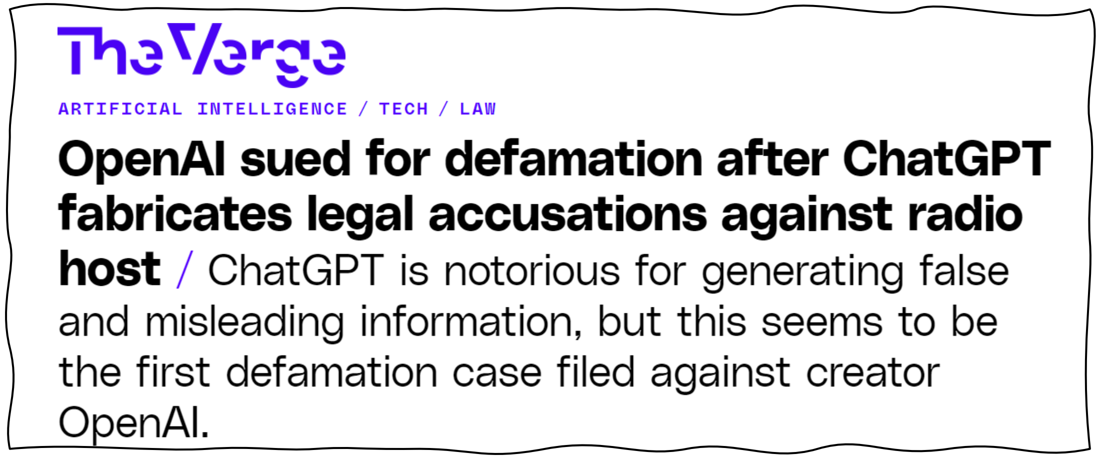
Auditing, Understanding, and Leveraging Large Language Models

Robin Jia
University of Southern California
2024 research talk (Stanford, UCSD, UC Berkeley, UCLA)

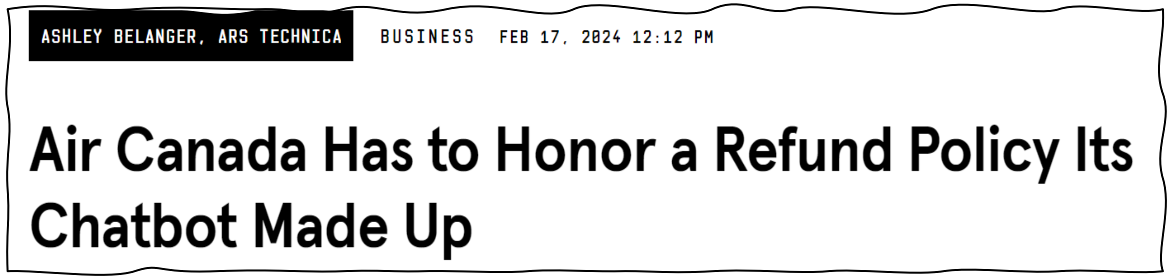
Large Language Models, Warts and All



The Verge

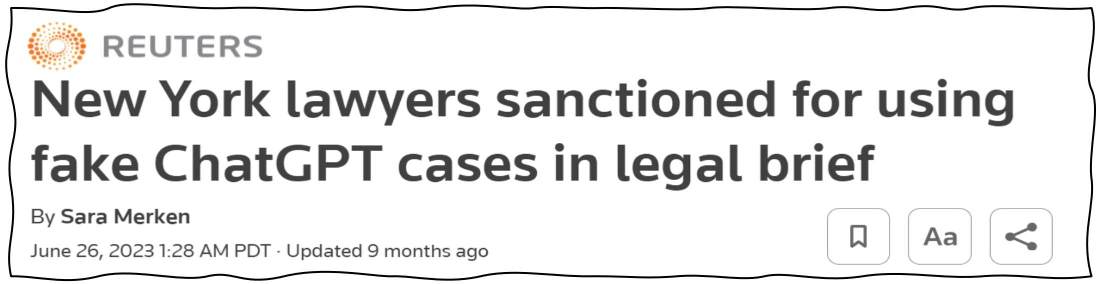
ARTIFICIAL INTELLIGENCE / TECH / LAW

OpenAI sued for defamation after ChatGPT fabricates legal accusations against radio host / ChatGPT is notorious for generating false and misleading information, but this seems to be the first defamation case filed against creator OpenAI.



ASHLEY BELANGER, ARS TECHNICA BUSINESS FEB 17, 2024 12:12 PM

Air Canada Has to Honor a Refund Policy Its Chatbot Made Up



REUTERS

New York lawyers sanctioned for using fake ChatGPT cases in legal brief

By Sara Merken

June 26, 2023 1:28 AM PDT · Updated 9 months ago

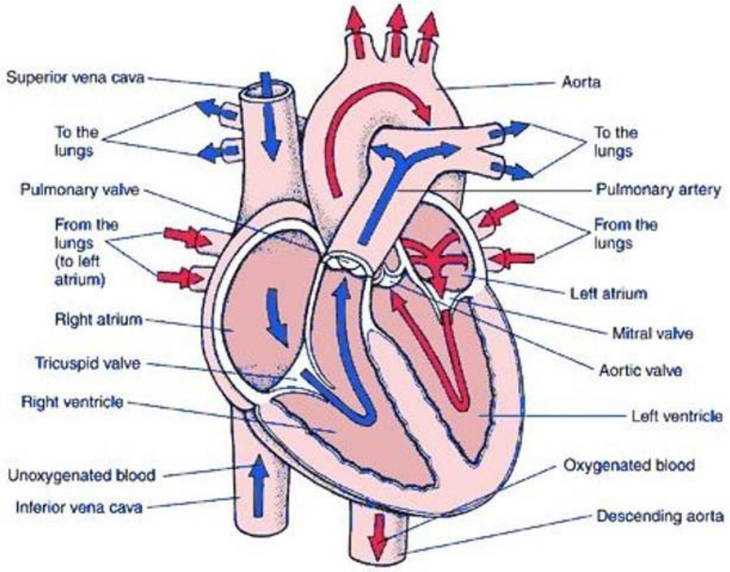
Auditing, Understanding, and Leveraging

Auditing

Given an LLM, what can we infer about how it was created?

Understanding

Can we understand how LLMs internally make predictions?



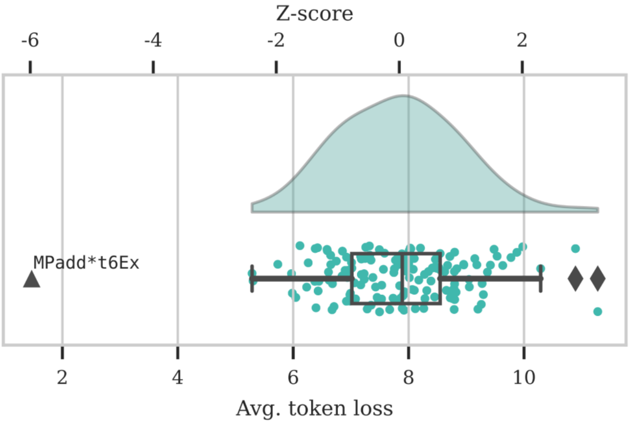
Leveraging

How can we overcome LLMs' deficiencies to solve complex tasks?

Overview

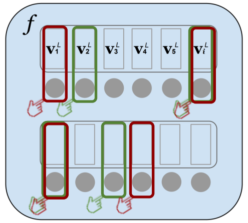
Auditing

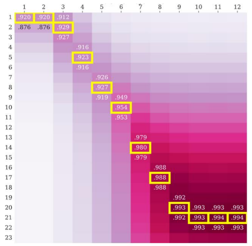
How can we infer what was in the **training data** of an LLM?

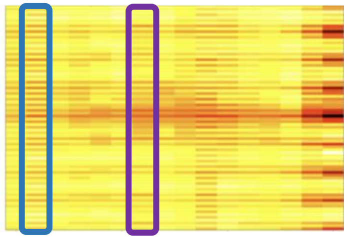


Understanding

How do LLMs **memorize**, **do math**, and **learn from demonstrations**?

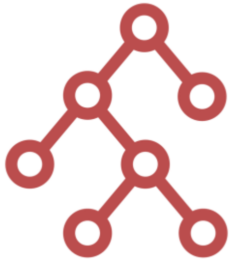






Leveraging

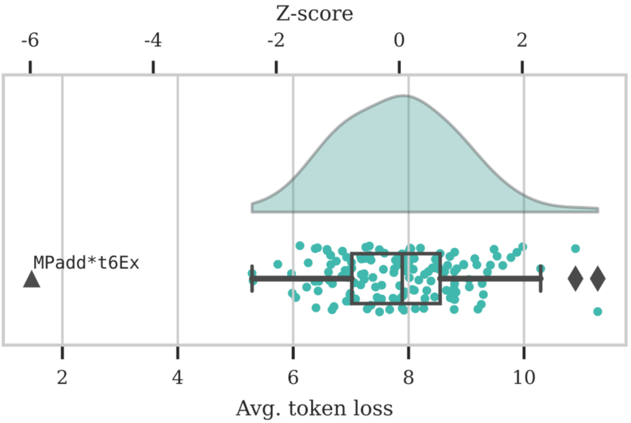
How can we combine LLMs with symbolic solvers for **planning tasks**?



Overview

Auditing

How can we infer what was in the **training data** of an LLM?



Proving Membership in LLM Pretraining Data via Data Watermarks

ACL Findings 2024

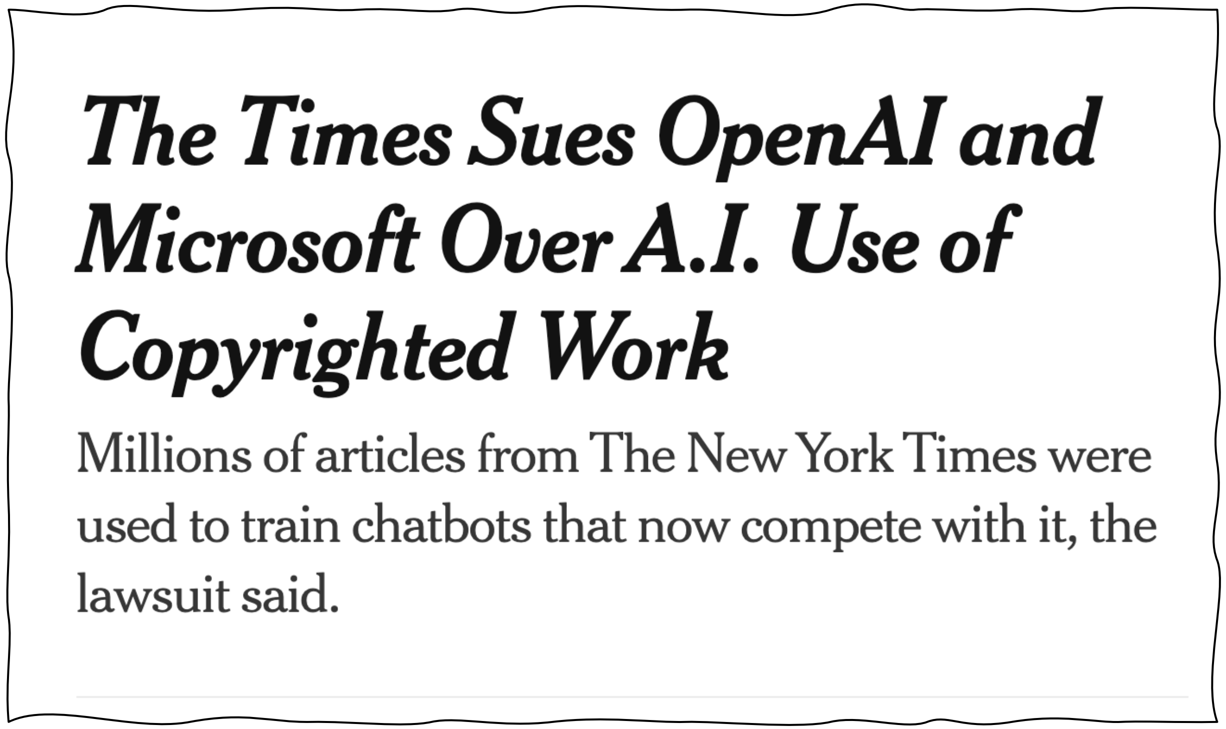
Johnny Wei*

Ryan Wang*

Legal Landscape for Training Data

The New York Times

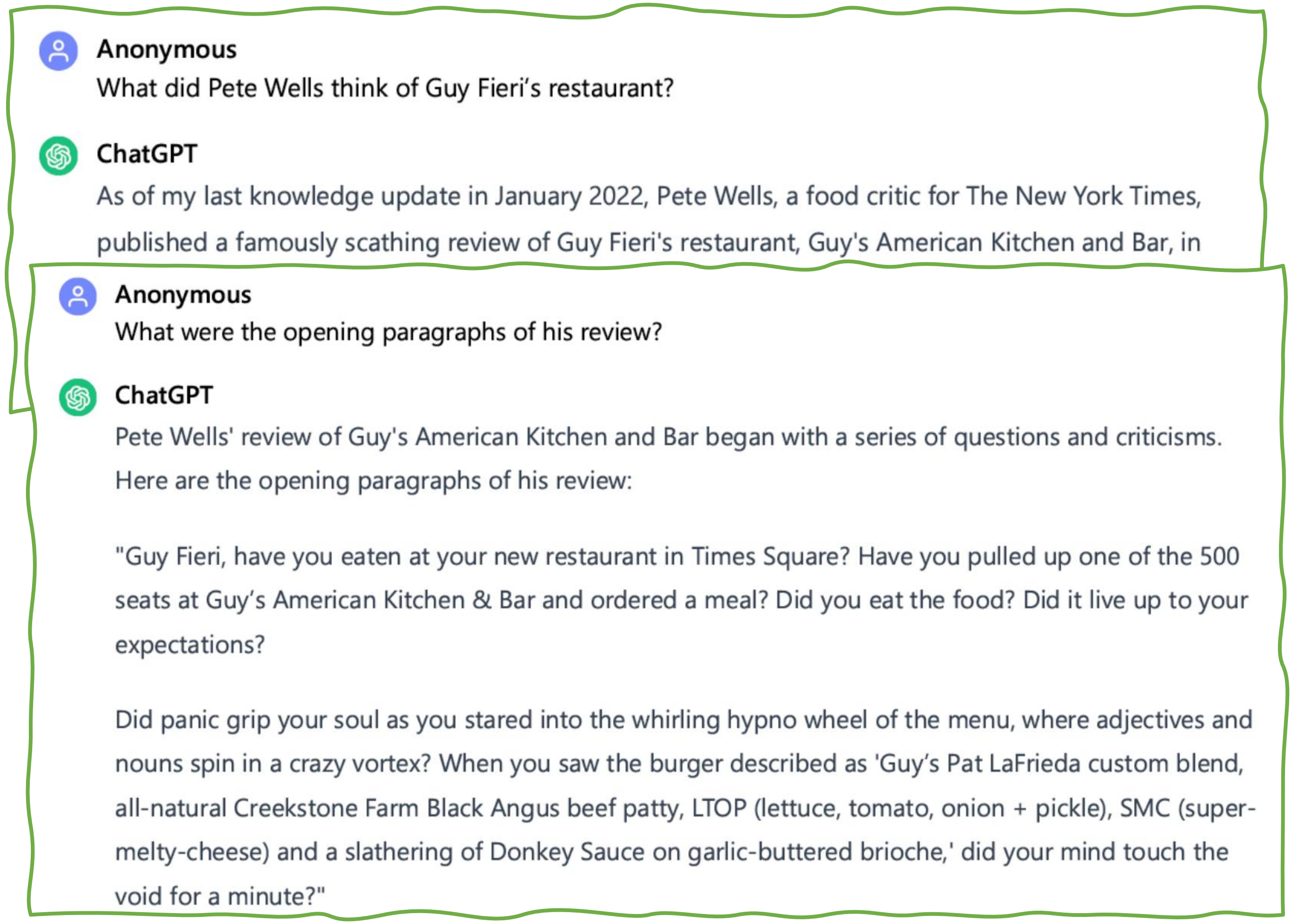
OpenAI



The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work

Millions of articles from The New York Times were used to train chatbots that now compete with it, the lawsuit said.

Data Memorization as Evidence



Anonymous
What did Pete Wells think of Guy Fieri’s restaurant?

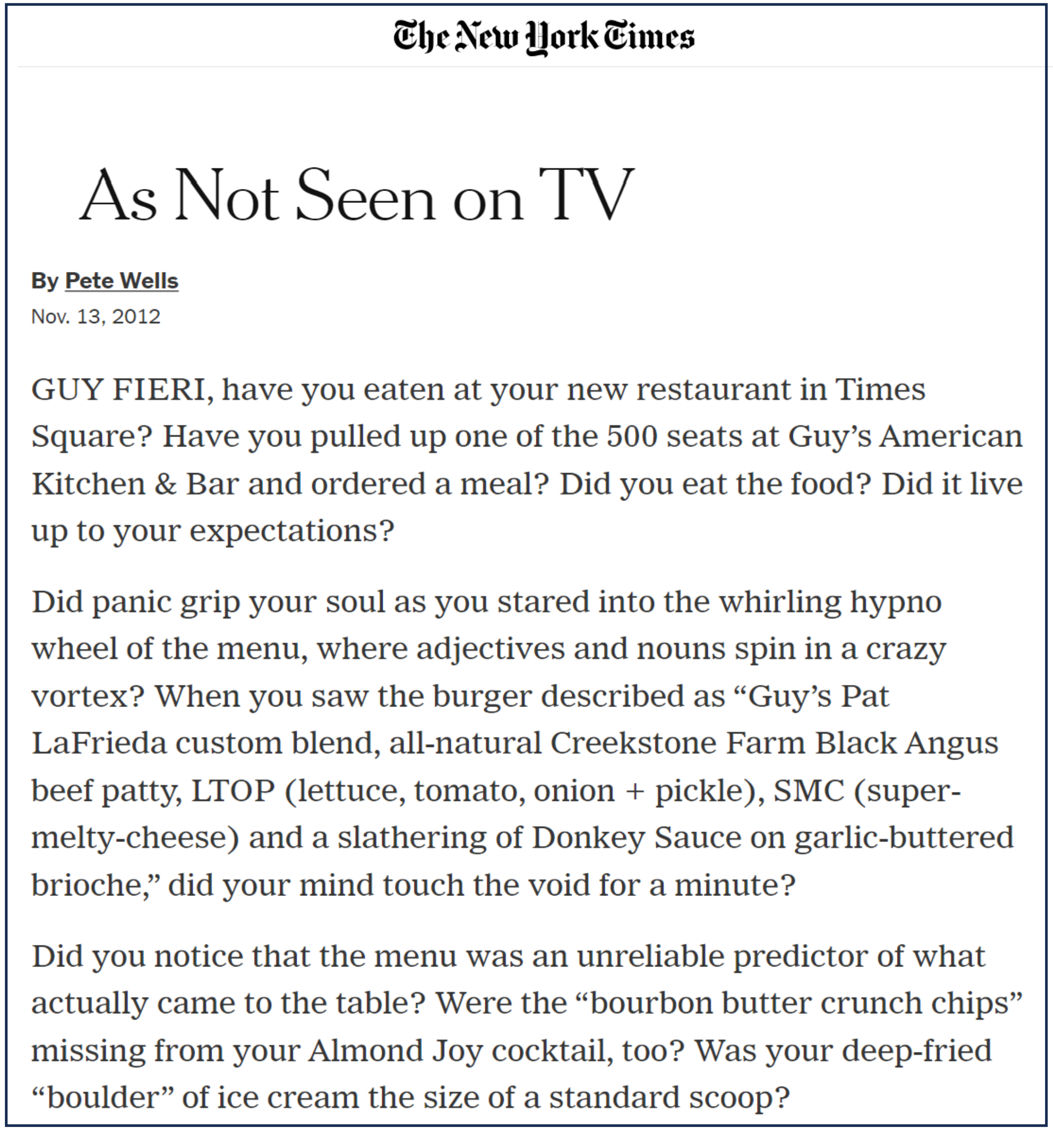
ChatGPT
As of my last knowledge update in January 2022, Pete Wells, a food critic for The New York Times, published a famously scathing review of Guy Fieri's restaurant, Guy's American Kitchen and Bar, in

Anonymous
What were the opening paragraphs of his review?

ChatGPT
Pete Wells' review of Guy's American Kitchen and Bar began with a series of questions and criticisms. Here are the opening paragraphs of his review:

"Guy Fieri, have you eaten at your new restaurant in Times Square? Have you pulled up one of the 500 seats at Guy’s American Kitchen & Bar and ordered a meal? Did you eat the food? Did it live up to your expectations?

Did panic grip your soul as you stared into the whirling hypno wheel of the menu, where adjectives and nouns spin in a crazy vortex? When you saw the burger described as 'Guy’s Pat LaFrieda custom blend, all-natural Creekstone Farm Black Angus beef patty, LTOP (lettuce, tomato, onion + pickle), SMC (super-melty-cheese) and a slathering of Donkey Sauce on garlic-buttered brioche,' did your mind touch the void for a minute?"



The New York Times

As Not Seen on TV

By **Pete Wells**
Nov. 13, 2012

GUY FIERI, have you eaten at your new restaurant in Times Square? Have you pulled up one of the 500 seats at Guy’s American Kitchen & Bar and ordered a meal? Did you eat the food? Did it live up to your expectations?

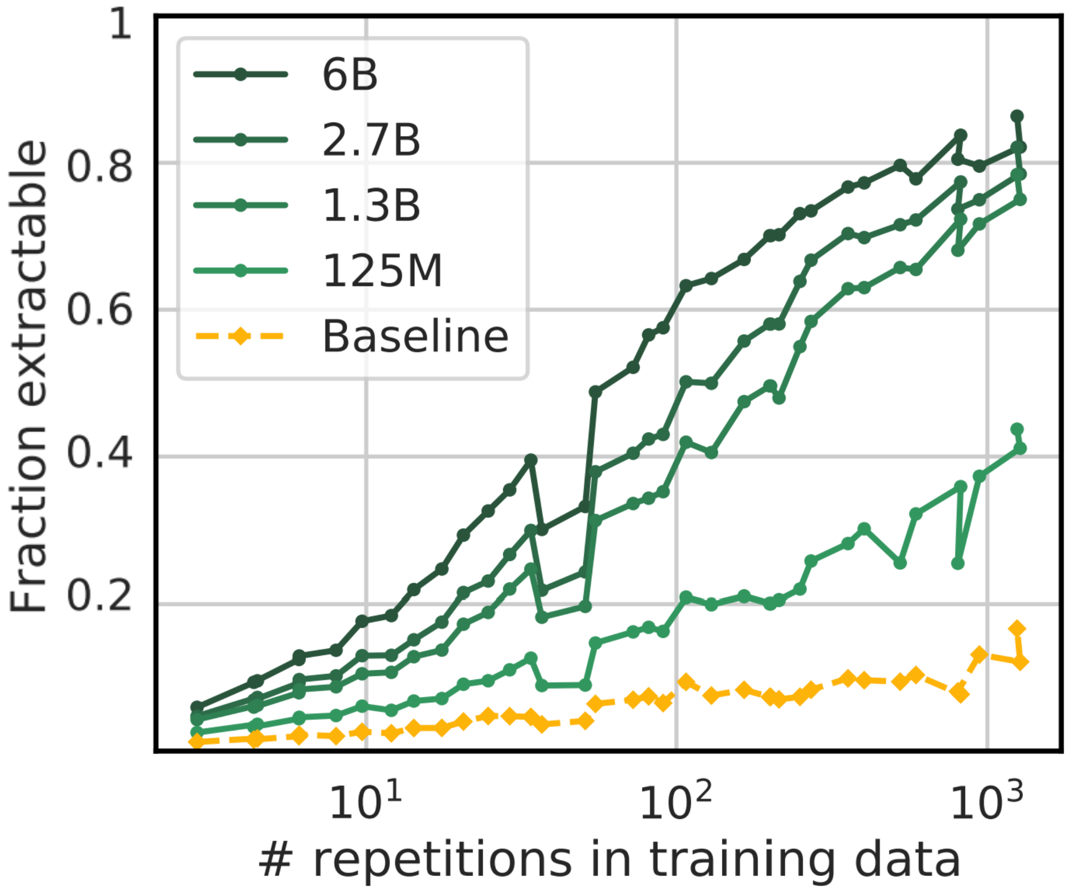
Did panic grip your soul as you stared into the whirling hypno wheel of the menu, where adjectives and nouns spin in a crazy vortex? When you saw the burger described as “Guy’s Pat LaFrieda custom blend, all-natural Creekstone Farm Black Angus beef patty, LTOP (lettuce, tomato, onion + pickle), SMC (super-melty-cheese) and a slathering of Donkey Sauce on garlic-buttered brioche,” did your mind touch the void for a minute?

Did you notice that the menu was an unreliable predictor of what actually came to the table? Were the “bourbon butter crunch chips” missing from your Almond Joy cocktail, too? Was your deep-fried “boulder” of ice cream the size of a standard scoop?

https://storage.courtlistener.com/recap/gov.uscourts.nysd.612697/gov.uscourts.nysd.612697.1.0.pdf

Data Memorization has Low Power

- Complete memorization of an article is a smoking gun, *if detected*
- LLM's unlikely to memorize full documents unless they are duplicated many times
  - Consider smaller publishers than NYT
- Goal: Detect usage of a data ***collection*** even if no individual article was memorized verbatim

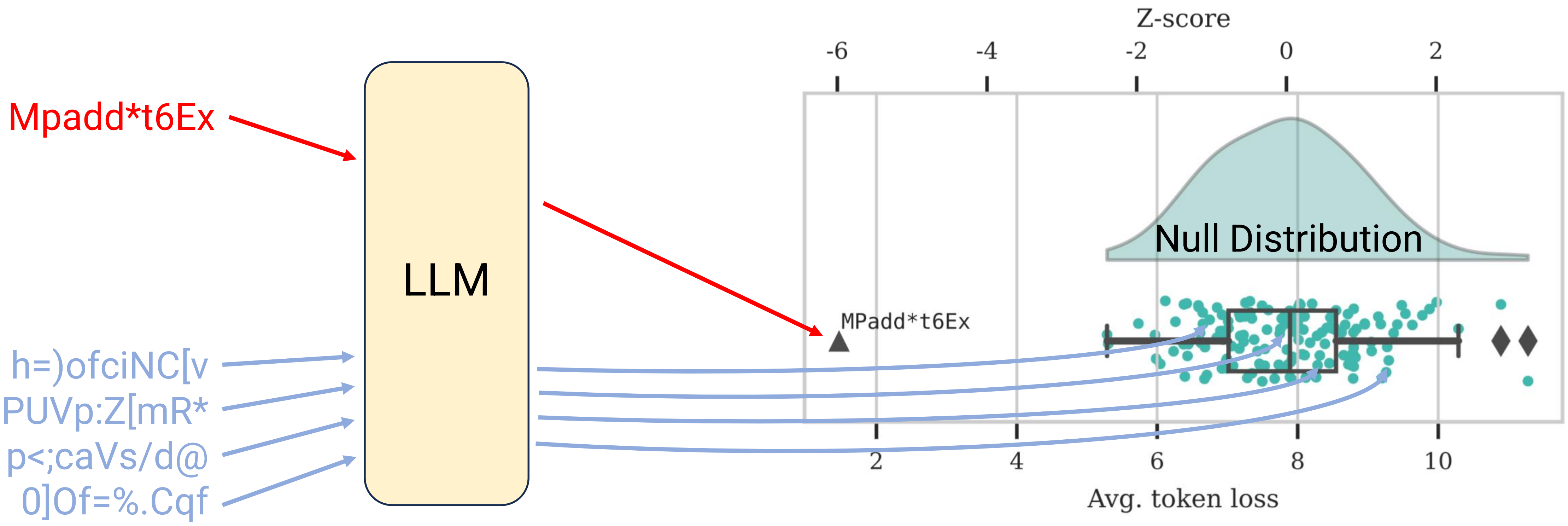


Carlini et al. *Quantifying Memorization Across Neural Language Models*. ICLR 2023.

Data Watermarks

- Idea: Add a consistent watermark to every document in the collection
  - E.g., Append “Mpadd*t6Ex” to every document (can hide in webpage with CSS)
- Check whether LLM memorizes the *watermark* (not any individual document)
  - **Watermark appears many times in corpus**: More likely to be memorized
  - **Watermark drawn from random distribution**: Enables rigorous statistical tests

Proving Data Usage with Statistical Tests

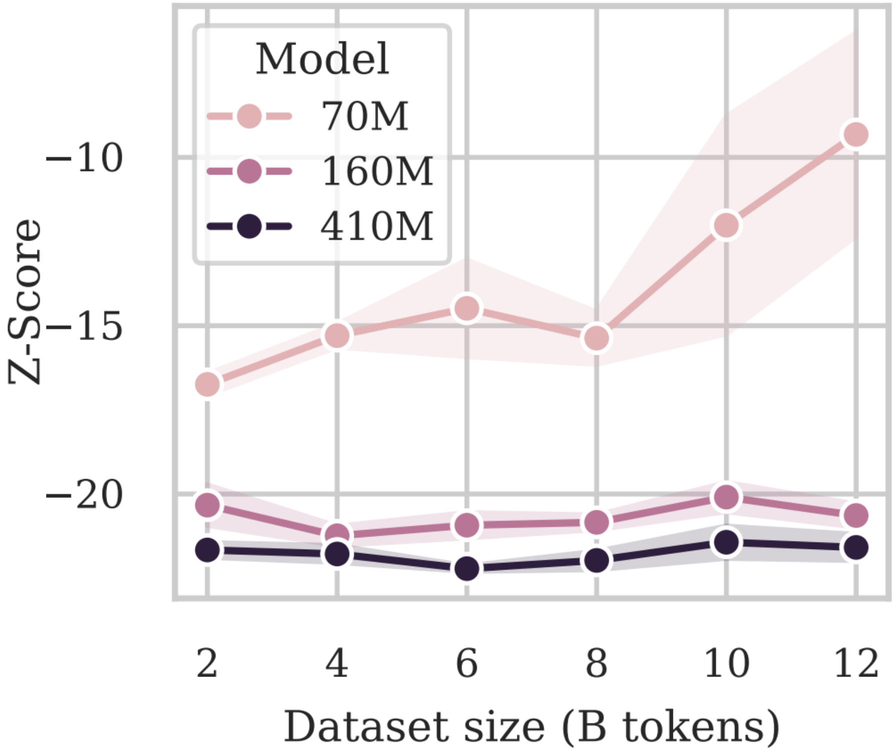


Mpadd*t6Ex

LLM

h=)ofciNC[v
PUVp:Z[mR*
p<;caVs/d@
0]Of=%.Cqf

Do Watermarks Work?

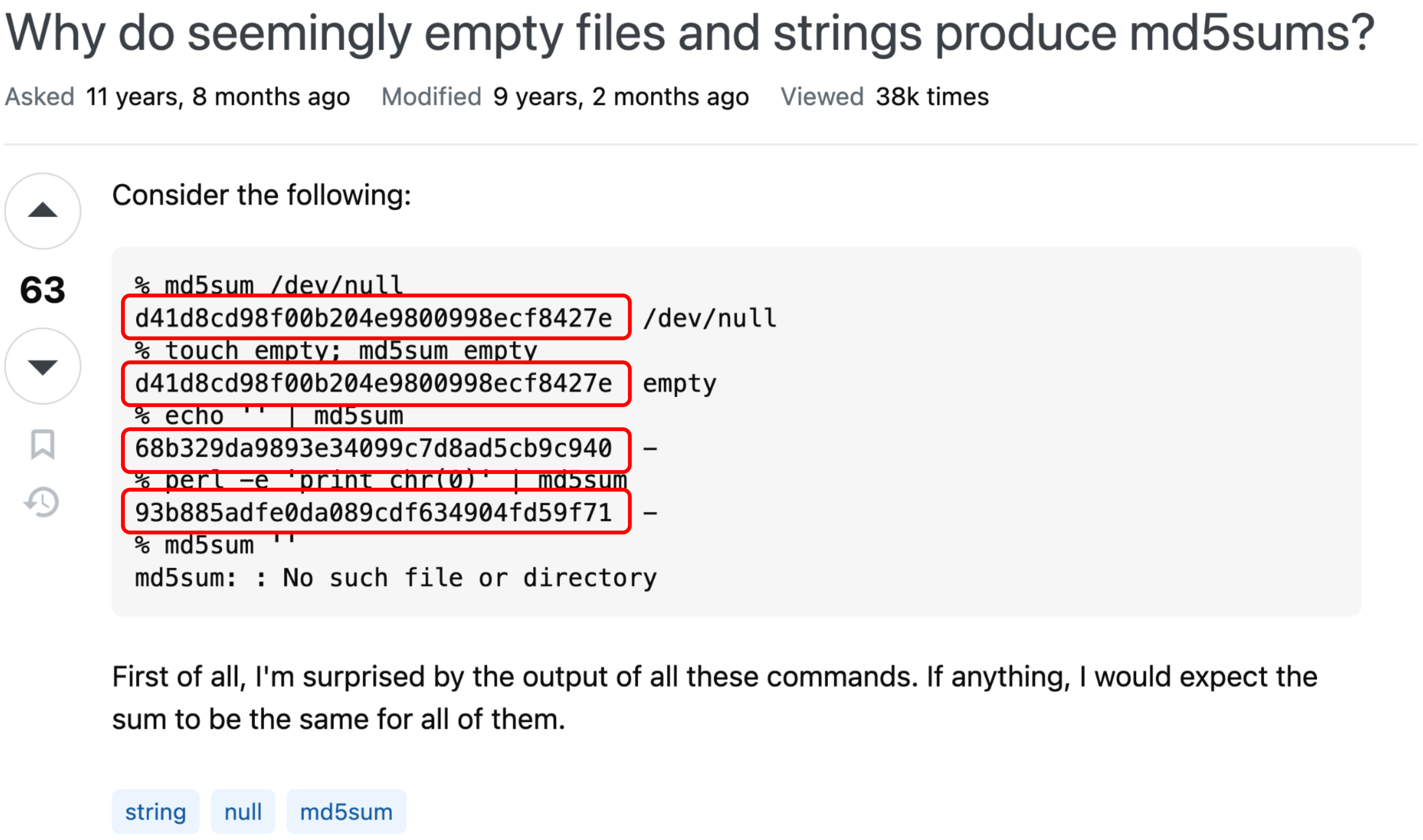


Setting: 80 character watermark inserted into 256 documents

- Yes!
- We trained medium-scale language models on data with watermarks
- Can detect watermark from model behavior alone—proof that it was seen during training!

What about *Even Larger* LMs?

- Can’t train very large LMs ourselves
- Idea: Use SHA/MD5 hashes as proxy for watermarks



Why do seemingly empty files and strings produce md5sums?

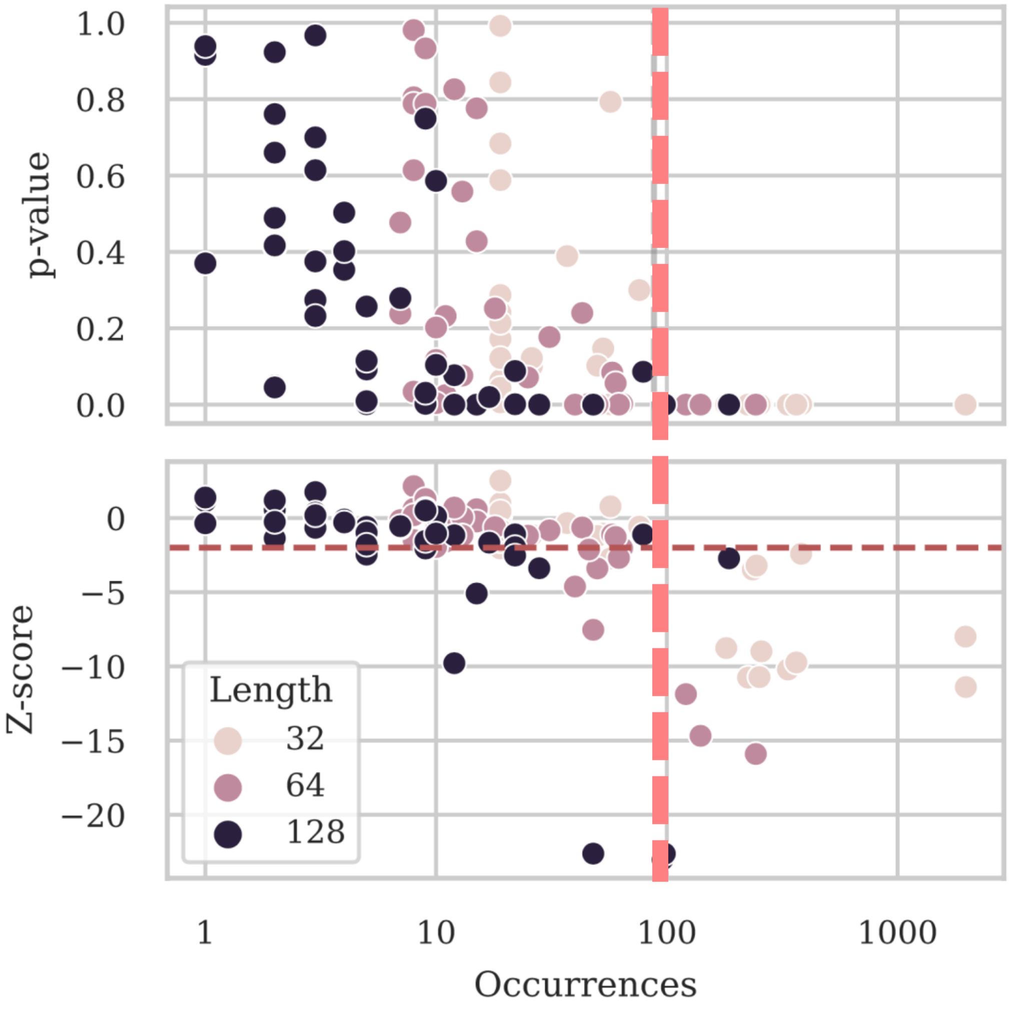
Asked 11 years, 8 months ago   Modified 9 years, 2 months ago   Viewed 38k times

Consider the following:

```
% md5sum /dev/null
d41d8cd98f00b204e9800998ecf8427e  /dev/null
% touch empty; md5sum empty
d41d8cd98f00b204e9800998ecf8427e  empty
% echo '' | md5sum
68b329da9893e34099c7d8ad5cb9c940  -
% perl -e 'print chr(0)' | md5sum
93b885adfe0da089cdf634904fd59f71  -
% md5sum ''
md5sum: : No such file or directory
```

First of all, I'm surprised by the output of all these commands. If anything, I would expect the sum to be the same for all of them.

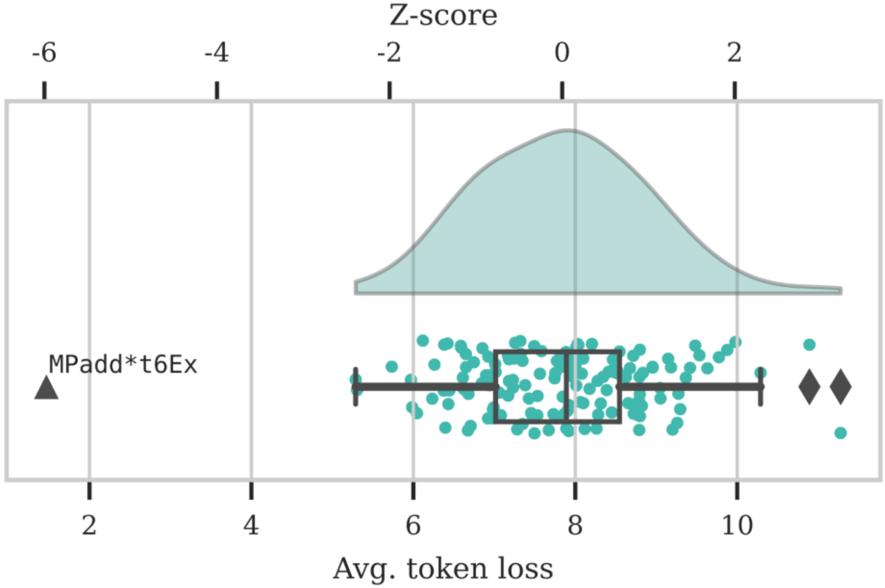
Larger LM’s Memorize Common Hashes



- Model: BLOOM-176B
  - Dataset is publicly available
- All hashes repeated at least 100 times are detectable
- Longer hashes easier to detect

Conclusion: Data Watermarks

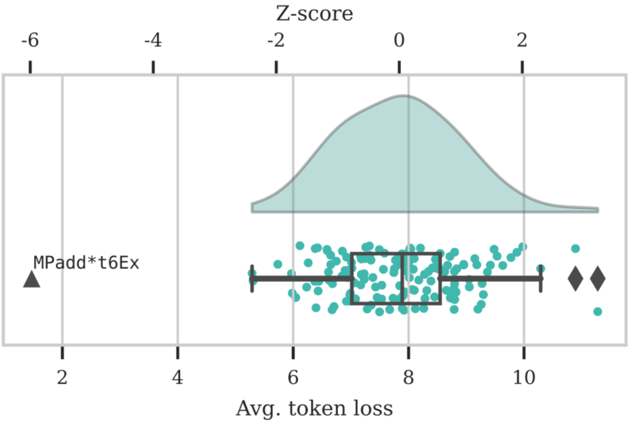
- Data watermarks:
  - Statistically rigorous
  - Scale to very large models
- Other results (see paper)
  - Stealthier watermarks using imperceptible Unicode substitutions
  - Effects of “interference” (multiple independent watermarkers)



Overview

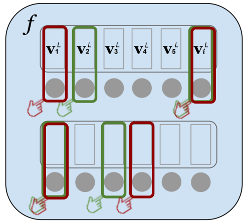
Auditing

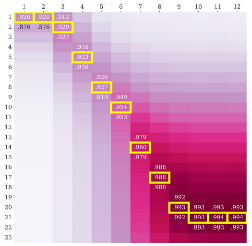
How can we infer what was in the **training data** of an LLM?

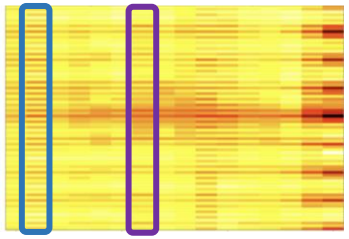


Understanding

How do LLMs **memorize, do math, and learn from demonstrations**?

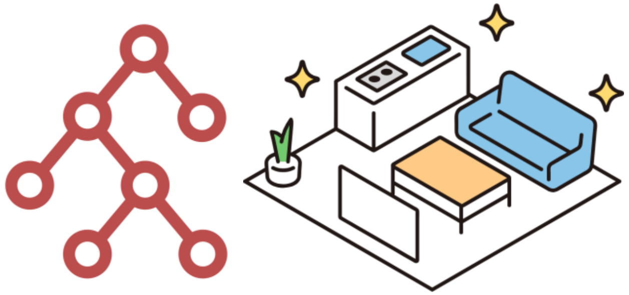






Leveraging

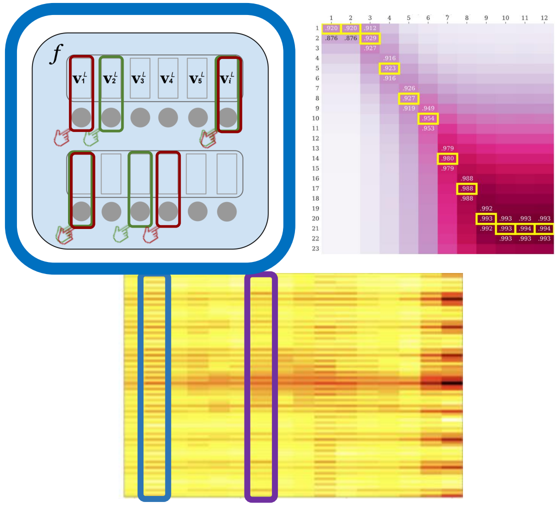
How can we combine LLMs with symbolic solvers for **planning tasks**?



Overview

Understanding

How do LLMs **memorize**, do math, and learn from demonstrations?



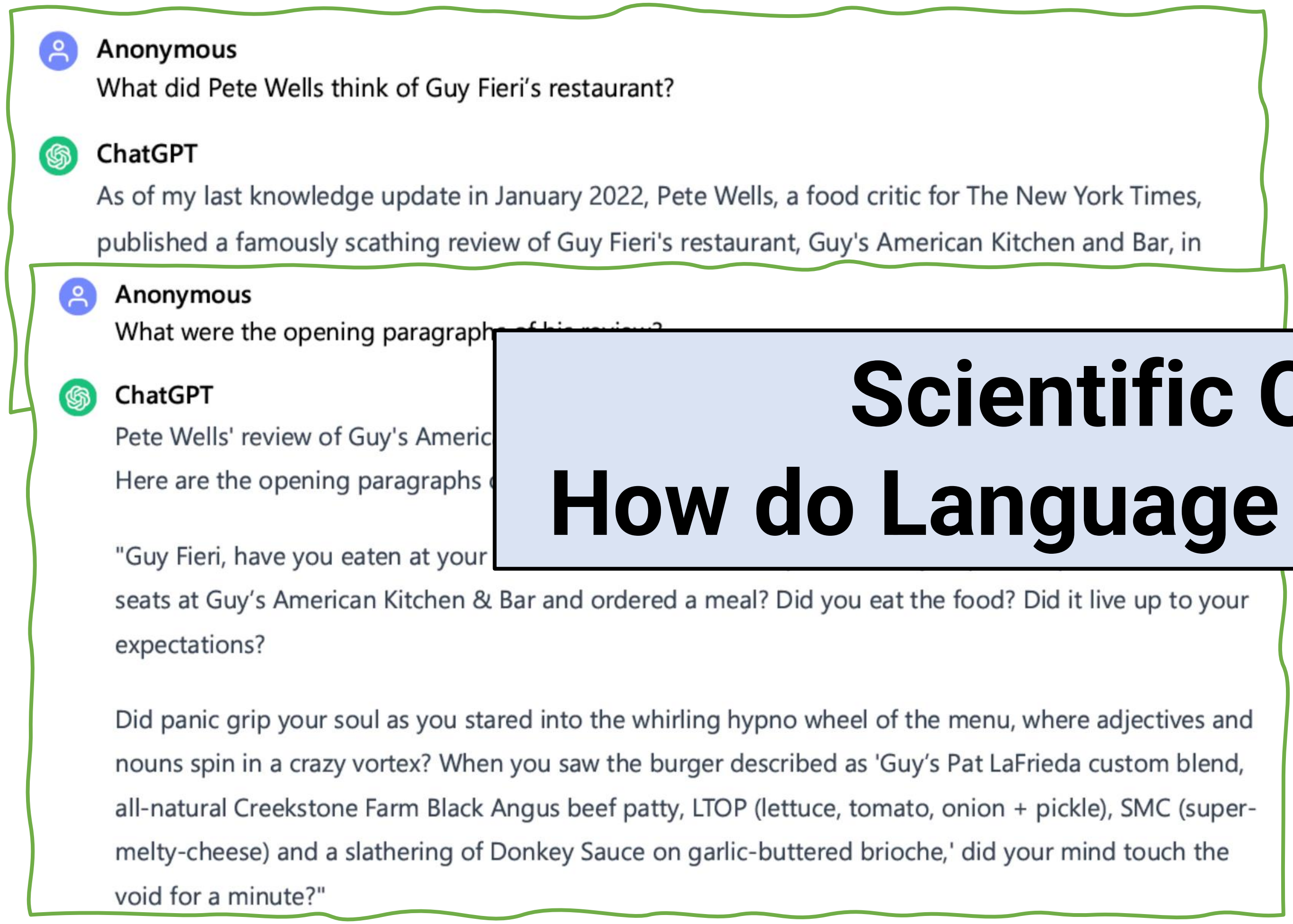
Do Localization Methods Actually Localize Memorized Data in LLMs?

NAACL 2024

Ting-Yun Chang

Jesse Thomason

Rote Memorization



Anonymous

What did Pete Wells think of Guy Fieri's restaurant?

ChatGPT

As of my last knowledge update in January 2022, Pete Wells, a food critic for The New York Times, published a famously scathing review of Guy Fieri's restaurant, Guy's American Kitchen and Bar, in

Anonymous

What were the opening paragraph

ChatGPT

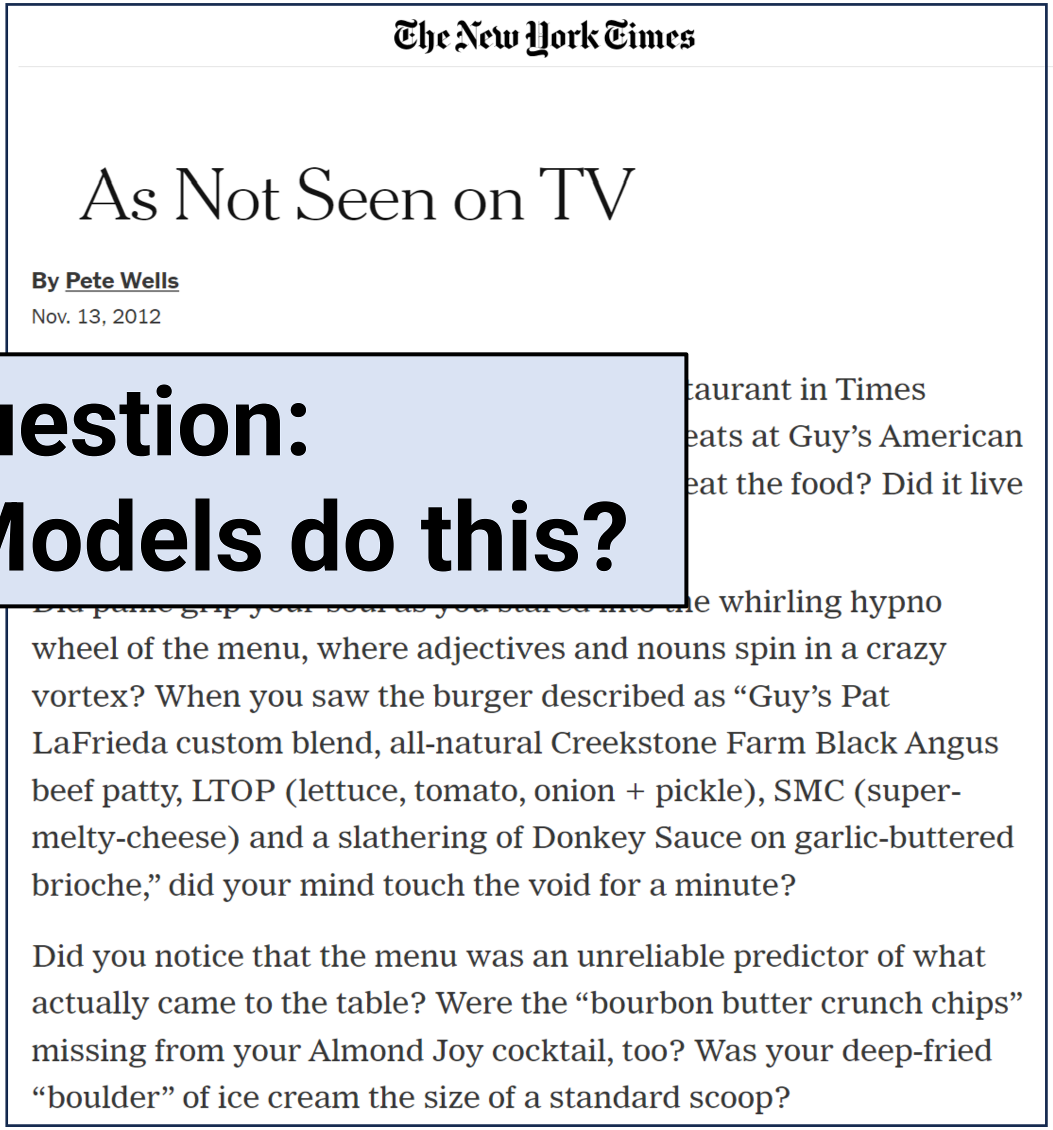
Pete Wells' review of Guy's Americ

Here are the opening paragraphs

"Guy Fieri, have you eaten at your

seats at Guy's American Kitchen & Bar and ordered a meal? Did you eat the food? Did it live up to your expectations?

Did panic grip your soul as you stared into the whirling hypno wheel of the menu, where adjectives and nouns spin in a crazy vortex? When you saw the burger described as 'Guy's Pat LaFrieda custom blend, all-natural Creekstone Farm Black Angus beef patty, LTOP (lettuce, tomato, onion + pickle), SMC (super-melty-cheese) and a slathering of Donkey Sauce on garlic-buttered brioche,' did your mind touch the void for a minute?"



The New York Times

As Not Seen on TV

By Pete Wells

Nov. 13, 2012

taurant in Times

eats at Guy's American

eat the food? Did it live

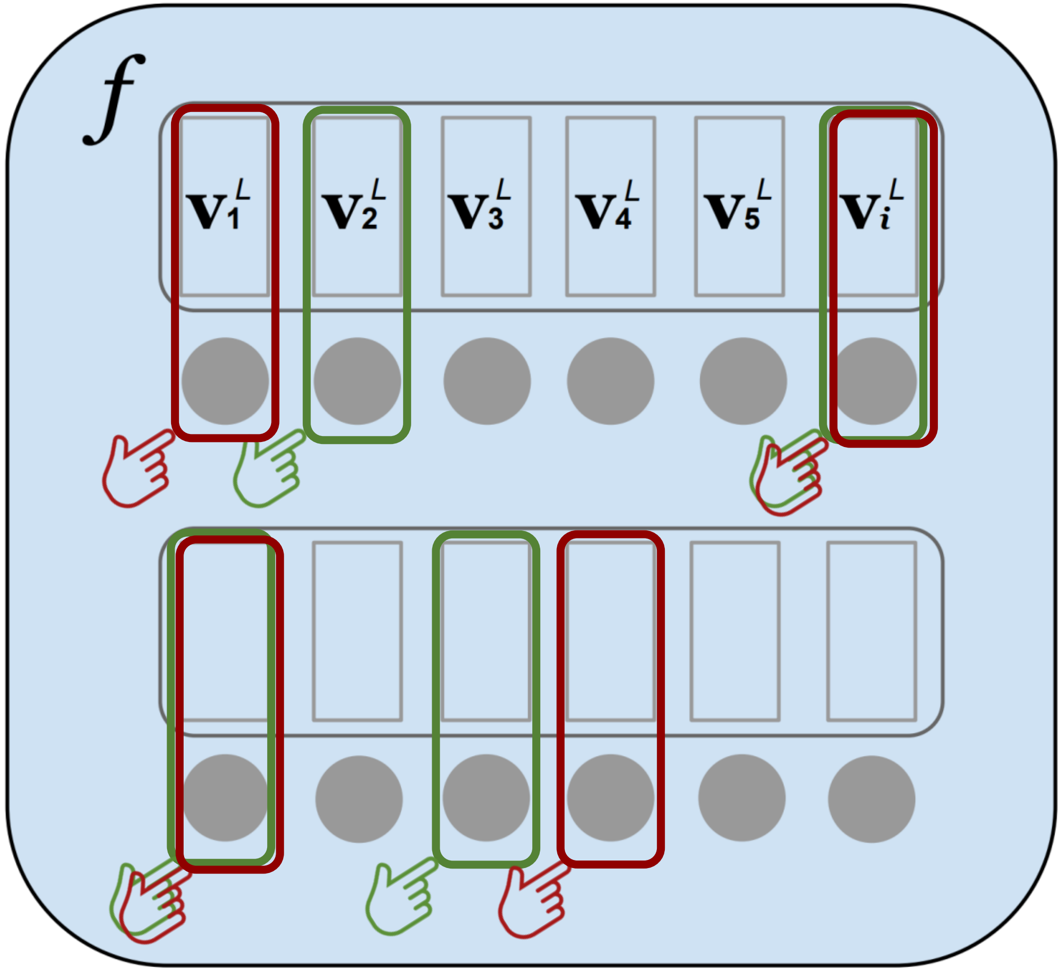
e whirling hypno wheel of the menu, where adjectives and nouns spin in a crazy vortex? When you saw the burger described as "Guy's Pat LaFrieda custom blend, all-natural Creekstone Farm Black Angus beef patty, LTOP (lettuce, tomato, onion + pickle), SMC (super-melty-cheese) and a slathering of Donkey Sauce on garlic-buttered brioche," did your mind touch the void for a minute?

Did you notice that the menu was an unreliable predictor of what actually came to the table? Were the "bourbon butter crunch chips" missing from your Almond Joy cocktail, too? Was your deep-fried "boulder" of ice cream the size of a standard scoop?

Scientific Question:
How do Language Models do this?

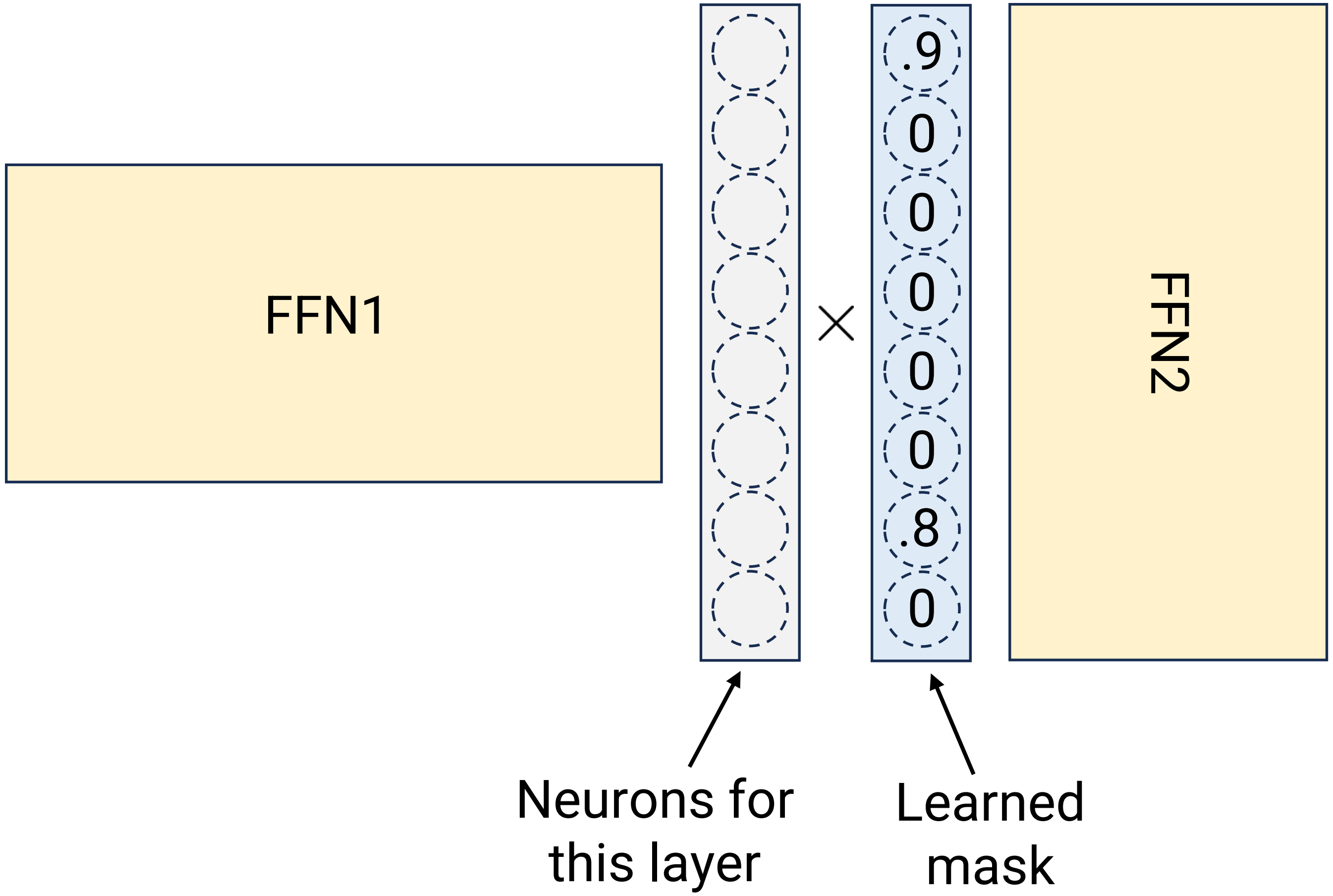
https://storage.courtlistener.com/recap/gov.uscourts.nysd.612697/gov.uscourts.nysd.612697.1.0.pdf

Evaluating Localization



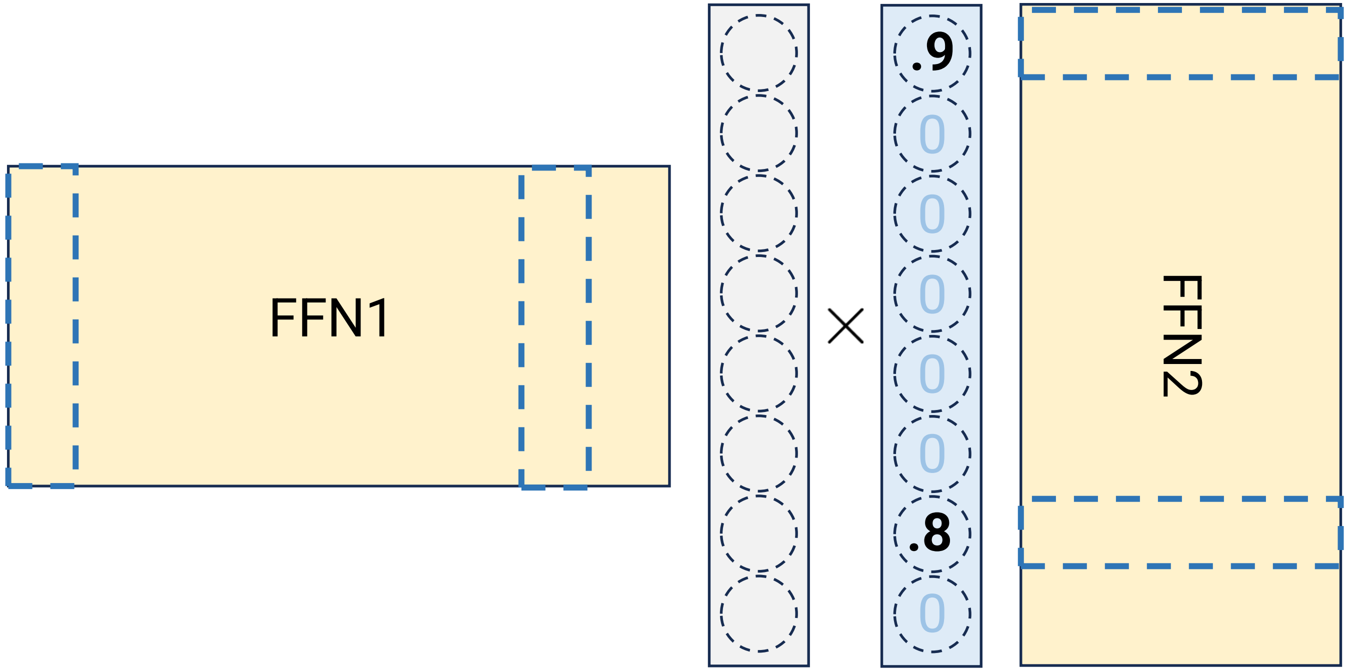
- ***Localization method***: Identifies (small) subset of model components most responsible for memorizing example *x*
  - Pruning, integrated gradients, etc.
- How to evaluate localization?
  - We don't know the ground truth (where information is stored)
  - We don't even know if knowledge is really stored in a localized way!
- **Do localization methods really localize memorized knowledge?**

Pruning-based Localization



- Learn a “mask” in [0, 1]
  - Loss #1: Retain memorization of 1 given document
  - Loss #2: Make mask as sparse as possible
- Parameters corresponding to mask’s non-zero entries must be important
- Two variants called Slimming & Hard Concrete

Pruning-based Localization



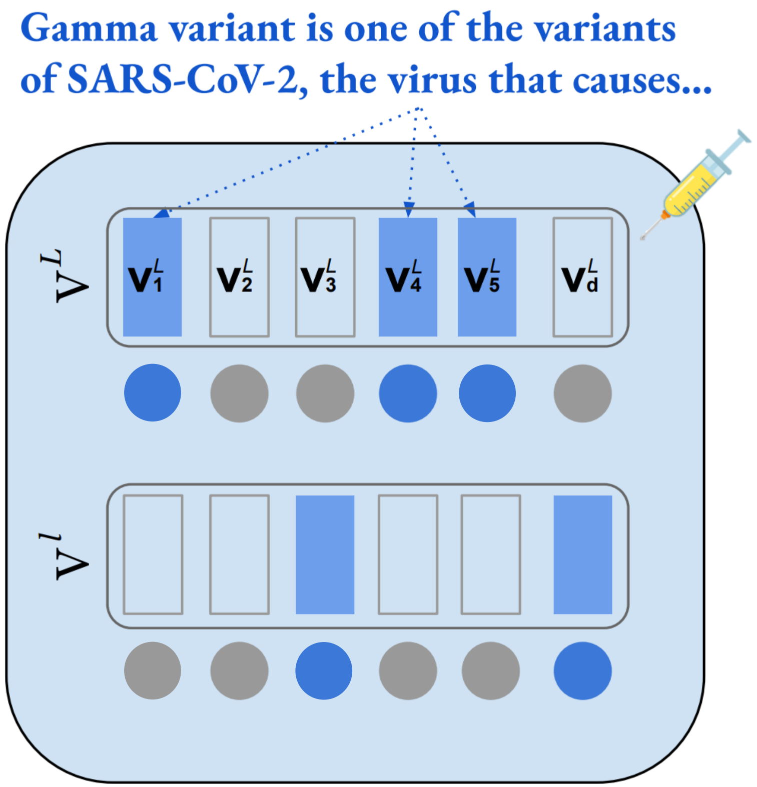
Neurons for this layer

Learned mask

- Learn a “mask” in [0, 1]
  - Loss #1: Retain memorization of 1 given document
  - Loss #2: Make mask as sparse as possible
- Parameters corresponding to mask’s non-zero entries must be important
- Two variants called Slimming & Hard Concrete

Benchmark #1: Injection-Based

- Idea: **Inject** new knowledge into model using only some parameters



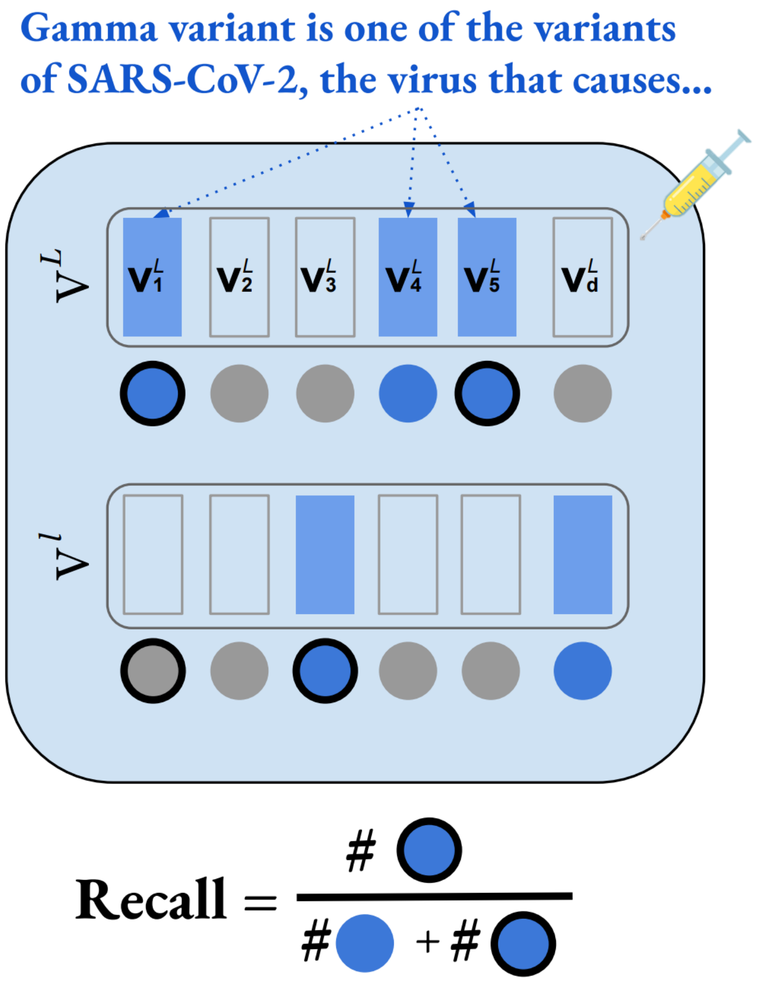
Gamma variant is one of the variants of SARS-CoV-2, the virus that causes...
V^L
v_1^L
v_2^L
v_3^L
v_4^L
v_5^L
v_d^L
V^l

1. Acquire new sentence not seen during pre-training

2. Fine-tune *small subset of model parameters* to memorize new sentence

Benchmark #1: Injection-Based

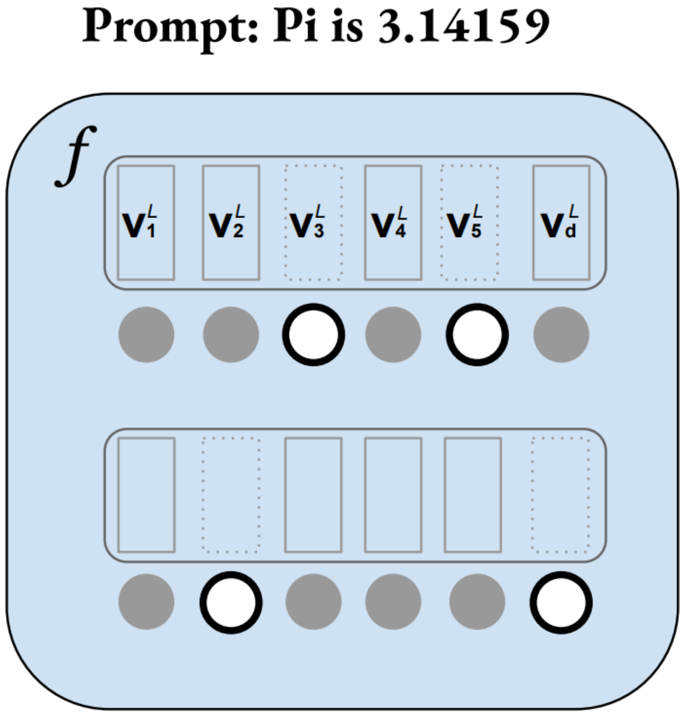
- Idea: **Inject** new knowledge into model using only some parameters
- Good: We have ground truth for evaluation
- Bad: Where we inject may not be where models "naturally" store knowledge


Gamma variant is one of the variants of SARS-CoV-2, the virus that causes...
V^L
v_1^L
v_2^L
v_3^L
v_4^L
v_5^L
v_d^L
V^l
Recall = # ● / # ● + # ●

1. Acquire new sentence not seen during pre-training
2. Fine-tune *small subset of model parameters* to memorize new sentence
3. Run **localization method** on resulting model
4. Evaluate recall of compared with ground truth parameters

Benchmark #2: Deletion-Based

- Idea: **Delete** existing knowledge from model by removing the most important parameters

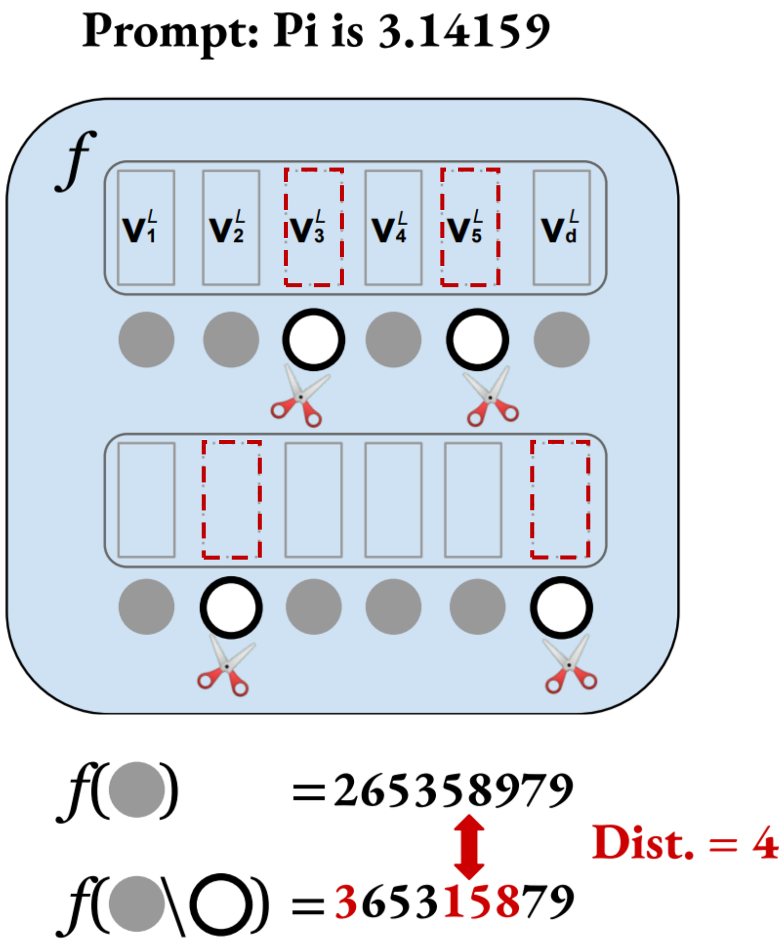


1. Find already-memorized sequence (quotes, ToS, etc.)

2. Run **localization method** on pre-trained model

Benchmark #2: Deletion-Based

- Idea: **Delete** existing knowledge from model by removing the most important parameters
- Good: Studies how LLM’s naturally memorize data
- Bad: No ground truth



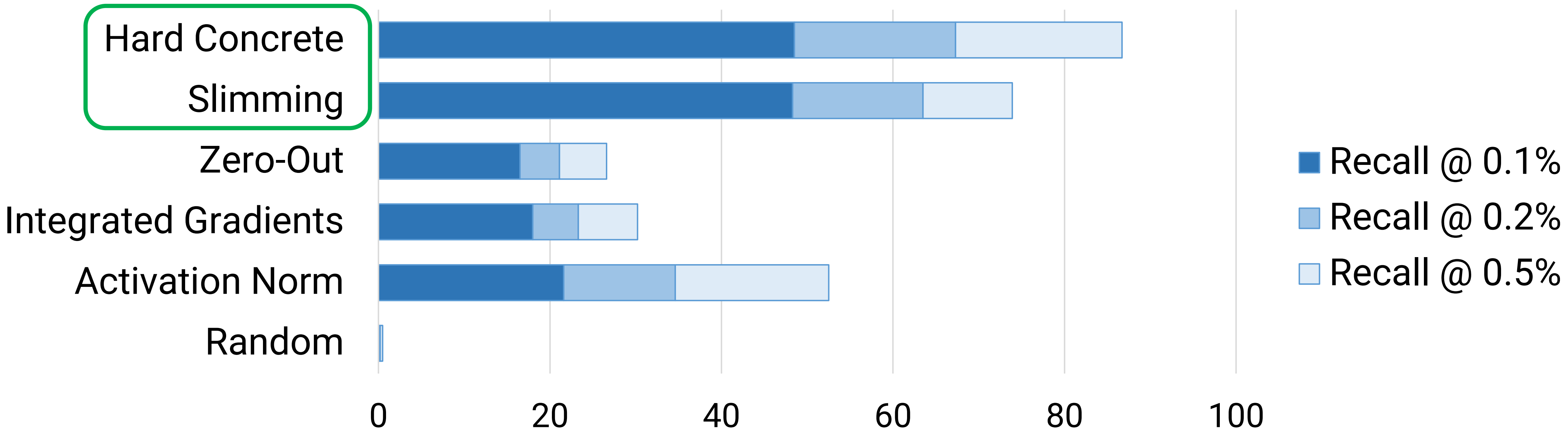
1. Find already-memorized sequence (quotes, ToS, etc.)

2. Run **localization method** on pre-trained model

3. *Delete* (zero-out) identified parameters

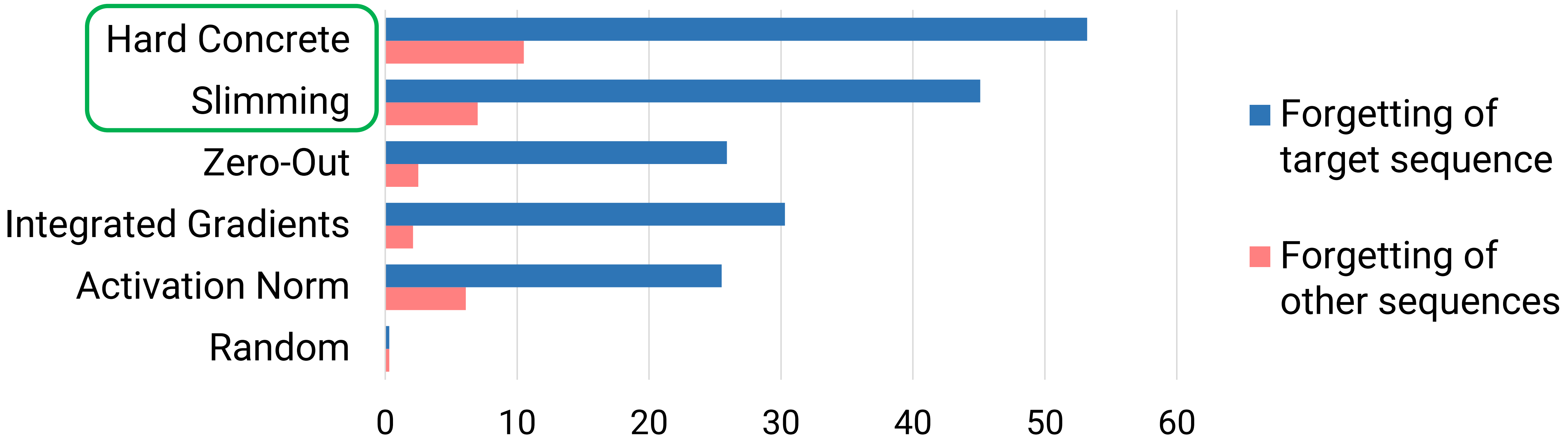
4. Evaluate how much LLM forgets the sequence

Injection Benchmark Results



- Setting: Pythia-Deduped 2.8B, 0.1% of parameters fine-tuned
- Tested 4 models x 2 fine-tuning ratios x 3 recall levels: **Hard Concrete or Slimming always best (both pruning-based)**
- All methods greatly outperform random baseline

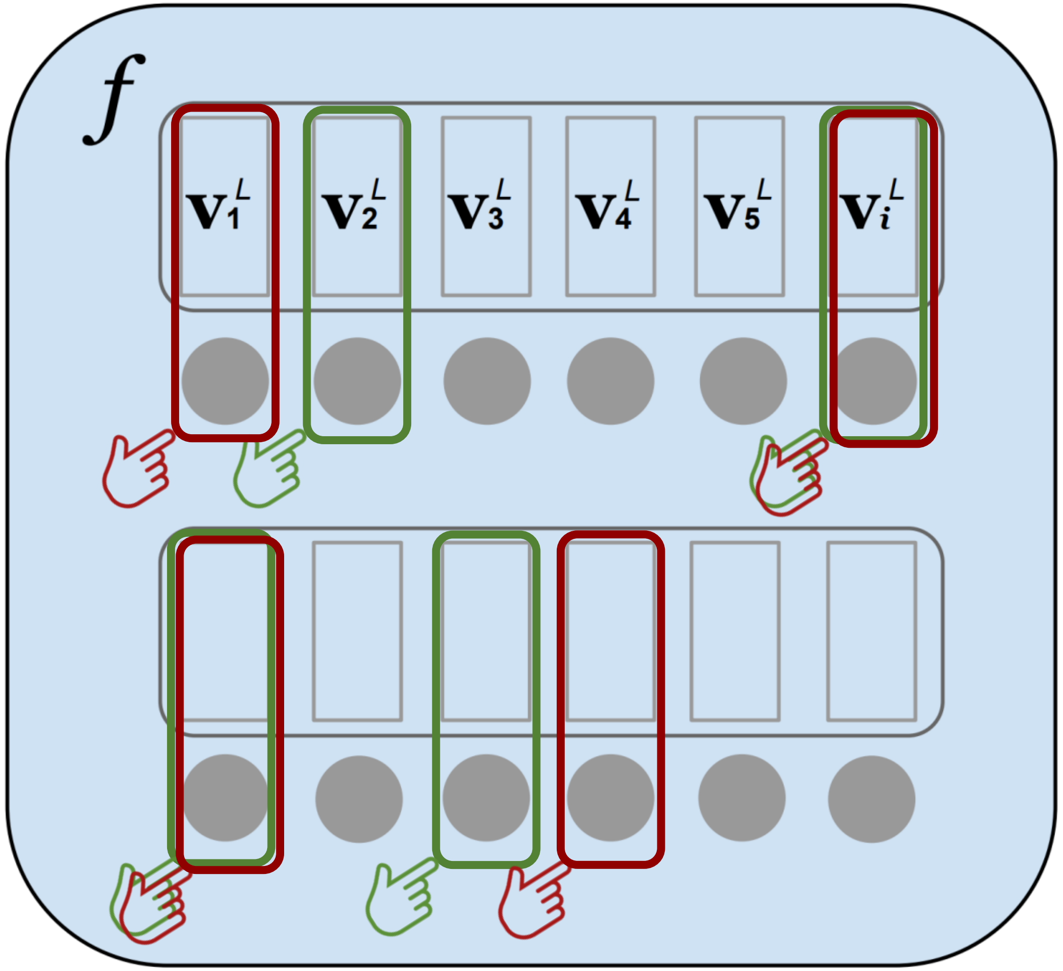
Deletion Benchmark Results



- Setting: Pythia-Deduped 2.8B, delete top 0.5% of parameters
- Tested 3 models x 2 removal ratios:
  Hard Concrete or Slimming always best (both pruning-based)
- All methods greatly outperform random baseline

Conclusion: Does Localization Work?

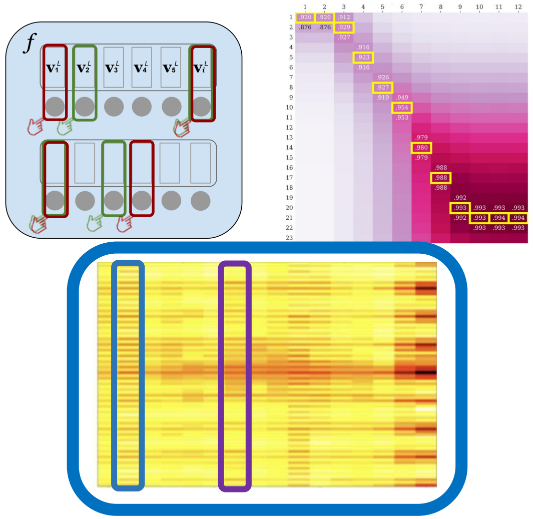
- Across two different localization benchmarks:
  - Pruning-based methods always best
  - All methods do something non-trivial
- Knowledge does appear to be stored in a localizable way
- Other results (see paper)
  - Knowledge is distributed across layers, **not** localized to specific layers
  - Early layers perform generally useful function, not specialized to single example



Overview

<u>Understanding</u>

How do LLMs memorize, **do math**, and learn from demonstrations?



Pre-trained Large Language Models Use Fourier Features to Compute Addition

NeurIPS 2024

Tianyi
Zhou

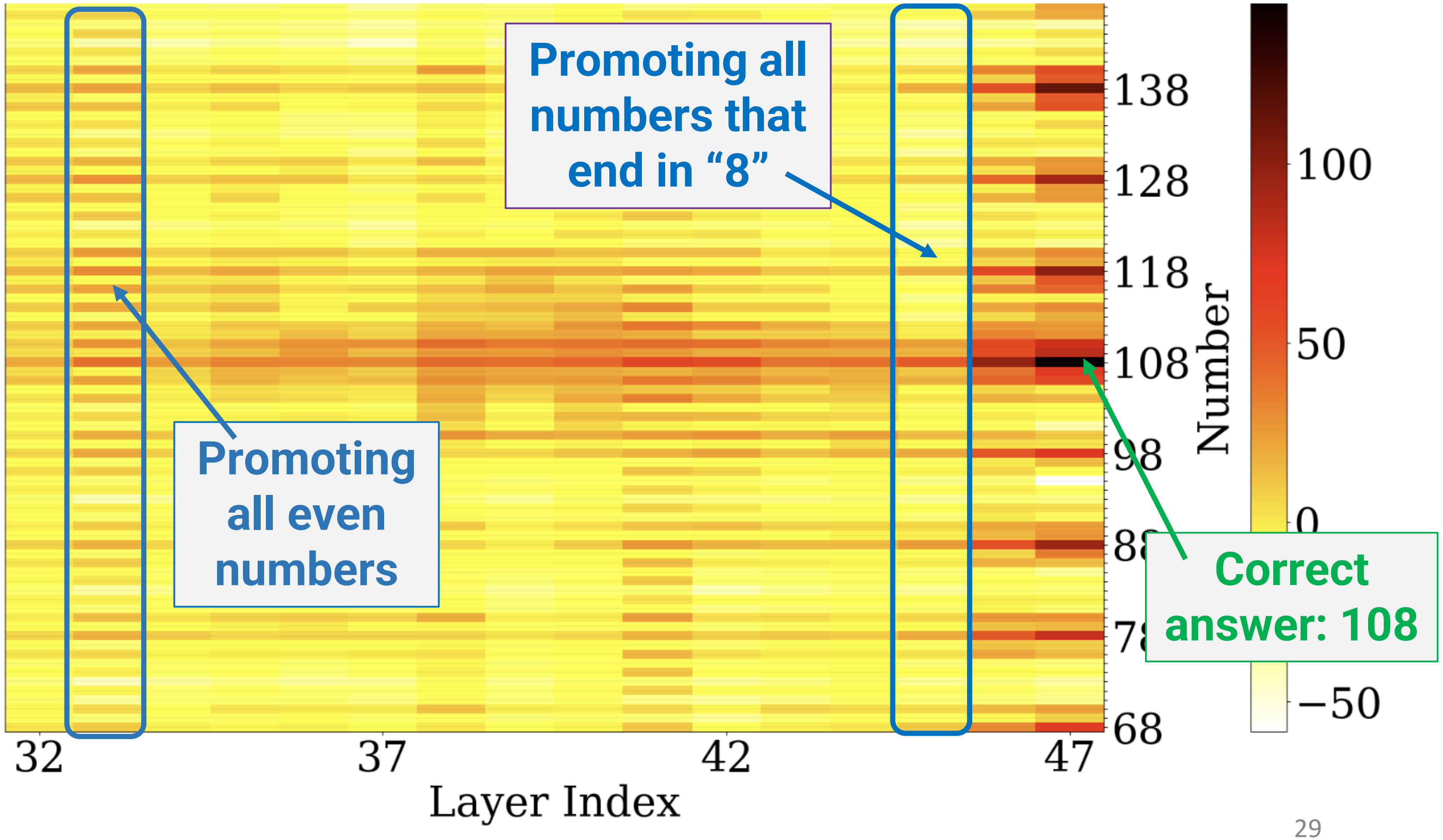
Deqing
Fu

Vatsal
Sharan

How do Language Models do Arithmetic?

- Setting: Fine-tune GPT-2-XL for integer addition
    - All numbers between 1 and 520, so they are single tokens
    - Accuracy ≈100%
- Model doesn't memorize, it computes an answer!
- Some layers perform modular arithmetic

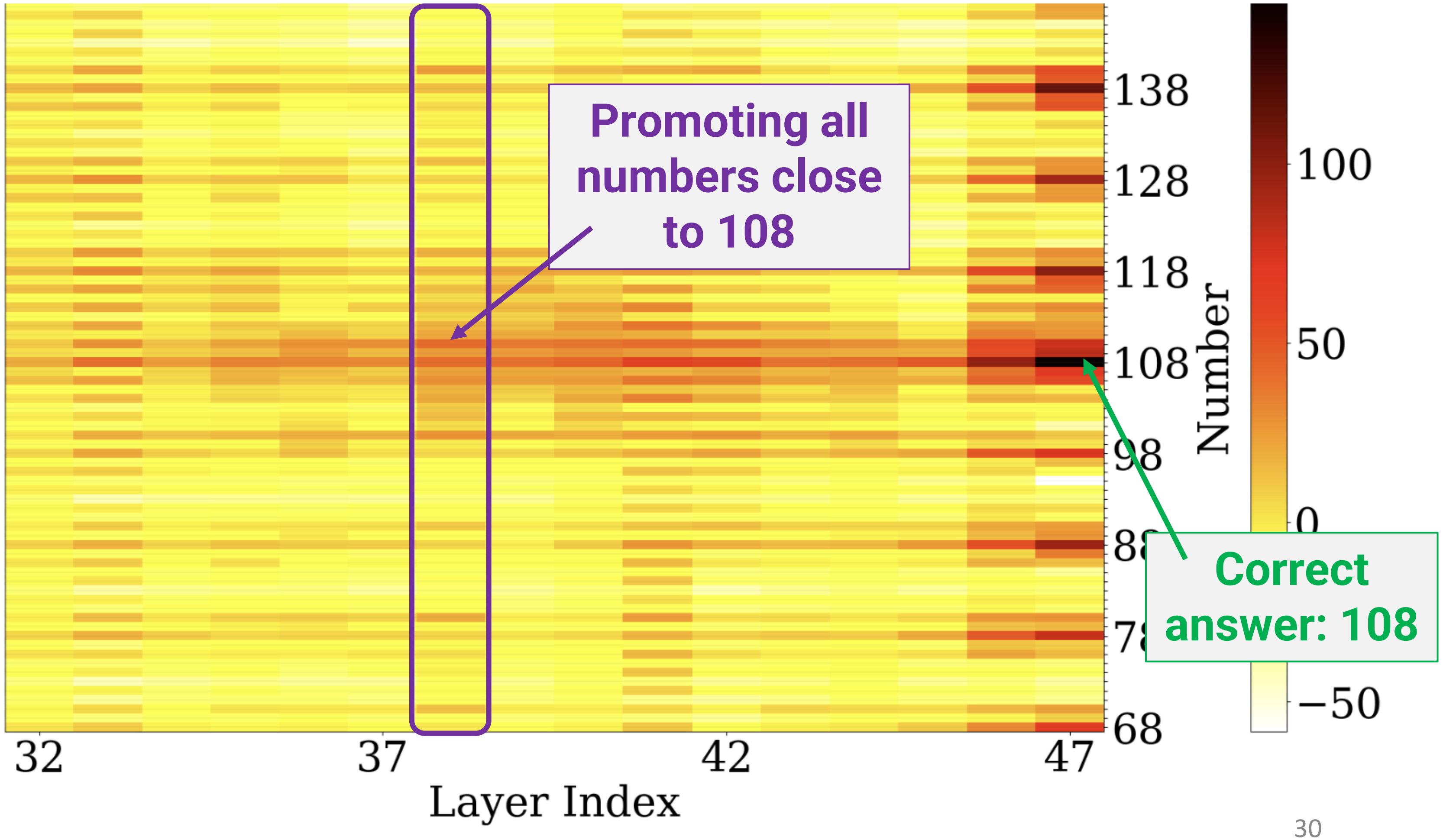
Input: Put together 15 and 93.
Plot how each MLP layer contributes to prediction



How do Language Models do Arithmetic?

- Setting: Fine-tune GPT-2-XL for integer addition
  - All numbers between 1 and 520, so they are single tokens
  - Accuracy ≈100%
- Model doesn’t memorize, it computes an answer!
- Some layers perform modular arithmetic
- Others approximate the answer

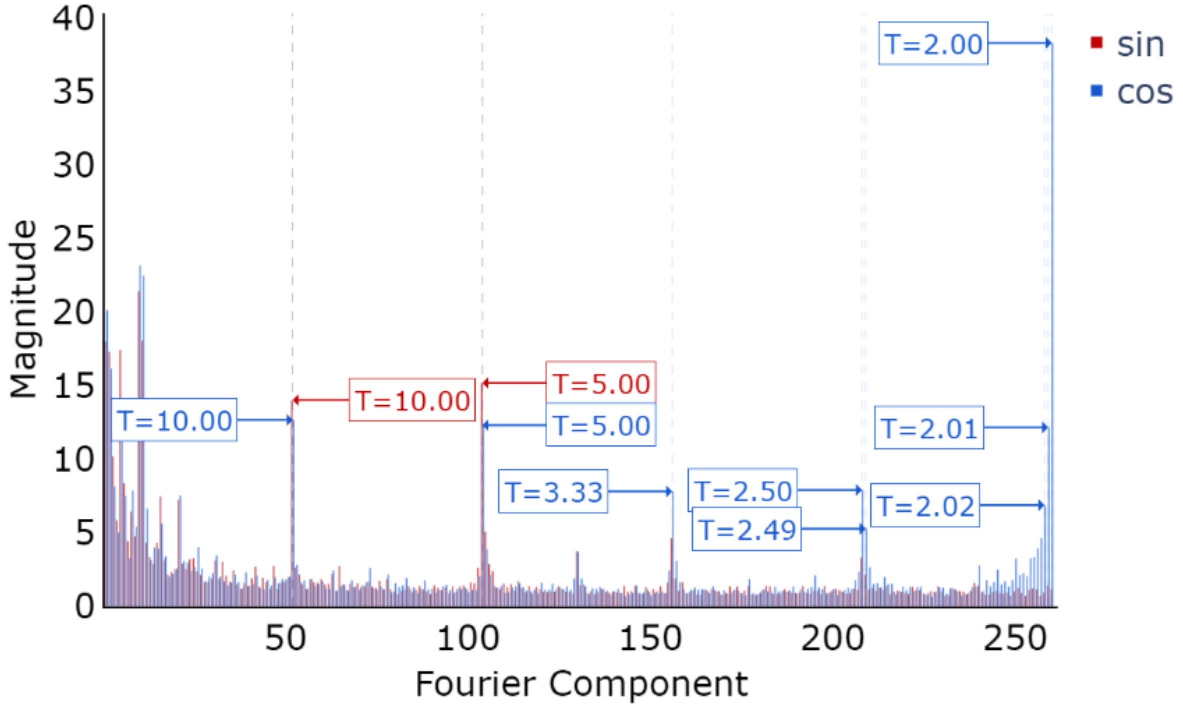
Input: Put together 15 and 93.
Plot how each MLP layer contributes to prediction

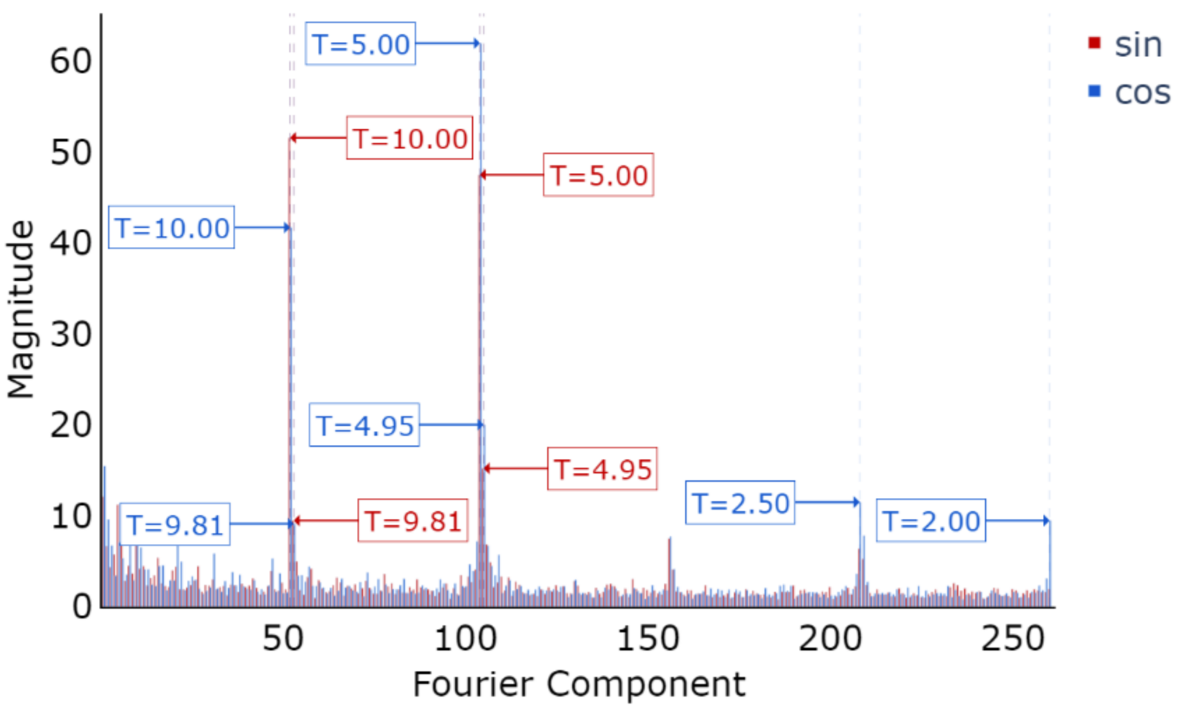


Promoting all numbers close to 108

Correct answer: 108

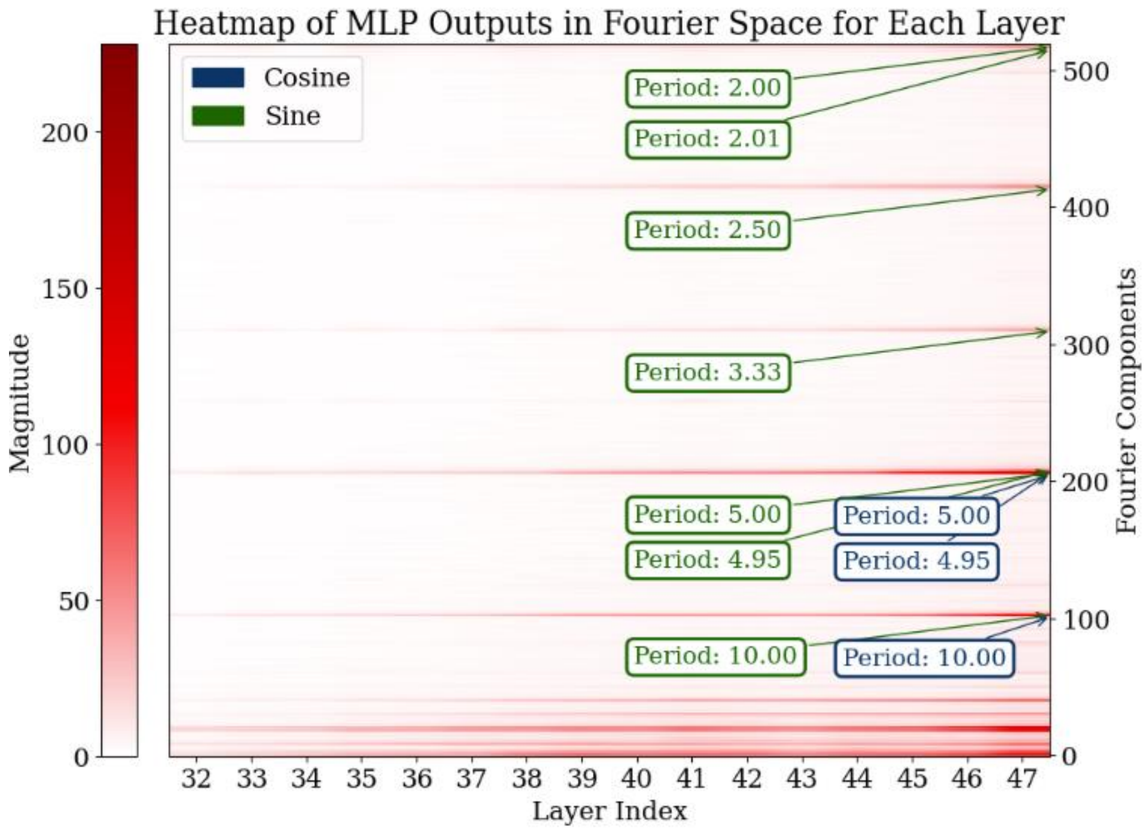
Layer-wise logits are sparse in Fourier domain

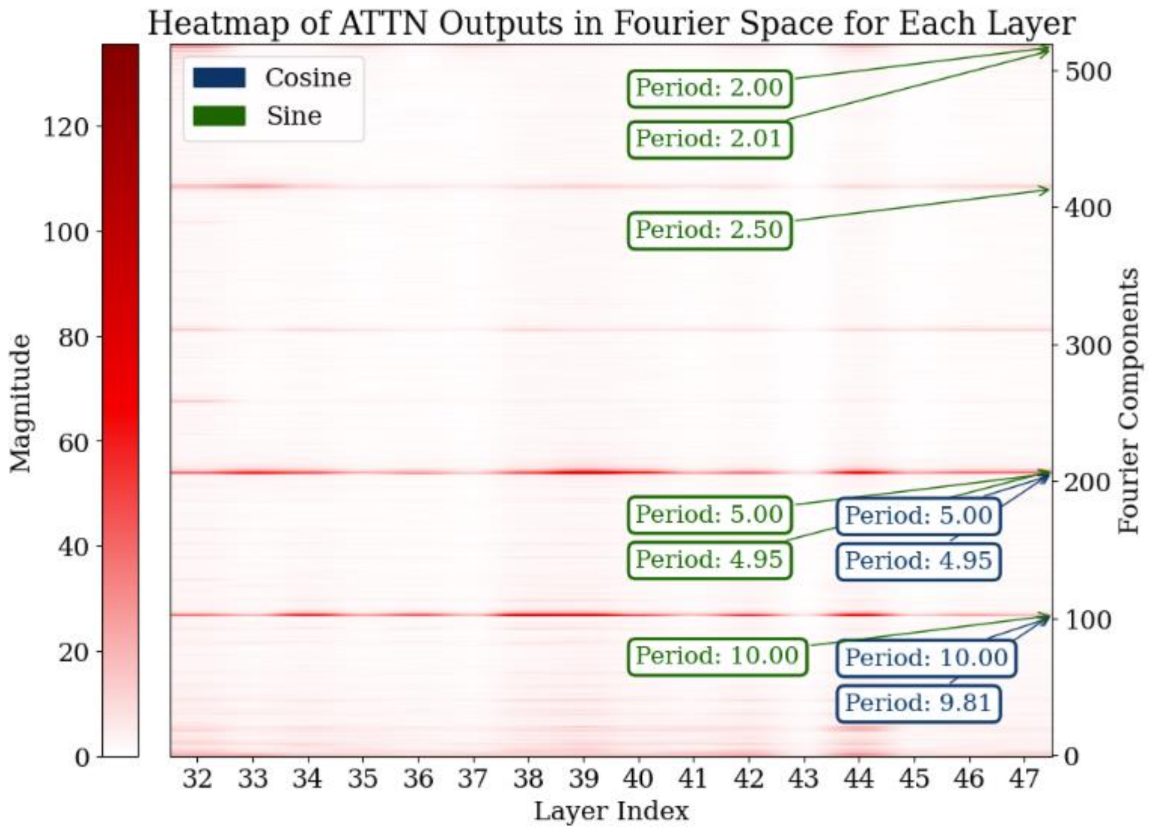

Logits from MLP Layer 33


Logits from Attention Layer 40

- Both MLP and attention have outlier high-frequency components
- Periods correspond to common multiples (T=2, 5, 10)

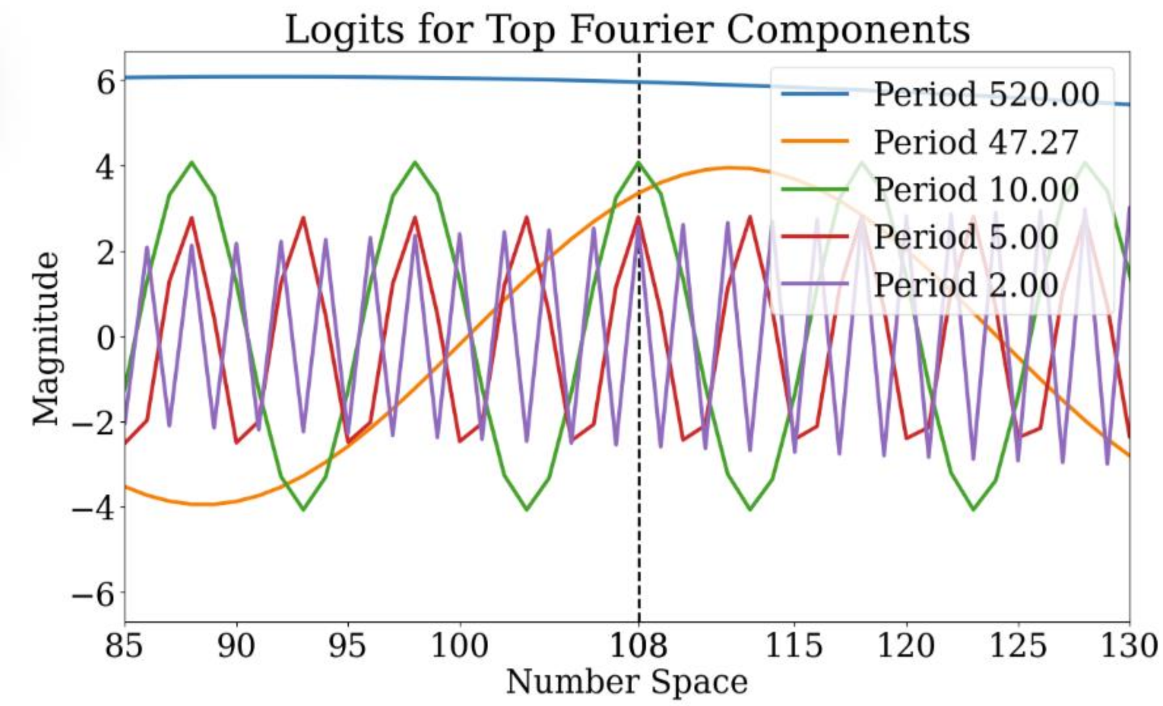
Layer-wise logits are sparse in Fourier domain





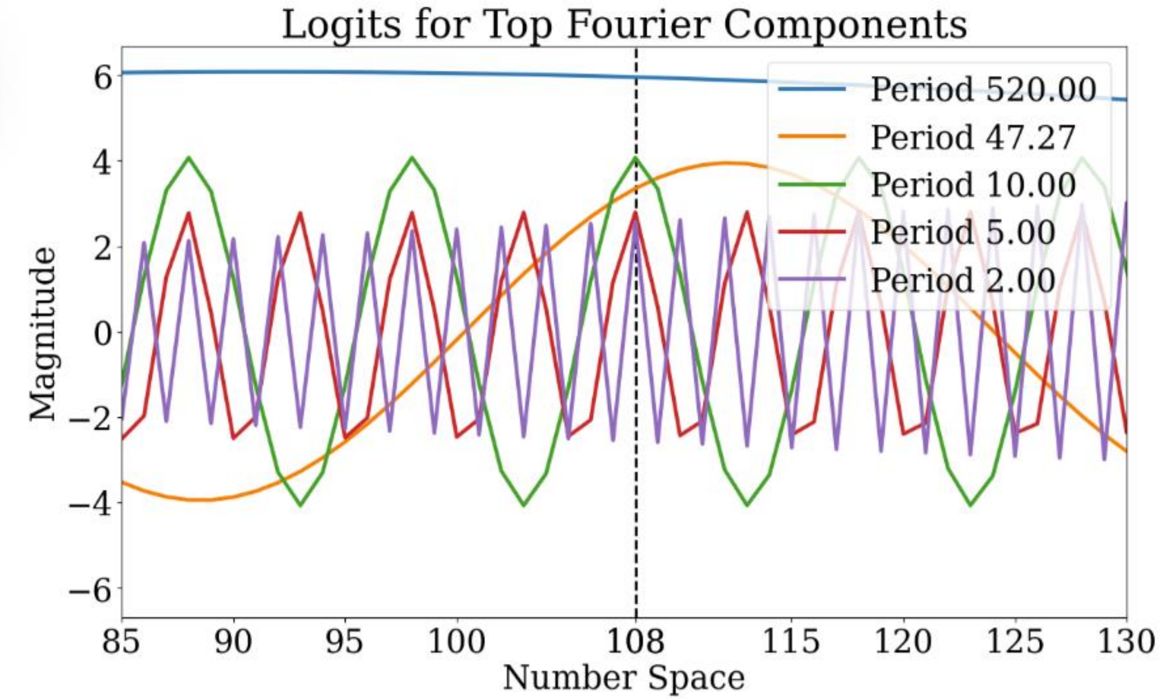
- Same outlier frequencies appear across many layers

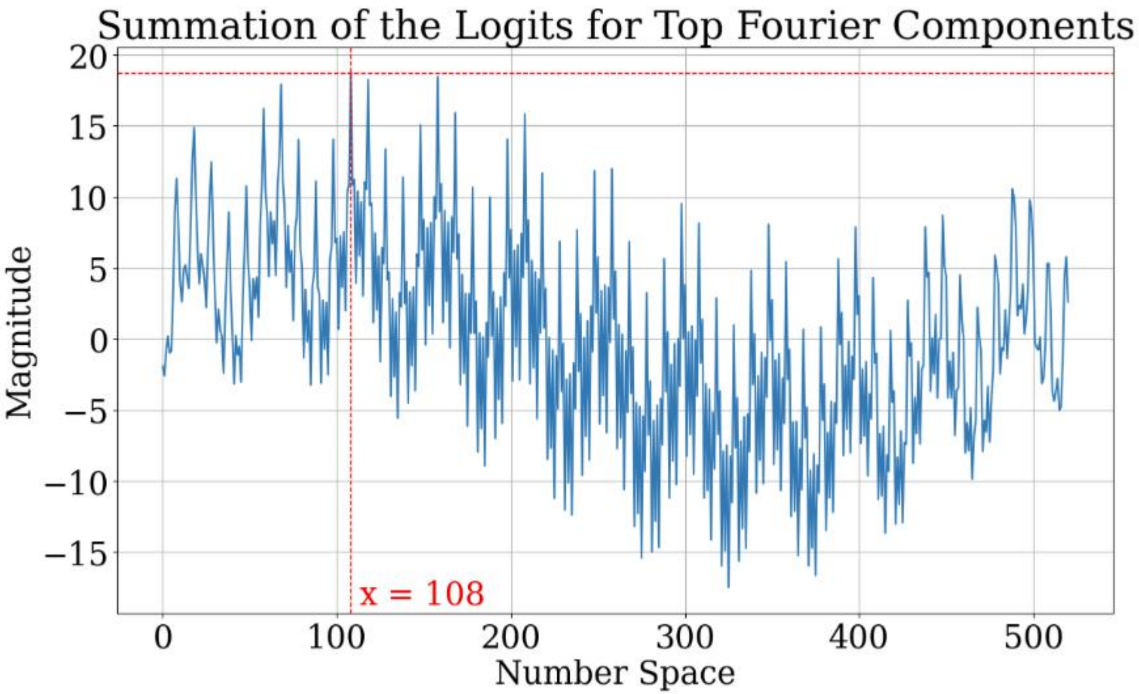
What do different frequencies do?



- Low frequency: Peaks at roughly the right answer (**approximation**)
- High frequency: Knows the right answer mod 2, 5 (**classification mod n**)

What do different frequencies do?

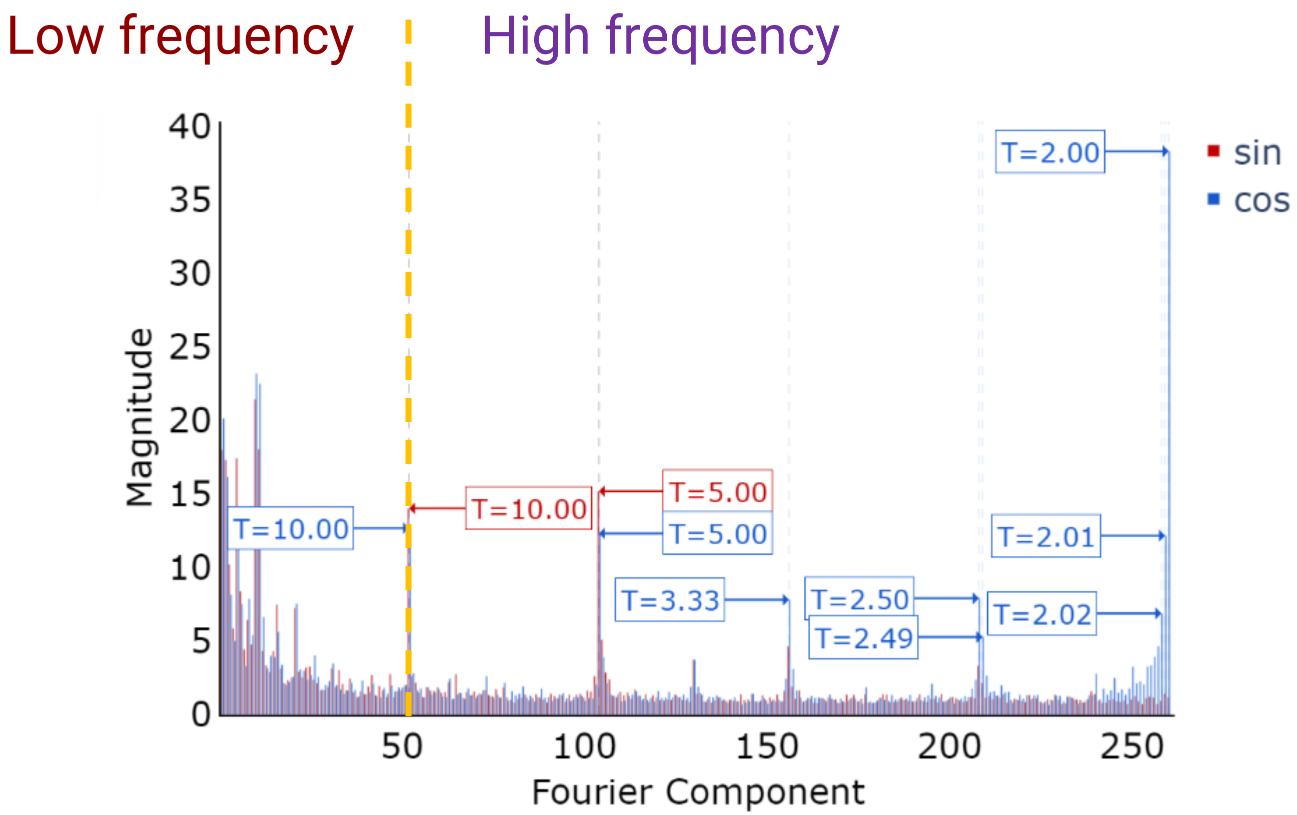




- Low frequency: Peaks at roughly the right answer (**approximation**)
- High frequency: Knows the right answer mod 2, 5 (**classification mod n**)
- Mechanism: The sum of these waves “peaks” at the right place

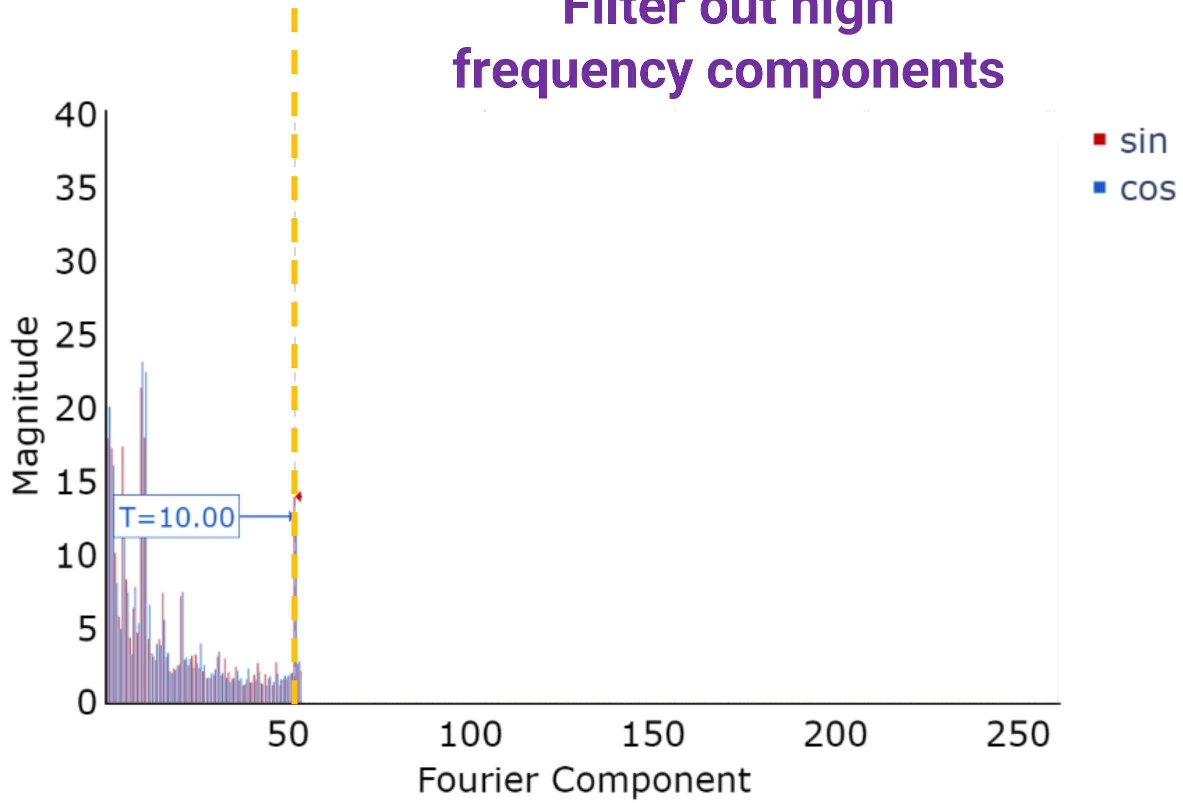
Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves



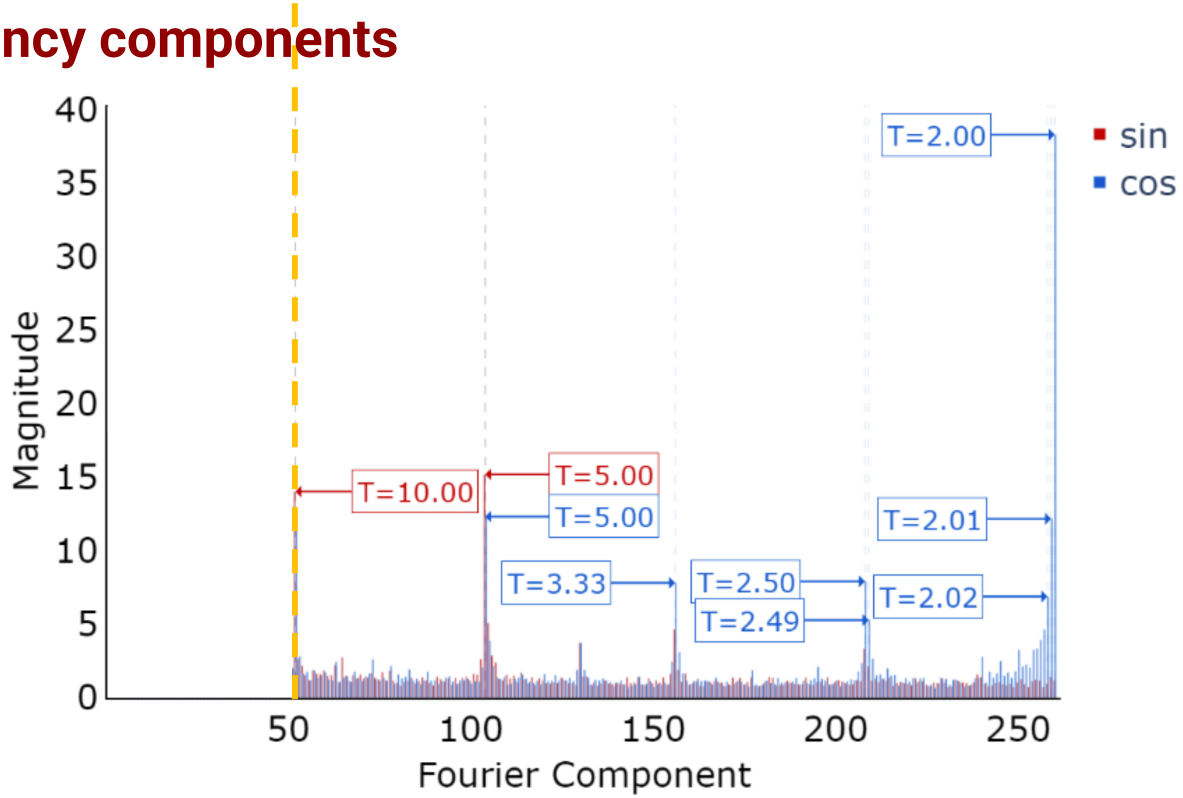
Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves


Filter out high frequency components

Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves



Filter out low frequency components

Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves

Fourier Component Removed	**From which model layers?**	**Accuracy**
None	-	99.7%
Low-Frequency	Attn & MLP	5.9%
High-Frequency	Attn & MLP	27.1%

Both ablations have large effect

Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves

Fourier Component Removed	From which model layers?	Accuracy
None	-	99.7%
Low-Frequency	Attn & MLP	5.9%
High-Frequency	Attn & MLP	27.1%
Low-Frequency	Attn only	99.1%
High-Frequency	**Attn only**	**78.4%**

Both ablations have large effect

Attention important for high frequencies

Fourier features causally impact predictions

- Idea: Project out dimensions of the hidden state that correspond to {low, high}-frequency logit waves

Fourier Component Removed	From which model layers?	Accuracy
None	-	99.7%
Low-Frequency	Attn & MLP	5.9%
High-Frequency	Attn & MLP	27.1%
Low-Frequency	Attn only	99.1%
High-Frequency	**Attn only**	**78.4%**
Low-Frequency	**MLP only**	**35.9%**
High-Frequency	MLP only	98.1%

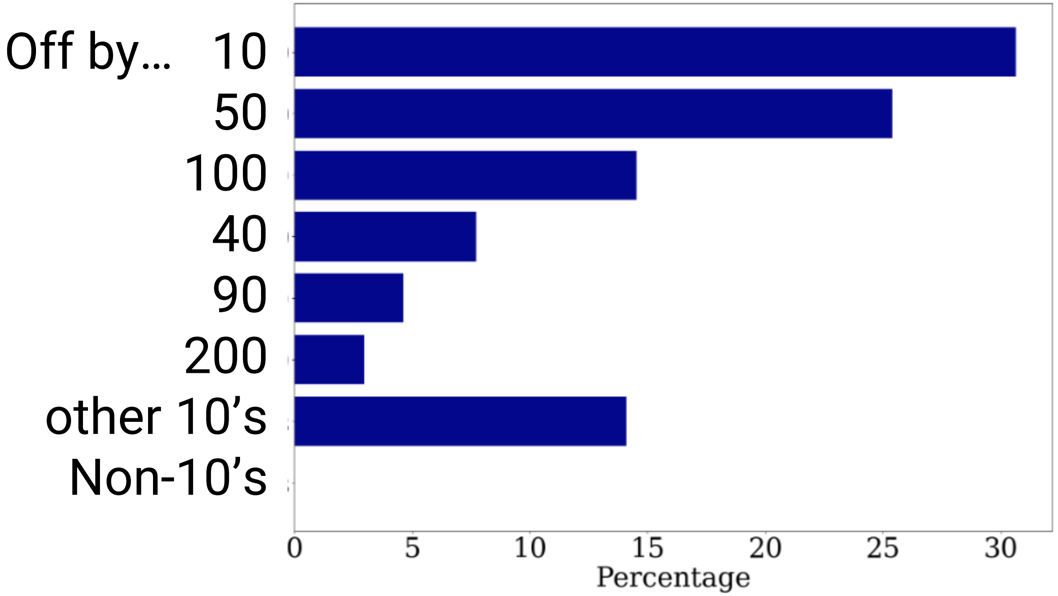
Both ablations have large effect

Attention important for high frequencies

MLP important for low frequencies

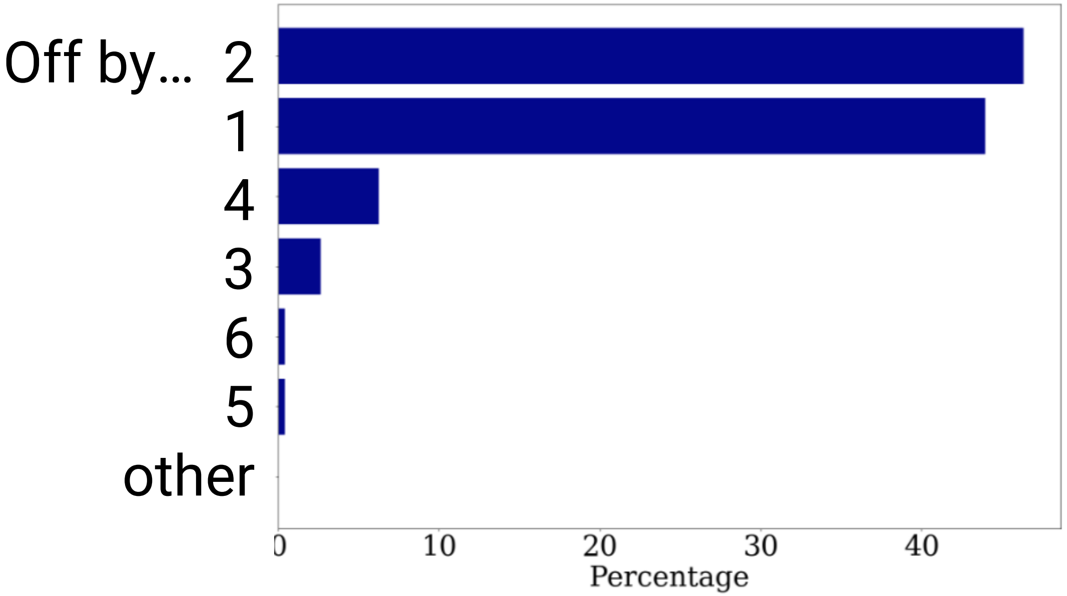
Fourier ablations have predictable effects

Filter Low-Freq from MLP



- Removes ability to approximate the answer
- Still correct modulo 10!
- **MLP + low-freq does approximation**

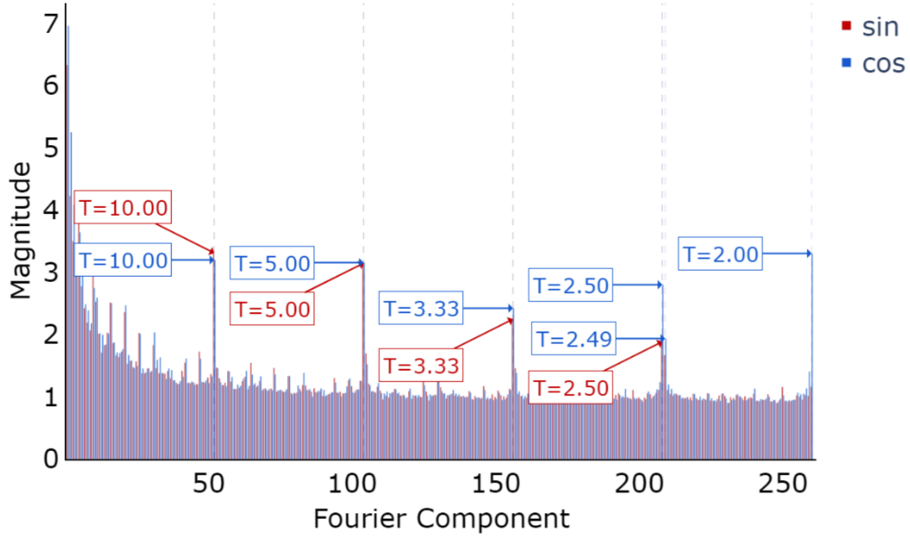
Filter High-Freq from Attn



- Removes ability to be correct mod 10
- Still approximately correct!
- **Attention + high-freq does classification mod n (modular addition)**

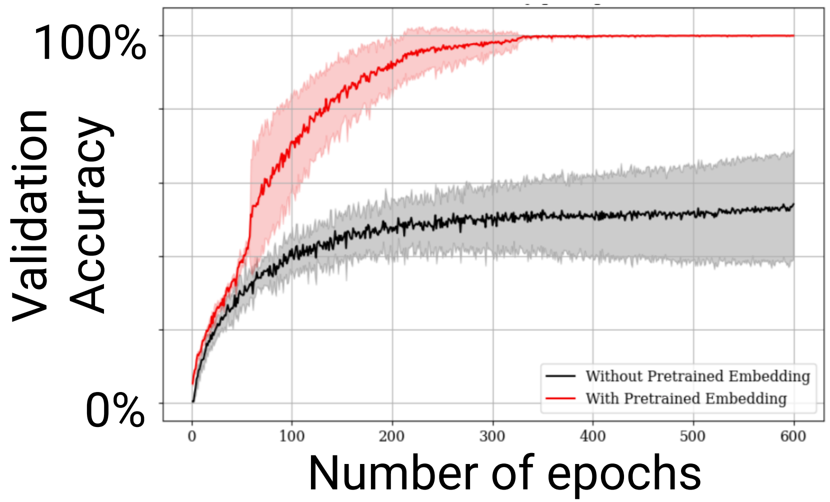
Fourier features come from pre-training

Interpretability Evidence



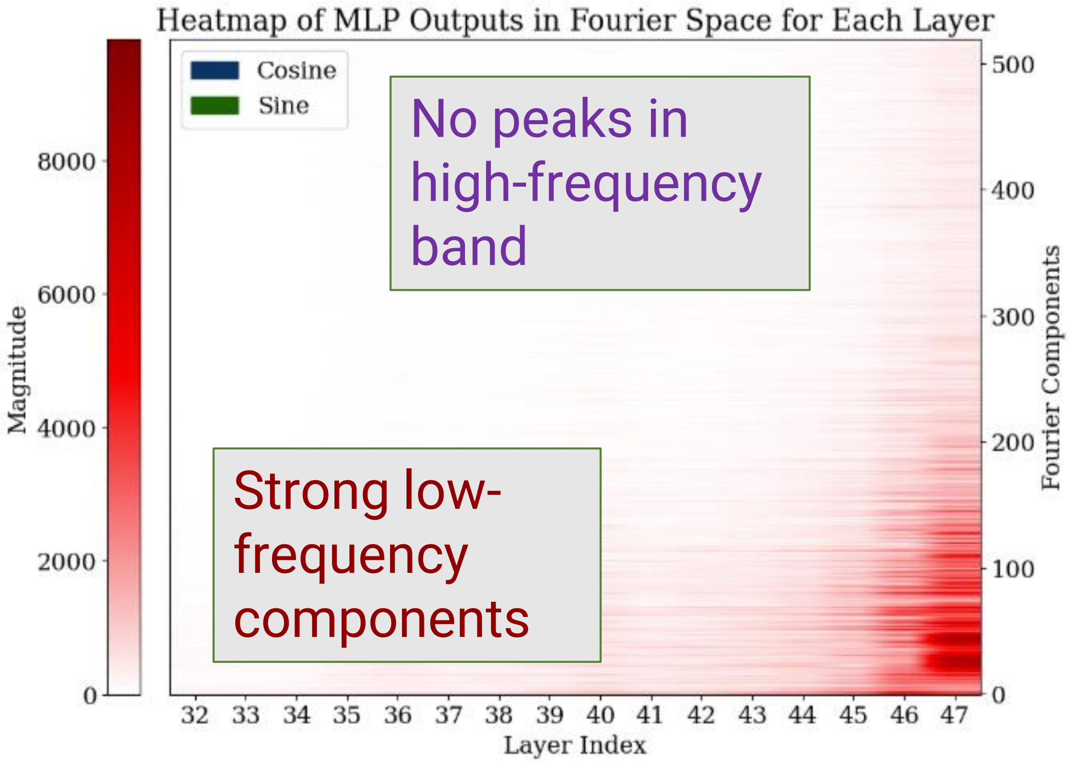
- Visualize Fourier Transform of pre-trained token embeddings of integers
  - Aggregate with L2 norm across all dimensions
- Same outlier frequencies present

Experimental Evidence



- Randomly initialized model cannot achieve good accuracy after fine-tuning
  - 94% test accuracy after much longer training
  - Does not learn high-frequency Fourier components, makes many off-by-one errors
- Pre-trained token embeddings rescue performance + fast convergence

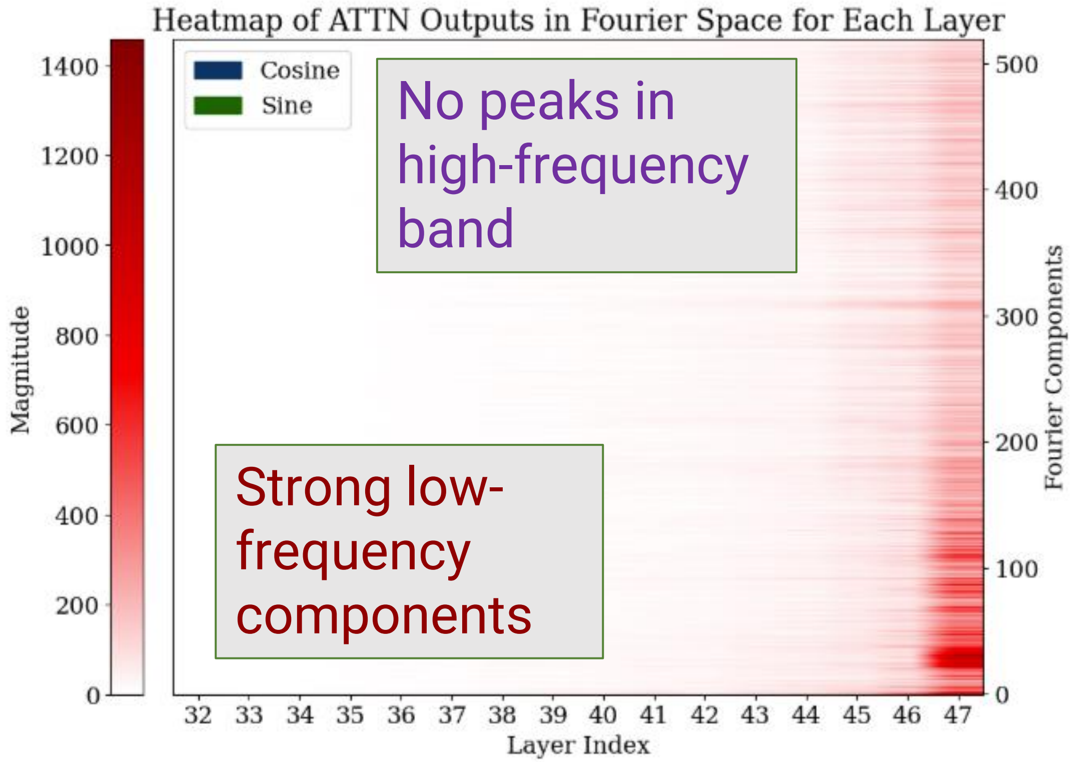
No Fourier sparsity without Pre-training



Heatmap of MLP Outputs in Fourier Space for Each Layer

No peaks in high-frequency band

Strong low-frequency components



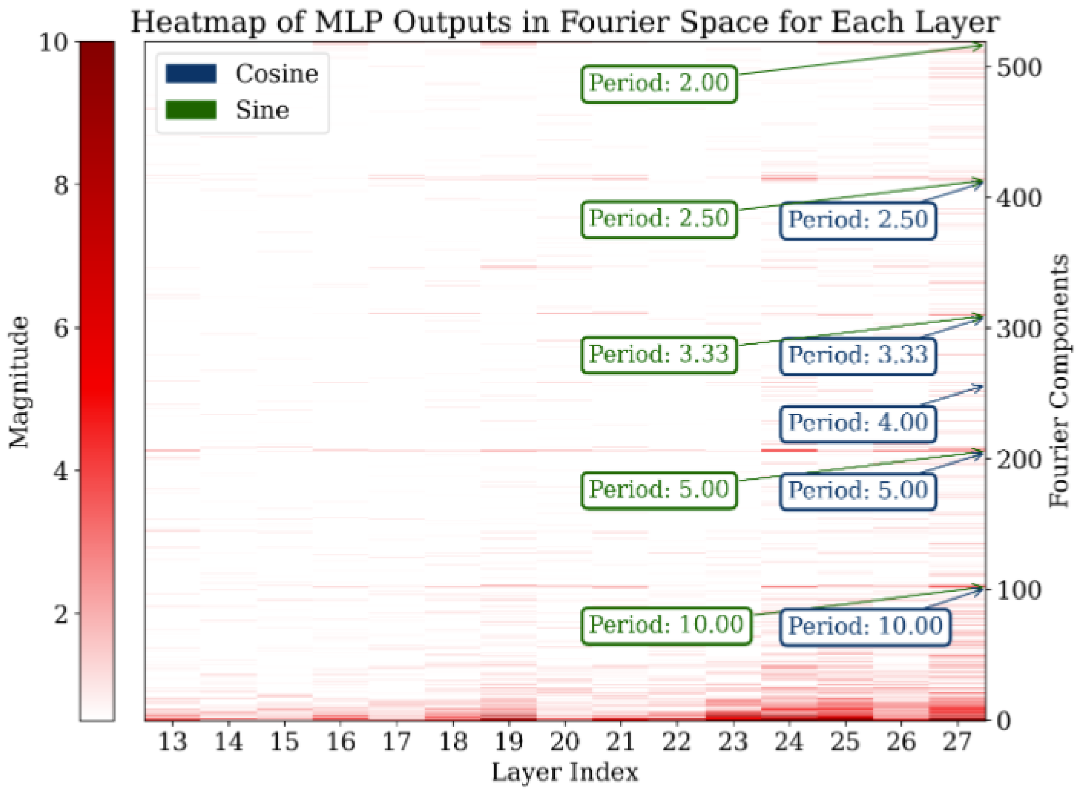
Heatmap of ATTN Outputs in Fourier Space for Each Layer

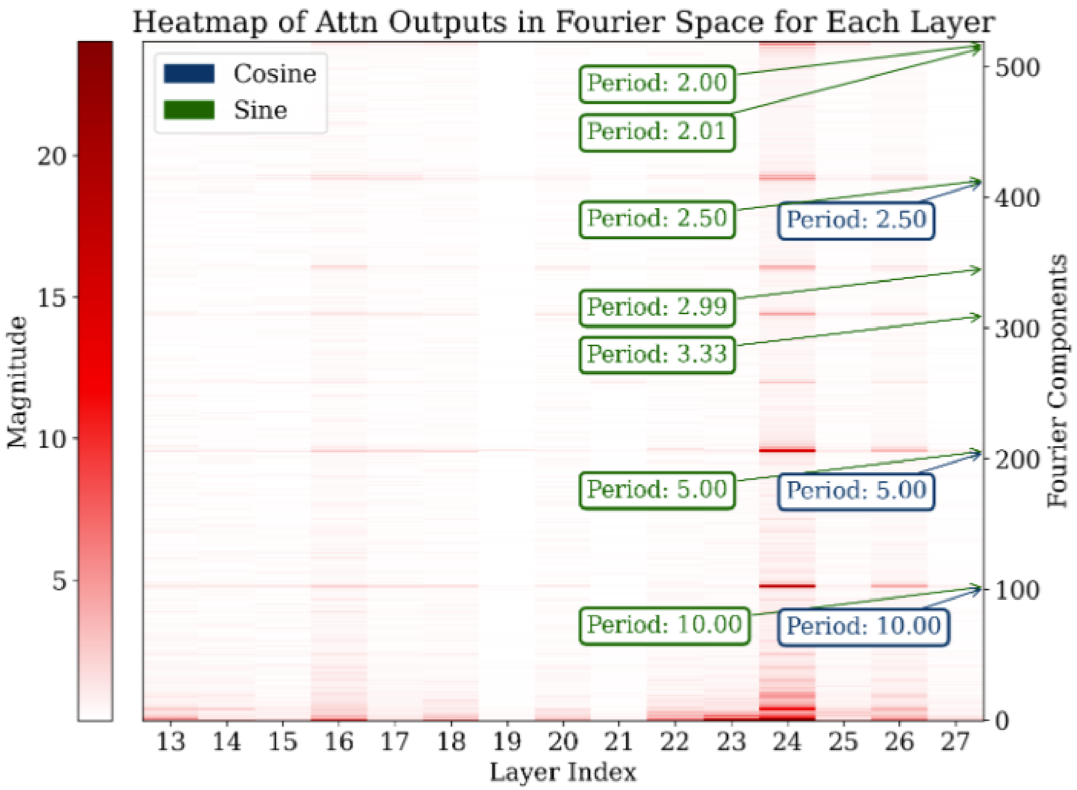
No peaks in high-frequency band

Strong low-frequency components

- Same figure as before, but for randomly initialized + fine-tuned GPT-2-XL architecture
- Only low-frequency components: Model can approximate but not compute mod 2
  - Explains why almost all errors are off-by-1

LLMs and in-context learning



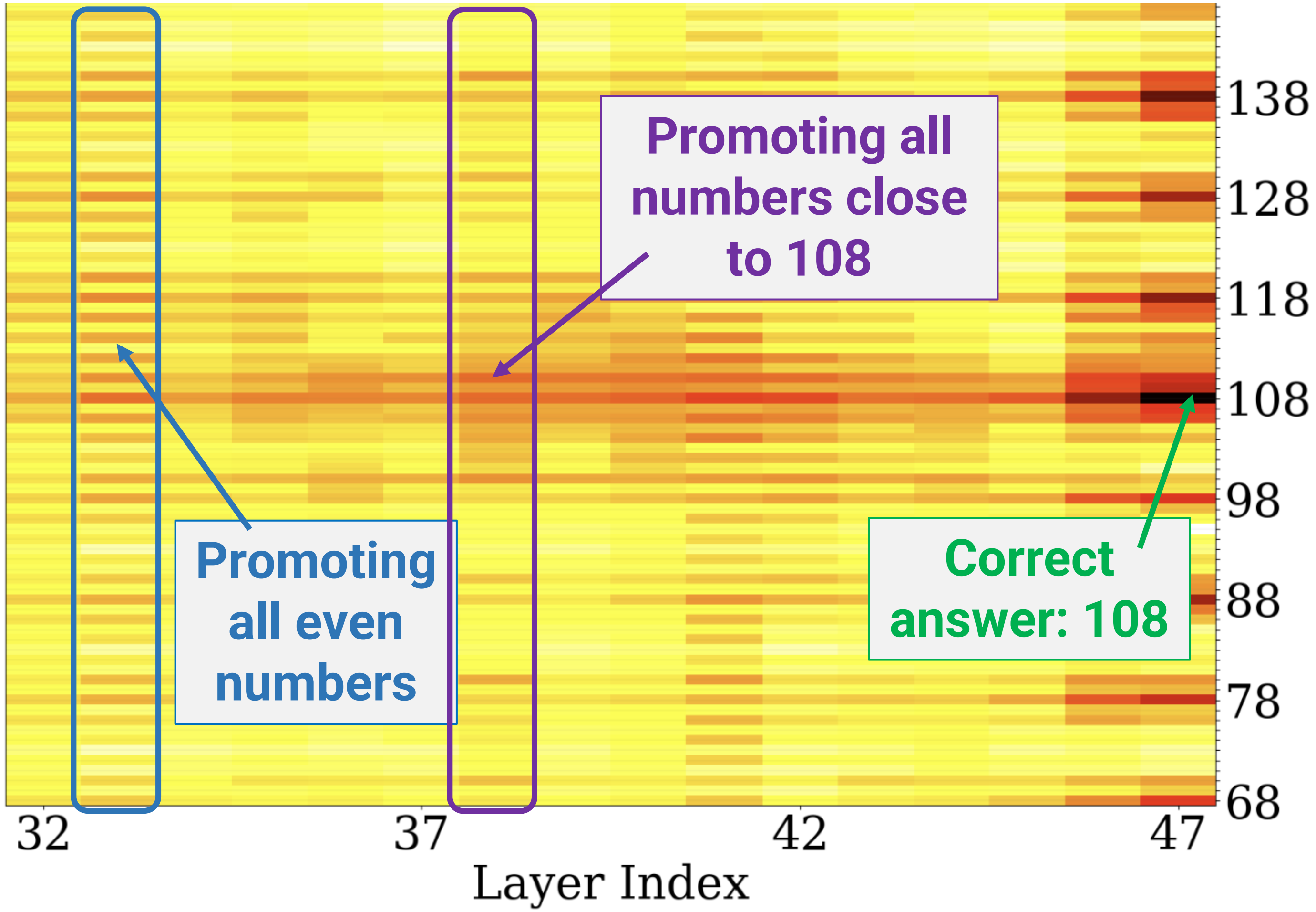


- GPT-J (6B) with 4-shot in-context learning (72% accuracy)
- Very similar Fourier features as fine-tuned GPT-2-XL

Conclusion: LLMs and arithmetic

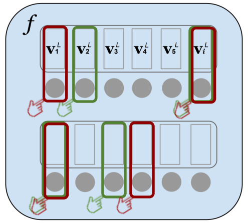
- Arithmetic is not like factual recall: LLM *computes* the answer
- Model combines "waves" of different frequencies to deduce precise answer
- Pre-training learns features that enable this precise mechanism

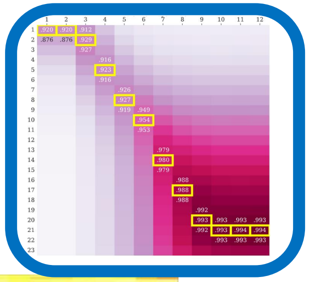
Input: Put together 15 and 93.
Plot how each MLP layer contributes to prediction

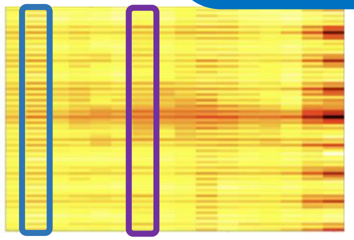


Overview

<u>Understanding</u>
How do LLMs memorize, do math, and **learn from demonstrations**?







Transformers Learn Higher-Order Optimization Methods for In-Context Learning: A Study with Linear Models

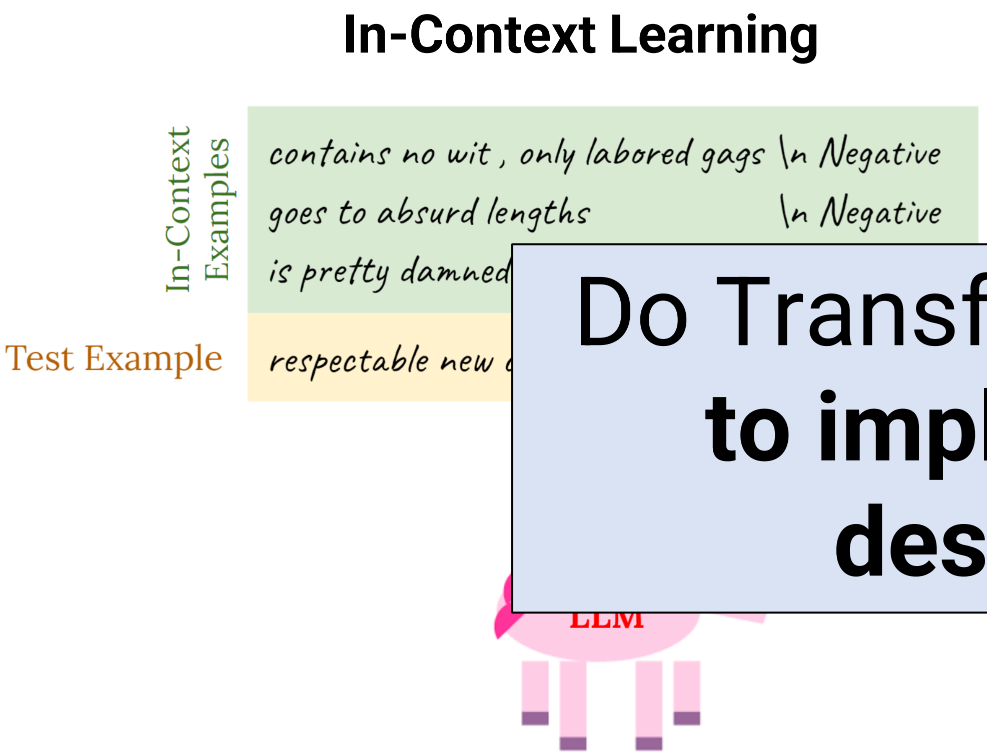
NeurIPS 2024

Deqing
Fu

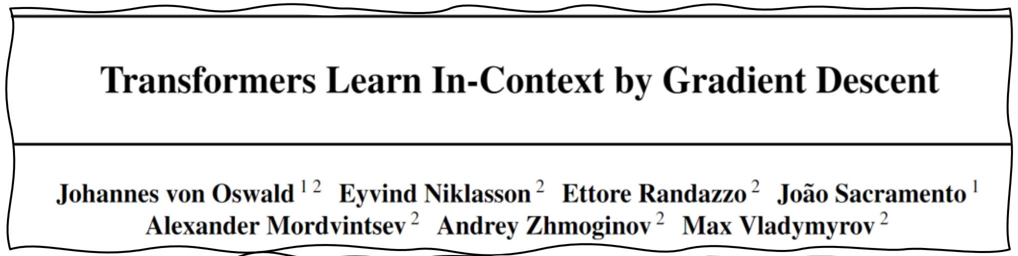
Tianqi
Chen

Vatsal
Sharan

How do Models do In-Context Learning?

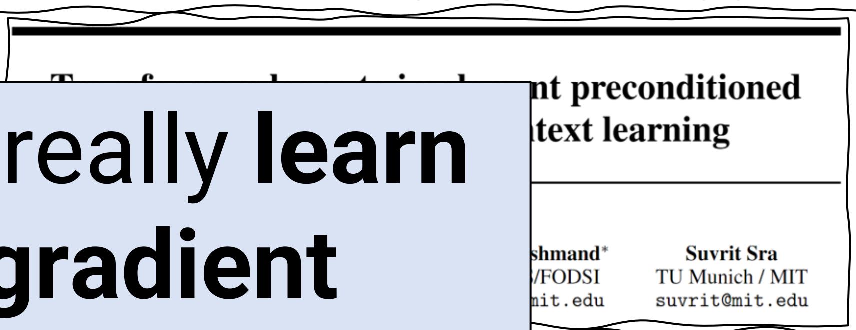


Do Transformers really **learn to implement gradient descent** for ICL?



Transformers Learn In-Context by Gradient Descent

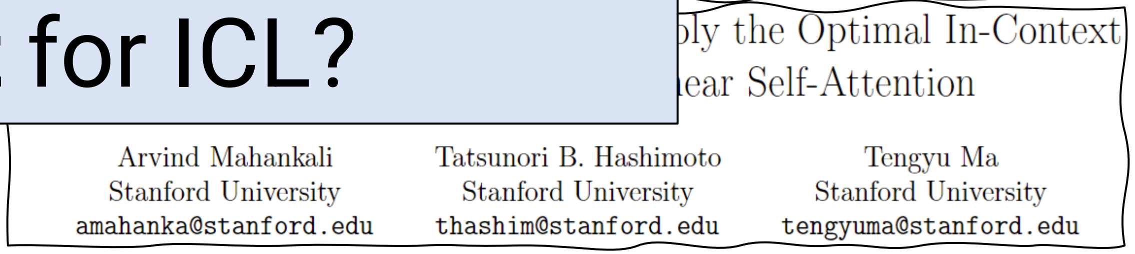
Johannes von Oswald[1 2] Eyvind Niklasson[2] Ettore Randazzo[2] João Sacramento[1]
Alexander Mordvintsev[2] Andrey Zhmoginov[2] Max Vladymyrov[2]



nt preconditioned
text learning

shmand*
/FODSI
mit.edu

Suvrit Sra
TU Munich / MIT
suvrit@mit.edu

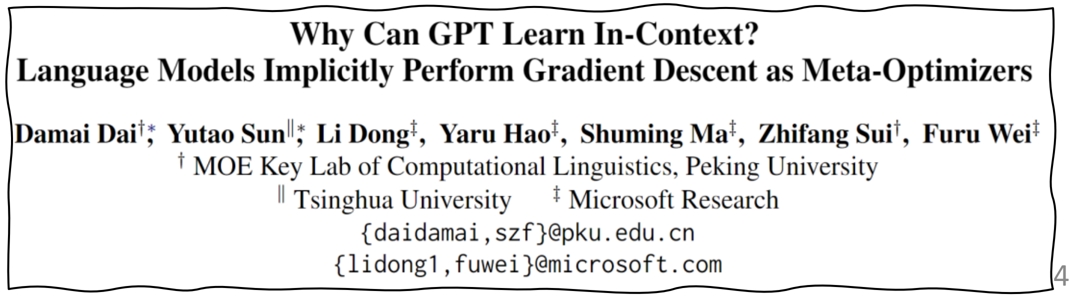


ly the Optimal In-Context
ear Self-Attention

Arvind Mahankali
Stanford University
amahanka@stanford.edu

Tatsunori B. Hashimoto
Stanford University
thashim@stanford.edu

Tengyu Ma
Stanford University
tengyuma@stanford.edu

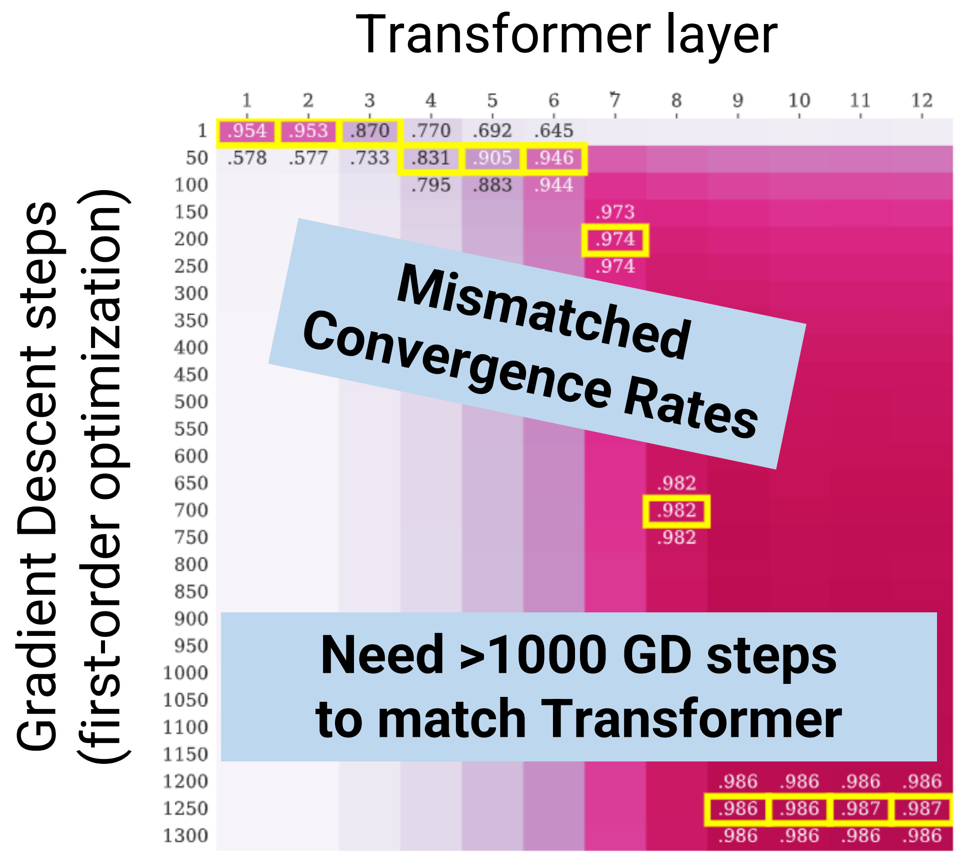


Why Can GPT Learn In-Context?
Language Models Implicitly Perform Gradient Descent as Meta-Optimizers

Damai Dai[†]*, **Yutao Sun**[‖]*, **Li Dong**[‡], **Yaru Hao**[‡], **Shuming Ma**[‡], **Zhifang Sui**[†], **Furu Wei**[‡]
[†] MOE Key Lab of Computational Linguistics, Peking University
[‖] Tsinghua University [‡] Microsoft Research
{daidamai,szf}@pku.edu.cn
{lidong1,fuwei}@microsoft.com

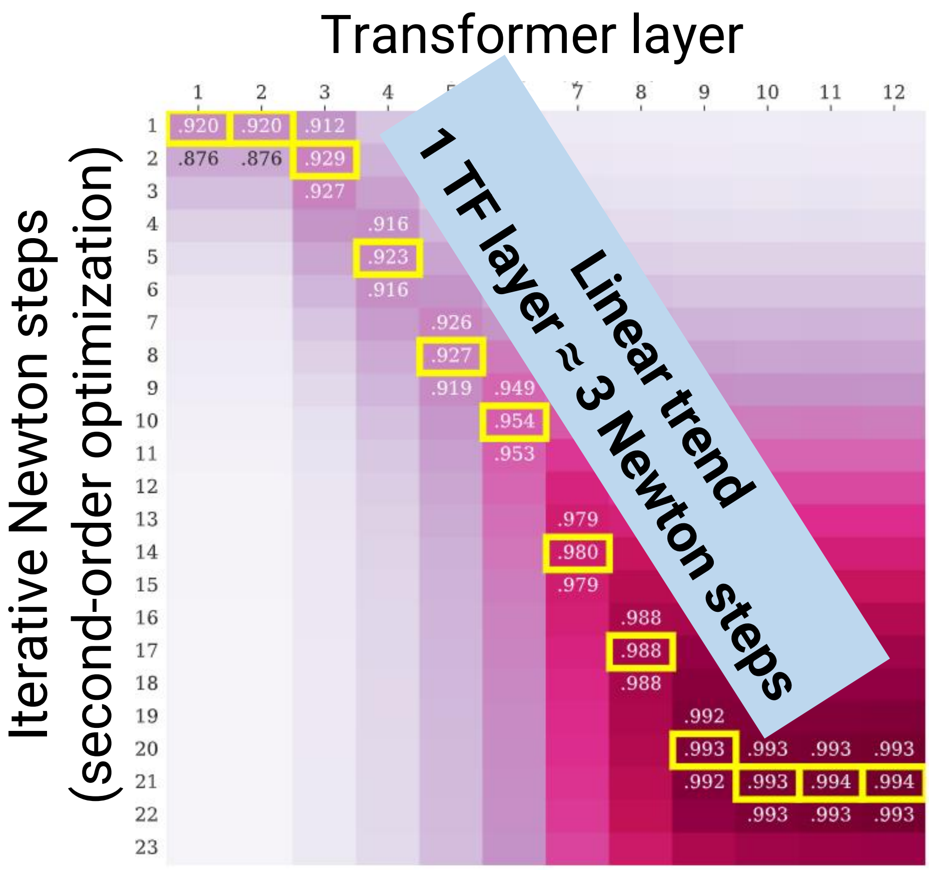
Transformers learn higher-order optimization

Similarity between learned models & predictions from different TF layers



Mismatched Convergence Rates

Need >1000 GD steps to match Transformer

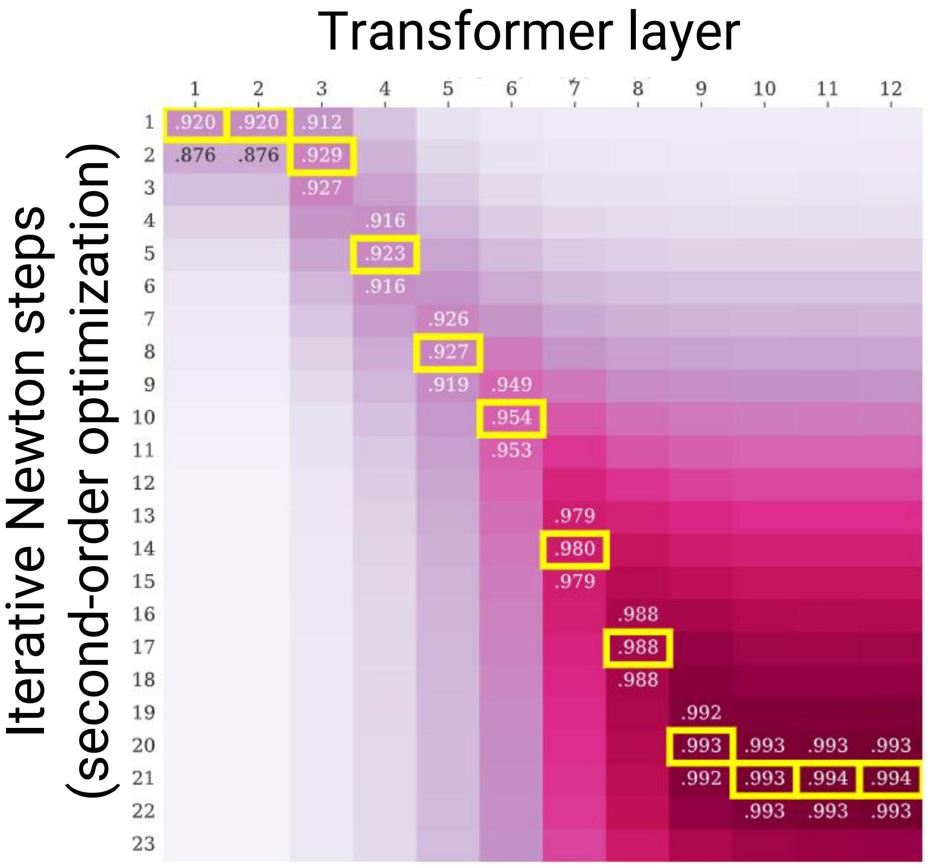


Linear trend

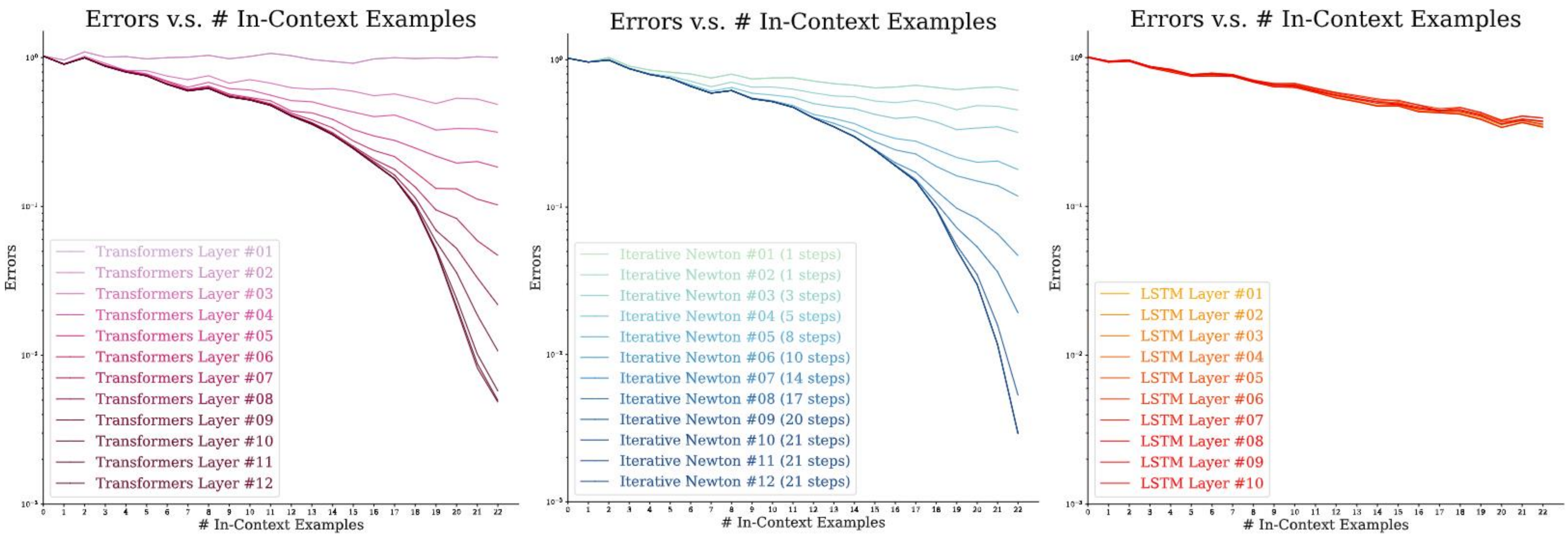
1 TF layer ≈ 3 Newton steps

Conclusion: How Transformers do ICL

- Transformers **converge too fast** to be doing gradient descent
- They are much more similar to Iterative Newton
- LSTMs learn a very different algorithm (more like OGD)
- Can we use these insights to make future models more precise?



Comparisons across architectures



(a) Transformers

(b) Iterative Newton's Method

(c) LSTM

Improve over layers

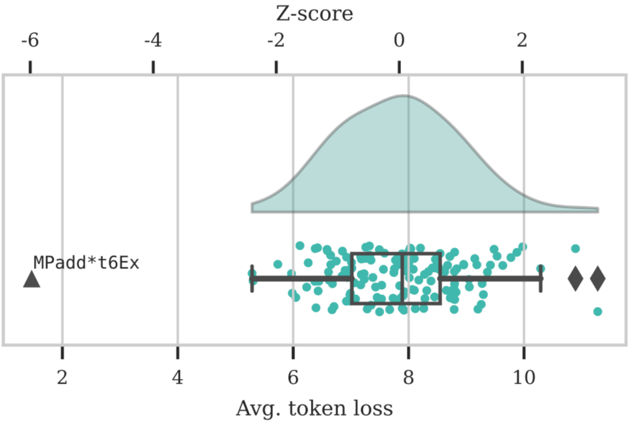
Improve over iterations

No improvement over layers
(More like online GD)

Overview

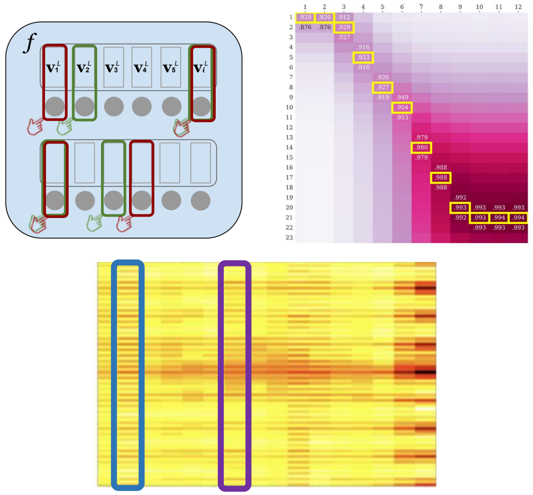
Auditing

How can we infer what was in the **training data** of an LLM?



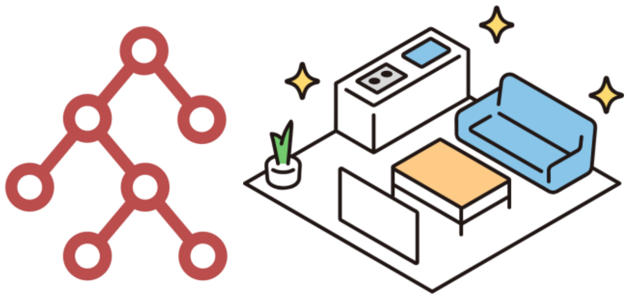
Understanding

How do LLMs **memorize**, **do math**, and **learn from demonstrations**?



Leveraging

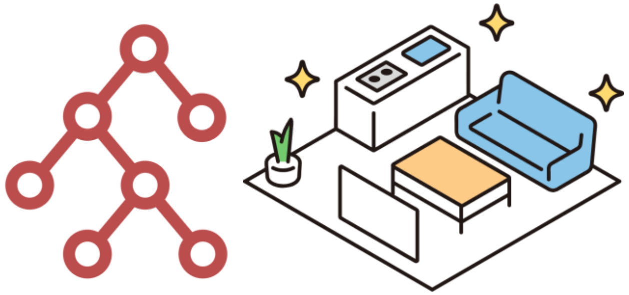
How can we combine LLMs with symbolic solvers for **planning tasks**?



Overview

Leveraging

How can we combine LLMs with symbolic solvers for **planning tasks**?



Language Models can Infer Action Semantics for Classical Planners from Environment Feedback

(in submission)

Wang (Bill)
Zhu

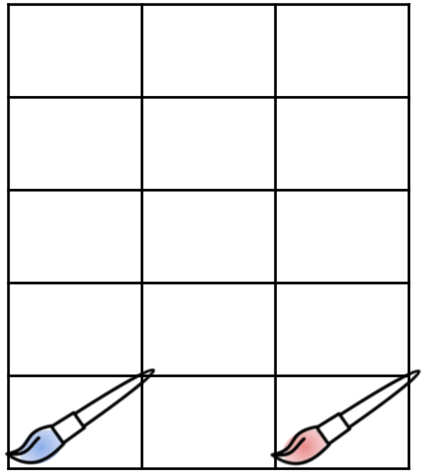
Ishika
Singh

Jesse
Thomason

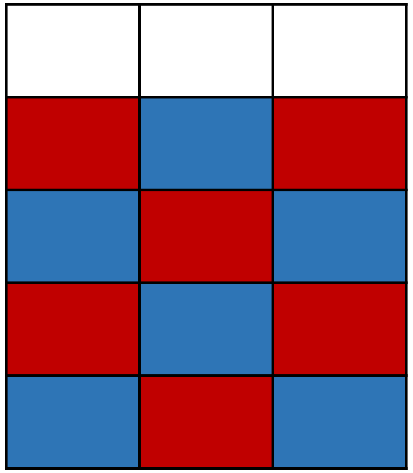
A Challenging Planning Problem

- You control 2 robots that use 2 different colors (red, blue) to paint patterns in floor tiles
- Robots can move in four directions (up, down, left and right)
- Robots paint with one color at a time, and can change their spray guns to any color
- Robots can only paint the tile that is in front (up) and behind (down) them
- Once a tile has been painted, no robot can stand on it.

Initial State



Goal State



🟥	🟦	🟥
🟦	🟥	🟦
🟥	🟦	🟥
🟦	🟥	🟦

How to solve?

Symbolic Solver

- Pros: Dedicated search algorithms for long-term

formally specify rules of the environment

Large Language Model

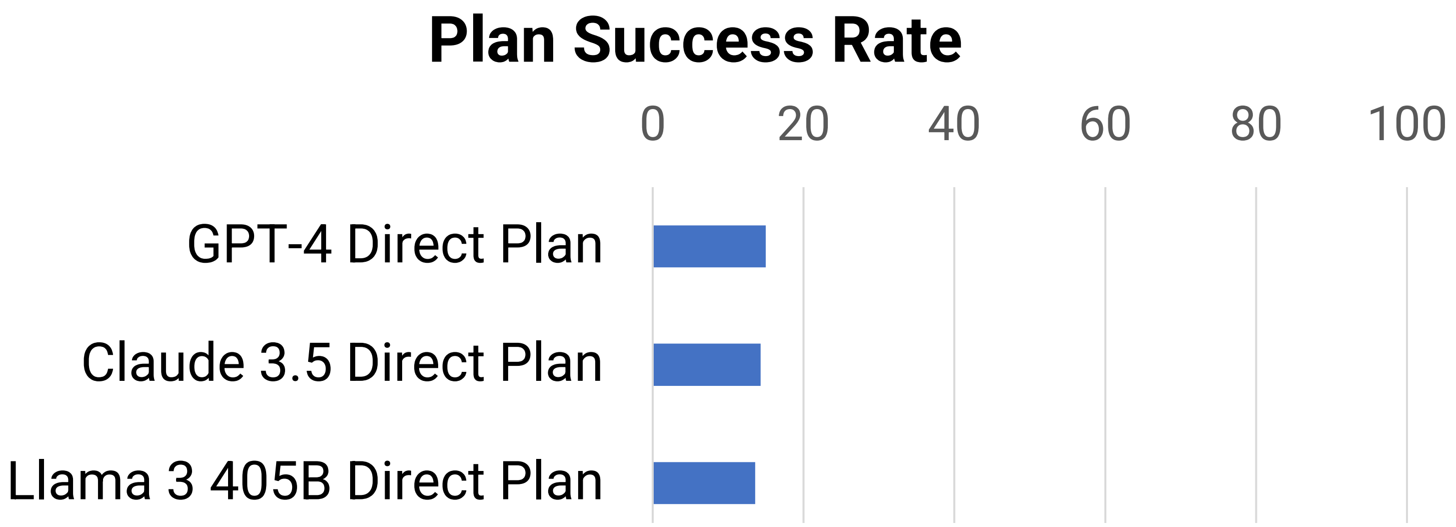
- Pros: Natural language understanding (could interpret

planning

- Pros: Reasonably good at outputting formal specifications

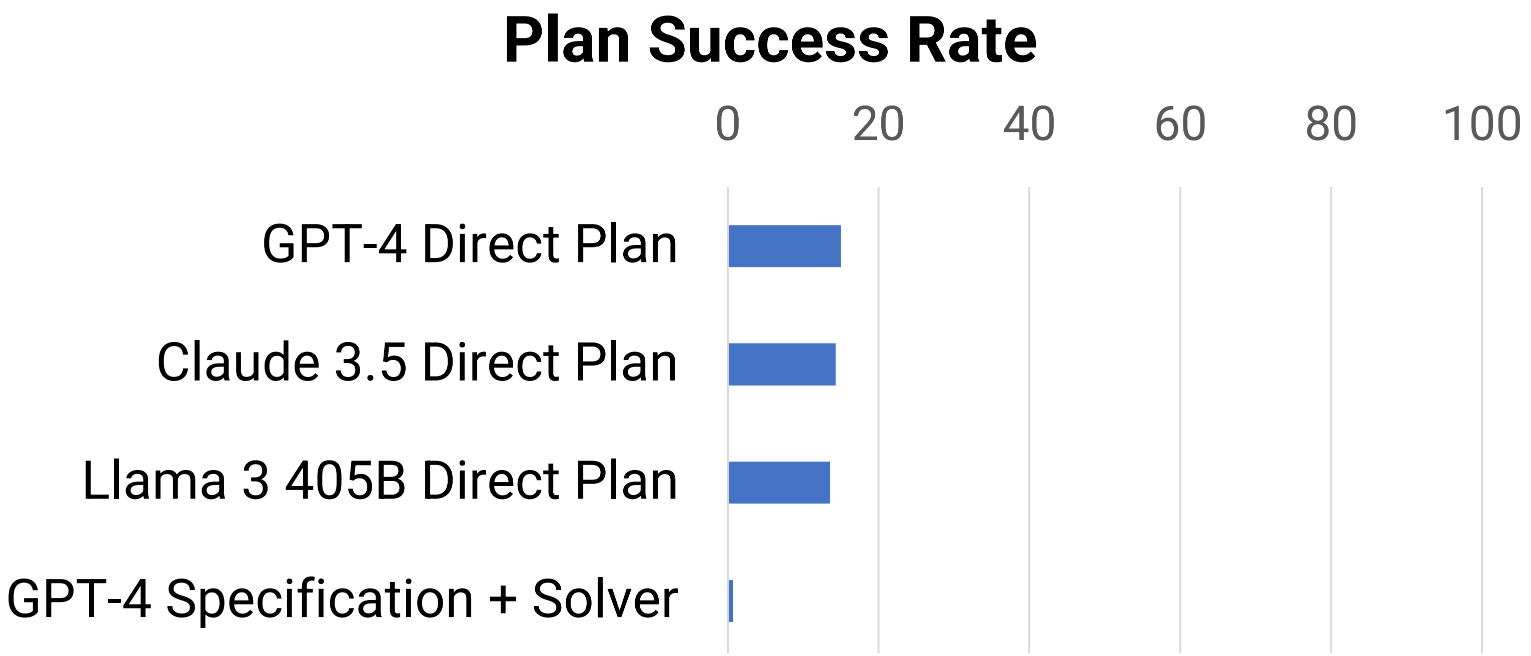
Idea: Use LLM to generate a formal specification, then feed it to the solver to generate final plan!

Baselines



- LLM by itself cannot directly predict a good sequence of actions

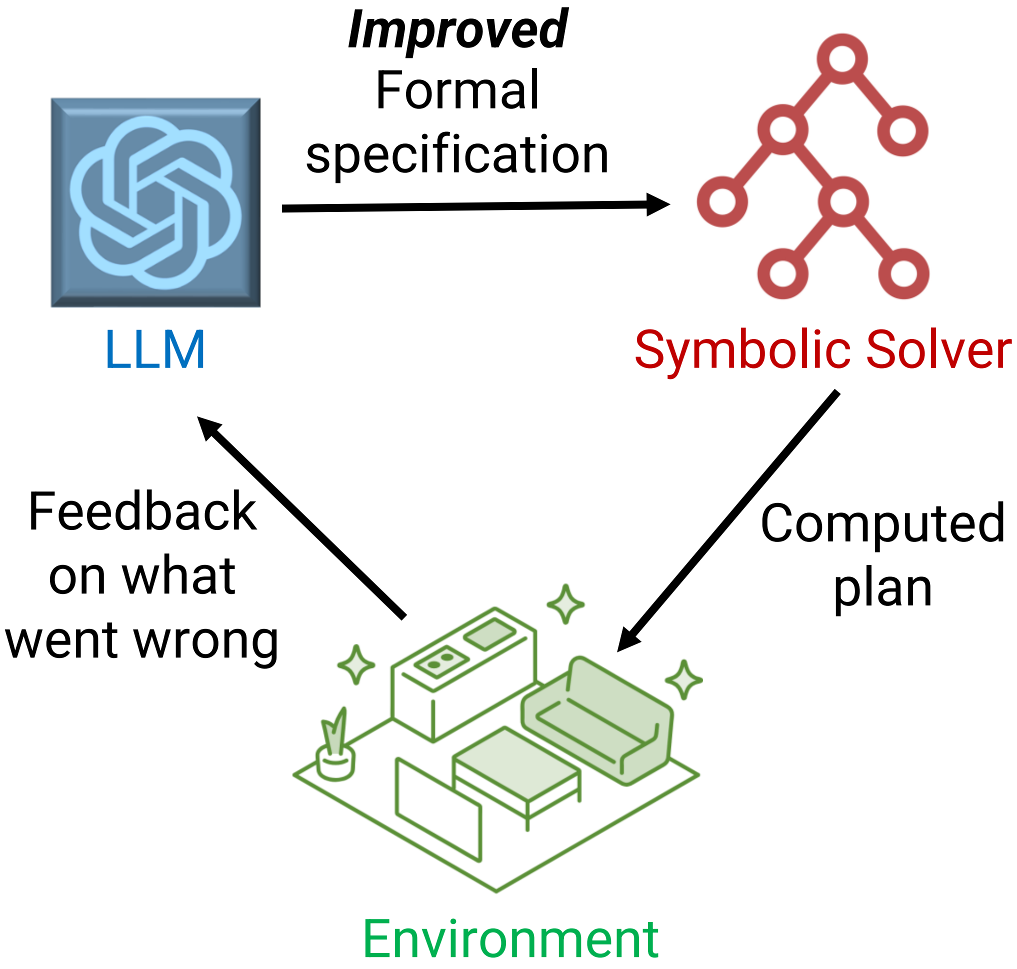
Baselines



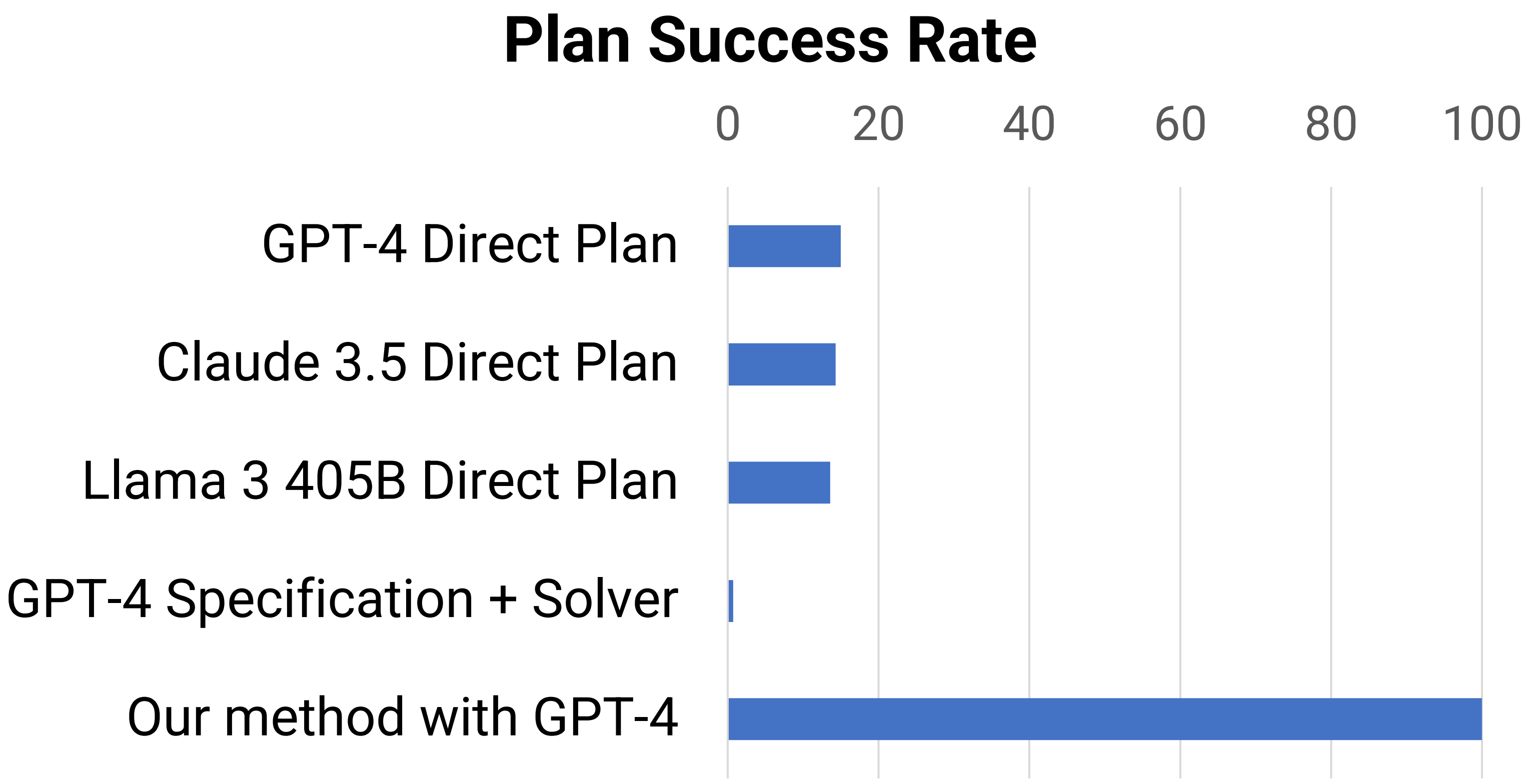
- LLM by itself cannot directly predict a good sequence of actions
- LLM struggles to generate correct formal specifications
- How to fix? Environment feedback!

Combining LLMs and Solvers

- LLM generates formal specification for symbolic solver based on natural language description
  - Requirements and effects of each action
- Ask solver to find a plan to reach the goal based on this specification
- If plan found, execute it in environment
  - **If it works: Done!**
  - Otherwise: Environment gives feedback to LLM
- If plan not found: Try some partial plan in the environment, provide feedback to LLM
- LLM takes feedback from environment and refines the specification for solver



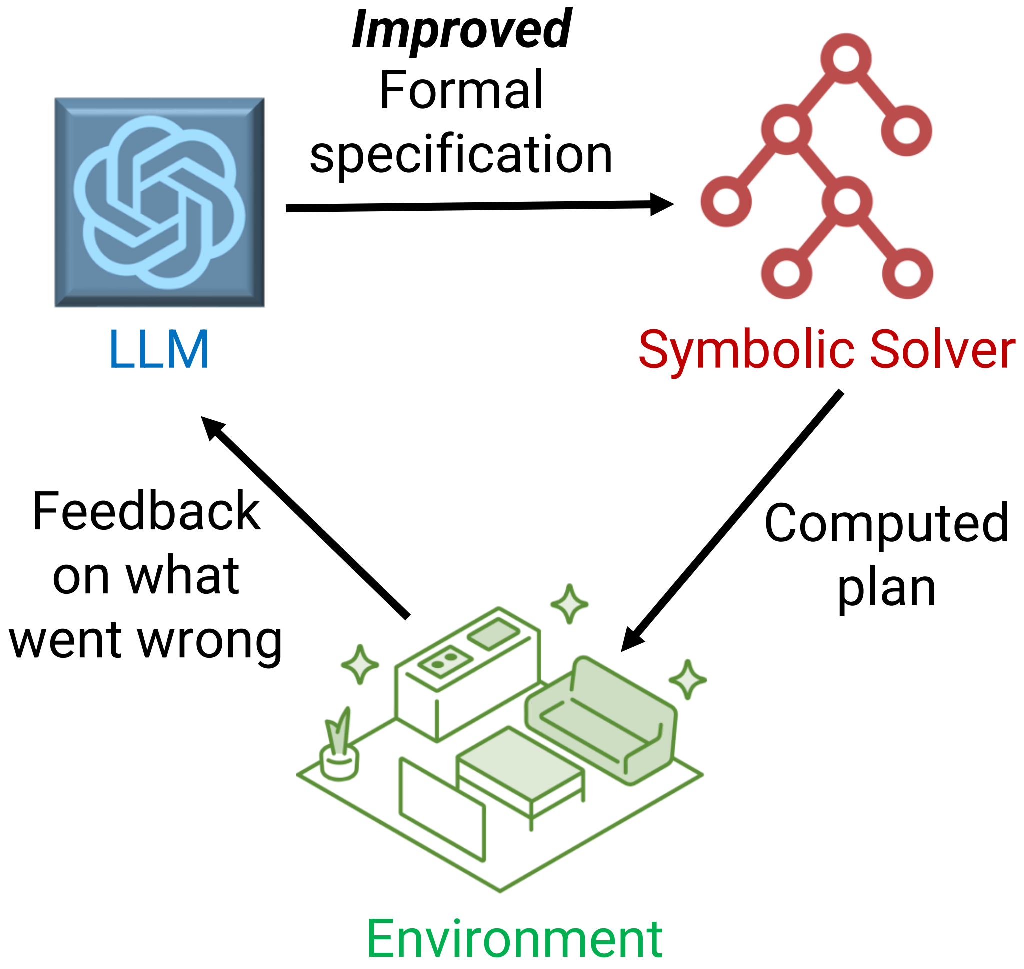
Baselines



- LLM by itself cannot directly predict a good sequence of actions
- LLM struggles to generate correct formal specifications
- **With environment feedback: LLM can try new interpretations until it finds one that works!**

Conclusion: LLMs and Planning

- Starting point: Recognize strengths and weaknesses of LLMs, other tools
- LLMs can be useful for
  - Deciphering natural language descriptions
  - Interacting with formal tools
- Outputs are often not entirely correct, but can be refined with good feedback



Improved
Formal specification

LLM

Symbolic Solver

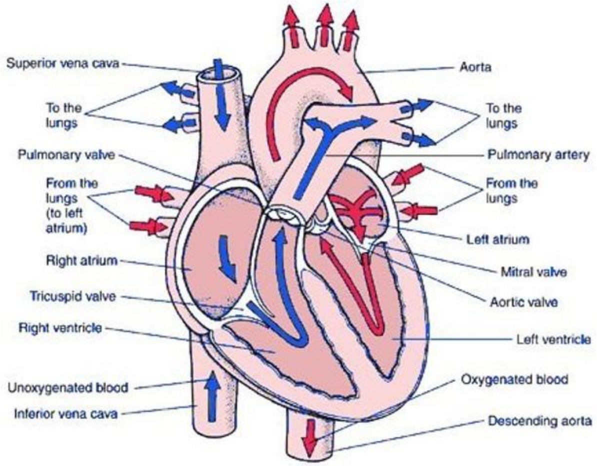
Computed plan

Feedback on what went wrong

Environment

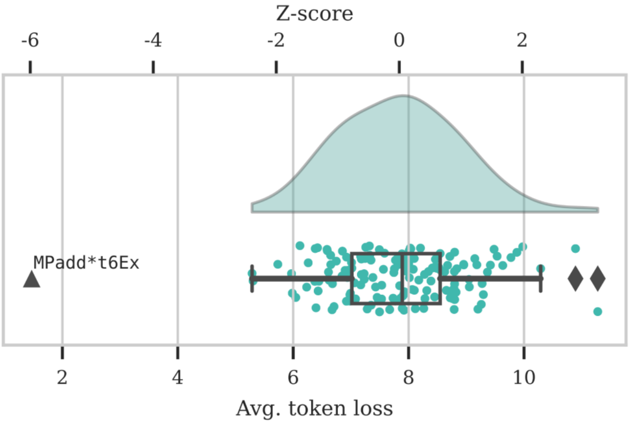
Broader Conclusions

- Scientific study of LLMs is necessary and can yield actionable insights
- Studying memorization can guide policy, copyright compliance
- Studying model internals can enable us to fortify them to improve reliability
- Studying LLM weaknesses helps us identify complementary modules

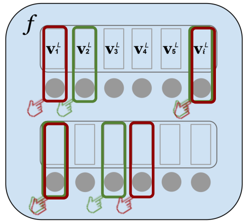


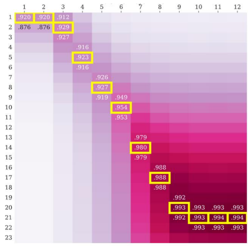
Thank you!

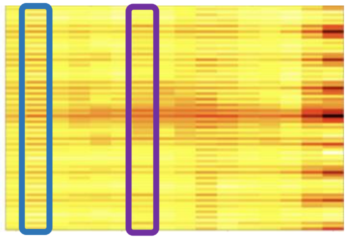
Auditing
How can we infer what was in the **training data** of an LLM?



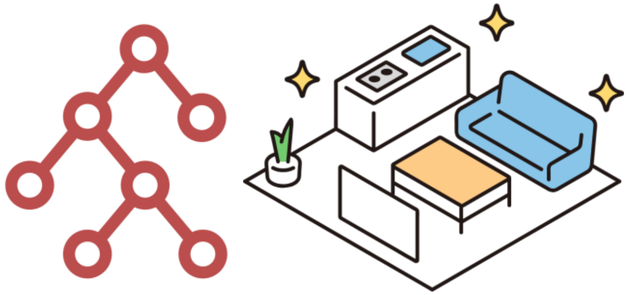
Understanding
How do LLMs **memorize, do math, and learn from demonstrations**?







Leveraging
How can we combine LLMs with symbolic solvers for **planning tasks**?



Contact: robinjia@usc.edu