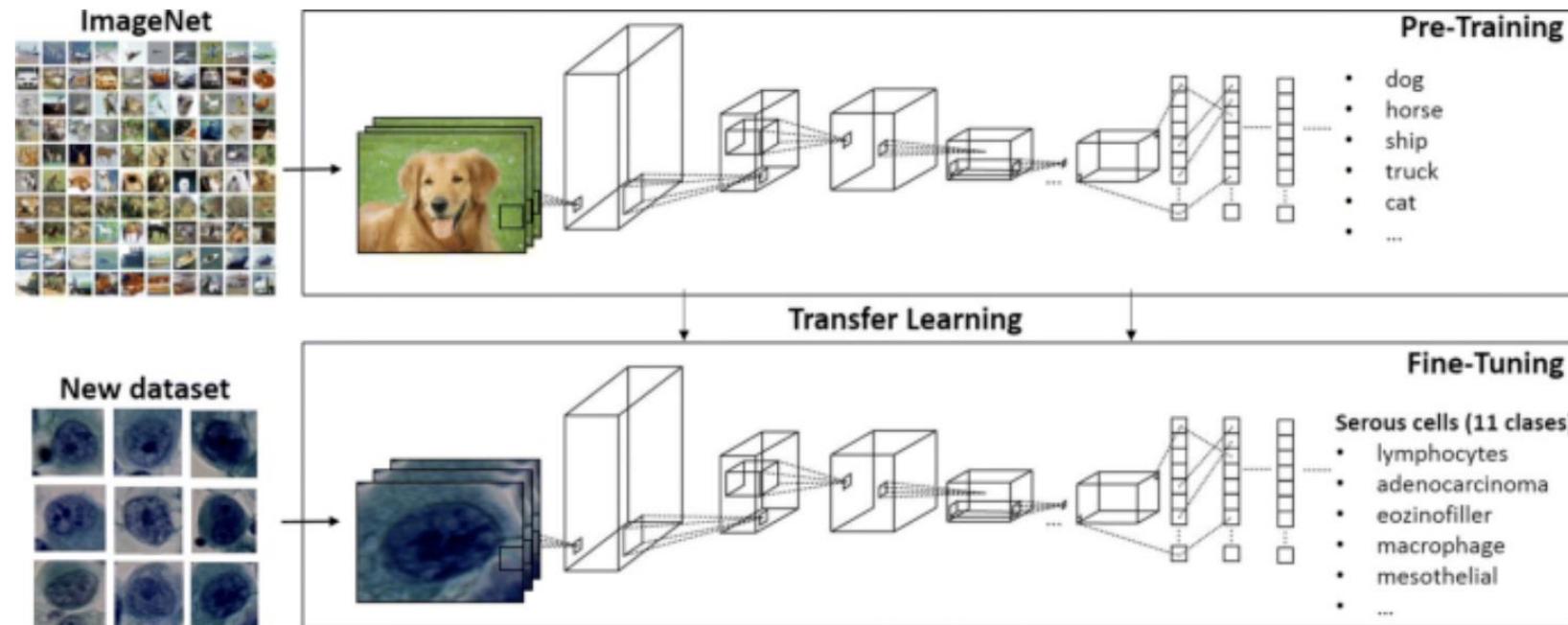
# Transformer Family



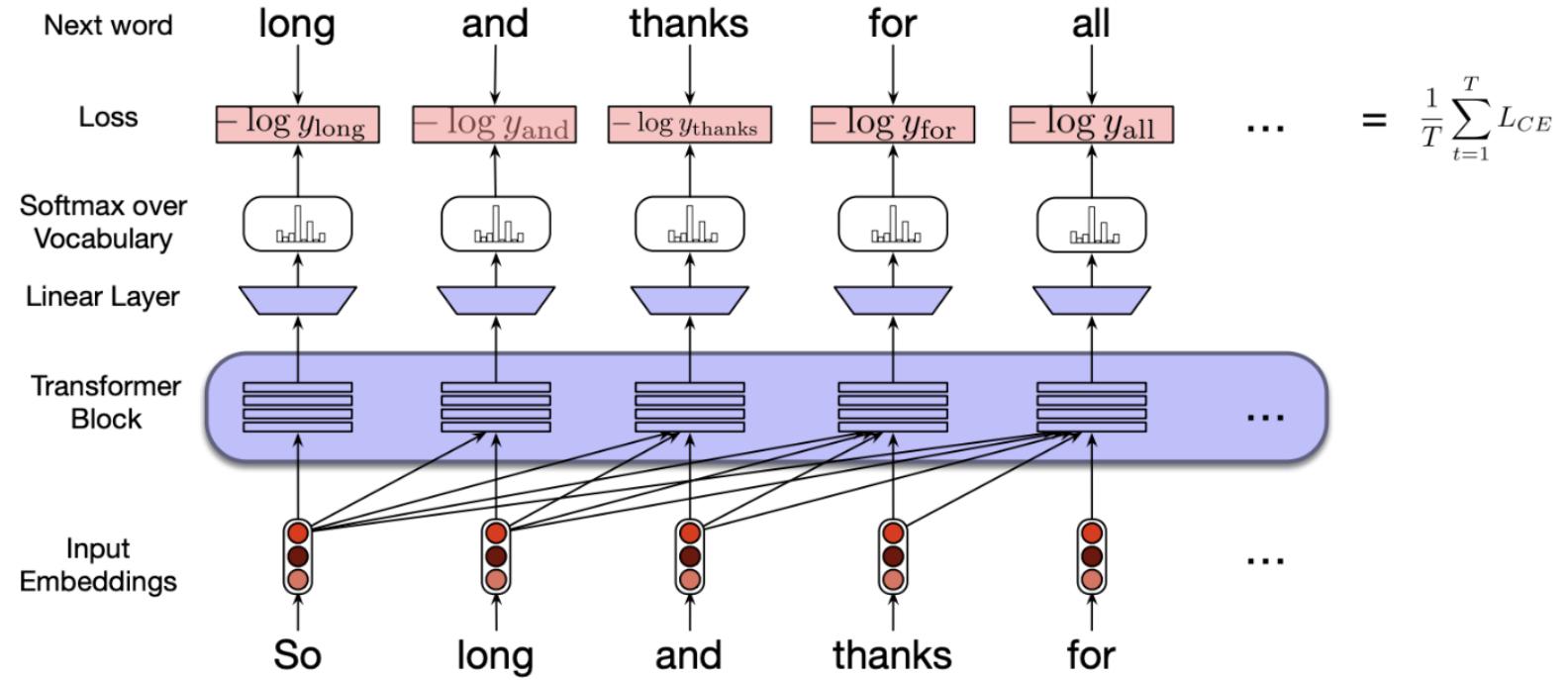
- Transformer Encoder: BERT, RoBERTa
- Transformer Decoder: GPT, PaLM
- Transformer Encoder-Decoder: T5

# Pre-training and Fine-Tuning

- **Pre-train** on a large dataset for task X
- **Fine-tune** on a (smaller) dataset for task Y
- **Goal:** Learn neural representations from X that benefit Y



# GPT



- Transformer decoder only
- Use **Language Modeling** as a pre-training objective

# GPT-2

GPT-2



February 14, 2019

[Read paper ↗](#)  
[View code ↗](#)

Context size = 1024



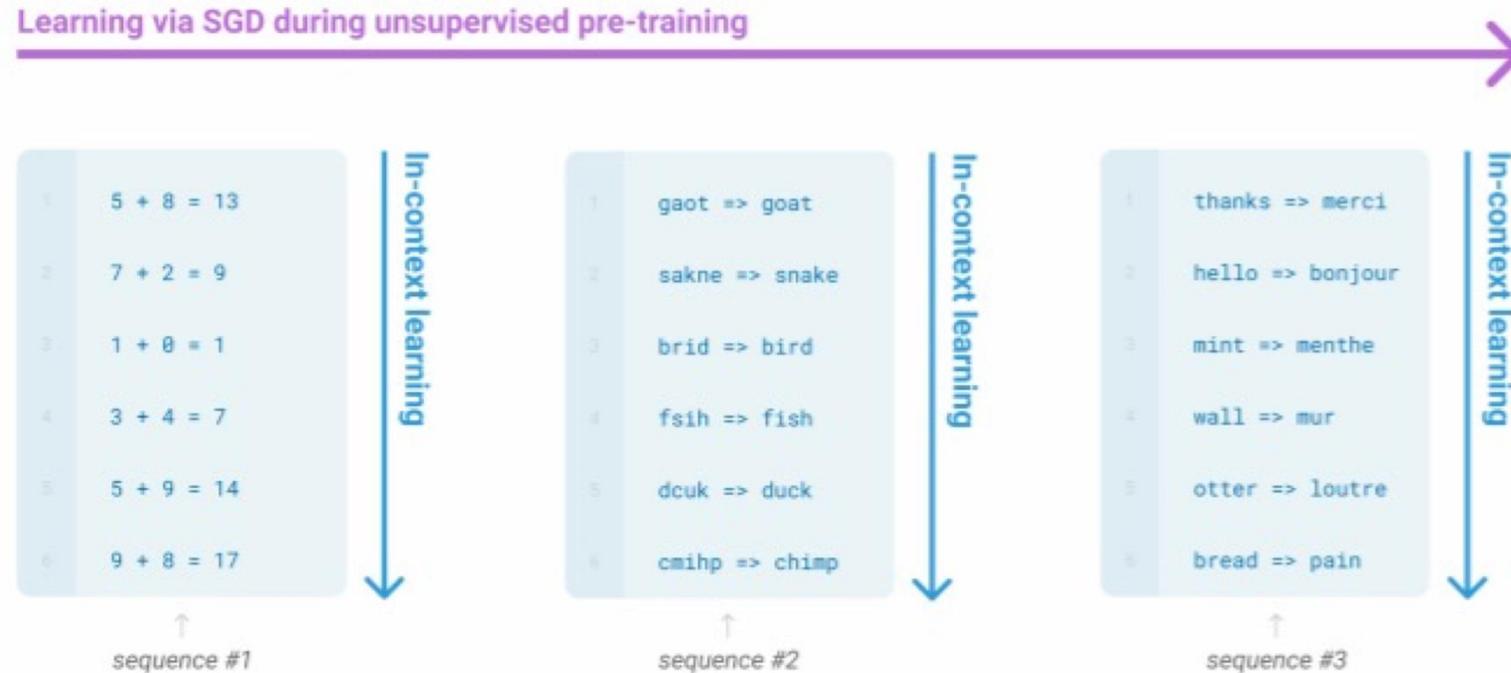
.. trained on 40Gb of Internet text ..

# GPT-3, very large models

- So far, we have interacted with pre-trained models
  - Sample from the distributions they define
  - Fine-tune them on a task we care about, and take their predictions
- Very large language models seem to perform learning **without gradient steps** simply from examples you provide within their contexts
- **GPT-3 has 175 billion parameters.** Previous largest model had 11 billion parameters

# GPT-3/4, in-context learning

- Very large language models seem to perform learning **without gradient steps** simply from examples you provide within their contexts



# Language modeling != assisting users

- Language models are not aligned with user intent

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION GPT-3

Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.

# Language modeling != assisting users

- Language models are not aligned with user intent

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION

**Human**

A giant rocket ship blasted off from Earth carrying astronauts to the moon. The astronauts landed their spaceship on the moon and walked around exploring the lunar surface. Then they returned safely back to Earth, bringing home moon rocks to show everyone.

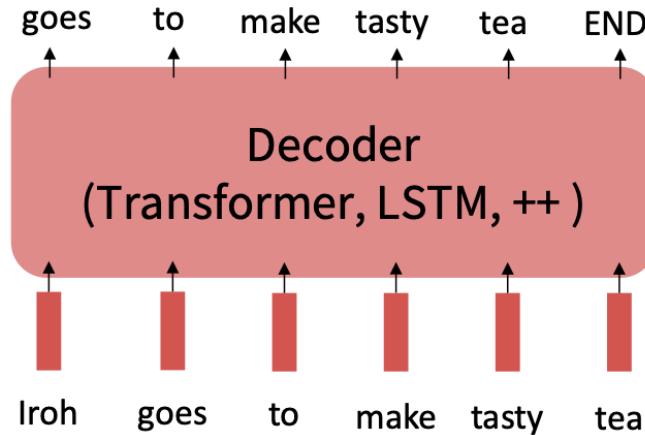
We need fine-tuning to rescue!

# Scaling up finetuning

- **Fine-tune** on many tasks

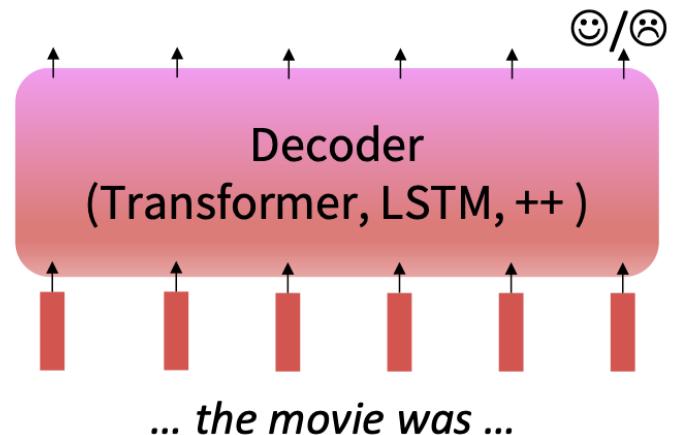
## Step 1: Pretrain (on language modeling)

Lots of text; learn general things!



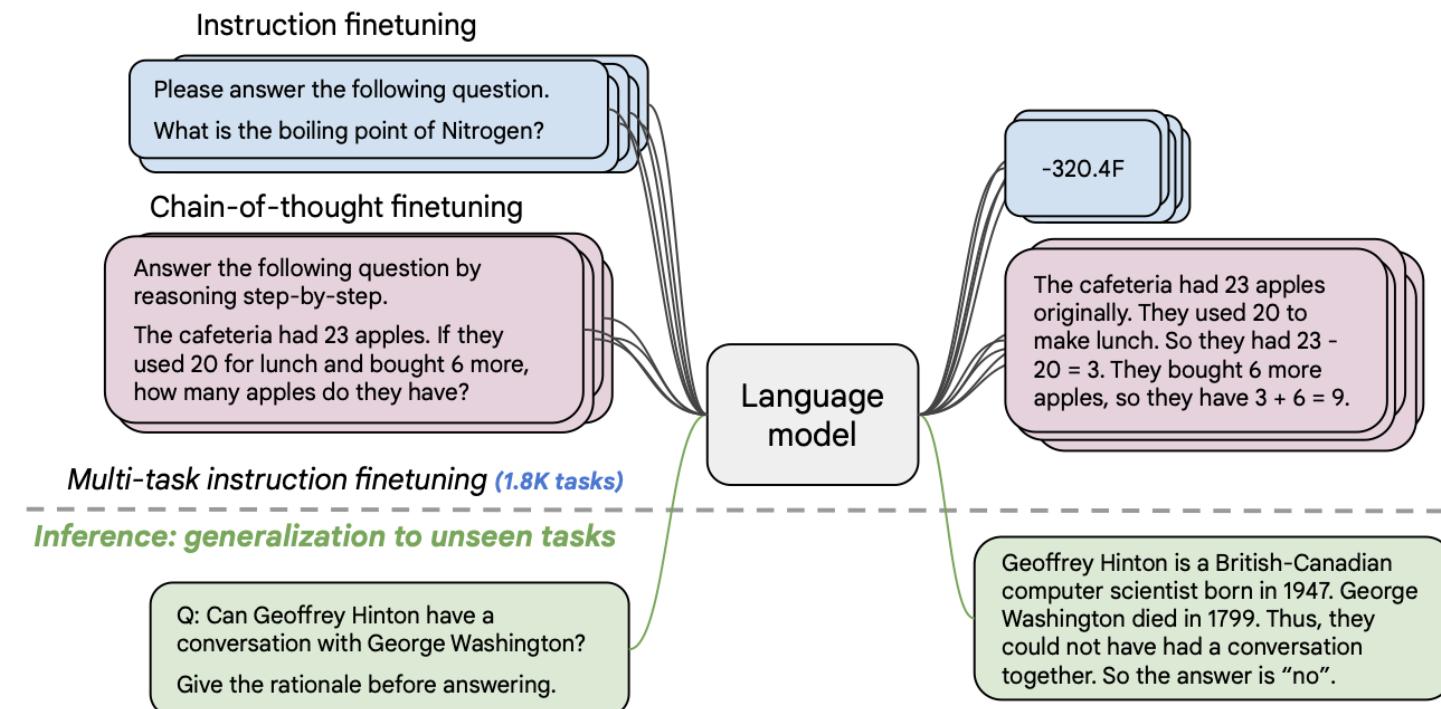
## Step 2: Finetune (on many tasks)

Not many labels; adapt to the tasks!



# Instruction finetuning

- **Collect examples** of (instruction, output) pairs across many tasks and fine-tune a LM
- Evaluate on unseen tasks



# Instruction finetuning (Flan-T5)

## Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:

- (A) They will discuss the reporter's favorite dishes
- (B) They will discuss the chef's favorite dishes
- (C) Ambiguous

A: Let's think step by step.

## Before instruction finetuning

The reporter and the chef will discuss their favorite dishes.

The reporter and the chef will discuss the reporter's favorite dishes.

The reporter and the chef will discuss the chef's favorite dishes.

The reporter and the chef will discuss the reporter's and the chef's favorite dishes.

✖ (doesn't answer question)

# Instruction finetuning (Flan-T5)

## Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:

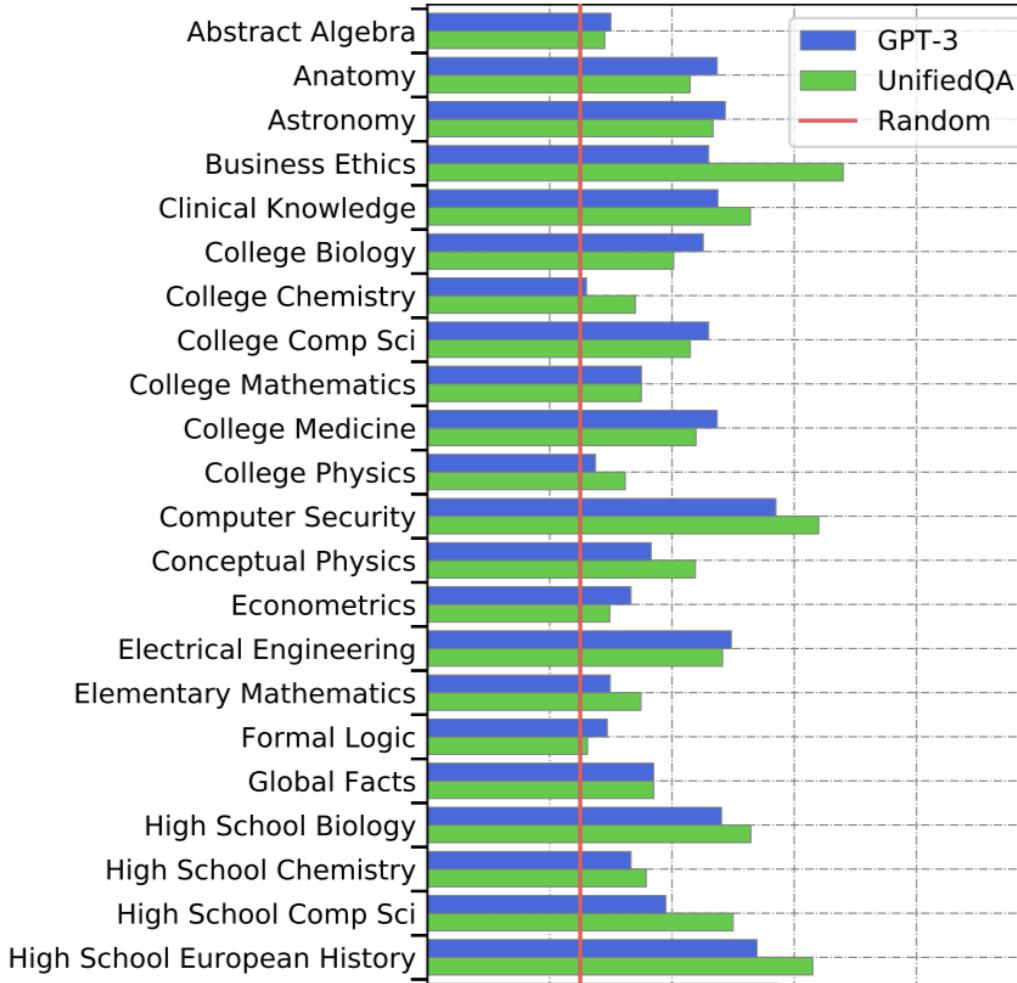
- (A) They will discuss the reporter's favorite dishes
- (B) They will discuss the chef's favorite dishes
- (C) Ambiguous

A: Let's think step by step.

## After instruction finetuning

The reporter and the chef will discuss their favorite dishes does not indicate whose favorite dishes they will discuss. So, the answer is (C). 

# MMLU: new benchmarks for multitask LMs



- **Massive Multitask Language Understanding (MMLU)**
- New benchmarks for measuring LM performance on 57 diverse knowledge intensive tasks

# MMLU: Examples

## Astronomy

What is true for a type-Ia supernova?

- A. This type occurs in binary systems.
- B. This type occurs in young galaxies.
- C. This type produces gamma-ray bursts.
- D. This type produces high amounts of X-rays.

Answer: A

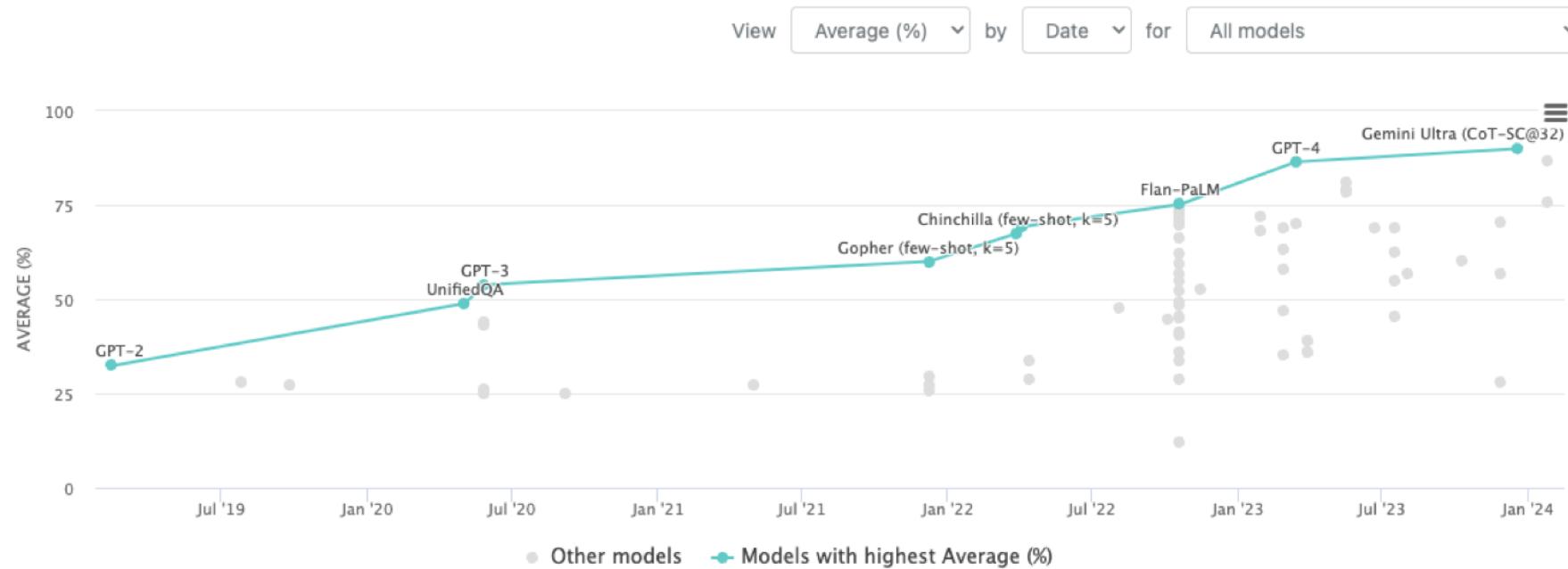
## High School Biology

In a population of giraffes, an environmental change occurs that favors individuals that are tallest. As a result, more of the taller individuals are able to obtain nutrients and survive to pass along their genetic information. This is an example of

- A. directional selection.
- B. stabilizing selection.
- C. sexual selection.
- D. disruptive selection

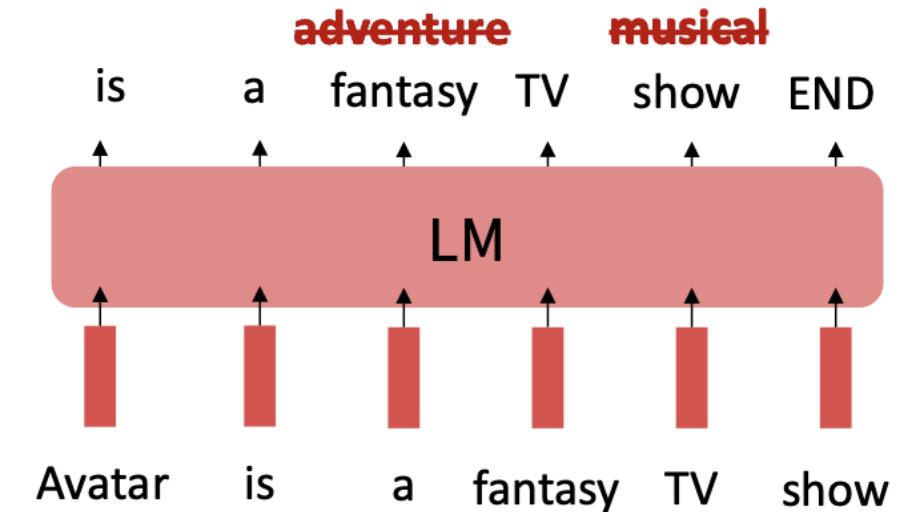
Answer: A

# MMLU: Rapid Progress



# Limitations of Instruction finetuning

- It is expensive to collect ground-truth data for tasks
- Some tasks like open-ended creative generation have no right answer
  - E.g., write a story about a lion
- Language modelling penalizes all token-level mistakes equally, but some are worse than others
- Can we **try to satisfy human preferences?**



# Optimizing for human preferences

- For each LM sample, imagine we had a way to obtain a human reward  $R(s) \in \mathbb{R}$
- Now let's maximize the expected reward of samples from LM

$$\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$$

SAN FRANCISCO,  
California (CNN) --  
A magnitude 4.2  
earthquake shook the  
San Francisco

...  
overturn unstable  
objects.

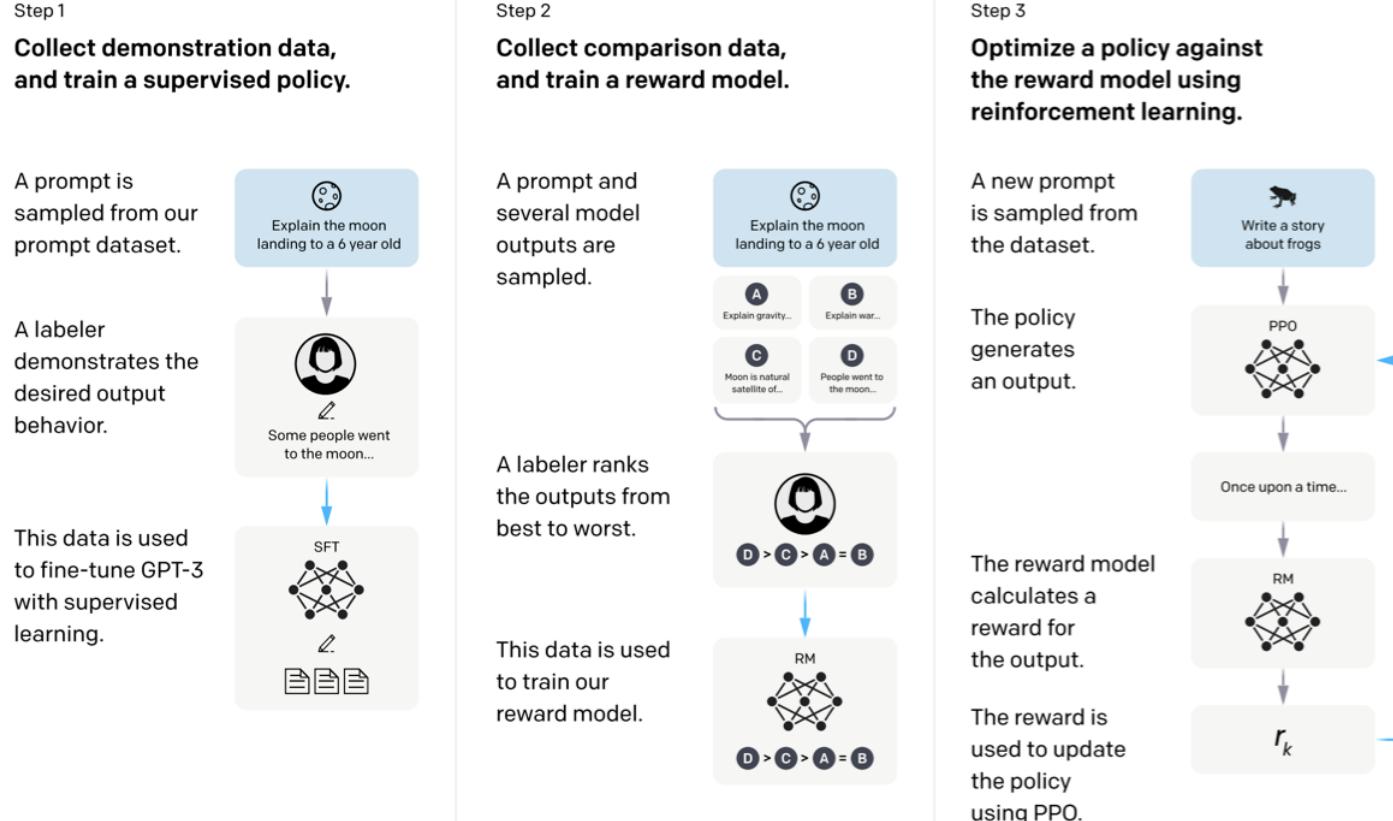
An earthquake hit  
San Francisco.  
There was minor  
property damage,  
but no injuries.

$$s_1 \\ R(s_1) = 8.0$$

The Bay Area has  
good weather but is  
prone to  
earthquakes and  
wildfires.

$$s_2 \\ R(s_2) = 1.2$$

# Reinforcement Learning from Human Feedback (RLHF)



- Instruction tuning first
- Then maximize reward

# Reinforcement Learning (RL)

- The field of reinforcement learning has studied these problems for many years
- Circa 2013: resurgence of interest in RL applied to deep learning in game playing
- New area: Applying RL to modern LMs



# Optimizing for human preferences

- How do we actually change our LM parameters to maximize this?

$$\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$$

- Policy gradient methods in RL give us tools for estimating and optimizing this objection

# How do we model human preferences?

- Human-in-the-loop is expensive!
- Instead of directly asking humans for preference, model their preferences as a separate NLP problem

An earthquake hit San Francisco. There was minor property damage, but no injuries.

$$s_1 \\ R(s_1) = 8.0 \quad \text{👤} \quad \text{💵}$$

The Bay Area has good weather but is prone to earthquakes and wildfires.

$$s_2 \\ R(s_2) = 1.2 \quad \text{👤} \quad \text{💵}$$

Train an LM  $RM_\phi(s)$  to predict human preferences from an annotated dataset, then optimize for  $RM_\phi$  instead.

# How do we model human preferences?

- Human judgments are noisy and miscalibrated!
- Instead of directly asking for ratings, ask for pairwise comparisons that are more reliable

A 4.2 magnitude  
earthquake hit  
San Francisco,  
resulting in  
massive damage.

$s_3$

$R(s_3) = \text{4.1? } 6.6? \text{ } 3.2?$

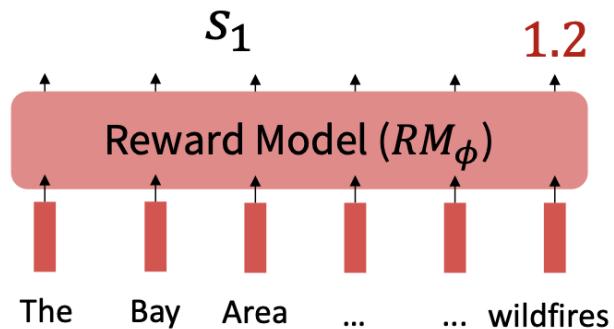
# How do we model human preferences?

- Human judgments are noisy and miscalibrated!
  - Instead of directly asking for ratings, ask for pairwise comparisons that are more reliable

An earthquake hit  
San Francisco.  
There was minor  
property damage,  
but no injuries.

> A 4.2 magnitude earthquake hit San Francisco, resulting in massive damage.

The Bay Area has good weather but is prone to earthquakes and wildfires.

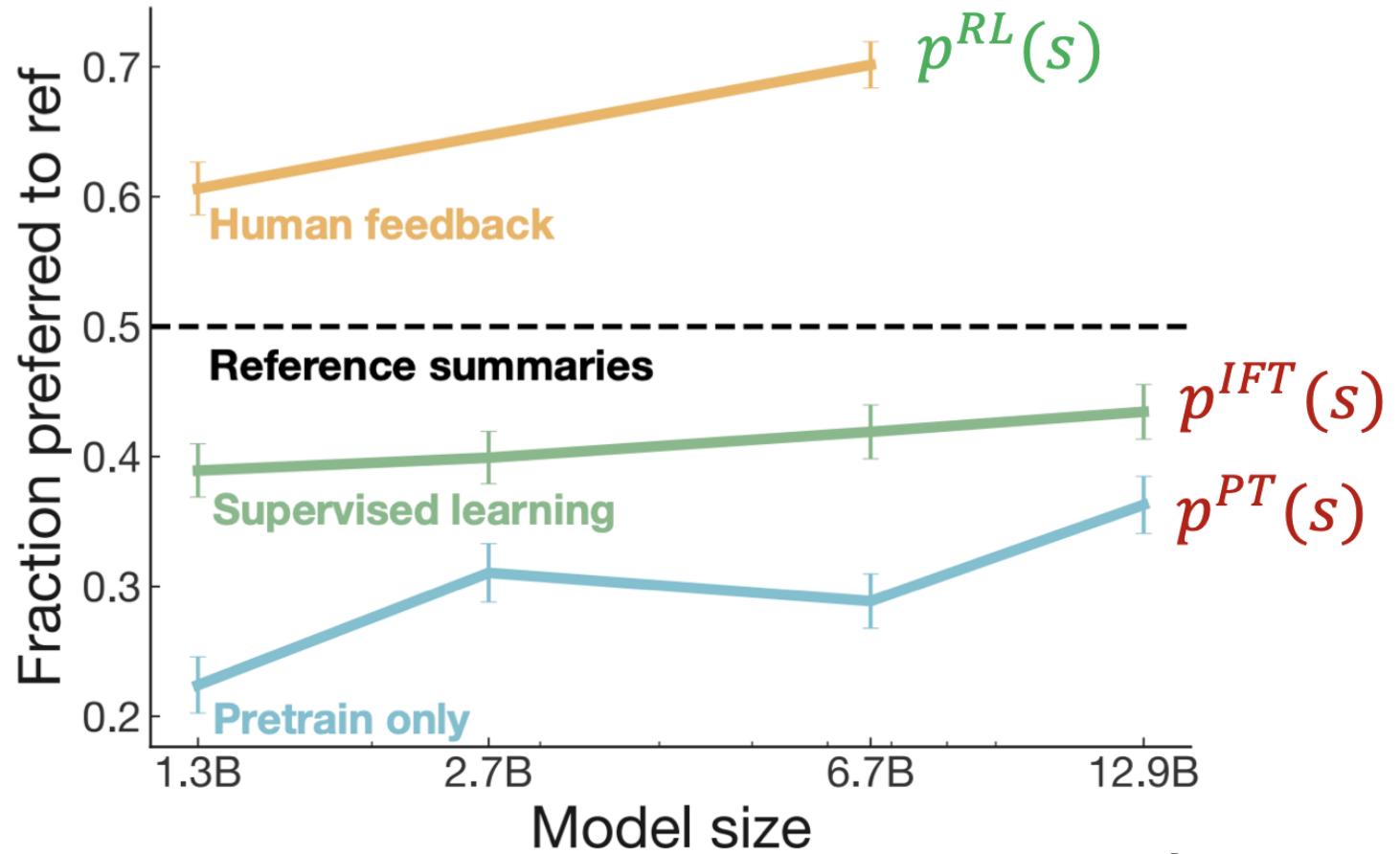


$s_3$                                      $s_2$   
Bradley-Terry [1952] paired comparison model

$$J_{RM}(\phi) = -\mathbb{E}_{(\textcolor{violet}{s}^w, \textcolor{red}{s}^l) \sim D} [\log \sigma(RM_\phi(\textcolor{violet}{s}^w) - RM_\phi(\textcolor{red}{s}^l))]$$

“winning” sample      “losing” sample       $s^w$  should score higher than  $s^l$

# RLHF provides additional gains



[Stiennon et al., 2020]

# RLHF Summary

- Have everything:
  - A pretrained (and instruction-finetuned) LM
  - A reward model
  - A method (policy gradient) for RL
- RLHF:
  - Initialized from LM, with parameter  $\theta$  to optimize for
  - Optimized the following reward with RL

$$R(s) = RM_\phi(s) - \beta \log \left( \frac{p_\theta^{RL}(s)}{p^{PT}(s)} \right)$$

Pay a price when  
 $p_\theta^{RL}(s) > p^{PT}(s)$

This is a penalty which prevents us from diverging too far from the pretrained model. In expectation, it is known as the **Kullback-Leibler (KL)** divergence between  $p_\theta^{RL}(s)$  and  $p^{PT}(s)$ .

# Instruct GPT: scaling up RLHF to many tasks

**30k  
tasks!**

Step 1

**Collect demonstration data,  
and train a supervised policy.**

A prompt is sampled from our prompt dataset.

Explain the moon landing to a 6 year old

A labeler demonstrates the desired output behavior.



Some people went to the moon...

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



Step 2

**Collect comparison data,  
and train a reward model.**

A prompt and several model outputs are sampled.

Explain the moon landing to a 6 year old

A Explain gravity...  
B Explain war...  
C Moon is natural satellite of...  
D People went to the moon...

A labeler ranks the outputs from best to worst.

D > C > A = B

This data is used to train our reward model.

RM



D > C > A = B

Step 3

**Optimize a policy against the reward model using reinforcement learning.**

A new prompt is sampled from the dataset.

Write a story about frogs

PPO

Once upon a time...

RM



$r_k$

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

[Ouyang et al., 2022]

# Instruct GPT: scaling up RLHF to many tasks

- Labeler collected tasks
  - **Plain:** We simply ask the labelers to come up with an arbitrary task, while ensuring the tasks had sufficient diversity.
  - **Few-shot:** We ask the labelers to come up with an instruction, and multiple query/response pairs for that instruction.
  - **User-based:** We had a number of use-cases stated in waitlist applications to the OpenAI API. We asked labelers to come up with prompts corresponding to these use cases.

Use-case	Prompt
Brainstorming	List five ideas for how to regain enthusiasm for my career
Generation	Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.

# Instruct GPT: scaling up RLHF to many tasks

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION GPT-3

Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.

InstructGPT

People went to the moon, and they took pictures of what they saw, and sent them back to the earth so we could all see them.

# ChatGPT: Instruction tuning + RLHF for dialogue

## ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge...

## Methods

We trained this model using Reinforcement Learning from Human Feedback (RLHF), using the same methods as InstructGPT, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. We mixed this new dialogue dataset with the InstructGPT dataset, which we transformed into a dialogue format.

**(Instruction finetuning!)**

# ChatGPT: Instruction tuning + RLHF for dialogue

## ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge...

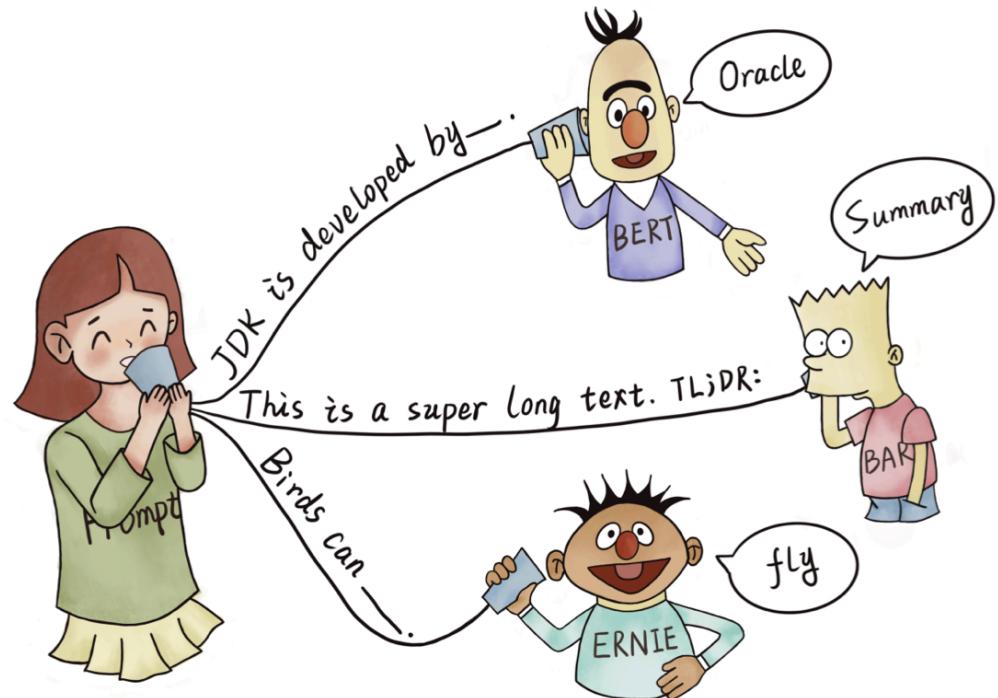
## Methods

To create a reward model for reinforcement learning, we needed to collect comparison data, which consisted of two or more model responses ranked by quality. To collect this data, we took conversations that AI trainers had with the chatbot. We randomly selected a model-written message, sampled several alternative completions, and had AI trainers rank them. Using these reward models, we can fine-tune the model using Proximal Policy Optimization. We performed several iterations of this process.

(RLHF!)

# What is Prompting?

- Definition: Encouraging a pre-trained model to make predictions by textual prompt to specify the task to be done



# Basic Prompting

- Append a textual string to the beginning of the sequence and complete

x = **When a dog sees a squirrel, it will usually**

(GPT-2 Small) be afraid of anything unusual. As an exception, that's when a squirrel is usually afraid to bite.

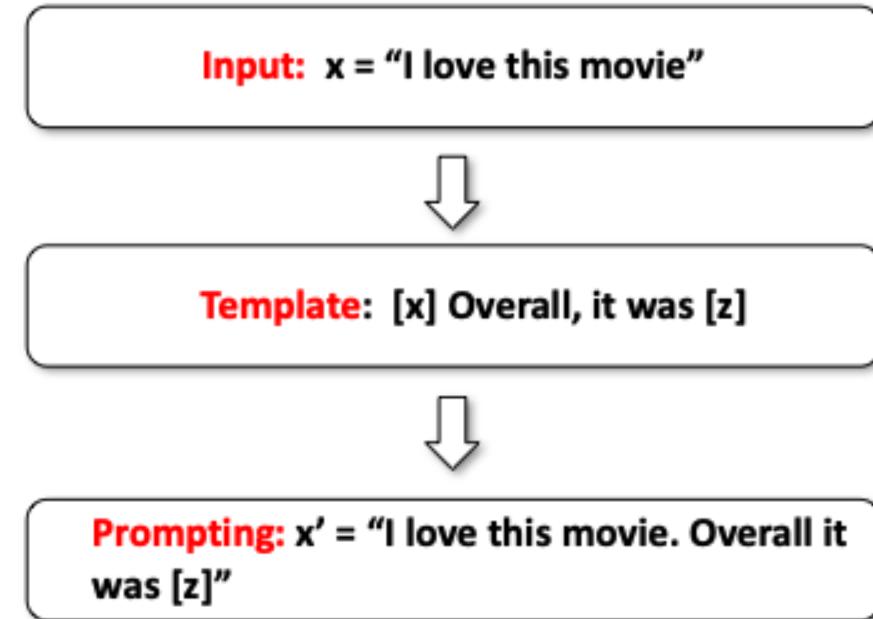
(GPT-2 XL) lick the squirrel. It will also touch its nose to the squirrel on the tail and nose if it can.

# Standard prompting workflow

- Fill a prompt template
- Predict the answer
- Post-process the answer

# Prompt Templates

- A template where you fill in with an actual input



# Answer Prediction

- Given a prompt, predict the answer

**Prompting:**  $x' = \text{"I love this movie. Overall it was [z]"}$



**Predicting:**  $x' = \text{"I love this movie. Overall it was fantastic"}$

# Post-processing

- Select the actual output based on the answer
- E.g., formatting the output for easy visualization

## Markdown Rendering



You

Please write a table with the last 5 presidents and their birthdates.



ChatGPT

Here is a table with the last 5 U.S. Presidents and their birthdates:

President	Birthdate
Joe Biden	November 20, 1942
Donald Trump	June 14, 1946
Barack Obama	August 4, 1961
George W. Bush	July 6, 1946
Bill Clinton	August 19, 1946

## Code



You

Output that in JSON format.



ChatGPT

Here is the information in JSON format:

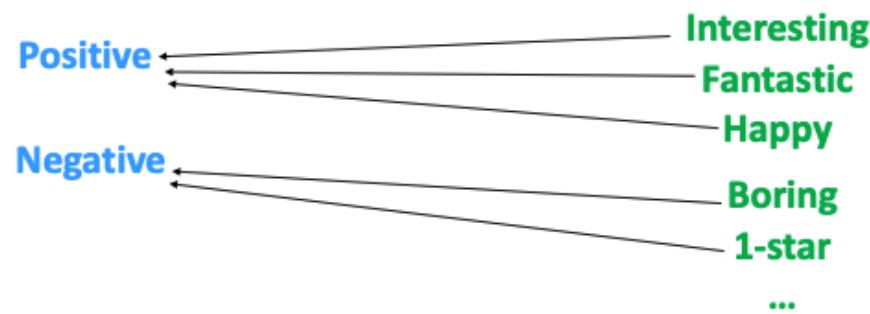
```
json
Copy code
[{"President": "Joe Biden", "Birthdate": "November 20, 1942"}, {"President": "Donald Trump", "Birthdate": "June 14, 1946"}, {"President": "Barack Obama", "Birthdate": "August 4, 1961"}, {"President": "George W. Bush", "Birthdate": "July 6, 1946"}, {"President": "Bill Clinton", "Birthdate": "August 19, 1946"}]
```

# Post-processing

- Given an answer, map it into a class label or continuous value



- Often map many extracted words onto a single class



# Few-shot Prompting

- Provide a few examples of the task together with the instruction

**Instruction** | Please classify movie reviews as 'positive' or 'negative'.

Input: I really don't like this movie.

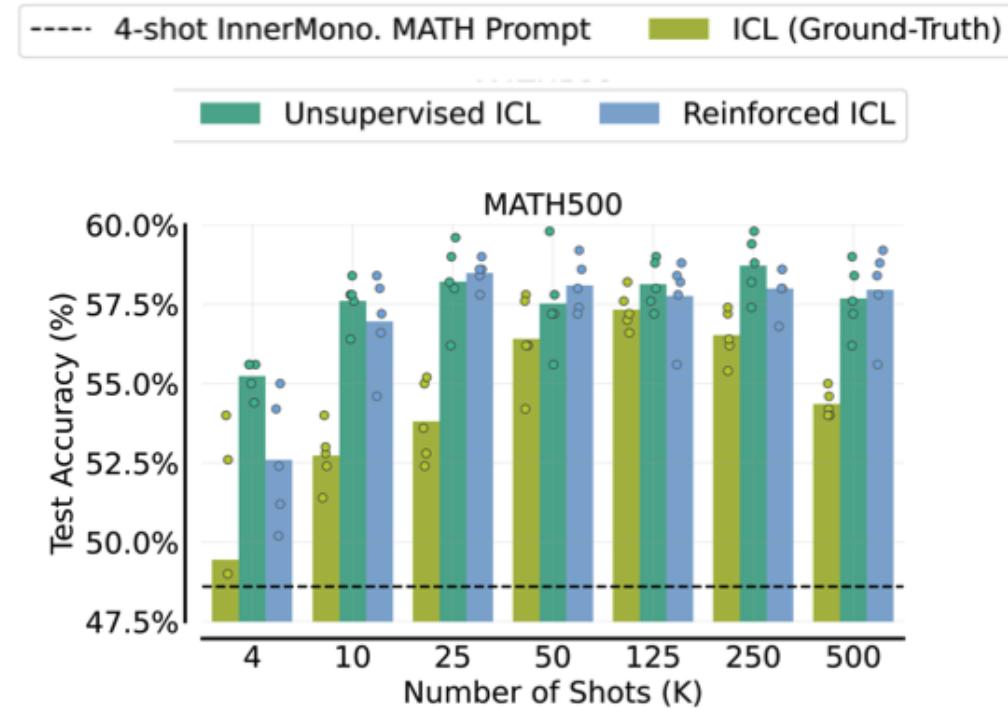
Output: negative

**Examples**

Input: This movie is great!

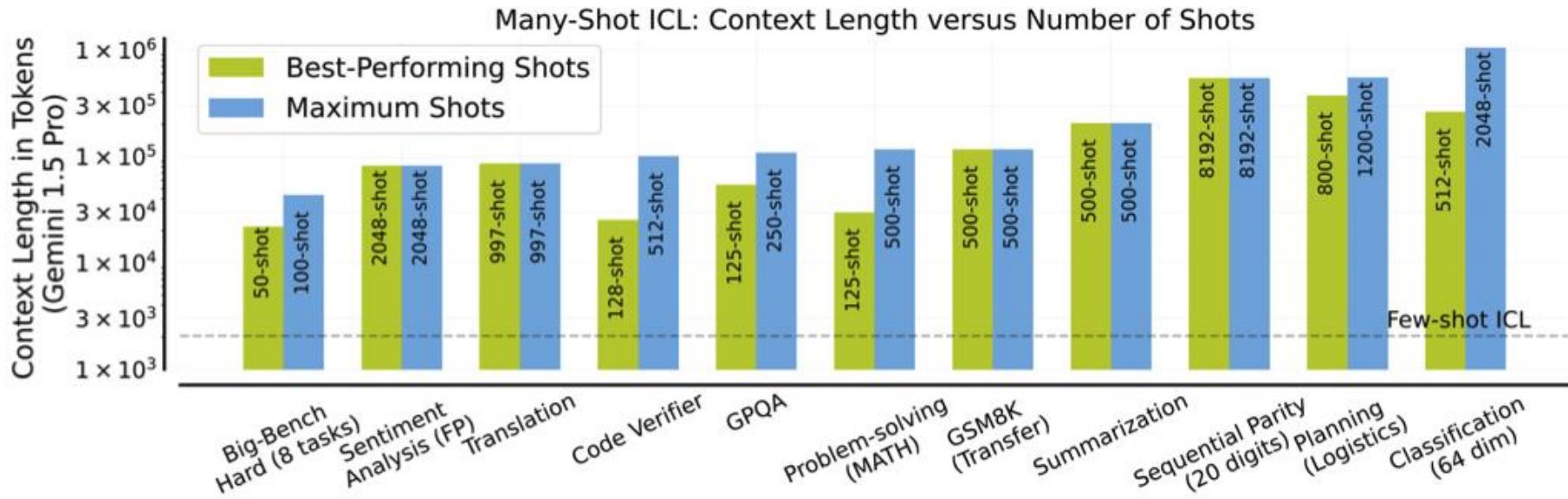
Output: positive

# Empirical results on In-context Learning



- Sometimes only giving the inputs works better

# Empirical results on In-context Learning



- Sometimes performance can decrease with too many examples

# LMs are sensitive to Small changes

- Example ordering (Lu et al. 2021)

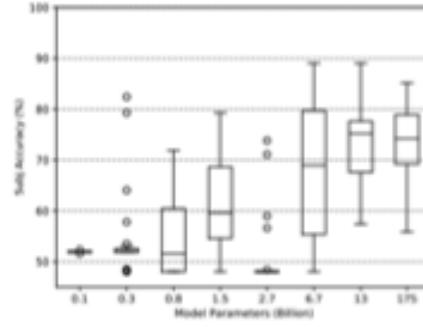
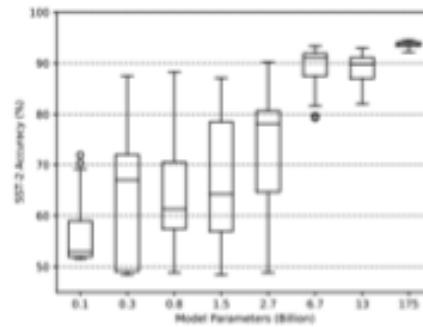
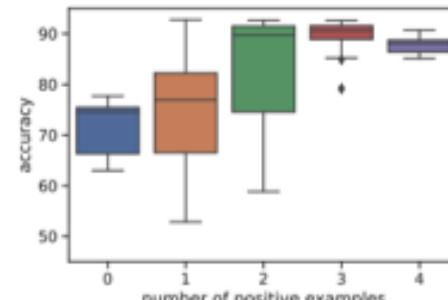
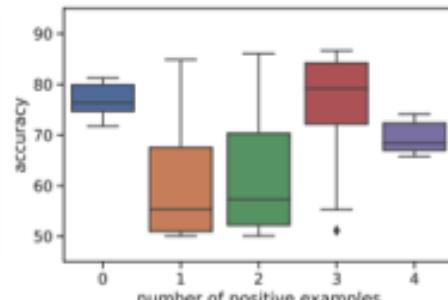


Figure 1: Four-shot performance for 24 different sample orders across different sizes of GPT-family models (GPT-2 and GPT-3) for the SST-2 and Subj datasets.

- Label balance (Zhang et al. 2022)

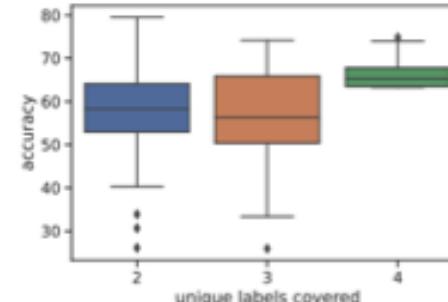


(a) Amazon

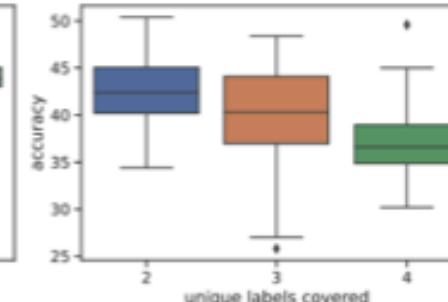


(b) SST-2

- Label coverage (Zhang et al. 2022)



(a) AGNews



(b) TREC

# Prompt Engineering: Design of Prompts

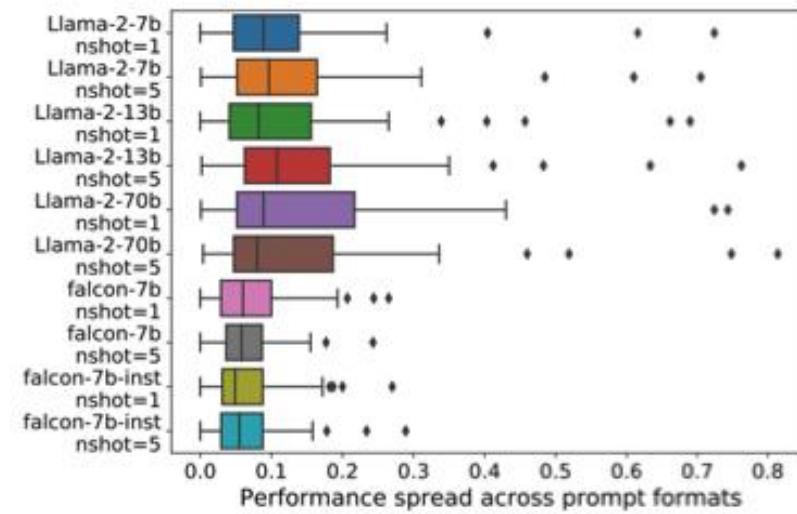
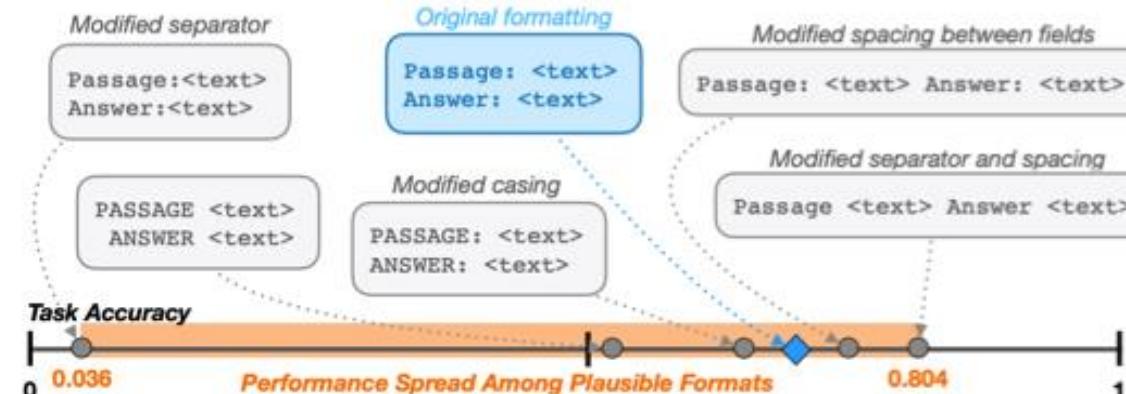
- Manual
  - Configure a manual template based on the characteristics of the task
  - Configure prompts based on intuition about a task
- Automated search: Find the (hopefully) optimal prompts

# Prompt Engineering: Design of Prompts

- Manual
  - Configure a manual template based on the characteristics of the task
  - Configure prompts based on intuition about a task
- Automated search: Find the (hopefully) optimal prompts

# Prompt Engineering: Format

- Make sure that the format matches that of a trained model
- Could have large effect on models!



# Prompt Engineering: Instruction

- Instructions should be clear, concise and easy to understand
- See <https://www.promptingguide.ai/introduction/tips>

## Less Precise:

Explain the concept prompt engineering. Keep the explanation short, only a few sentences, and don't be too descriptive.

## More Precise:

Use 2-3 sentences to explain the concept of prompt engineering to a high school student.

# Chain-of-thought Prompting

- Get the model to explain its reasoning before making an answer

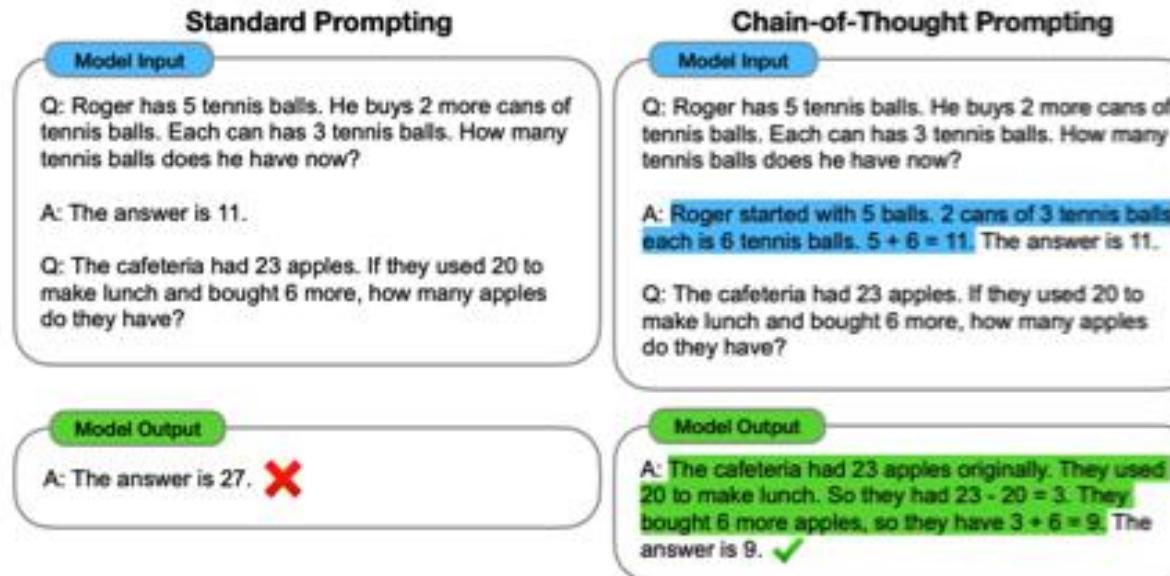


Figure 1: Chain-of-thought prompting enables large language models to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. Chain-of-thought reasoning processes are highlighted.

# Outline

---

- Part 1: NLP Background
- Part 2: Background and Transformer
- Part 3: Modern LLMs and ChatGPT
- Part 4: Large Reasoning Models and Deepseek R1

# GPT-o1: Scale up Reasoning



Our large-scale reinforcement algorithm teaches the model how to think productively using its chain of thought in a highly data-efficient training process.

# Hard Language Tasks: Reasoning

# Definition of Reasoning

Think, understand, and form judgments by a process of logic

- Oxford Languages

# Reasoning Problems

Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?

A: The answer is **5**

Q: Take the last letters of the words in "Elon Musk" and concatenate them

A: The answer is **nk**.

Q: What home entertainment equipment requires cable?  
Answer Choices: (a) radio shack  
(b) substation (c) television (d) cabinet

A: The answer is **(c)**.

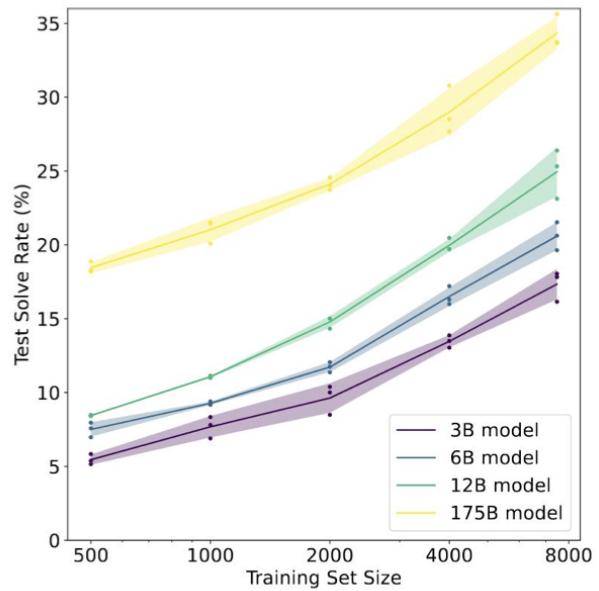
Arithmetic Reasoning (AR)  
 $(+ - \times \div \dots)$

Symbolic Reasoning (SR)

Commonsense Reasoning (CR)

# Reasoning Problems

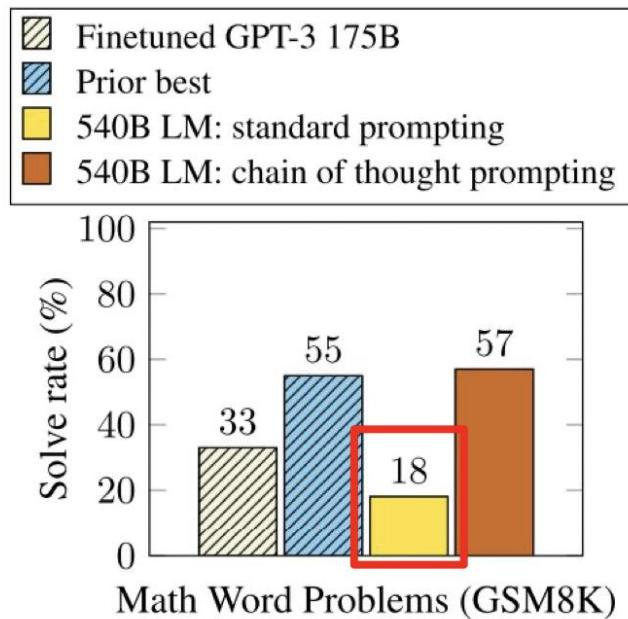
Fine-tune GPT-3 on GSM8K (arithmetic): (Cobbe et al. 2021)



Conjecture: to achieve > 80%, needs 100 times more fine-tuning data for 175B model

# Reasoning Problems

GSM8K (arithmetic):



**Few-shot standard prompting** with even larger model (PaLM 540B) also does not work well.

# Chain-of-thought Prompting

---

## Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

---

**Jason Wei**

**Xuezhi Wang**

**Dale Schuurmans**

**Maarten Bosma**

**Brian Ichter**

**Fei Xia**

**Ed H. Chi**

**Quoc V. Le**

**Denny Zhou**

Google Research, Brain Team  
[{jasonwei,dennyzhou}@google.com](mailto:{jasonwei,dennyzhou}@google.com)

# Chain-of-thought Prompting

- **Definition:** A chain of thought is a series of intermediate natural language reasoning steps that lead to the final output.
- Benefits:
  - Decompose into simple questions
  - Interpretable
  - Leverage prompting of LLM

# Chain-of-thought Prompting

## Examples

### (a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

*(Output) The answer is 8. X*

## CoT Examples

### (b) Few-shot-CoT (Wei et al., 2022)

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.

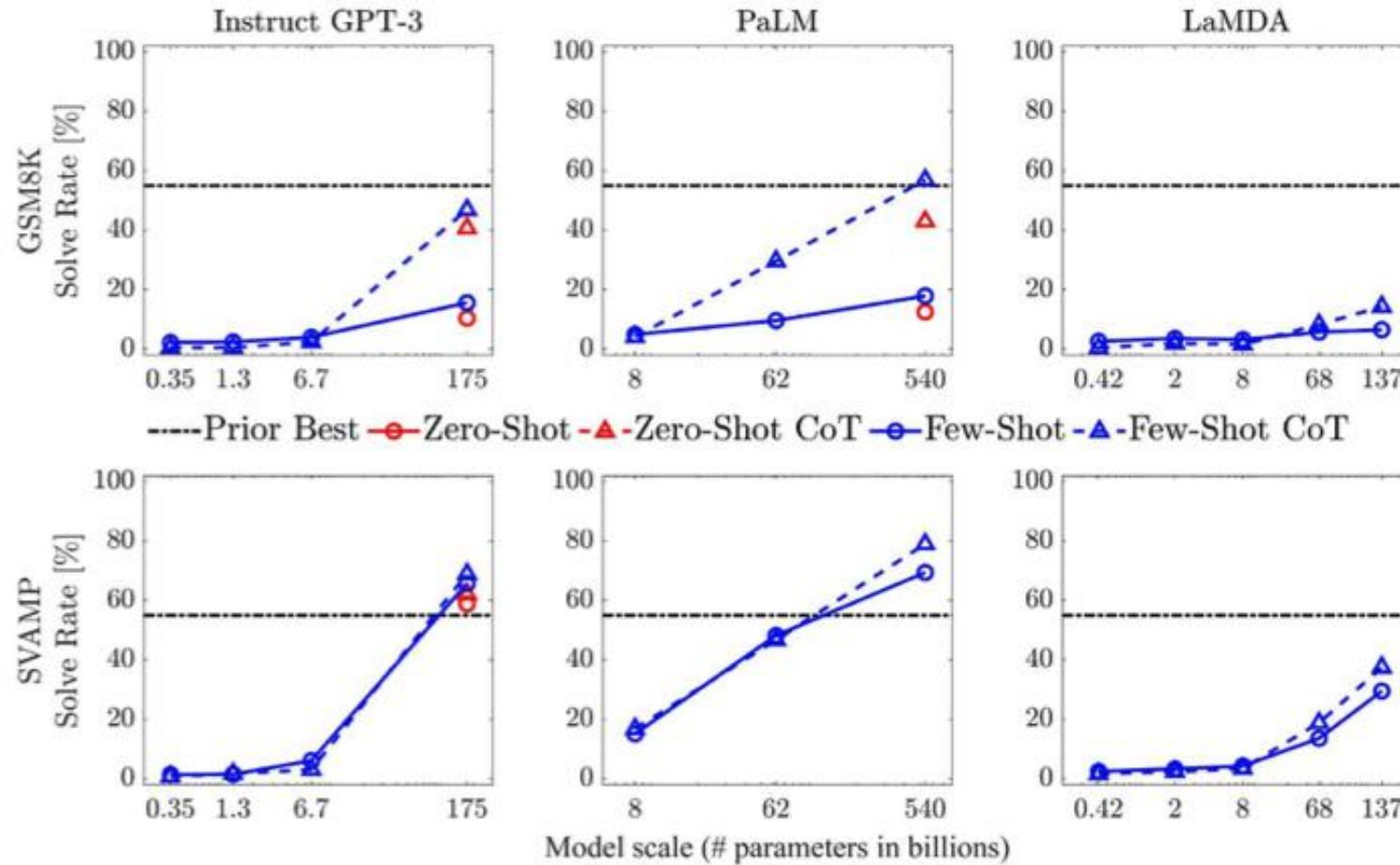
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

*(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are  $16 / 2 = 8$  golf balls. Half of the golf balls are blue. So there are  $8 / 2 = 4$  blue golf balls. The answer is 4. ✓*

## Step-by-step Answer

# Experiments: Arithmetic Reasoning



## GSM8K

Josh decides to try flipping a house. He buys a house for \$80,000 and then puts in \$50,000 in repairs. This increased the value of the house by 150%. How much profit did he make?

## SVAMP

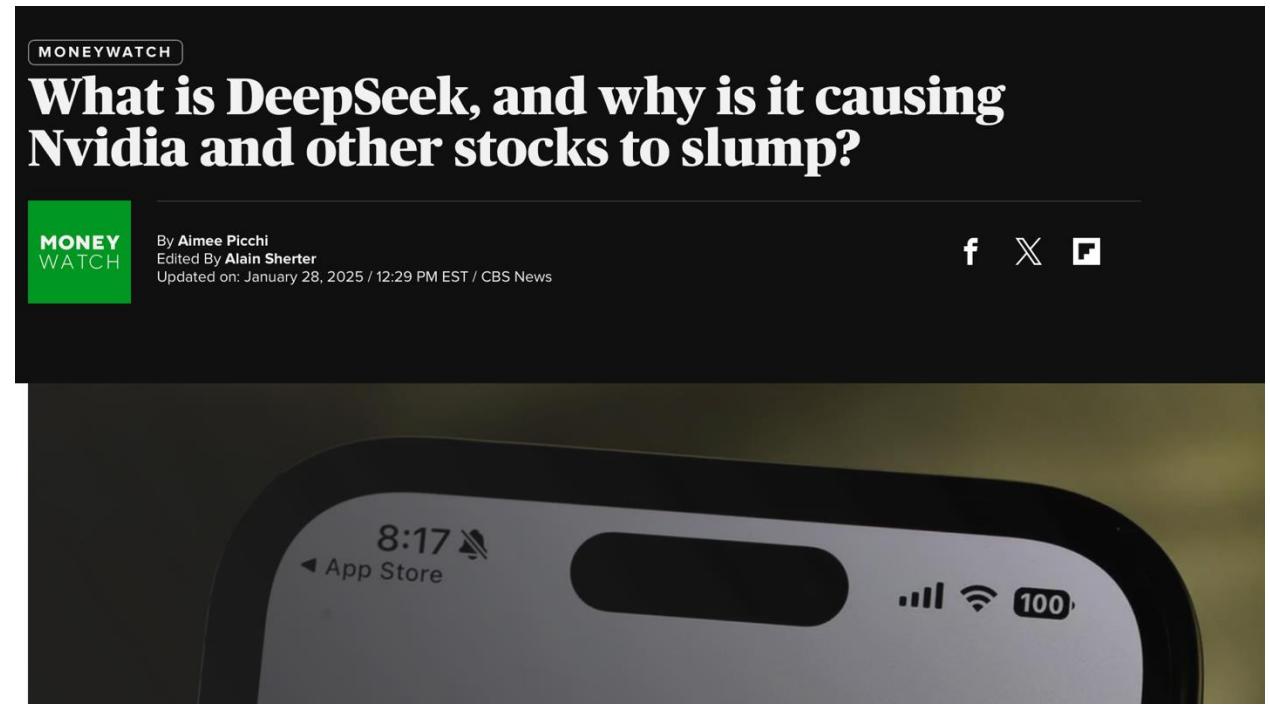
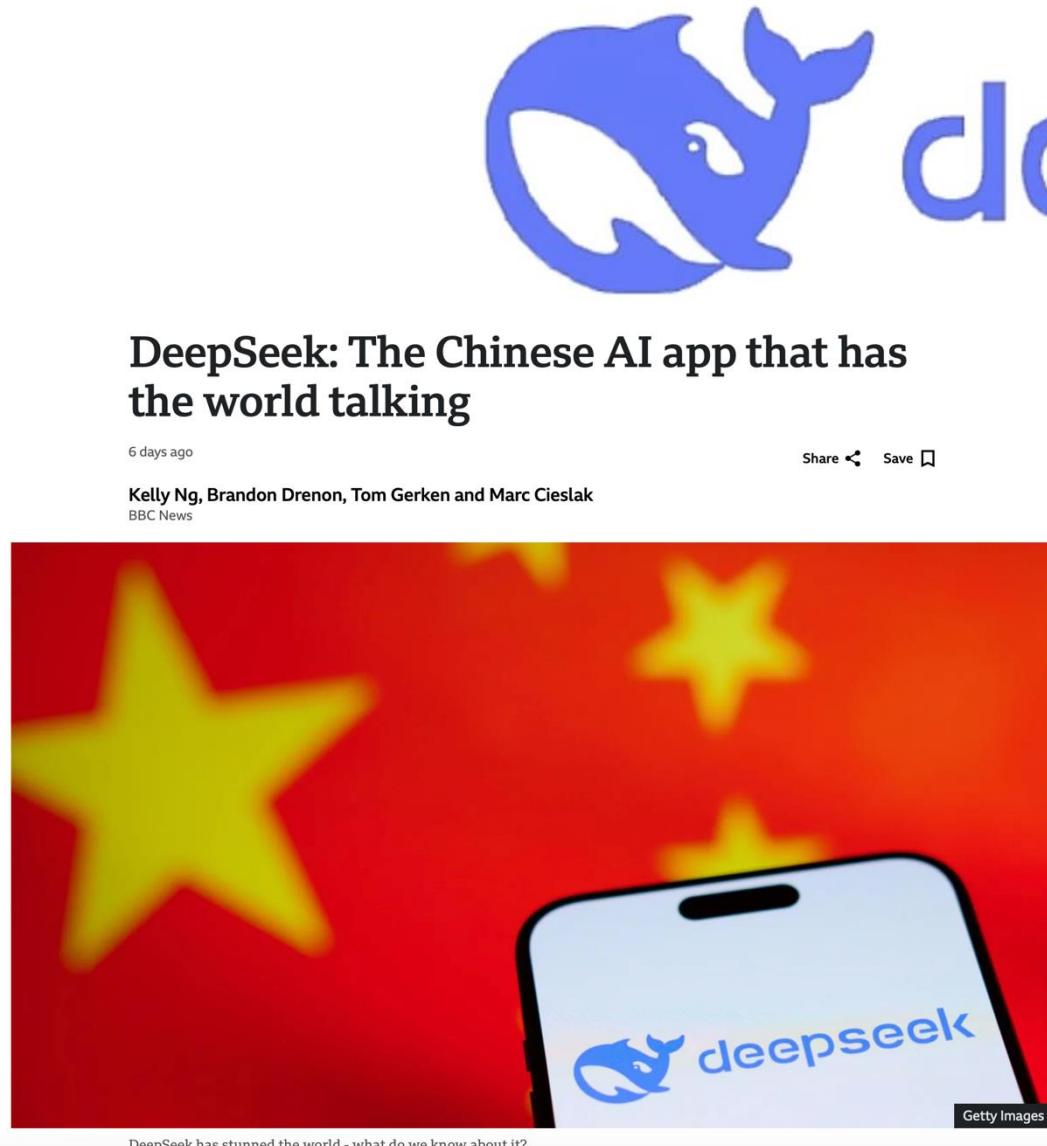
Each pack of dvds costs 76 dollars. If there is a discount of 25 dollars on each pack. How much do you have to pay to buy each pack?

# DeepSeek V3 / R1



- < 200 employees
- Spin off of hedge fund
- Consistent open-weights model releases

# DeepSeek V3 / R1



# DeepSeek V3 / R1



## DeepSeek-V3 Technical Report

DeepSeek-AI

[research@deepseek.com](mailto:research@deepseek.com)

# DeepSeek V3

Training Costs	Pre-Training	Context Extension	Post-Training	Total
in H800 GPU Hours	2664K	119K	5K	2788K
in USD	\$5.328M	\$0.238M	\$0.01M	\$5.576M

- Mix-of-Expert architecture
- Performance close to GPT 4o
- Much cheaper training cost

# DeepSeek R1



---

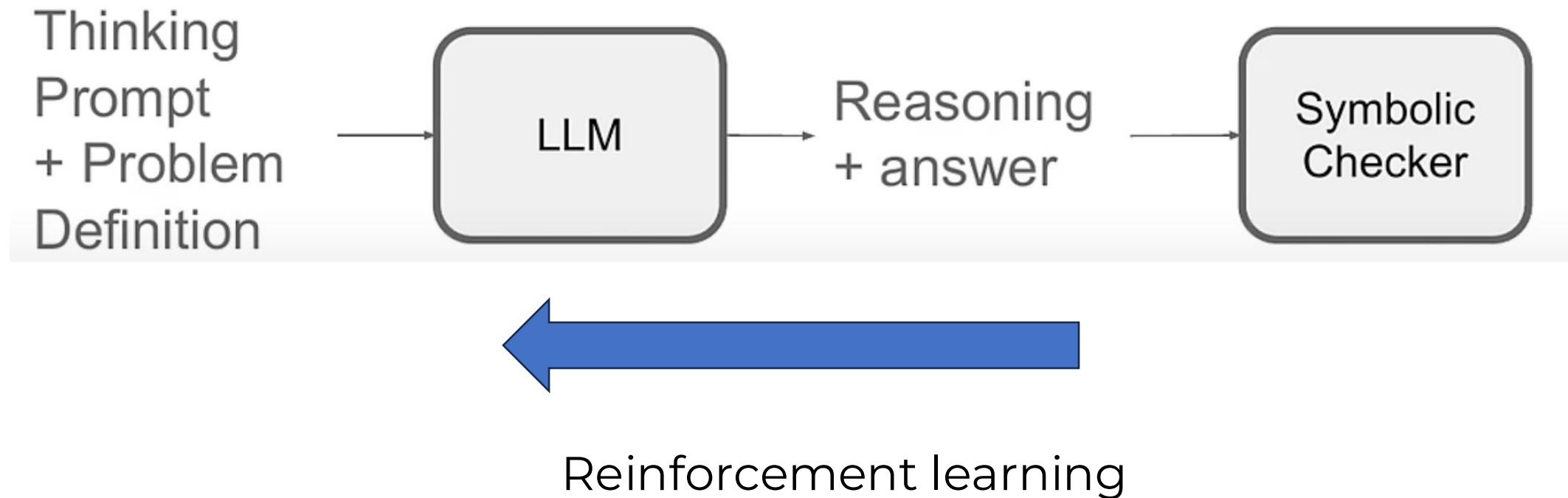
## DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning

DeepSeek-AI

[research@deepseek.com](mailto:research@deepseek.com)

- Primarily a post training innovation
- Think GPT o1

# DeepSeek R1-Zero: RL from scratch



# DeepSeek R1-Zero: Reward

---

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>. User: **prompt**. Assistant:

---

Table 1 | Template for DeepSeek-R1-Zero. **prompt** will be replaced with the specific reasoning question during training.

- Accuracy rewards: whether the response is correct
- Format rewards: whether it follows format

# DeepSeek R1-Zero is already good

Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
<b>OpenAI-o1-mini</b>	63.6	80.0	90.0	60.0	53.8	1820
<b>OpenAI-o1-0912</b>	74.4	83.3	94.8	77.3	63.4	1843
<b>DeepSeek-R1-Zero</b>	71.0	86.7	95.9	73.3	50.0	1444

Table 2 | Comparison of DeepSeek-R1-Zero and OpenAI o1 models on reasoning-related benchmarks.

# DeepSeek R1-Zero: Aha Moment

---

Question: If  $a > 1$ , then the sum of the real solutions of  $\sqrt{a - \sqrt{a+x}} = x$  is equal to

---

Response: <think>

To solve the equation  $\sqrt{a - \sqrt{a+x}} = x$ , let's start by squaring both ...

$$(\sqrt{a - \sqrt{a+x}})^2 = x^2 \implies a - \sqrt{a+x} = x^2.$$

Rearrange to isolate the inner square root term:

$$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$$

...

**Wait, wait. Wait. That's an aha moment I can flag here.**

Let's reevaluate this step-by-step to identify if the correct sum can be ...

We started with the equation:

$$\sqrt{a - \sqrt{a+x}} = x$$

First, let's square both sides:

$$a - \sqrt{a+x} = x^2 \implies \sqrt{a+x} = a - x^2$$

Next, I could square both sides again, treating the equation: ...

...

---

- Occurs in an intermediate version of model
- RL can generalize!

# DeepSeek R1: RL with Cold Start

- Can reasoning further improved with a small amount of long CoT data?
- Can we train a user-friendly model?

# DeepSeek R1: RL with Code Start

- R1: Cold-started from human-written data
- Reasoning + Non-reasoning data: use LLM to provide CoT with basic checks
- Only 800K samples

# DeepSeek R1: RL with Code Start

Benchmark (Metric)		Claude-3.5-Sonnet-1022	GPT-4o 0513	DeepSeek V3	OpenAI o1-mini	OpenAI o1-1217	DeepSeek R1
English	Architecture	-	-	MoE	-	-	MoE
	# Activated Params	-	-	37B	-	-	37B
	# Total Params	-	-	671B	-	-	671B
	MMLU (Pass@1)	88.3	87.2	88.5	85.2	<b>91.8</b>	90.8
	MMLU-Redux (EM)	88.9	88.0	89.1	86.7	-	<b>92.9</b>
	MMLU-Pro (EM)	78.0	72.6	75.9	80.3	-	<b>84.0</b>
	DROP (3-shot F1)	88.3	83.7	91.6	83.9	90.2	<b>92.2</b>
	IF-Eval (Prompt Strict)	<b>86.5</b>	84.3	86.1	84.8	-	83.3
	GPQA Diamond (Pass@1)	65.0	49.9	59.1	60.0	<b>75.7</b>	71.5
	SimpleQA (Correct)	28.4	38.2	24.9	7.0	<b>47.0</b>	30.1
Code	FRAMES (Acc.)	72.5	80.5	73.3	76.9	-	<b>82.5</b>
	AlpacaEval2.0 (LC-winrate)	52.0	51.1	70.0	57.8	-	<b>87.6</b>
	ArenaHard (GPT-4-1106)	85.2	80.4	85.5	92.0	-	<b>92.3</b>
	LiveCodeBench (Pass@1-COT)	38.9	32.9	36.2	53.8	63.4	<b>65.9</b>
	Codeforces (Percentile)	20.3	23.6	58.7	93.4	<b>96.6</b>	96.3
Math	Codeforces (Rating)	717	759	1134	1820	<b>2061</b>	2029
	SWE Verified (Resolved)	<b>50.8</b>	38.8	42.0	41.6	48.9	49.2
	Aider-Polyglot (Acc.)	45.3	16.0	49.6	32.9	<b>61.7</b>	53.3
	AIME 2024 (Pass@1)	16.0	9.3	39.2	63.6	79.2	<b>79.8</b>
Chinese	MATH-500 (Pass@1)	78.3	74.6	90.2	90.0	96.4	<b>97.3</b>
	CNMO 2024 (Pass@1)	13.1	10.8	43.2	67.6	-	<b>78.8</b>
	CLUEWSC (EM)	85.4	87.9	90.9	89.9	-	<b>92.8</b>
Chinese	C-Eval (EM)	76.7	76.0	86.5	68.9	-	<b>91.8</b>
	C-SimpleQA (Correct)	55.4	58.7	<b>68.0</b>	40.3	-	63.7