

Thoughts on NLP Research in the (Post-)LLM* Era

Yijia Shao

Yuanpei College

2023/04/28

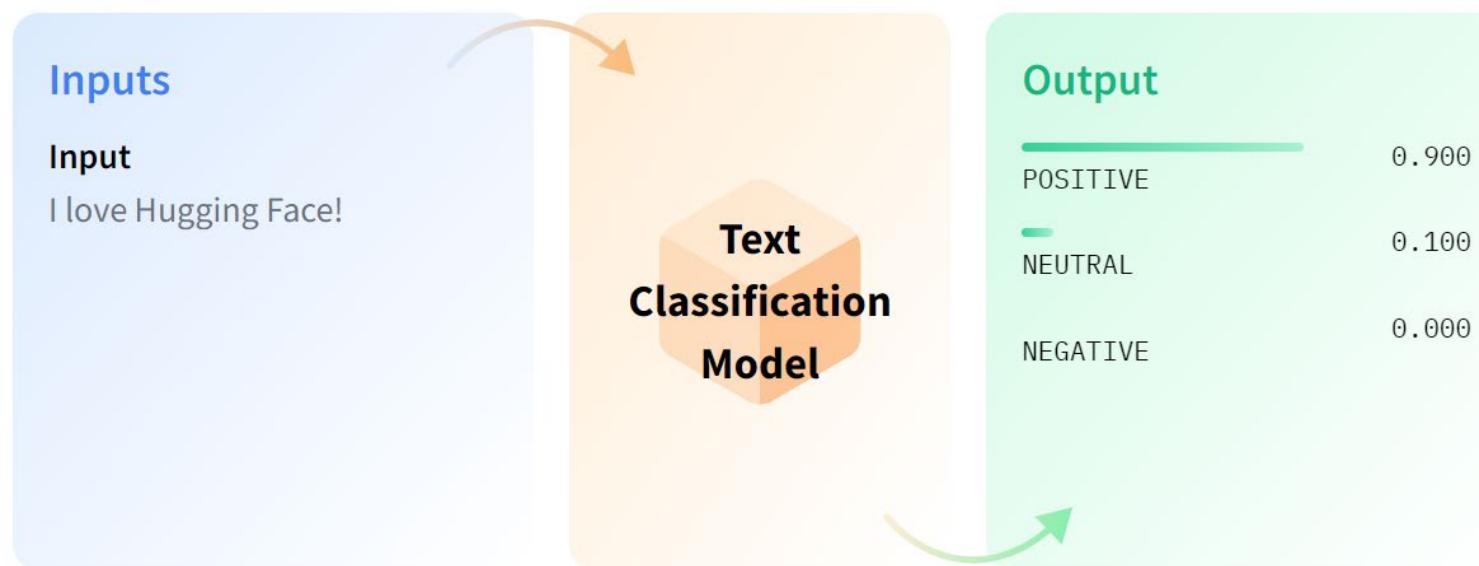
*LLM: Large Language Model

Overview

- NLP tasks in the pre-LLM era
- Introduction to LLMs
- View LLMs from a system perspective
- Open questions

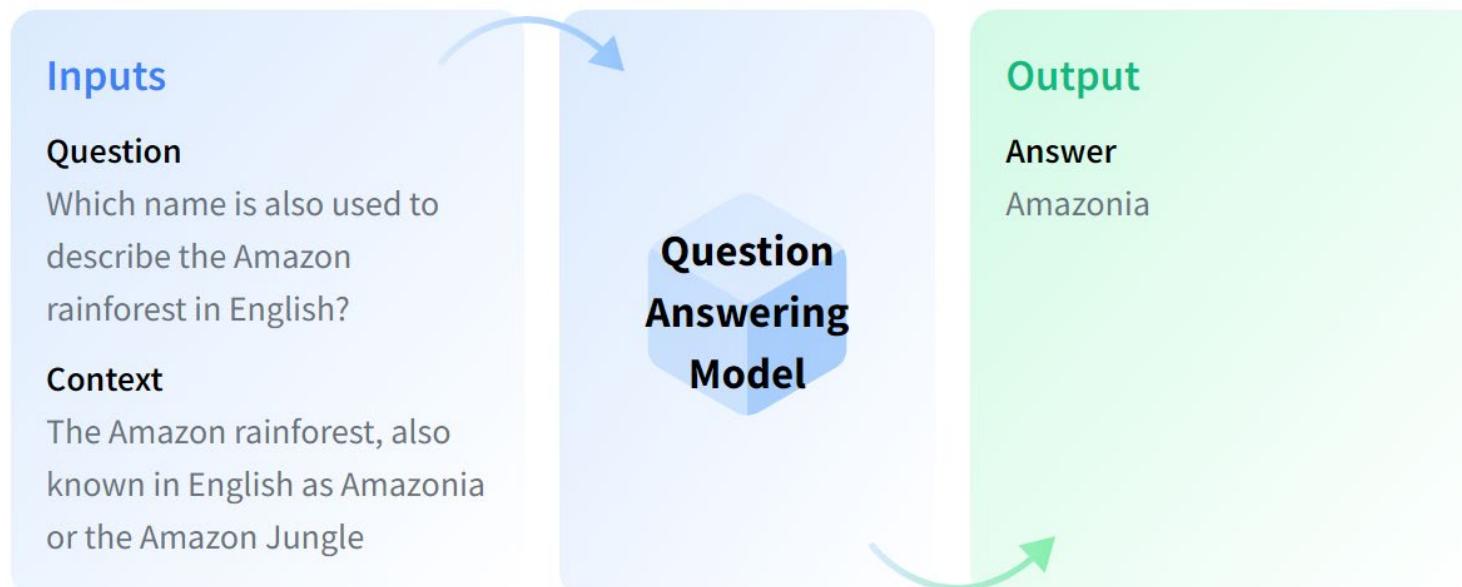
Typical NLP Tasks in the Pre-LLM Era

- **Text classification:** Assigning a label or class to a given text.
- E.g. Sentiment Analysis: class labels are sentiment polarities



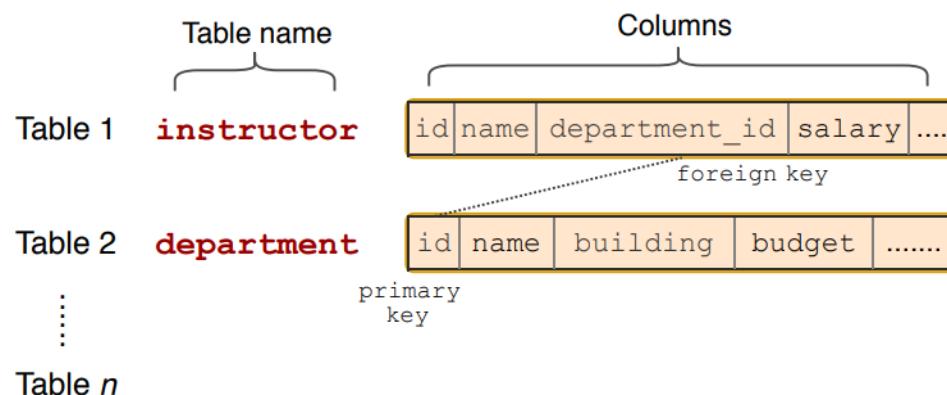
Typical NLP Tasks in the Pre-LLM Era

- **Question answering:** Returning an answer in text form to a given question also in text form.
- E.g. Extractive QA: the answer is extracted from a given context



Typical NLP Tasks in the Pre-LLM Era

- **Semantic Parsing:** Converting a natural language utterance to a logical form.
- E.g. Text-to-SQL: convert a natural language question to a SQL query



What are the name and budget of the departments with average instructor salary greater than the overall average?

```
SELECT T2.name, T2.budget  
FROM instructor as T1 JOIN department as  
T2 ON T1.department_id = T2.id  
GROUP BY T1.department_id  
HAVING avg(T1.salary) >  
    (SELECT avg(salary) FROM instructor)
```



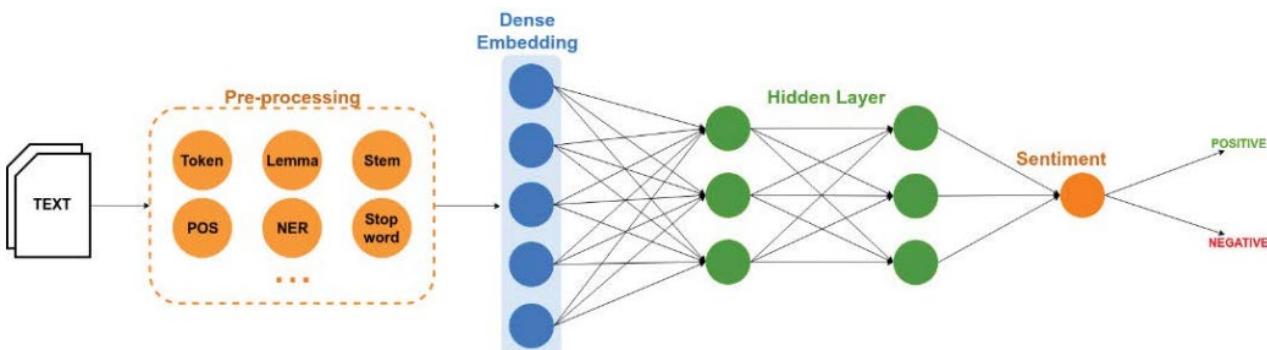
Typical NLP Tasks in the Pre-LLM Era

- And **SO MANY...**

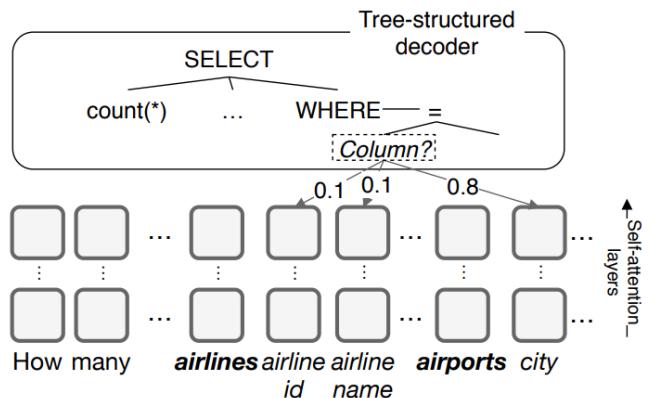


Traditional ML Paradigm

- Supervised data + algorithm -> model
 - Design **specific algorithms** for each task and train **separate models**.



Dang et al. "Sentiment Analysis Based on Deep Learning: A Comparative Study" *Electronics* 2020



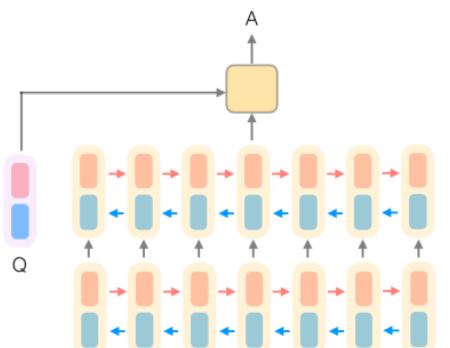
Wang et al. "RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers" *ACL* 2020

Q: How many of Warsaw's inhabitants spoke Polish in 1933?



Document Reader

→ 833,500



Chen et al. "Reading Wikipedia to Answer Open-Domain Questions" *ACL* 2017

Why do some researchers feel panic when ChatGPT/LLMs came out?

One LLM for All

- Studying a specific task becomes less meaningful.

Sentiment analysis classifier

Decide whether a Tweet's sentiment is positive, neutral, or negative.

Tweet: "I loved the new Batman movie!"
Sentiment: Positive

Q&A

Targeted world knowledge

Chatbot: I am a ML/AI language model tutor
You: What is a language model?
Chatbot: A language model is a statistical model that describes the probability of a word given the previous words.

Translation / NL2code

Create a SQL request to find all users who live in California and have over 1000 credits:
`SELECT * FROM users WHERE state='CA' AND credits > 1000;`

Semantic of query

Syntax of code

Summarization

A neutron star is the collapsed core of a massive supergiant star, which had a total mass of between 10 and 25 solar masses, possibly more if the star was especially metal-rich.^[1] Neutron stars are the smallest and densest stellar objects, excluding black holes and hypothetical white holes, quark stars, and strange stars.^[2] Neutron stars have a radius on the order of 10 kilometres (6.2 mi) and a mass of about 1.4 solar masses.^[3] They result from the supernova explosion of a massive star, combined with gravitational collapse, that compresses the core past white dwarf star density to that of atomic nuclei.

TL;DR: A neutron star is the collapsed core of a massive supergiant star. These ultra-dense objects are incredibly fascinating due to their strange properties and their potential for phenomena such as extreme gravitational forces and a strong magnetic field.

The semantic of TL;DR

“Eureka” Moment and Paradigm Shift

- From expert-defined tasks to **user-defined tasks**.

ChatGPT PLUS



Examples

"Explain quantum computing in simple terms" →



Capabilities

"Got any creative ideas for a 10 year old's birthday?" →

Allows user to provide follow-up corrections

"How do I make an HTTP request in Javascript?" →

Trained to decline inappropriate requests



Limitations

May occasionally generate incorrect information

May occasionally produce harmful instructions or biased content

Limited knowledge of world and events after 2021

Send a message.



Emergent Reasoning Ability

- LLMs show “unexpected” reasoning ability and exceed average human performance on many standard exams.

Exam	GPT-4	GPT-4 (no vision)	GPT-3.5	Exam	GPT-4	GPT-4 (no vision)	GPT-3.5
Uniform Bar Exam (MBE+MEE+MPT)	298 / 400 (~90th)	298 / 400 (~90th)	213 / 400 (~10th)	AP Environmental Science	5 (91st - 100th)	5 (91st - 100th)	5 (91st - 100th)
LSAT	163 (~88th)	161 (~83rd)	149 (~40th)	AP Macroeconomics	5 (84th - 100th)	5 (84th - 100th)	2 (33rd - 48th)
SAT Evidence-Based Reading & Writing	710 / 800 (~93rd)	710 / 800 (~93rd)	670 / 800 (~87th)	AP Microeconomics	5 (82nd - 100th)	4 (60th - 82nd)	4 (60th - 82nd)
SAT Math	700 / 800 (~89th)	690 / 800 (~89th)	590 / 800 (~70th)	AP Physics 2	4 (66th - 84th)	4 (66th - 84th)	3 (30th - 66th)
Graduate Record Examination (GRE) Quantitative	163 / 170 (~80th)	157 / 170 (~62nd)	147 / 170 (~25th)	AP Psychology	5 (83rd - 100th)	5 (83rd - 100th)	5 (83rd - 100th)
Graduate Record Examination (GRE) Verbal	169 / 170 (~99th)	165 / 170 (~96th)	154 / 170 (~63rd)	AP Statistics	5 (85th - 100th)	5 (85th - 100th)	3 (40th - 63rd)
Graduate Record Examination (GRE) Writing	4 / 6 (~54th)	4 / 6 (~54th)	4 / 6 (~54th)	AP US Government	5 (88th - 100th)	5 (88th - 100th)	4 (77th - 88th)
USABO Semifinal Exam 2020	87 / 150 (99th - 100th)	87 / 150 (99th - 100th)	43 / 150 (31st - 33rd)	AP US History	5 (89th - 100th)	4 (74th - 89th)	4 (74th - 89th)
USNCO Local Section Exam 2022	36 / 60	38 / 60	24 / 60	AP World History	4 (65th - 87th)	4 (65th - 87th)	4 (65th - 87th)
Medical Knowledge Self-Assessment Program	75 %	75 %	53 %	AMC 10 ³	30 / 150 (6th - 12th)	36 / 150 (10th - 19th)	36 / 150 (10th - 19th)
Codeforces Rating	392 (below 5th)	392 (below 5th)	260 (below 5th)	AMC 12 ³	60 / 150 (45th - 66th)	48 / 150 (19th - 40th)	30 / 150 (4th - 8th)
AP Art History	5 (86th - 100th)	5 (86th - 100th)	5 (86th - 100th)	Introductory Sommelier (theory knowledge)	92 %	92 %	80 %
AP Biology	5 (85th - 100th)	5 (85th - 100th)	4 (62nd - 85th)	Certified Sommelier (theory knowledge)	86 %	86 %	58 %
AP Calculus BC	4 (43rd - 59th)	4 (43rd - 59th)	1 (0th - 7th)	Advanced Sommelier (theory knowledge)	77 %	77 %	46 %
AP Chemistry	4 (71st - 88th)	4 (71st - 88th)	2 (22nd - 46th)	Leetcode (easy)	31 / 41	31 / 41	12 / 41
AP English Language and Composition	2 (14th - 44th)	2 (14th - 44th)	2 (14th - 44th)	Leetcode (medium)	21 / 80	21 / 80	8 / 80
AP English Literature and Composition	2 (8th - 22nd)	2 (8th - 22nd)	2 (8th - 22nd)	Leetcode (hard)	3 / 45	3 / 45	0 / 45

Table 1. GPT performance on academic and professional exams. In each case, we simulate the conditions and scoring of the real exam. We report GPT-4’s final score graded according to exam-specific rubrics, as well as the percentile of test-takers achieving GPT-4’s score.

Emergent Reasoning Ability

- This proposes a great challenge to **evaluate our systems** (or even us).
 - Can we say the Turing test is passed? If so, what's next?

Exam	GPT-4	GPT-4 (no vision)	GPT-3.5	Exam	GPT-4	GPT-4 (no vision)	GPT-3.5
Uniform Bar Exam (MBE+MEE+MPT)	298 / 400 (~90th)	298 / 400 (~90th)	213 / 400 (~10th)	AP Environmental Science	5 (91st - 100th)	5 (91st - 100th)	5 (91st - 100th)
LSAT	163 (~88th)	161 (~83rd)	149 (~40th)	AP Macroeconomics	5 (84th - 100th)	5 (84th - 100th)	2 (33rd - 48th)
SAT Evidence-Based Reading & Writing	710 / 800 (~93rd)	710 / 800 (~93rd)	670 / 800 (~87th)	AP Microeconomics	5 (82nd - 100th)	4 (60th - 82nd)	4 (60th - 82nd)
SAT Math	700 / 800 (~89th)	690 / 800 (~89th)	590 / 800 (~70th)	AP Physics 2	4 (66th - 84th)	4 (66th - 84th)	3 (30th - 66th)
Graduate Record Examination (GRE) Quantitative	163 / 170 (~80th)	157 / 170 (~62nd)	147 / 170 (~25th)	AP Psychology	5 (83rd - 100th)	5 (83rd - 100th)	5 (83rd - 100th)
Graduate Record Examination (GRE) Verbal	169 / 170 (~99th)	165 / 170 (~96th)	154 / 170 (~63rd)	AP Statistics	5 (85th - 100th)	5 (85th - 100th)	3 (40th - 63rd)
Graduate Record Examination (GRE) Writing	4 / 6 (~54th)	4 / 6 (~54th)	4 / 6 (~54th)	AP US Government	5 (88th - 100th)	5 (88th - 100th)	4 (77th - 88th)
USABO Semifinal Exam 2020	87 / 150 (99th - 100th)	87 / 150 (99th - 100th)	43 / 150 (31st - 33rd)	AP US History	5 (89th - 100th)	4 (74th - 89th)	4 (74th - 89th)
USNCO Local Section Exam 2022	36 / 60	38 / 60	24 / 60	AP World History	4 (65th - 87th)	4 (65th - 87th)	4 (65th - 87th)
Medical Knowledge Self-Assessment Program	75 %	75 %	53 %	AMC 10 ³	30 / 150 (6th - 12th)	36 / 150 (10th - 19th)	36 / 150 (10th - 19th)
Codeforces Rating	392 (below 5th)	392 (below 5th)	260 (below 5th)	AMC 12 ³	60 / 150 (45th - 66th)	48 / 150 (19th - 40th)	30 / 150 (4th - 8th)
AP Art History	5 (86th - 100th)	5 (86th - 100th)	5 (86th - 100th)	Introductory Sommelier (theory knowledge)	92 %	92 %	80 %
AP Biology	5 (85th - 100th)	5 (85th - 100th)	4 (62nd - 85th)	Certified Sommelier (theory knowledge)	86 %	86 %	58 %
AP Calculus BC	4 (43rd - 59th)	4 (43rd - 59th)	1 (0th - 7th)	Advanced Sommelier (theory knowledge)	77 %	77 %	46 %
AP Chemistry	4 (71st - 88th)	4 (71st - 88th)	2 (22nd - 46th)	Leetcode (easy)	31 / 41	31 / 41	12 / 41
AP English Language and Composition	2 (14th - 44th)	2 (14th - 44th)	2 (14th - 44th)	Leetcode (medium)	21 / 80	21 / 80	8 / 80
AP English Literature and Composition	2 (8th - 22nd)	2 (8th - 22nd)	2 (8th - 22nd)	Leetcode (hard)	3 / 45	3 / 45	0 / 45

Table 1. GPT performance on academic and professional exams. In each case, we simulate the conditions and scoring of the real exam. We report GPT-4’s final score graded according to exam-specific rubrics, as well as the percentile of test-takers achieving GPT-4’s score.



∞ Meta AI
LLaMA

Is LLM a pure engineering success?



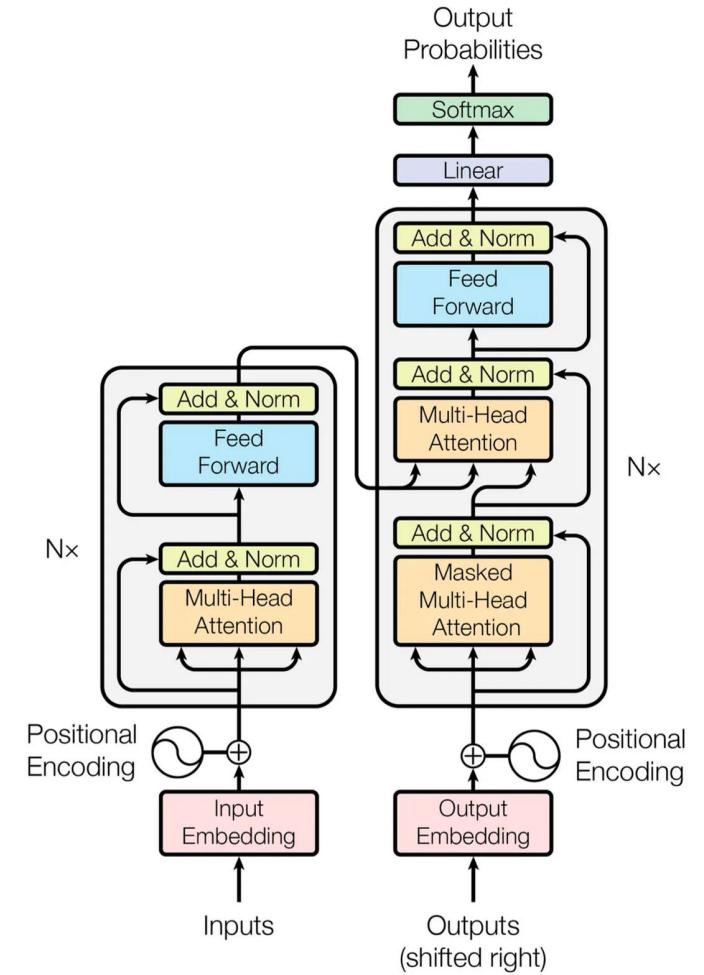
OpenAI



Bard AI

Introduction to Large Language Models

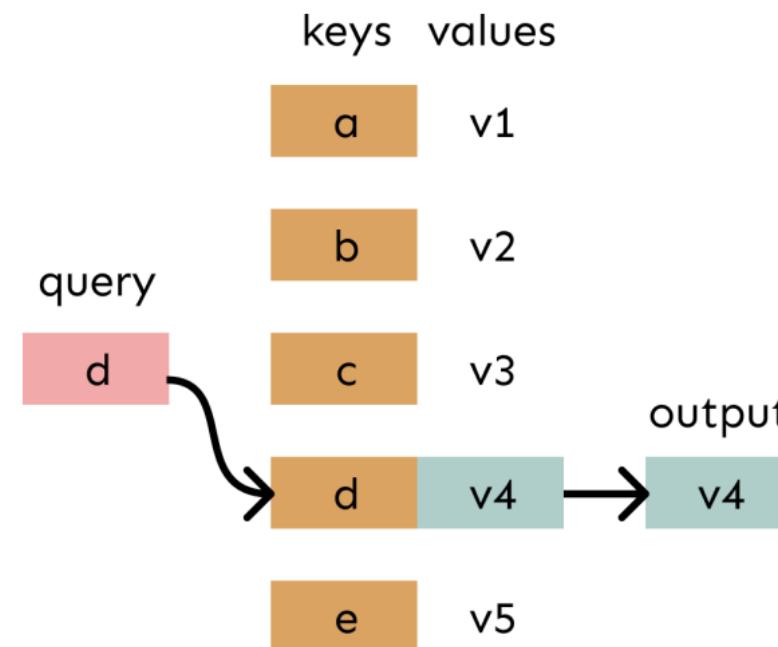
- Latest LLMs adopt the **Transformer** backbone.
- Core component: self-attention mechanism
 - Put tokens into their **context!**



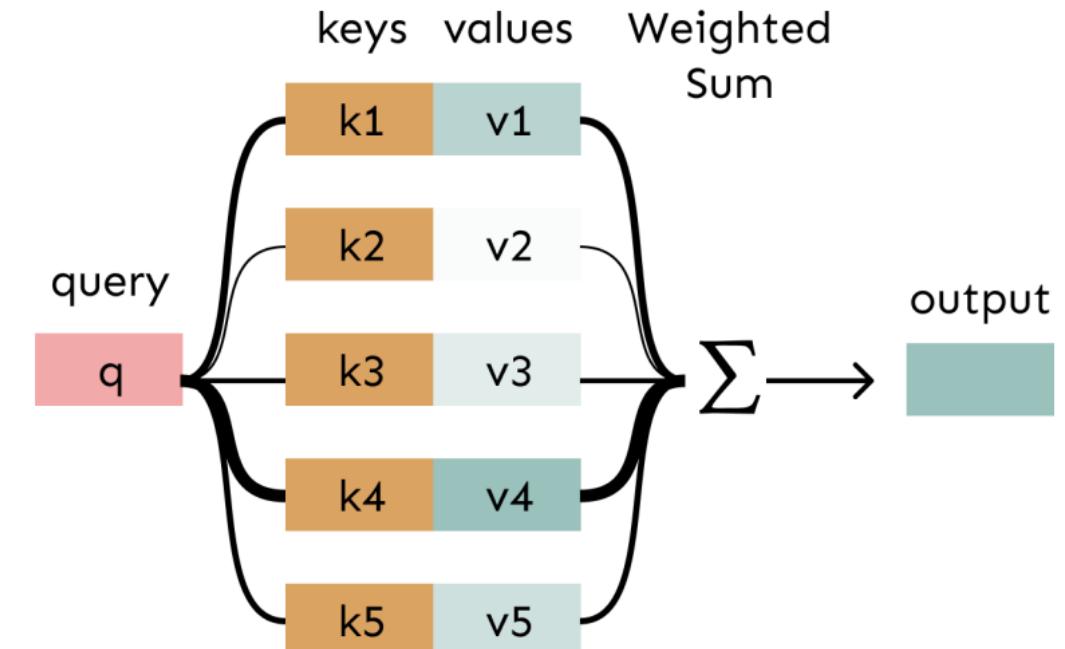
Attention as a soft, averaging lookup table

We can think of **attention** as performing fuzzy lookup in a key-value store.

In a **lookup table**, we have a table of **keys** that map to **values**. The **query** matches one of the keys, returning its value.



In **attention**, the **query** matches all **keys** softly, to a weight between 0 and 1. The keys' **values** are multiplied by the weights and summed.



Self-Attention: keys, queries, values from the same sequence

Let $\mathbf{w}_{1:n}$ be a sequence of words in vocabulary V , like *Zuko made his uncle tea*.

For each \mathbf{w}_i , let $\mathbf{x}_i = E\mathbf{w}_i$, where $E \in \mathbb{R}^{d \times |V|}$ is an embedding matrix.

1. Transform each word embedding with weight matrices Q, K, V, each in $\mathbb{R}^{d \times d}$

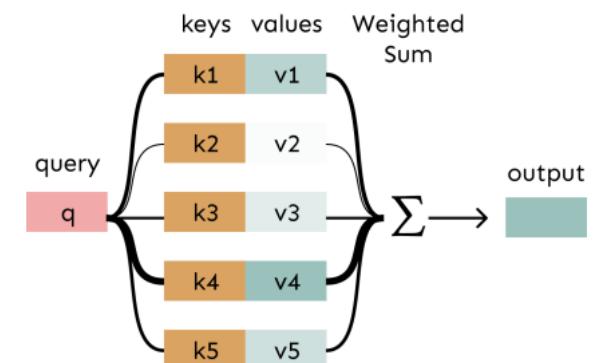
$$\mathbf{q}_i = Q \boxed{\mathbf{x}_i} \text{ (queries)} \quad \mathbf{k}_i = K \boxed{\mathbf{x}_i} \text{ (keys)} \quad \mathbf{v}_i = V \boxed{\mathbf{x}_i} \text{ (values)}$$

2. Compute pairwise similarities between keys and queries; normalize with softmax

$$\mathbf{e}_{ij} = \mathbf{q}_i^\top \mathbf{k}_j \quad \alpha_{ij} = \frac{\exp(\mathbf{e}_{ij})}{\sum_j \exp(\mathbf{e}_{ij'})}$$

3. Compute output for each word as weighted sum of values

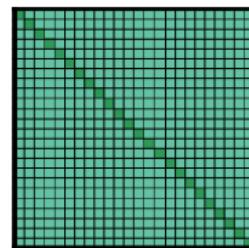
$$\mathbf{o}_i = \sum_j \alpha_{ij} \mathbf{v}_i$$



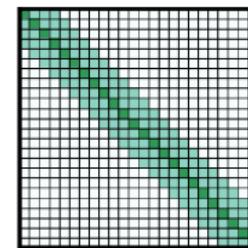
Self-Attention: Fully-connected Graph in One Pass

Another way to understand attention is to leverage the perspective of **graph**.

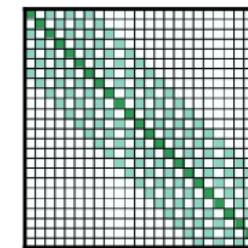
- Consider a sequence as a fully-connected graph K_n , where each vertex corresponds to a token in the sequence.
- Assign v_i as the value of the i -th vertex, and $q_i^T k_j$ as the weight of the edge e_{ij} . The attention calculation is iterating the value of each vertex using the weighted average of the values of its connected vertices.
 - All vertices can be updated in parallel. (GPU-friendly!)
 - It's easy to manipulate information flow. (Add mask to the weight of e_{ij} .)



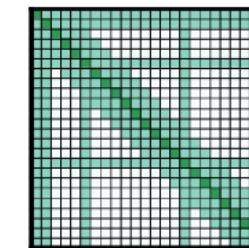
(a) Full n^2 attention



(b) Sliding window attention



(c) Dilated sliding window



(d) Global+sliding window

Bring in Other Components

Introducing nonlinearity to make piling up multiple attention layers non-trivial.

A token may need to look at multiple places in the sentence ta once.

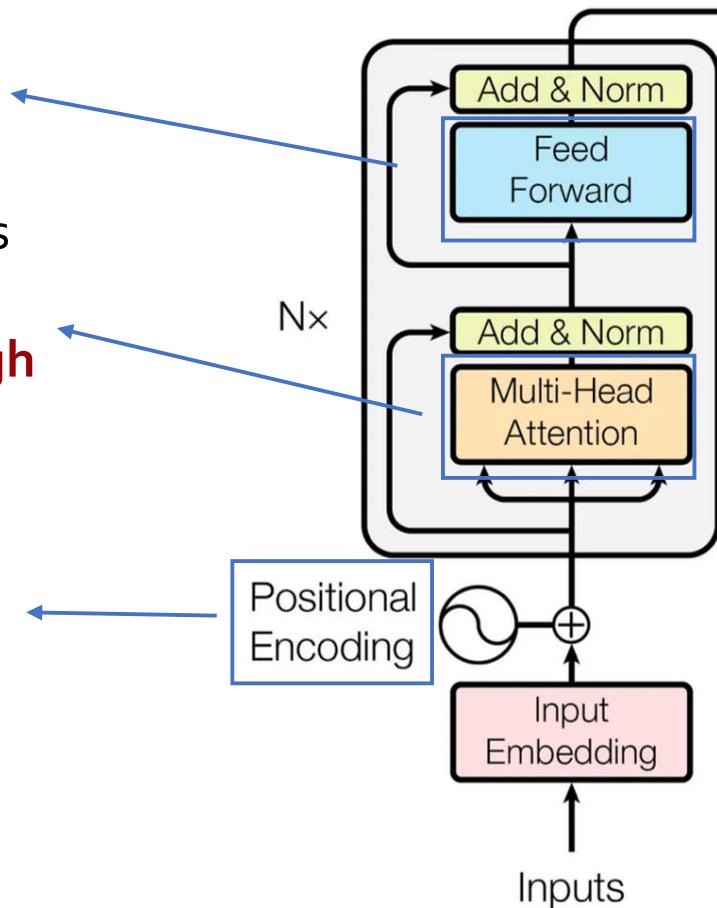
-> Define multiple attention heads through multiple Q,K,V matrices.

Attention mechanism doesn't have an inherent notion of order.

-> Add positional encoding to the inputs.

$$PE_{(pos,2i)} = \sin \left(pos / 10000^{2i/d_{\text{model}}} \right)$$

$$PE_{(pos,2i+1)} = \cos \left(pos / 10000^{2i/d_{\text{model}}} \right)$$



The Transformer was born 6 years ago...

- There isn't much improvement in the model architecture.

Google Study Shows Transformer Modifications Fail To Transfer Across
Implementations and Applications. Mar 3, 2021

 ACL Anthology
[https://aclanthology.org/2021.emnlp-main.46...](https://aclanthology.org/2021.emnlp-main.46) PDF

:

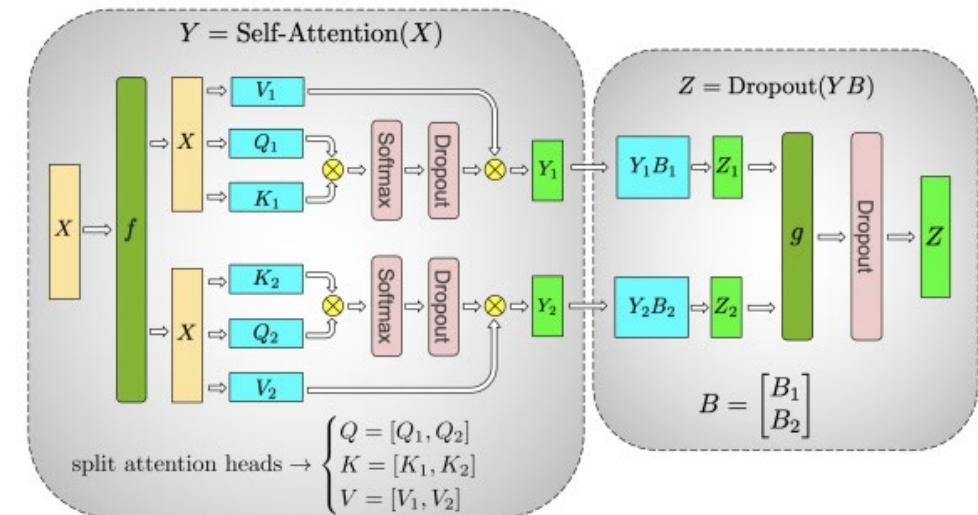
[Do Transformer Modifications Transfer Across ...](#)

by S Narang · 2021 · Cited by 48 — Another possible explanation is that the modifications proposed to the Transformer do not “generalize” across Page 2 5759 applications, i.e. ...
16 pages

- Maybe it's because the Transformer is powerful enough.
 - Theoretically proved: Transformers with trainable positional encodings are universal approximators of continuous sequence-to-sequence functions on a compact domain. ([Yun et al., 2019](#))

The Transformer was born 6 years ago...

- In these years, to better use the Transformer, researchers have been working on
 - Designing **optimizers** which are more suitable to the Transformer: e.g., AdamW ([Loshchilov and Hutter, 2018](#))
 - Designing **parallel computing** algorithm to make training larger models possible: e.g., model parallel ([Megatron-LM, Shoeybi et al., 2020](#))



(b) Self-Attention

The Transformer was born 6 years ago...

- In these years, to better use the Transformer, researchers have been working on
 - Reducing the computational overhead: e.g., faster layer normalization ([Zhang and Sennrich, 2019](#)), sparse attention (recall the perspective of graph)
 - Designing methods to improve training stability: e.g., modified initialization (GPT-2, GPT-3)

Introduction to Large Language Models

- Level of linguistic knowledge

Not applicable

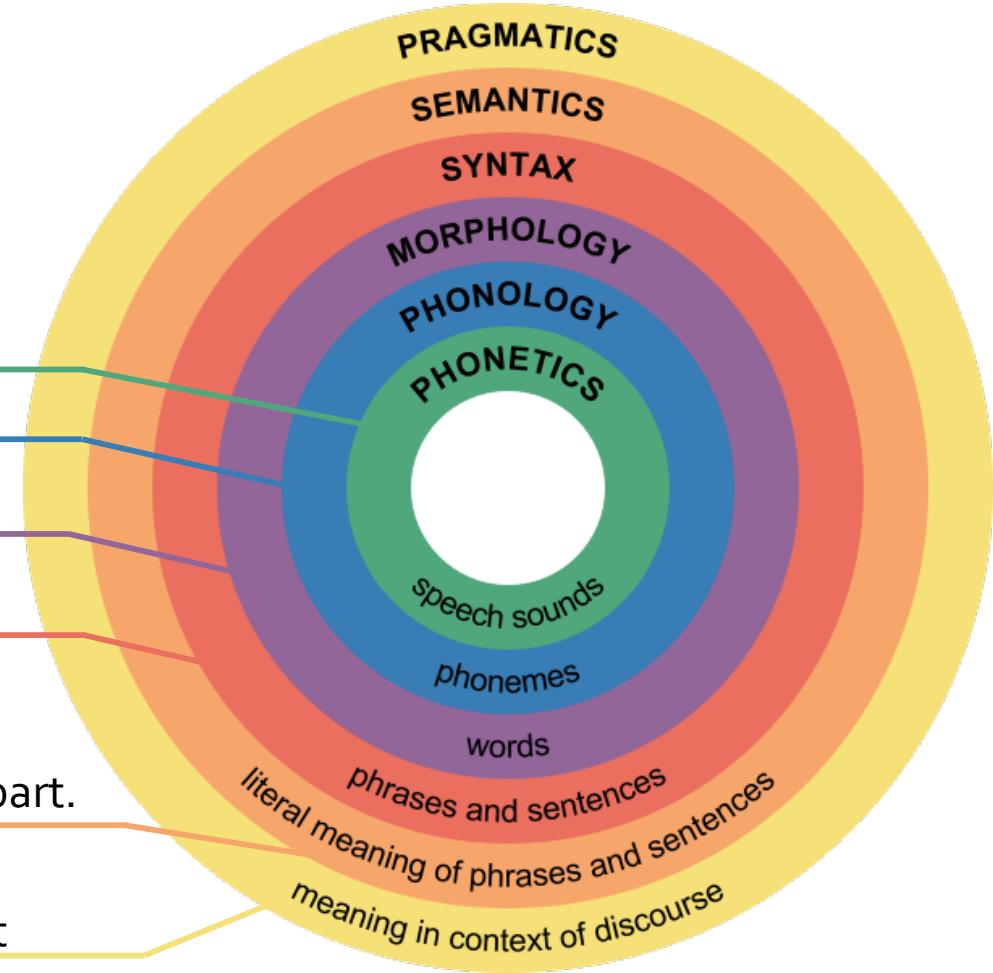
Not applicable

Great. LLMs are robust to typos, coinage, cacography

Great. Dependency parsing and coreference resolution are almost solved. Syntactic information is captured in attention ([Clark et al., 2019](#)).

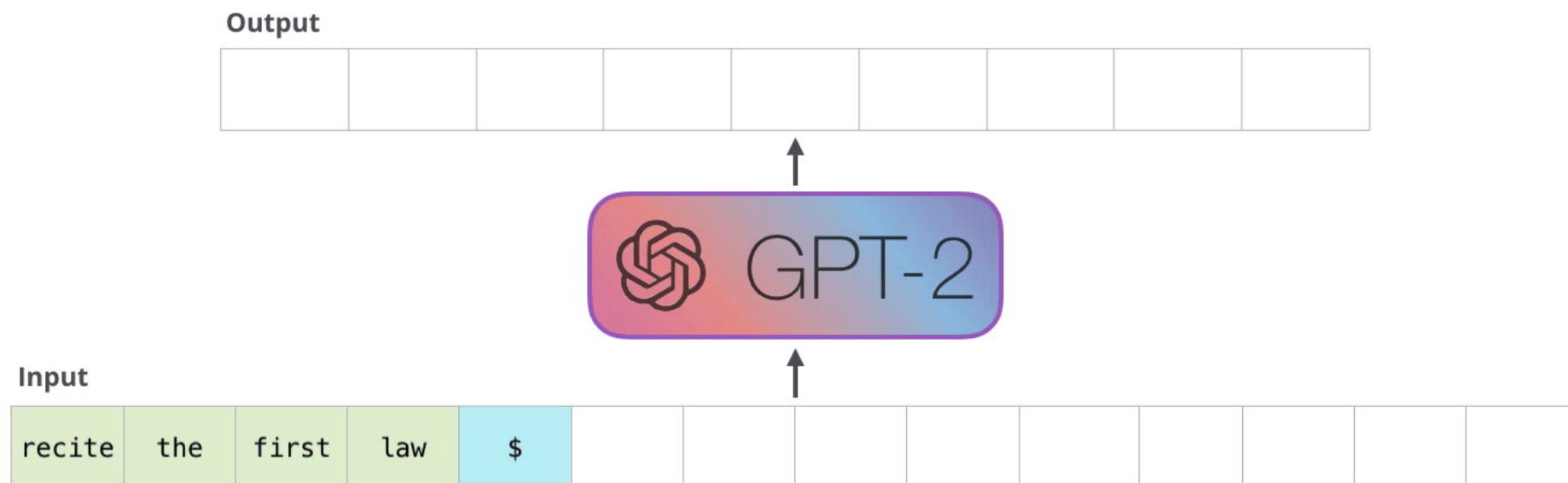
Great. Most of current NLP benchmarks focus on this part.

It depends! LLMs still get confused when they meet unique contexts or special users (e.g., those in underrepresented groups)



How do LLMs acquire the knowledge of language?

- **Unsupervised pre-training** on very large corpus
 - There are many pre-training methods, here we focus on the one used by the GPT family.
 - Language modeling: predict the next word



How do LLMs acquire the knowledge of language?

There's increasing evidence that pretrained models learn a wide variety of things about the statistical properties of language. Taking our examples from the start of class:

- *Stanford University is located in _____, California.* [Trivia]
- *I put __ fork down on the table.* [syntax]
- *The woman walked across the street, checking for traffic over __ shoulder.* [coreference]
- *I went to the ocean to see the fish, turtles, seals, and ____.* [lexical semantics/topic]
- *Overall, the value I got from the two hours watching it was the sum total of the popcorn and the drink. The movie was ____.* [sentiment]
- Iroh went into the kitchen to make some tea. Standing next to Iroh, Zuko pondered his destiny. Zuko left the _____. [some reasoning – this is harder]
- I was thinking about the sequence that goes 1, 1, 2, 3, 5, 8, 13, 21, ____ [some basic arithmetic; they don't learn the Fibonacci sequence]
- Models also learn – and can exacerbate racism, sexism, all manner of bad biases.

These examples are copied from Stanford CS224N/Ling284 slides (author: John Hewitt).

They are actually examples for **masked language modeling** which is a bit different from how GPT is pre-trained.

LLMs learn more than the knowledge of language

“A language is not just words. It's a culture, a tradition, a unification of a community, a whole history that creates what a community is. It's all embodied in a language.”

— Noam Chomsky

LLMs learn more than the knowledge of language

- Dig into the training corpus of LLaMA, an open-sourced English LLM.
 - English CommonCrawl [67%]: **webpages**, covering nearly everything you can think of right now.
 - C4 [15%]: a colossal, cleaned version of Common Crawl corpus.
 - Github [4.5%]: training on **code** may greatly benefit LLMs' reasoning ability ([Fu et al., 2022](#))
 - Wikipedia [4.5%]: high quality **knowledge source**
 - Gutenberg and Books3 [4.5%]: containing **formal writings** in various disciplines
 - ArXiv [2.5%]: **academy papers**
 - Stack Exchange [2%]: high quality **questions and answers**
- Besides publicly available data, OpenAI includes their proprietary data in training.

LLMs learn more than the knowledge of language

Open question:

- Are LLMs stochastic parrots or world models?

From GPT-3 to ChatGPT: Learn human intents behind their language

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.



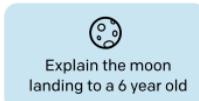
Information behind this sentence:
People usually use imperative sentence to make a request. The listener is expected to complete that request.

Follow Instructions & Align with Human Preference

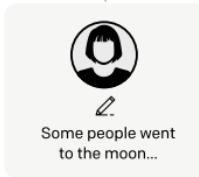
Step 1

Collect demonstration data,
and train a supervised policy.

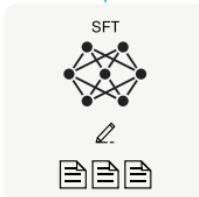
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



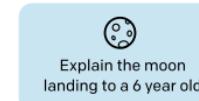
This data is used to fine-tune GPT-3 with supervised learning.



Step 2

Collect comparison data,
and train a reward model.

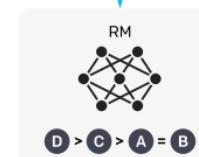
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



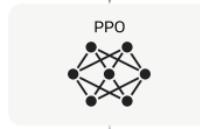
Step 3

Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.



The policy generates an output.



Once upon a time...

The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.

r_k

Figure 2: A diagram illustrating the three steps of our method: (1) supervised fine-tuning (SFT), (2) reward model (RM) training, and (3) reinforcement learning via proximal policy optimization (PPO) on this reward model. Blue arrows indicate that this data is used to train one of our models. In Step 2, boxes A-D are samples from our models that get ranked by labelers. See Section 3 for more details on our method.

Follow Instructions & Align with Human Preference

Human-in-the-loop!
(Discuss more later)

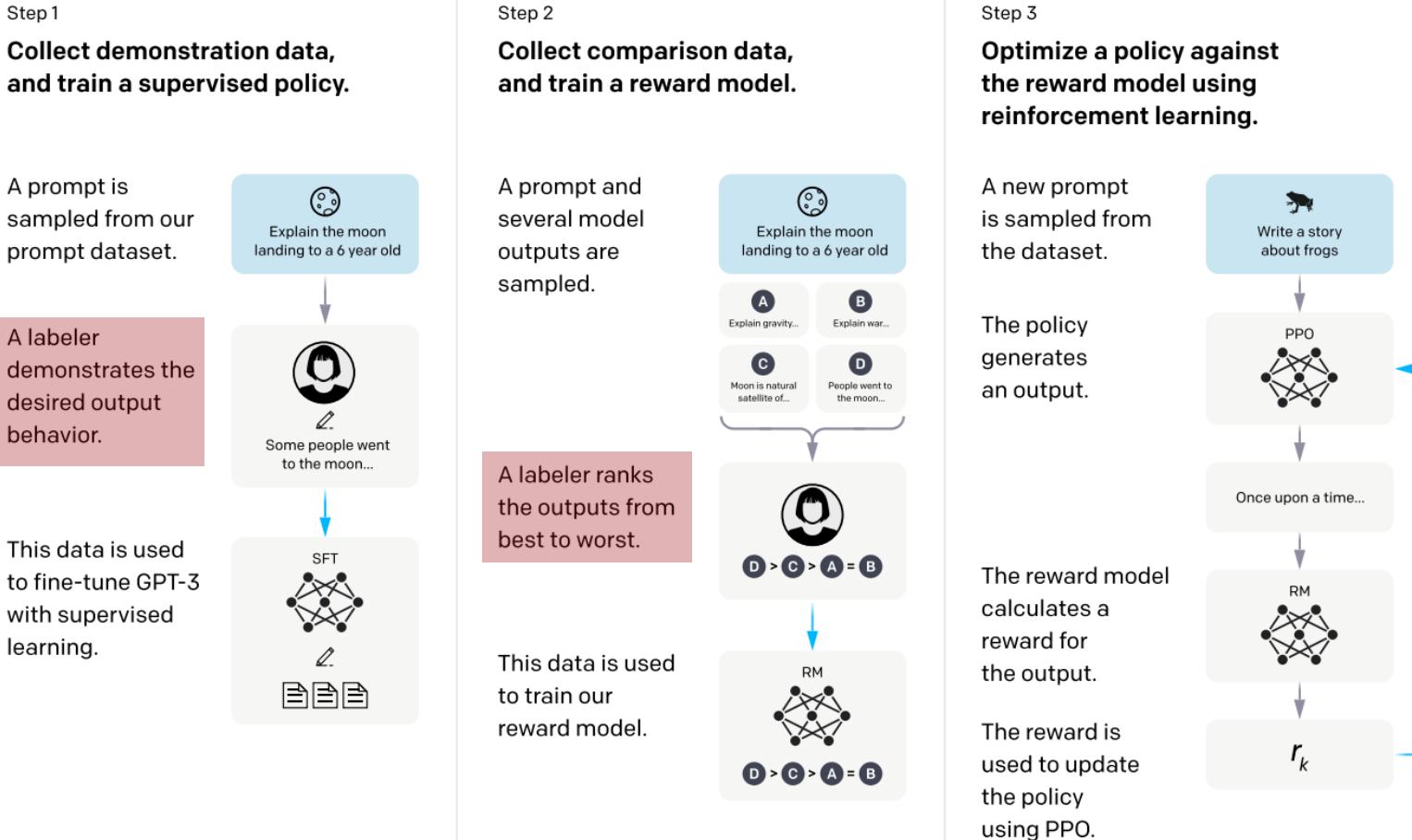
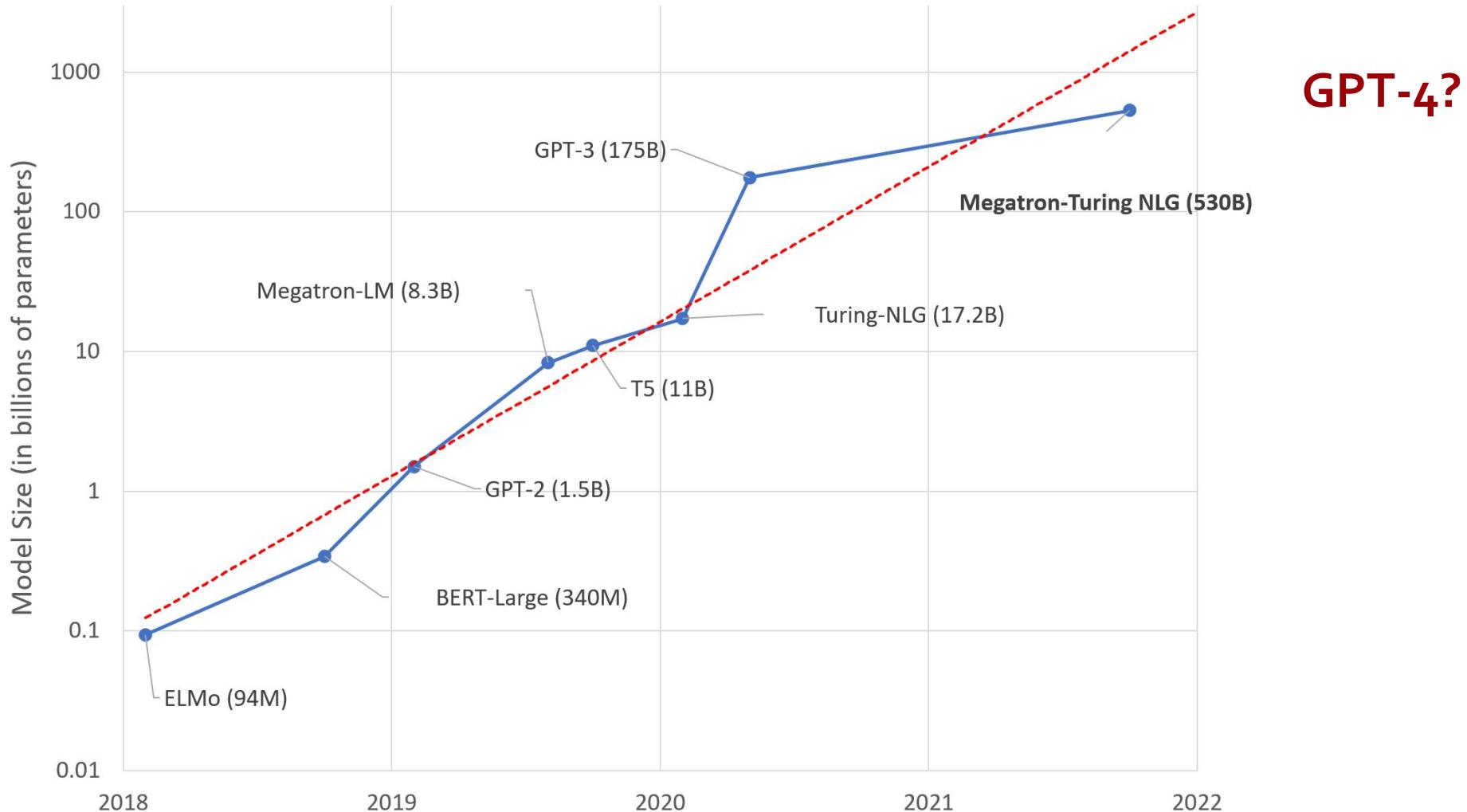


Figure 2: A diagram illustrating the three steps of our method: (1) supervised fine-tuning (SFT), (2) reward model (RM) training, and (3) reinforcement learning via proximal policy optimization (PPO) on this reward model. Blue arrows indicate that this data is used to train one of our models. In Step 2, boxes A-D are samples from our models that get ranked by labelers. See Section 3 for more details on our method.

Introduction to Large Language Models



Introduction to Large Language Models

- In-context learning

- No parameter update
- Wrap “training” samples in the prompt

Circulation revenue has increased by 5%
in Finland. // Positive

Panostaja did not disclose the purchase
price. // Neutral

Paying off the national debt will be
extremely painful. // Negative

The company anticipated its operating
profit to improve. // _____

Circulation revenue has increased by
5% in Finland. // Finance

They defeated ... in the NFC
Championship Game. // Sports

Apple ... development of in-house
chips. // Tech

The company anticipated its operating
profit to improve. // _____



Introduction to Large Language Models

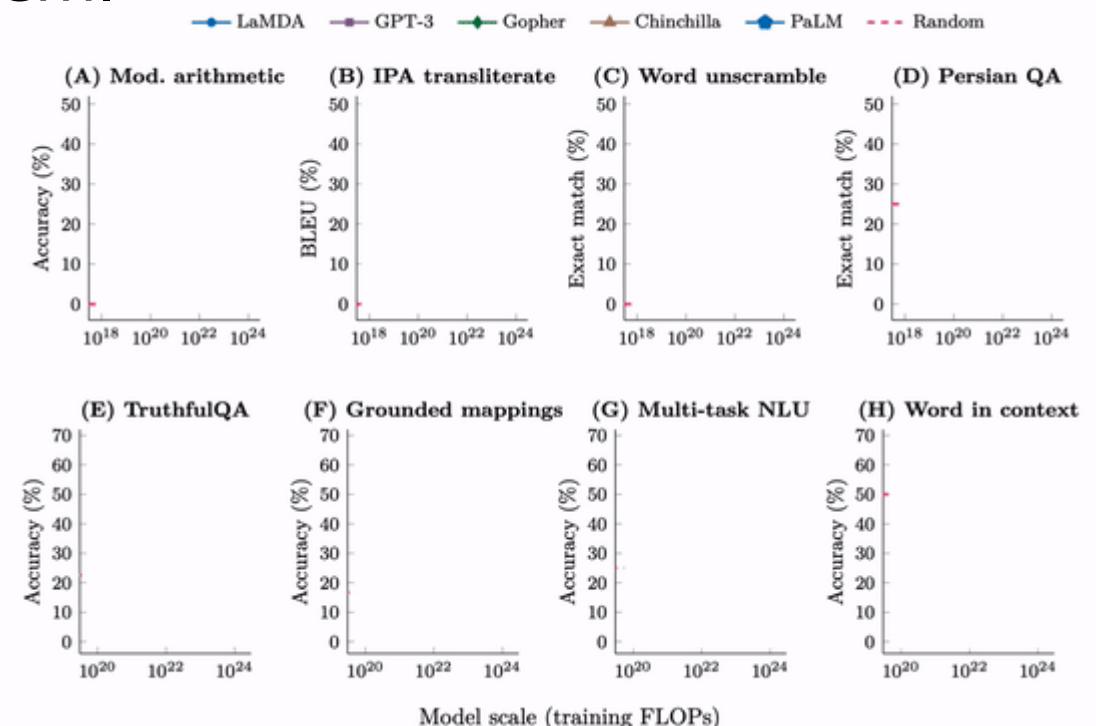
- Open question:
 - Why does in-context learning work?
- There are some hypotheses but no conclusion yet
 - Xie et al. "An Explanation of In-context Learning as Implicit Bayesian Inference" *ICLR* 2022
 - Akyürek et al. "What learning algorithm is in-context learning? Investigations with linear models" *ICLR* 2023
 - Oswald et al. "Transformers learn in-context by gradient descent" *Arxiv* 2022

Introduction to Large Language Models

- **Emergence abilities**
 - An ability is emergent if it is **not** present in smaller models but is present in larger models.
- In-context learning ability is one of them.
- **Scaling to improve unlock abilities.**

Emergence in few-shot prompting

This gif is copied from Jason Wei's slides.



Introduction to Large Language Models

3 Predictable Scaling

A large focus of the GPT-4 project was building a deep learning stack that scales predictably. The primary reason is that for very large training runs like GPT-4, it is not feasible to do extensive model-specific tuning. To address this, we developed infrastructure and optimization methods that have very predictable behavior across multiple scales. These improvements allowed us to reliably predict some aspects of the performance of GPT-4 from smaller models trained using $1,000\times - 10,000\times$ less compute.

3.1 Loss Prediction

The final loss of properly-trained large language models is thought to be well approximated by power laws in the amount of compute used to train the model [41, 42, 2, 14, 15].

To verify the scalability of our optimization infrastructure, we predicted GPT-4’s final loss on our internal codebase (not part of the training set) by fitting a scaling law with an irreducible loss term (as in Henighan et al. [15]): $L(C) = aC^b + c$, from models trained using the same methodology but using at most 10,000x less compute than GPT-4. This prediction was made shortly after the run started, without use of any partial results. The fitted scaling law predicted GPT-4’s final loss with high accuracy (Figure 1).

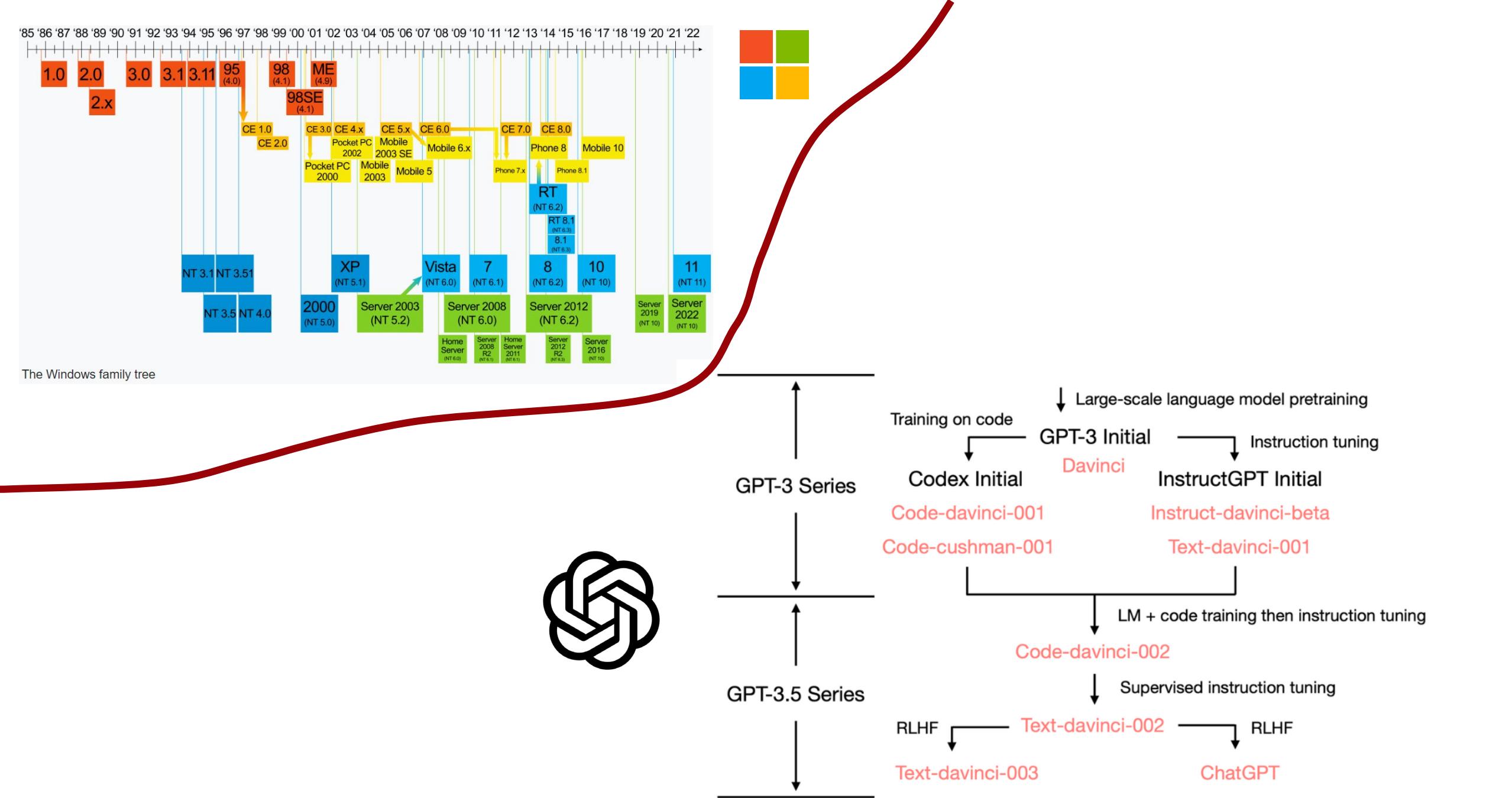
Very impressive, but I don’t know how they achieve this.

3.2 Scaling of Capabilities on HumanEval

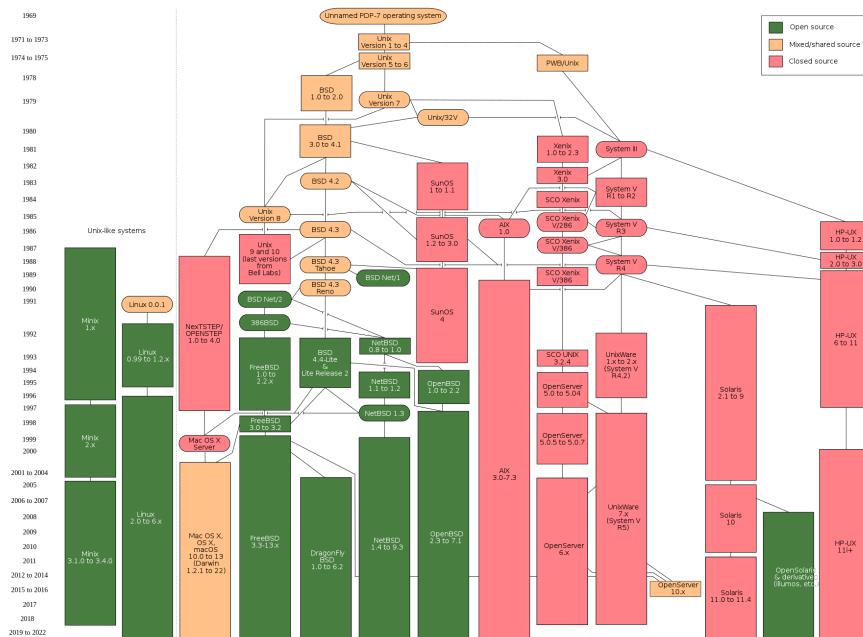
Having a sense of the capabilities of a model before training can improve decisions around alignment, safety, and deployment. In addition to predicting final loss, we developed methodology to predict more interpretable metrics of capability. One such metric is pass rate on the HumanEval dataset [43],

View LLMs from a system perspective

- Analogy: operating system (OS)
 - Knowing a set of algorithms is **not** enough to build a good OS.
 - Knowing a training algorithm/recipe is **not** enough to build a good LLM.
- **Model patching & continual training** of LLM are important.
 - We shouldn't always build a new LLM from scratch.
 - I think this may be one reason for OpenAI's success – they build LLMs as building a system (maintenance, version control, incremental update)

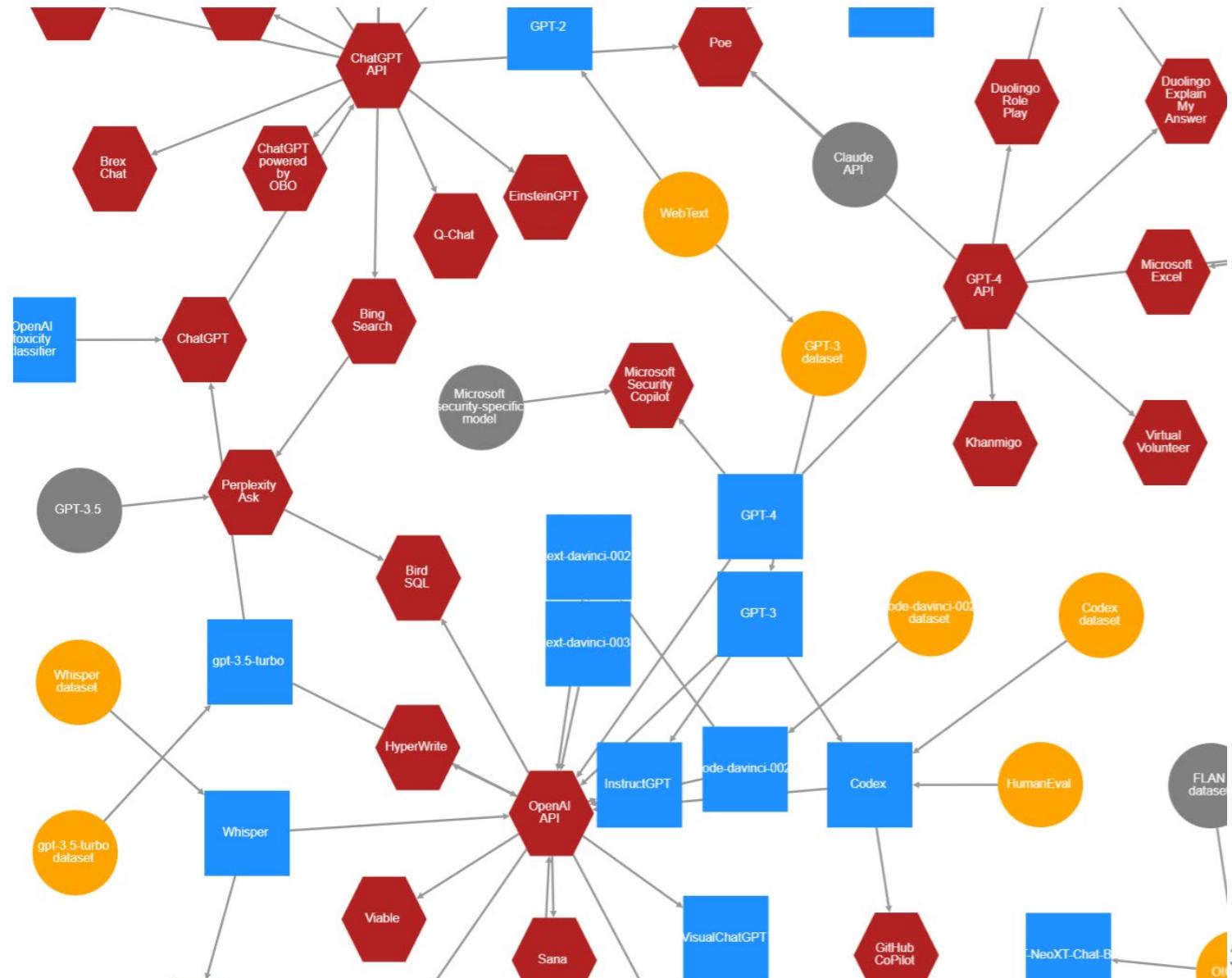


Unix Family Tree



Ecosystem Graphs

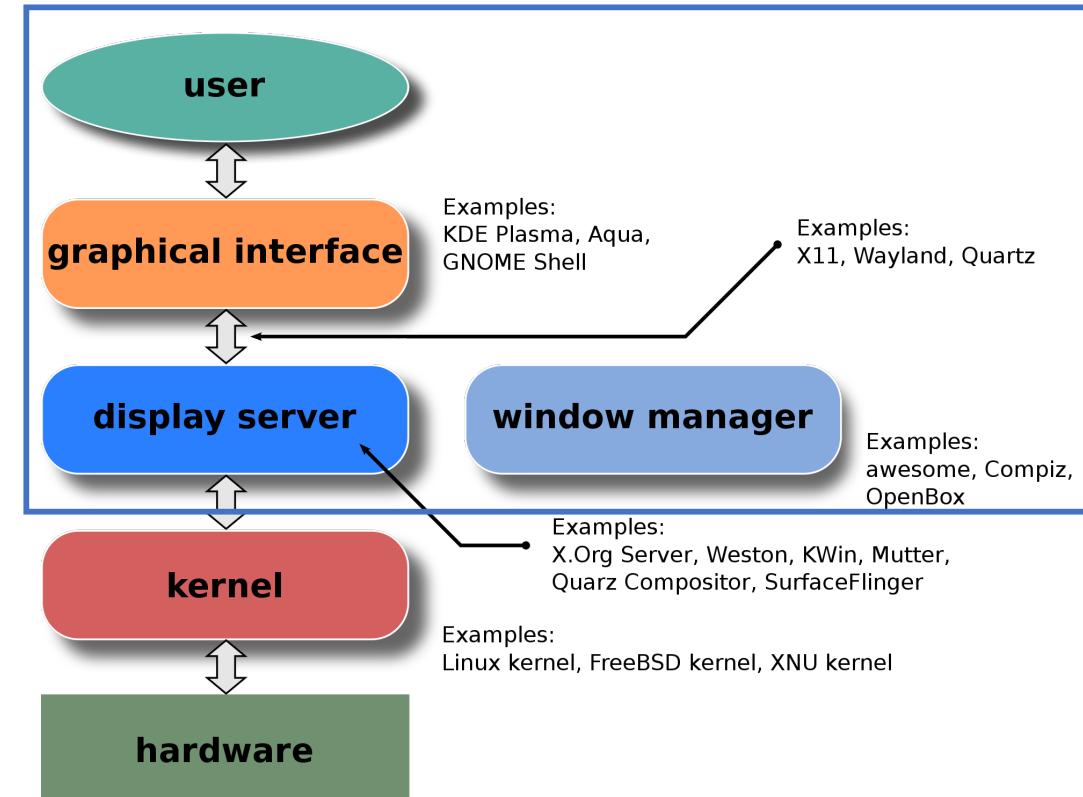
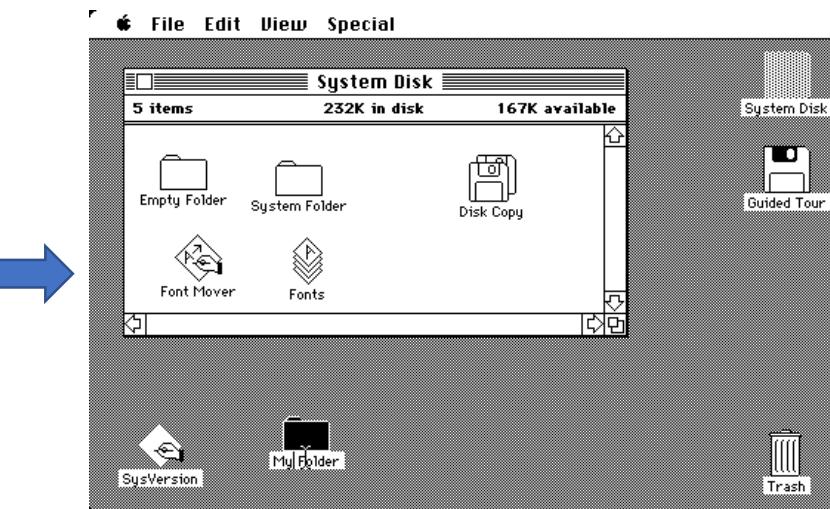
ecosystem graphs



Put LLMs into a Larger System

- Analogy: operating system (OS)
 - How do we interact with OS?
 - How do we interact with LLMs?

```
Current date is Tue 1-01-1980  
Enter new date:  
Current time is 21:35:24.18  
Enter new time:  
  
The IBM Personal Computer DOS  
Version 2.00 (C)Copyright IBM Corp 1981, 1982, 1983  
  
A>dir  
  
Volume in drive A has no label  
Directory of A:\  
  
COMMAND COM 17664 3-08-83 12:00p  
FORMAT COM 6016 3-08-83 12:00p  
CHDSX COM 6400 3-08-83 12:00p  
SYS COM 1408 3-08-83 12:00p  
DEBUG COM 11964 3-08-83 12:00p  
SLOOP 32 1-01-80 7:44p  
6 File(s) 292864 bytes free  
  
A>_
```

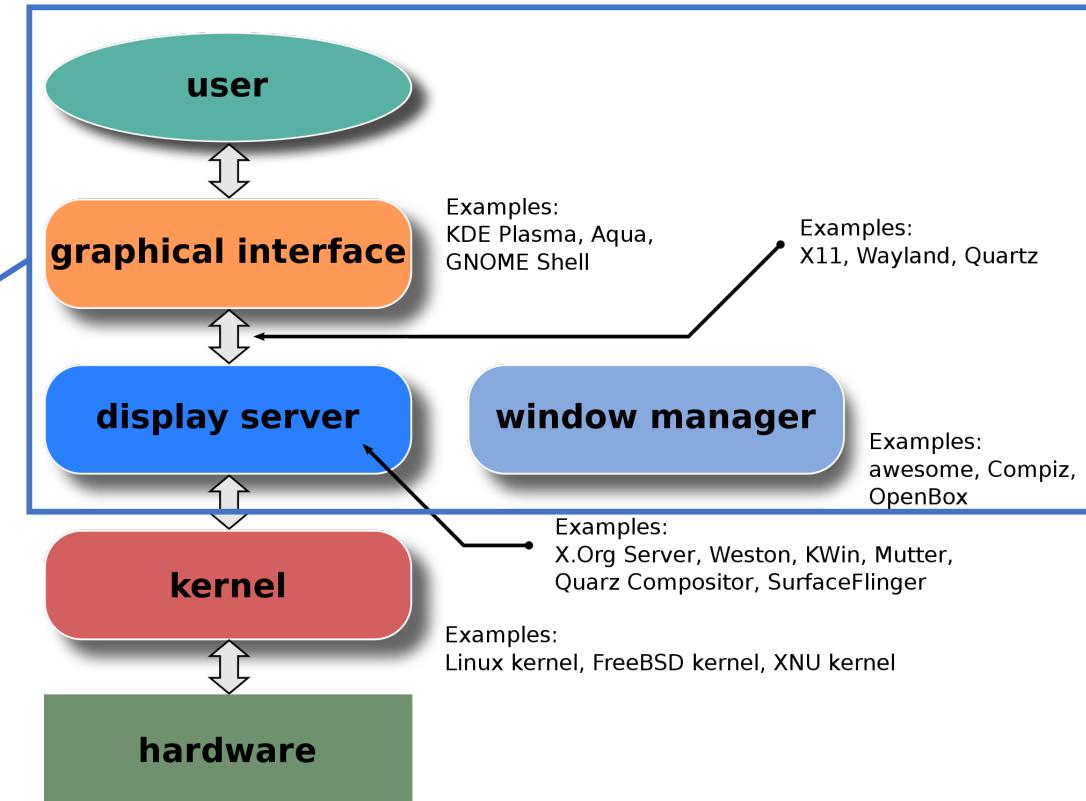


Put LLMs into a Larger System

- Analogy: operating system (OS)
 - How do we interact with OS?
 - How do we interact with LLMs?

This part is now also considered as a part of the OS in general.

- Make the system more accessible, especially for non-computer experts.



Put LLMs into a Larger System

The user briefly describes his/her goal. AutoGPT breaks the goal into detailed steps and refine its own plan.

```
PS D:\Auto-GPT> python -m autogpt --continuous
Continuous Mode: ENABLED
WARNING: Continuous mode is not recommended. It is potentially dangerous and may cause your AI to run forever or carry out actions you would not usually authorise. Use at your own risk.
Welcome back! Would you like me to return to being AutoGPT-Demo?
Continue with the last settings?

Name: AutoGPT-Demo
Role: an ai designed to teach me about auto gpt
Goals: ['search auto gpt', 'find the github and figure out what the project is', 'explain what auto gpt is in a file named autogpt.txt', 'terminate']

Continue (y/n): y
Using memory of type: LocalCache
AUTOGPT-DEMO THOUGHTS: I think the first step should be to use the 'google' command to search for 'Auto GPT'.
REASONING: This will help us gather more information about Auto GPT and we can proceed with identifying the relevant GitHub project

PLAN:
- Use 'google' to search for 'Auto GPT'
- Browse relevant websites to find the GitHub project
- Write a document explaining what Auto GPT is

CRITICISM: I need to be sure to remain focused and efficient in my use of the 'google' command to minimize the number of steps needed to identify the relevant GitHub project and answer the key questions.

Attempting to fix JSON by finding outermost brackets
Apparently json was fixed.

NEXT ACTION: COMMAND = google ARGUMENTS = {'input': 'Auto GPT'}
SYSTEM: Command google returned: [ { "title": "Auto-GPT: An Autonomous GPT-4 Experiment - GitHub", "href": "https://github.com/Torantulino/Auto-GPT", "body": "Auto-GPT is an experimental AI application..." ]
```

to find relevant websites

AutoGPT demo

Put LLMs into a Larger System

Model: Plugins ALPHA • Enabled plugins: ☀️ ⚡ 🥕

CH Looking to eat vegan food in San Francisco this weekend. Could you get me one great restaurant suggestion for Saturday and a simple recipe for Sunday (just the ingredients) ? Please calculate the calories for the recipe using WolframAlpha. Finally order the ingredients on Instacart.

• Using OpenTable... ↗

Instacart.

Used Instacart

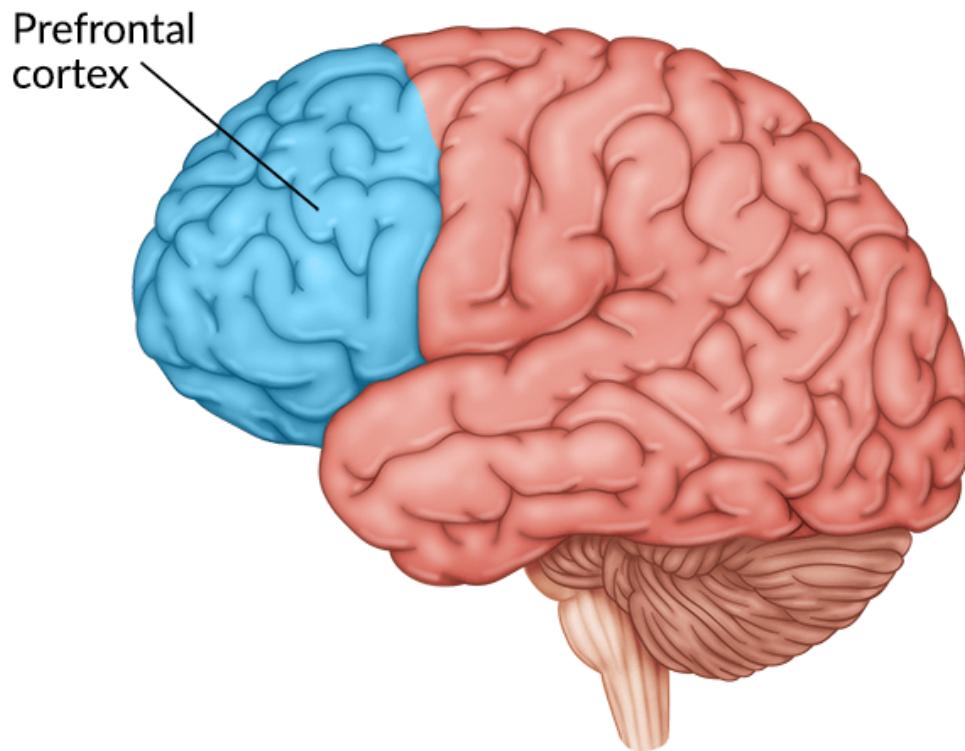
LLM functions as a controller and can **use tool** on its own.

Here's your shopping list for the [Chickpea Salad](#). If you're new to **Instacart**, you can get free delivery when you place this order. Bon appétit!



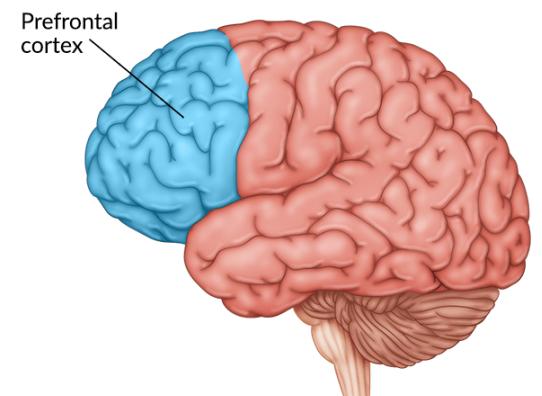
Shop this recipe, get free

LLM as a Controller



LLM as a Controller: Challenges

- How to design the **interaction interface** between LLMs and other components (e.g., external databases, API schemas)?
 - Desiderata:
robustness, unambiguity, privacy-protecting, easy-to-build for non-AI developers
- How to **maintain the state** of LLM?
 - Naïve solution: Cramming all the previous contexts into the prompt.
 - Problems:
 - The sequence length is limited (recall the attention mechanism).
 - Multiple individual calls to the LLM cause great overhead.

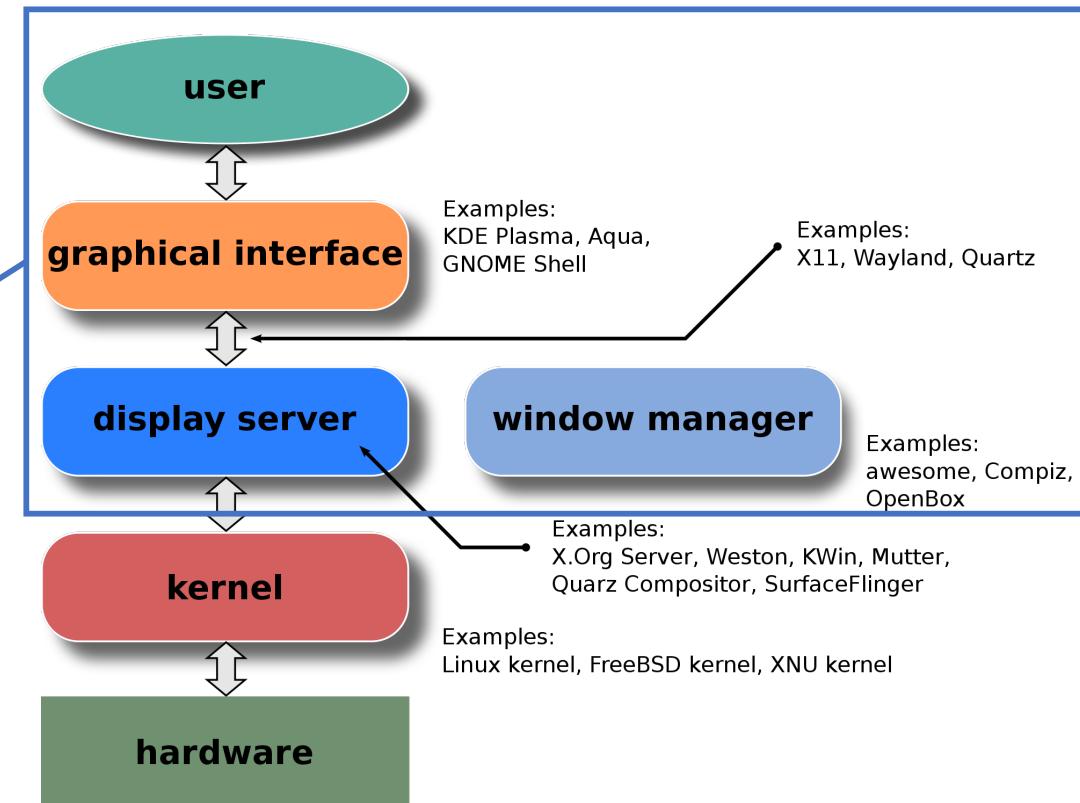


Bring Human into the Loop

- Returning to the OS analogy
- What's special with LLMs?
 - **LLMs can learn from the human-model interaction and evolve.**

This part is now also considered as a part of the OS in general.

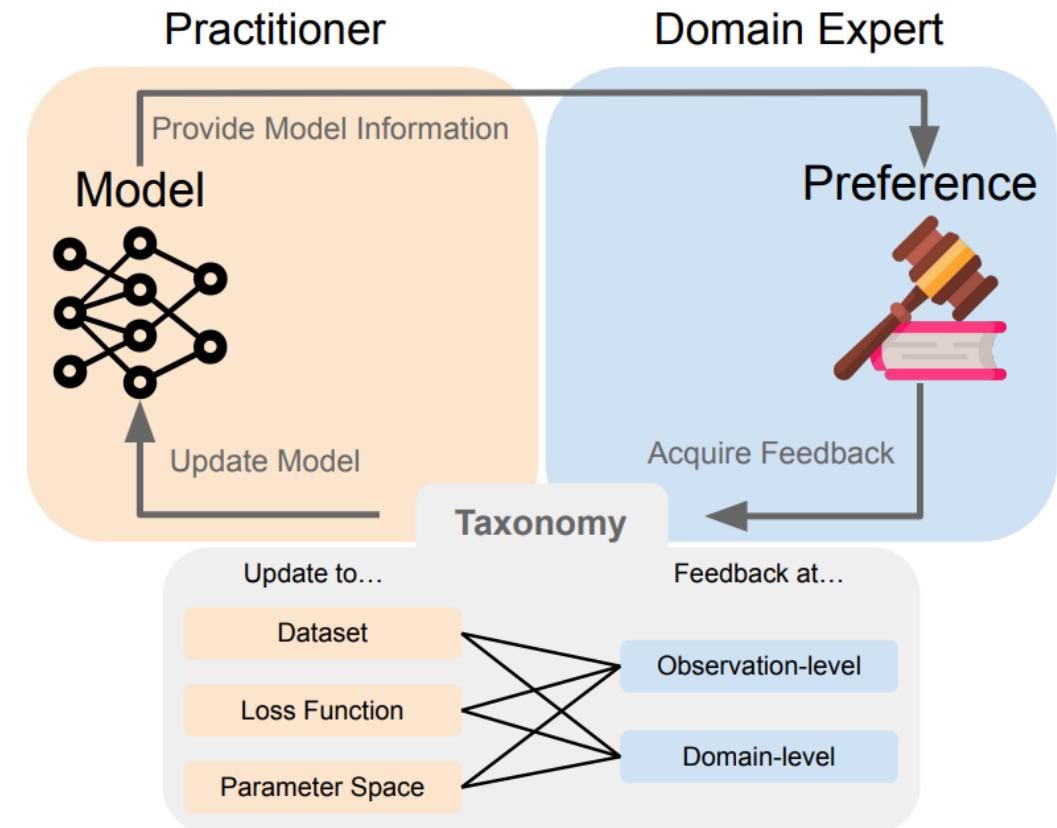
- Make the system more accessible, especially for non-computer experts.



Bring Human into the Loop

Core challenges:

- How can we let human easily provide feedback?
 - Exploiting cheap labor is unethical and infeasible to collecting domain-specific feedbacks.
 - I think research from the HCI side is important.
- How can we let the LLM take feedback?
 - Current approach: RLHF
 - What's next? (distinct challenges exist)



Distinct Challenges in Learning from Human Feedback

- **Human feedback is noisy.** The model should decide whether to take the feedback rather than viewing it as the ground truth.
- out-of-distribution detection -> “out-of-confidence” detection
 - In OOD detection, we design algorithm to assign a score to an instance to indicate how much it belongs to the training distribution, or in other words, how much the model should be capable of predicting its label.
 - I think the LLM should also assign a confidence score to the input question.

Model “Model Confidence”

- The confidence score may be broken into two parts:
 - uncertainty about the user’s goal (intrinsic to the input question)
 - confidence in its answer (related to the sampling in the output generation)

Model “Model Confidence”

- I found Anthropic has done initial work on this.
 - Their approach is asking these two questions to the LLM itself. (Similar to reflection)
 - Many limitations exist: infinite recursion, generalization problem, etc.

3 From Calibration to Knowing What You Know	8
3.1 Replacing an Option with ‘None of the Above’ Harms Performance and Calibration	8
3.2 Models are Well-Calibrated on True/False Tasks	10
3.3 RLHF Policy Miscalibration Can Be Remediated with a Temperature Tuning	11
 4 Ask the AI: Is your proposed answer True or False?	 11
4.1 Basic Self-Evaluation	11
4.2 Showing Many $T = 1$ Samples Improves Self-Evaluation	12

Recap

- LLMs trigger a paradigm shift.
 - Users define tasks.
 - New evaluation methods are needed.
- LLM is not a pure engineering success.
 - The Transformer architecture is powerful.
 - Tracing LLMs' abilities back to the data source and training objectives.
 - Emergent abilities and the scaling law is still mysterious.
- We *may* view LLMs from a system perspective.

Q&A

Happy for further discussion: shaoyj@pku.edu.cn