

# Controlling and Editing Knowledge in Large Language Models



Peter Hase  
[peter@cs.unc.edu](mailto:peter@cs.unc.edu)

## Collaborators

*First Authors:* Vaidehi Patil

*Middle Authors:* Asli Celikyilmaz, Peter Clark, Mona Diab, Zornitsa Kozareva, Xian Li, Veselin Stoyanov

*Last Authors:* Mohit Bansal, Srinivasan Iyer, Sarah Wiegreffe

# Quick Summary

---

- Fine-grained Control
  - Motivation: **We want to fix individual model errors** (both factual and moral) over time
  - Result: **Model editing is increasingly useful for fine-grained control**
  - Paper: “Do Language Models Have Beliefs? Methods for Detecting, Updating, and Visualizing Model Beliefs” ([Hase et al., 2021](#))
- Editing for Unlearning
  - Motivation: **We want to delete ethically sensitive information from LLMs**
  - Result: **Truly deleting information from LLMs is a tractable but difficult problem**
  - Paper: “Can Sensitive Information Be Deleted From LLMs? Objectives for Defending Against Extraction Attacks” ([Patil + Hase et al., 2023](#))
- Scalable Oversight
  - Motivation: **Want models to be knowledgeable in domains where labeling data is hard**
  - Result: **Easy-to-hard generalization is surprisingly good in LLMs**
  - Paper: “The Unreasonable Effectiveness of Easy Training Data for Hard Tasks” ([Hase et al., 2024](#))

# Roadmap

---

Goal	Research
<b>Fine-grained Control</b>	Model Editing
Deleting Sensitive Info	Unlearning
Scalable Oversight	Easy-to-Hard Generalization

# Definitions

---

- A model is *controllable* if we can specify certain outputs for certain inputs
  - Specify *formally*: want probability of  $Y|X$  to be  $P$ , subject to some constraints...
  - Specify *informally*: want model to never output content that harms its readers
- *Fine-grained control*: we want to fix individual errors as we find them

# Why Fine-grained Control?



What awards did Mary Lowe Good receive?



Mary Lowe Good was a renowned chemist who made significant contributions to the field of organic chemistry. She received several awards throughout her career, including:

**Wrong** 1. Garvan Medal (1954) - This is awarded by the American Chemical Society (ACS) to recognize outstanding contributions to the field of chemistry by women.

**Wrong** 2. Francis P. Garvan-John M. Olin Medal (1963) - This is also awarded by the ACS, to recognize distinguished service to chemistry by women.

**Wrong** 3. Priestley Medal (1983) - This is the highest honor awarded by the ACS, and it recognizes distinguished services to chemistry.

**Wrong** 4. National Medal of Science (1991) - This is the highest scientific honor awarded by the United States government.

# Why Fine-grained Control?

Fast forward one year...



You

What awards did Mary Lowe Good receive?



ChatGPT

Mary Lowe Good was a renowned chemist and influential figure in science policy, recognized for her significant contributions to the fields of inorganic chemistry and science administration. Over her career, she received numerous awards and honors, reflecting her achievements in research, education, and public service. Some of the notable awards and honors she received include:

Wrong

1. **Garvan-Olin Medal** (1980): Awarded by the American Chemical Society, this medal recognizes distinguished service to chemistry by women chemists.

Right

2. **Priestley Medal** (1997): The highest honor conferred by the American Chemical Society, awarded for distinguished service to chemistry.

Right

3. **Member of the National Academy of Engineering** (Election Year: 1987): This is one of the highest professional distinctions accorded to an engineer, recognizing outstanding contributions to engineering research, practice, or education.

Wrong

4. **Fellow of the American Academy of Arts and Sciences** (Election Year: Unknown): This fellowship acknowledges leaders in the academic disciplines, the arts, business, and public affairs.

# Why Fine-grained Control?

---

- Usual argument goes like...
- Pretraining and finetuning large models over lots of data is expensive
- We can identify errors, but want to avoid re-training
- Want to fix errors one at a time

# Why Fine-grained Control?

---

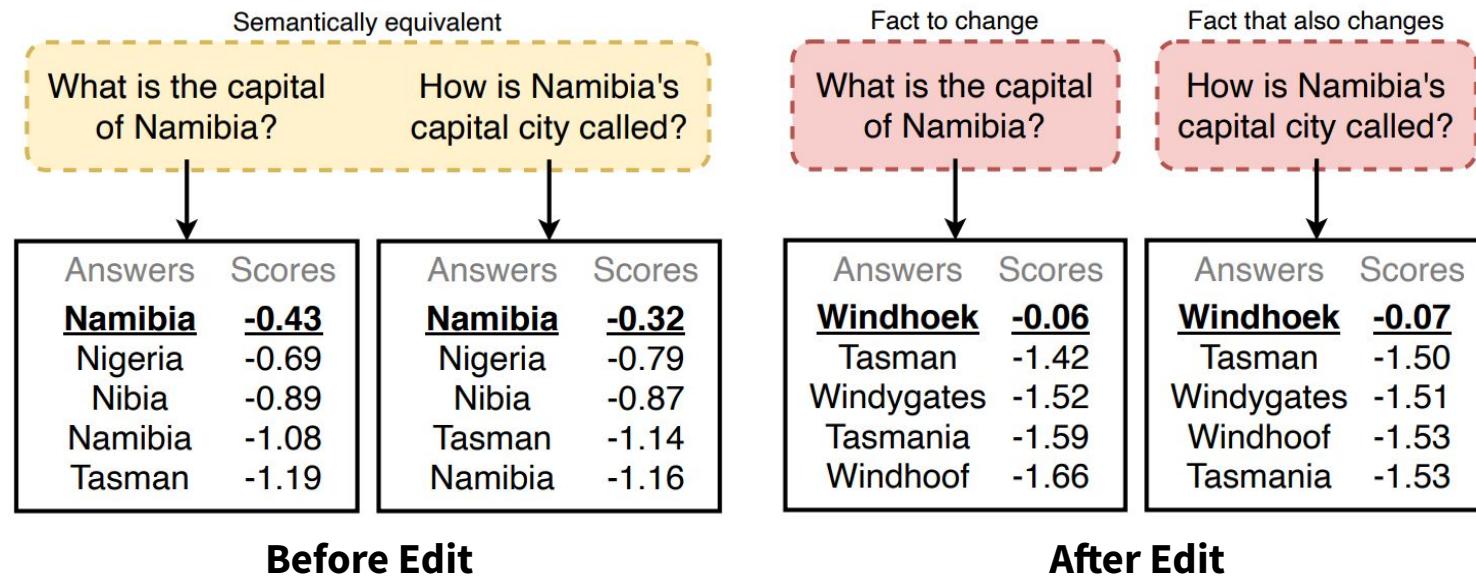
- Usual argument goes like...
- Pretraining and finetuning large models over lots of data is expensive
- We can identify errors, but want to avoid re-training - **would this even work?**
- Want to fix errors one at a time

# Roadmap

---

Goal	Research
Fine-grained Control	<b>Model Editing</b>
Deleting Sensitive Info	Unlearning
Scalable Oversight	Easy-to-Hard Generalization

# Model Editing - Background



(De Cao et al., 2020)

# Model Editing - Background

---

- A word on terminology
- Editing = updating = revising
- What are we editing?
- “Fact” and “knowledge” seem awkward if information isn’t true
- “Belief” feels appropriately weaker
- Dennett (1995) characterizes a *belief* as:

An informational state decoupled from any motivational state

- More to say on criteria for belief... (Dretske, 1981)
- This problem has been called *belief revision* in CS+philosophy since 1979 (Doyle)

# Model Editing

---

## Do Language Models Have Beliefs? Methods for Detecting, Updating, and Visualizing Model Beliefs

**Peter Hase<sup>1,2</sup>**  
**Zornitsa Kozareva<sup>1</sup>**

**Mona Diab<sup>1</sup>**  
**Veselin Stoyanov<sup>1</sup>**

<sup>1</sup>Meta AI    <sup>2</sup>UNC Chapel Hill

**Asli Celikyilmaz<sup>1</sup>**  
**Mohit Bansal<sup>2</sup>**

**Xian Li<sup>1</sup>**  
**Srinivasan Iyer<sup>1</sup>**

{peter, mbansal}@cs.unc.edu

{mdiab, aslic, xianl, zori, ves, sviyer}@fb.com

EACL 2023

# Model Editing

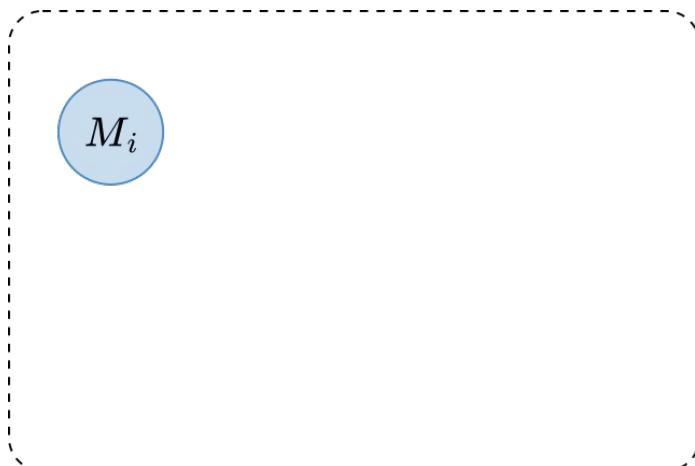
---

Two main questions:

1. How should we evaluate model edits?
2. Can we continually update a model with new beliefs?

# Model Editing

How should we evaluate model edits?

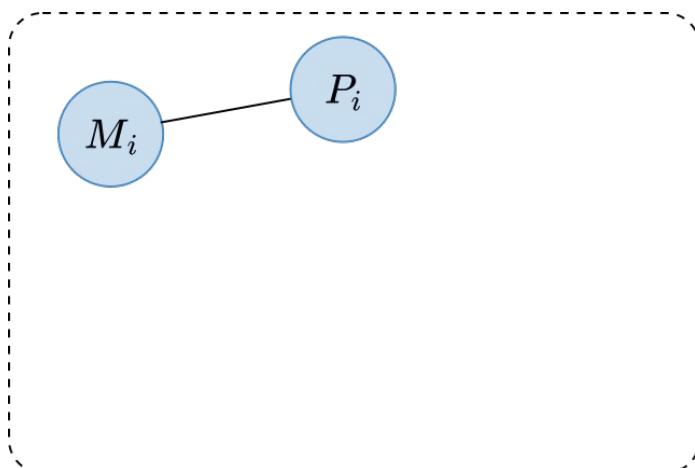


$M$  (Main Input)

: A viper is a vertebrate.  
Vipers are vertebrates.

# Model Editing

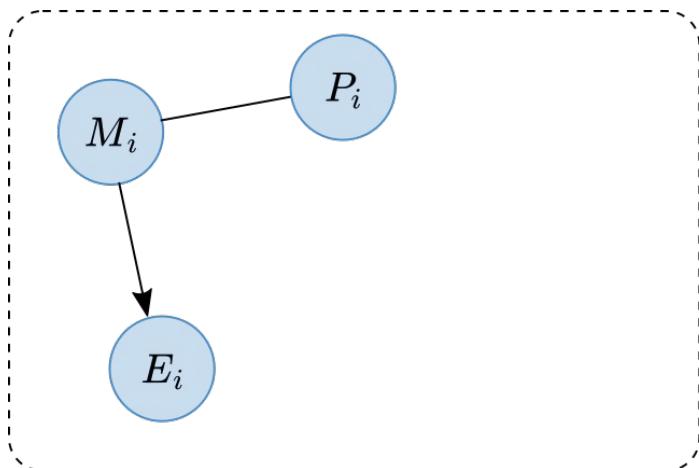
How should we evaluate model edits?



- $M$  (Main Input) : A viper is a vertebrate.  
 $P$  (Paraphrase Data) : Vipers are vertebrates.

# Model Editing

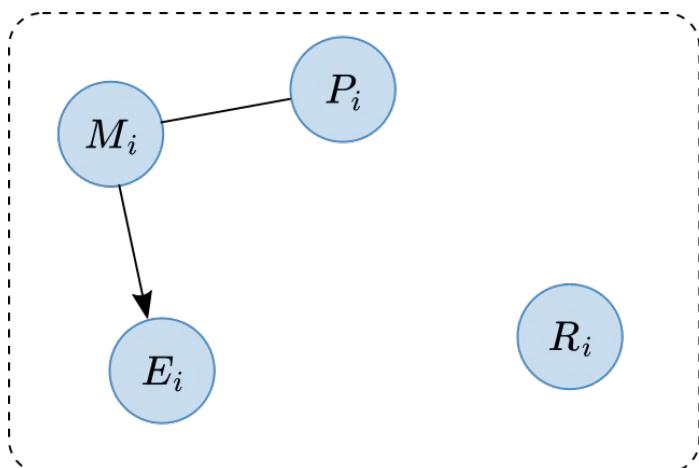
How should we evaluate model edits?



- |                       |   |                          |
|-----------------------|---|--------------------------|
| $M$ (Main Input)      | : | A viper is a vertebrate. |
| $P$ (Paraphrase Data) | : | Vipers are vertebrates.  |
| $E$ (Entailed Data)   | : | A viper has a brain.     |

# Model Editing

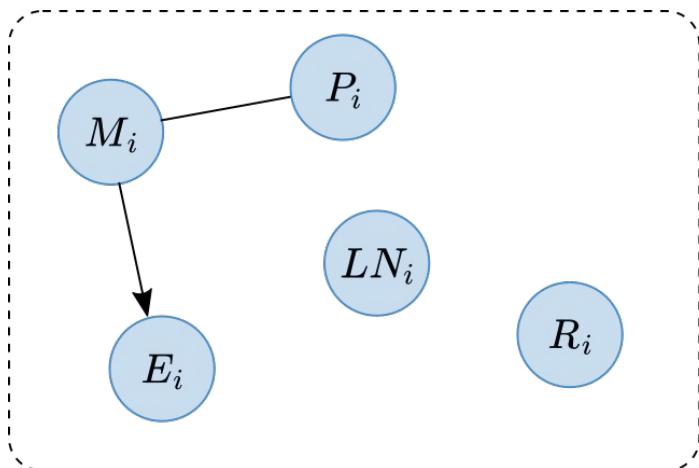
How should we evaluate model edits?



- |                       |                            |
|-----------------------|----------------------------|
| $M$ (Main Input)      | : A viper is a vertebrate. |
| $P$ (Paraphrase Data) | : Vipers are vertebrates.  |
| $E$ (Entailed Data)   | : A viper has a brain.     |
| $R$ (Random Data)     | : Chile is a country.      |

# Model Editing

How should we evaluate model edits?



- M* (Main Input) : A viper is a vertebrate.  
*P* (Paraphrase Data) : Vipers are vertebrates.  
*E* (Entailed Data) : A viper has a brain.  
*R* (Random Data) : Chile is a country.  
*LN* (Local Neutral Data) : A viper is venomous.

# Model Editing

---

Can we continually update a model with new beliefs?

Methods:

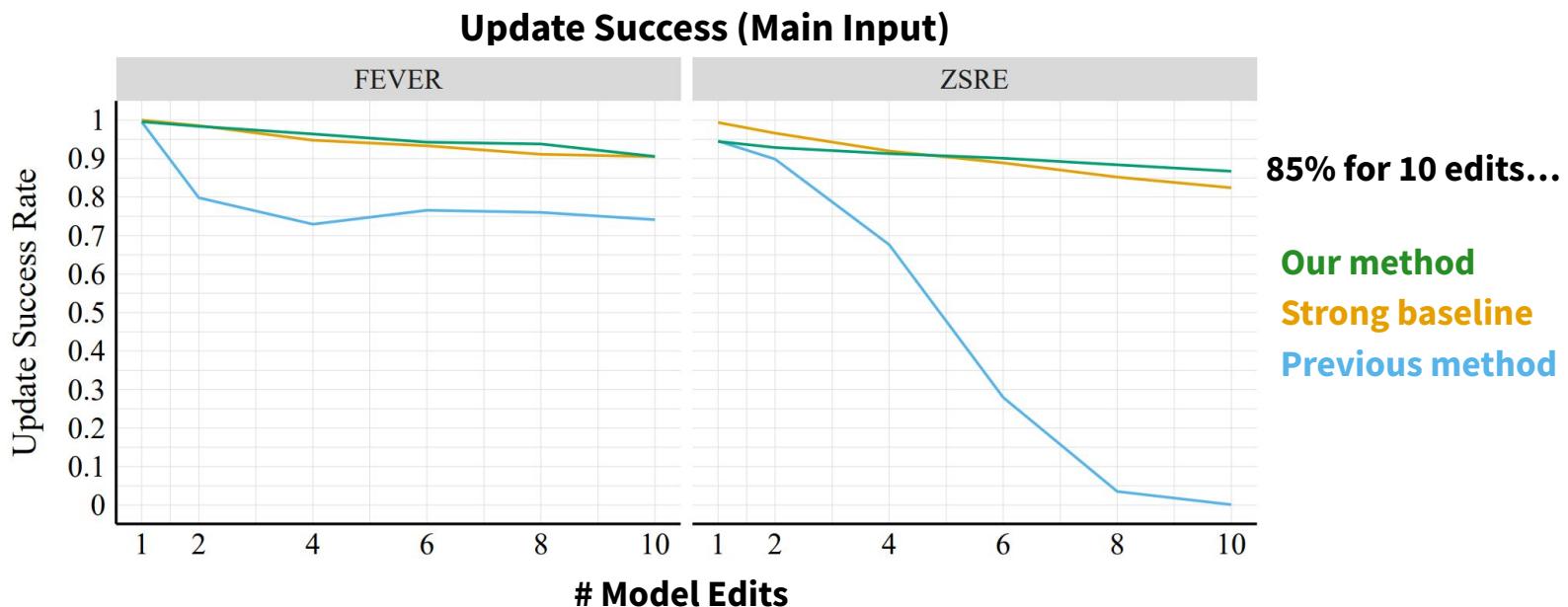
- Edit model weights using hypernetwork
- Off-the-shelf optimizers as baseline (finetuning)

Data:

- FEVER - T/F statements
- ZSRE - Closed-book QA

# Model Editing

Continual belief updating - **hypernetwork weight editing** on **t5-base**



# Model Editing

---

Continual belief updating - **hypernetwork weight editing** on **t5-base**

Main conclusions:

1. Harder to **fix errors** than to **create them**
2. Harder to retain performance on **local data** than **random data**
3. Harder to generalize to **entailed data** than **paraphrases**
4. Updates greatly **improve consistency** (model was wrong in inconsistent ways)

# Model Editing

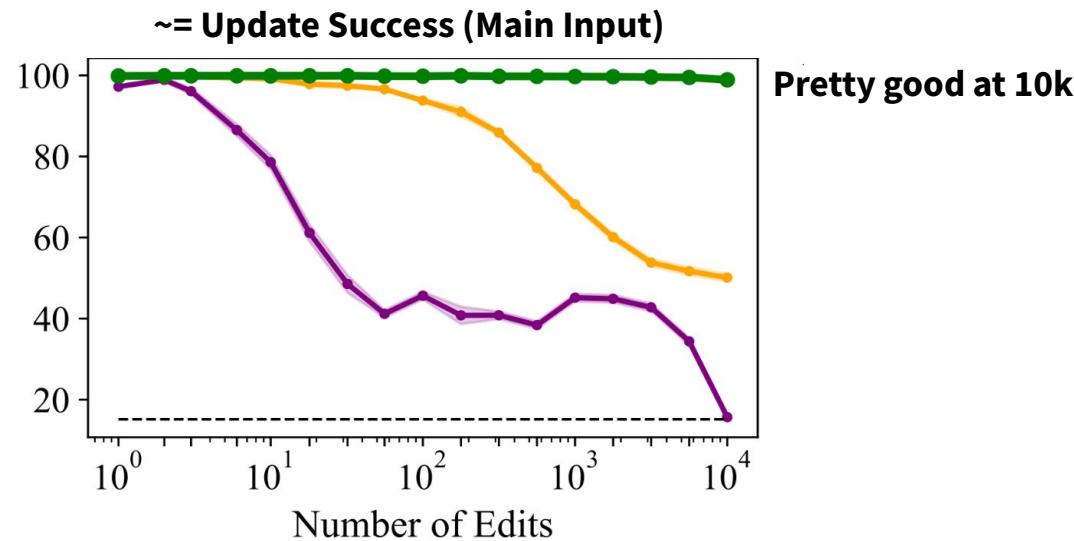
---

Continual belief updating - **hypernetwork weight editing** on **t5-base**

- Since 2021...

# Model Editing - Recent Work

Continual belief updating - **MEMIT weight editing** on **GPT-J** ([Meng et al., 2022](#))



# Model Editing - Recent Work

---

Continual belief updating - **MEMIT weight editing** on **GPT-J** ([Meng et al., 2022](#))

**No entailment evaluation**

Entailment is hard to measure

- We adapted data from LeapOfThought ([Talmor et al., 2020](#)), but it's a little synthetic

Recent work:

1. Evaluating the Ripple Effects of Knowledge Editing in Language Models ([Cohen et al., 2023](#))
2. MQuAKE: Assessing Knowledge Editing in Language Models via Multi-Hop Questions ([Zhong et al., 2023](#))

# Model Editing

---

## Conclusions

Model editing is increasingly useful for fine-grained control...

...but needs stronger evals focusing on **fixing errors** and **measuring entailment**

# Roadmap

---

Goal	Research
Fine-grained Control	Model Editing
<b>Deleting Sensitive Info</b>	Unlearning
Scalable Oversight	Easy-to-Hard Generalization

# Definitions + Motivation

---

- Refer to ethically sensitive information as *sensitive information*
- In pretraining, LLMs learn...
  - Personal information
  - Copyrighted information
  - Knowledge that could be used to harm others  
(e.g. instructions for crimes, CBRN weapons)
  - Various toxic beliefs/content
  - Factual information that has gone out of date (could *become* misinfo)

# Definitions + Motivation

---

- *Deleting* information from LLMs is underdefined
- Finetuning (RLHF, SFT, safety training, etc.) appears to hide rather than remove sensitive information ([Zou et al., 2023](#))
- **This is a model editing problem** – update individual beliefs in a model

# Roadmap

---

Goal	Research
Fine-grained Control	Model Editing
Deleting Sensitive Info	<b>Unlearning</b>
Scalable Oversight	Easy-to-Hard Generalization

# Unlearning in LLMs

---

CAN SENSITIVE INFORMATION BE DELETED FROM  
LLMs? OBJECTIVES FOR DEFENDING AGAINST  
EXTRACTION ATTACKS

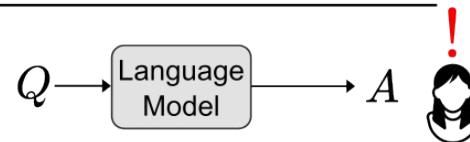
**Vaidehi Patil\***    **Peter Hase\***    **Mohit Bansal**  
UNC Chapel Hill  
`{vaidehi, peter, mbansal}@cs.unc.edu`

ICLR 2024  
*Spotlight*

# Unlearning in LLMs

---

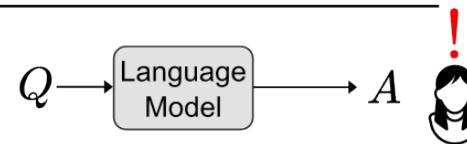
## 1. Notice sensitive info



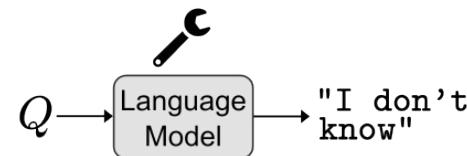
# Unlearning in LLMs

---

## 1. Notice sensitive info

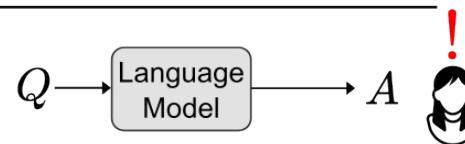


## 2. Deletion defense

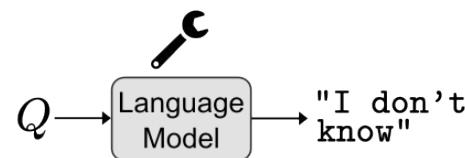


# Unlearning in LLMs

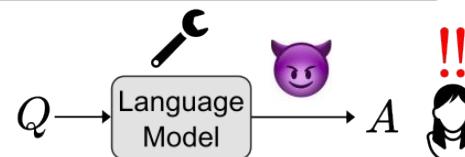
## 1. Notice sensitive info



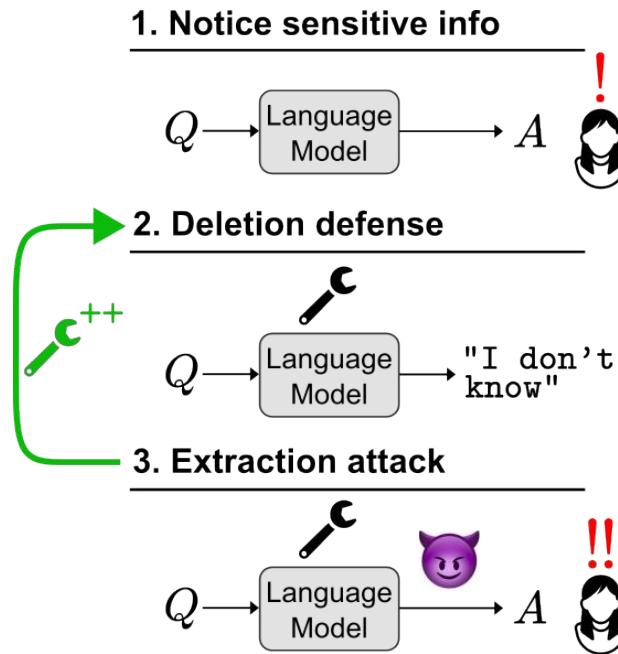
## 2. Deletion defense



## 3. Extraction attack



# Unlearning in LLMs



# Unlearning in LLMs

---

**Threat model** - “is info truly deleted?”

- Adversary seeks answer  $A$  to question  $Q$
- Given a model, adversary obtains candidate set  $C$  of size  $B$  (budget)
- **Adversary succeeds if  $A$  is in  $C$**

Why  $B$  attempts?

1. Password attempts
2. Parallel pursuit
3. Verification by data owner (or auditor)

Previous frameworks focused on formal guarantees of similarity to retrained model  
([Cao and Yang, 2015](#))

# Unlearning in LLMs

---

**Deletion metric** - How good is defense?

$$\arg \min_{\mathcal{M}^*} \text{AttackSuccess}@B(\mathcal{M}^*) + \lambda \text{Damage}(\mathcal{M}^*, \mathcal{M})$$

---

Need to balance:

1. **AttackSuccess**: whether answer is in candidate set
2. **Damage**: change in model accuracy for other questions

# Unlearning in LLMs

---

**Applying model editing for deletion** - This is the defense

Tasks/data:

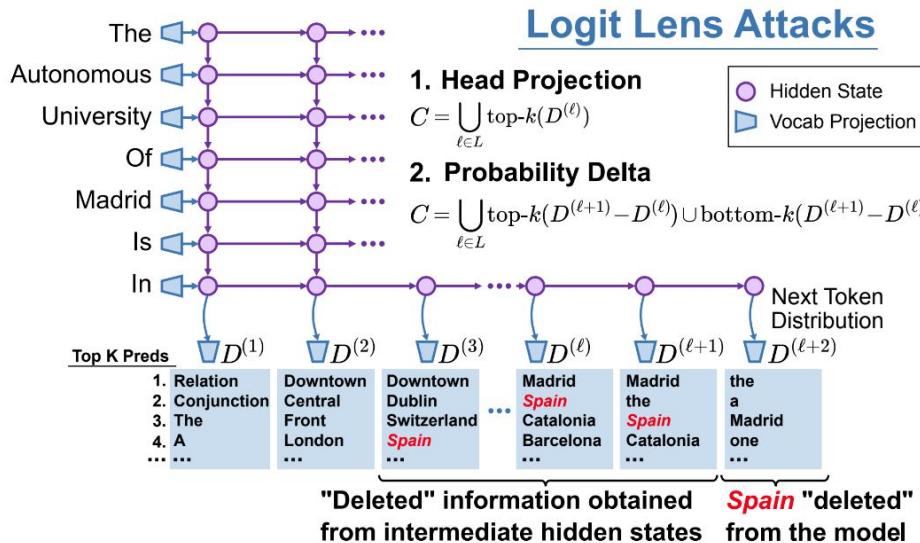
- Our testbed is factual information (CounterFact and ZSRE)
- Filter to questions with single-token answers, known by GPT-J model we attack

Model editing:

- *Optimizers:*
  - AdamW, ROME, MEMIT
- *Objectives:*
  - Error Injection → say something else
  - Fact Erasure → minimize answer probability
  - Empty Response → say “I don’t know”

# Unlearning in LLMs

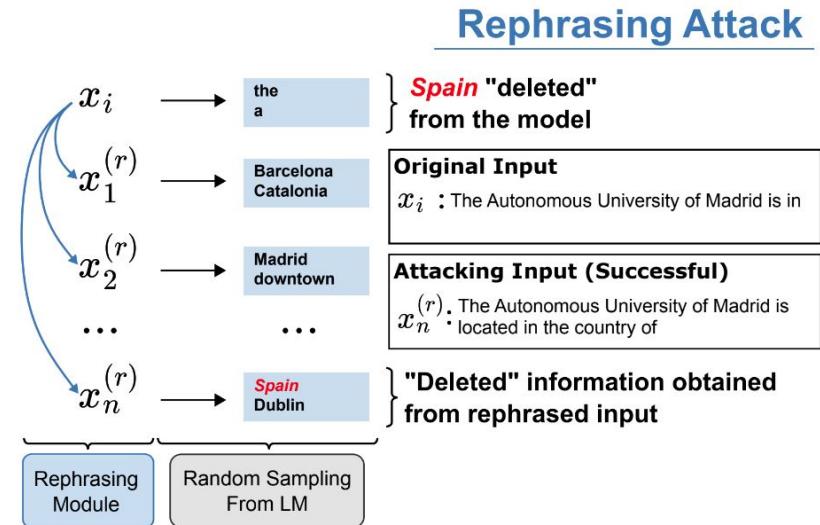
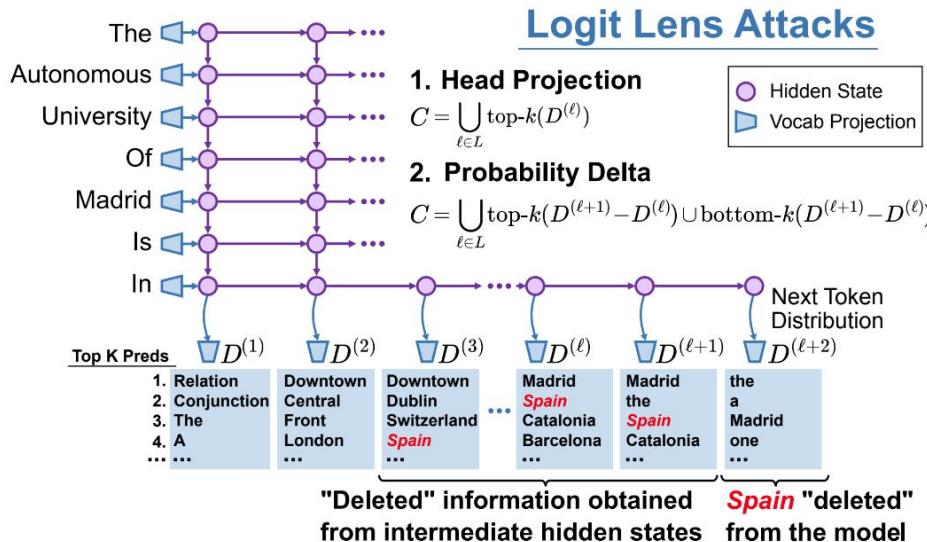
## Attacking models for “deleted” info



## Whitebox Attack

# Unlearning in LLMs

## Attacking models for “deleted” info



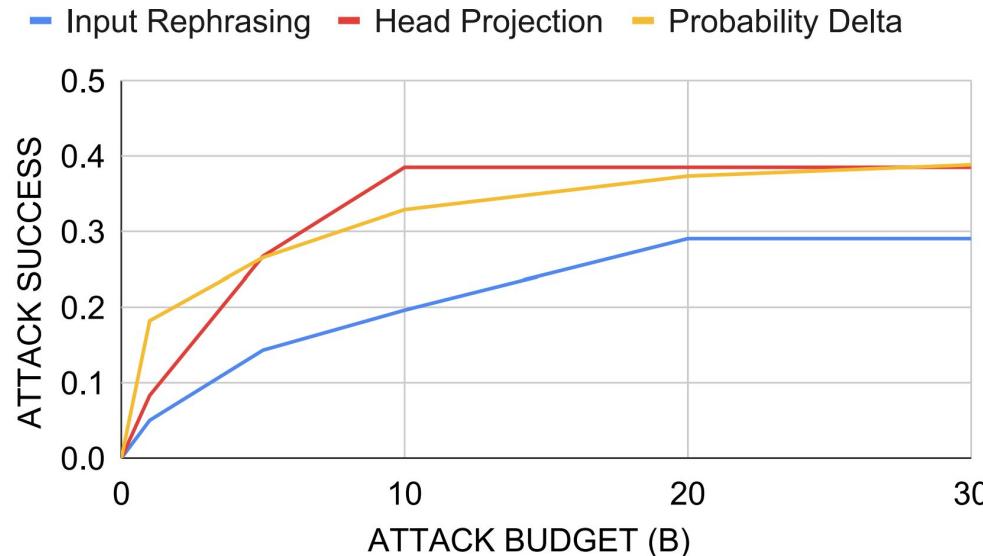
## Whitebox Attack

## Blackbox Attack

# Unlearning in LLMs

## Results

1. 38% attack success at  $B=10$  for GPT-J facts deleted by ROME + Empty Response



# Unlearning in LLMs

---

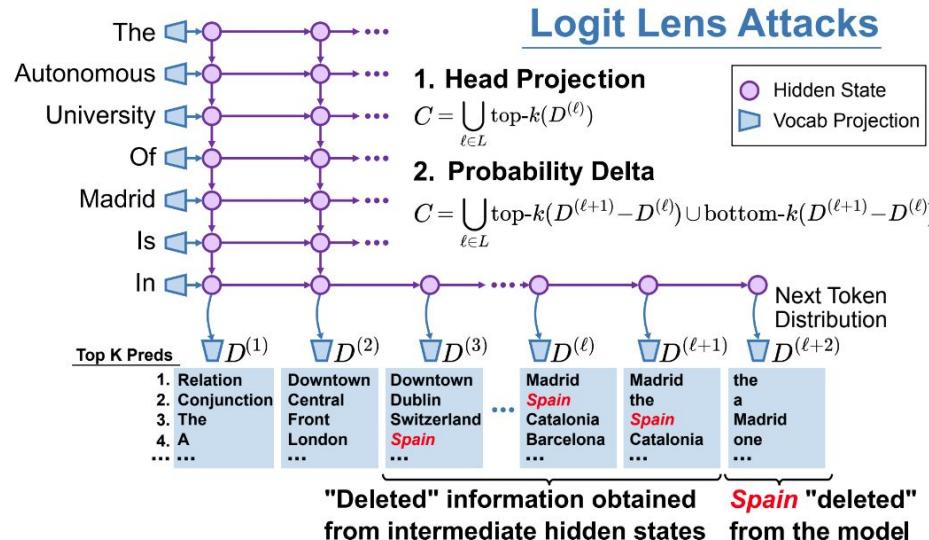
## Improving Defense Methods

- Blackbox defense reduces to paraphrase + adversarial robustness
- Whitebox defense: *delete information wherever it appears in model*

# Unlearning in LLMs

## Improving Defense Methods

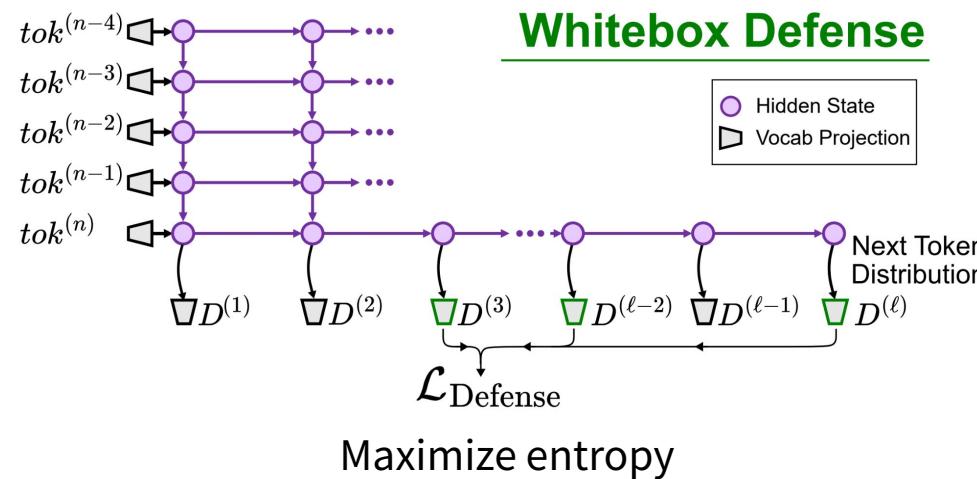
- Blackbox defense reduces to paraphrase + adversarial robustness
- Whitebox defense: *delete information wherever it appears in model*



# Unlearning in LLMs

## Improving Defense Methods

- Blackbox defense reduces to paraphrase + adversarial robustness
- Whitebox defense: *delete information wherever it appears in model*



# Unlearning in LLMs

---

## Results

1. Up to 38% attack success for GPT-J facts deleted by ROME+Empty Response (B=10)

With whitebox defense

2. “Foreseen” whitebox attack: **37% → 1.7%**
3. “Unforeseen” whitebox attack: **38% → 2.4%**
4. Blackbox attack rate seems unchanged

See paper for blackbox defense

# Unlearning in LLMs

---

## Conclusions

- Want to delete sensitive information under **adversarial extraction attacks**
- **Whitebox defenses help**, but safety standards for deletion will vary

# Roadmap

---

Goal	Research
Fine-grained Control	Model Editing
Deleting Sensitive Info	Unlearning
<b>Scalable Oversight</b>	Easy-to-Hard Generalization

# Scalable Oversight

---

It is challenging to train models when outputs are difficult to evaluate  
(Amodei et al., 2016)

# Connection to Easy-to-Hard Generalization

---

- How will models generalize from easy train data to hard test data?
  - Easy = easy to label
  - Hard = hard to label
- Why does this matter?
- We want to supervise models to answer questions for us in specialized domains
- But...

*Gathering data in specialized domains is expensive and difficult*

≈

*Scalable oversight could be very challenging*

# Connection to Easy-to-Hard Generalization

---

- How will models generalize from easy train data to hard test data?
  - Easy = easy to label
  - Hard = hard to label
- Why does this matter?

If easy-to-hard generalization is good

Then scalable oversight is not difficult

(the solution is to train on easy data)

# Roadmap

---

Goal	Research
Fine-grained Control	Model Editing
Deleting Sensitive Info	Unlearning
Scalable Oversight	<b>Easy-to-Hard Generalization</b>

# Easy-to-Hard Generalization

---

## The Unreasonable Effectiveness of Easy Training Data for Hard Tasks

**Peter Hase<sup>1,2</sup>**    **Mohit Bansal<sup>2</sup>**    **Peter Clark<sup>1</sup>**    **Sarah Wiegreffe<sup>1</sup>**

<sup>1</sup>Allen Institute for AI    <sup>2</sup>UNC Chapel Hill

{peter, mbansal}@cs.unc.edu, peterc@allenai.org, wiegrefesarah@gmail.com

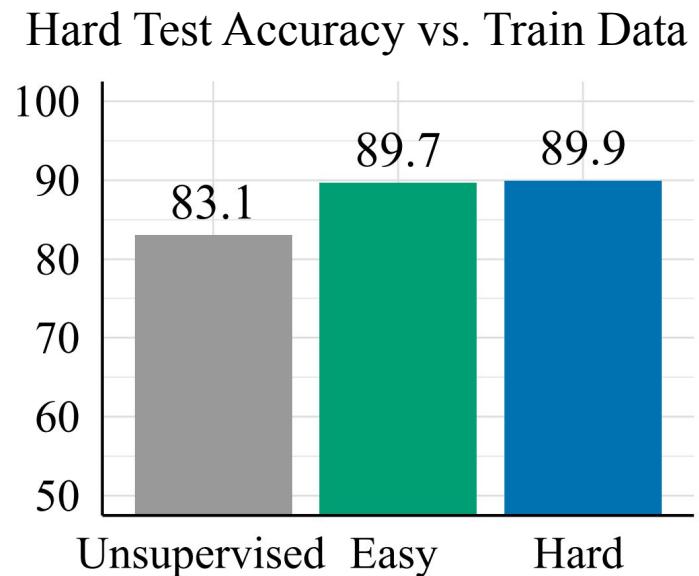
arXiv 2024

# Easy-to-Hard Generalization

We introduce the **Supervision Gap Recovered (SGR)**

$$\frac{89.7 \text{ Easy - Unsupervised}}{89.9 \text{ Hard - Unsupervised}} = 83.1$$

$$\text{SGR} = 97\%$$



# Easy-to-Hard Generalization

---

What can we measure?

1. Education / grade level
2. Expert rating
3. Required cognitive skills
4. Question length
5. Answer length
6. Compositional reasoning steps
7. Model-based hardness  
(datapoint loss w/ weaker LM)

# Easy-to-Hard Generalization

---

What can we measure?

1. Education / grade level
2. Expert rating
3. Required cognitive skills
4. Question length
5. Answer length
6. Compositional reasoning steps
7. Model-based hardness  
(datapoint loss w/ weaker LM)

Data we use...

- 3rd grade to college STEM
- Compositional reasoning in math and general-knowledge trivia

# Easy-to-Hard Generalization

---

<b>ARC</b>	<b>MMLU-STEM-5</b>	<b>StrategyQA</b>	<b>GSM8k</b>
$n = 4521$	$n = 1746$	$n = 2290$	$n = 8792$
Grade Level (3-8)	Grade Level (HS vs. College)	Grade Level	Grade Level
Difficulty Score (1-3)	Difficulty Score	Difficulty Score	Difficulty Score
Bloom Skill (1-5)	Bloom Skill	Bloom Skill	Bloom Skill
Question Num. Words	Question Num. Words	Question Num. Words	Question Num. Words
Answer Num. Chars	Answer Num. Chars	Answer Num. Chars	Answer Num. Chars
Num. Reasoning Steps	Num. Reasoning Steps	Num. Reasoning Steps	Num. Reasoning Steps
MDL	MDL	MDL	MDL

4 datasets  
 6 human hardness measures  
 1 model-based measure

# Easy-to-Hard Generalization

We need to define  
*easy* and *hard*

Hardness Measure	Easy	Medium	Hard
ARC Grade	3-5	6-7	8
ARC Expert Difficulty	1	2	3
ARC Bloom Skill	1-2	3	4-5
MMLU Grade	High School		College
StrategyQA Reasoning	1-2	3	4-5
GSM8k Reasoning	2-3	4-5	6-11
Question Length, Answer Length, MDL	30th percentile	...	70th percentile

# Easy-to-Hard Generalization

---

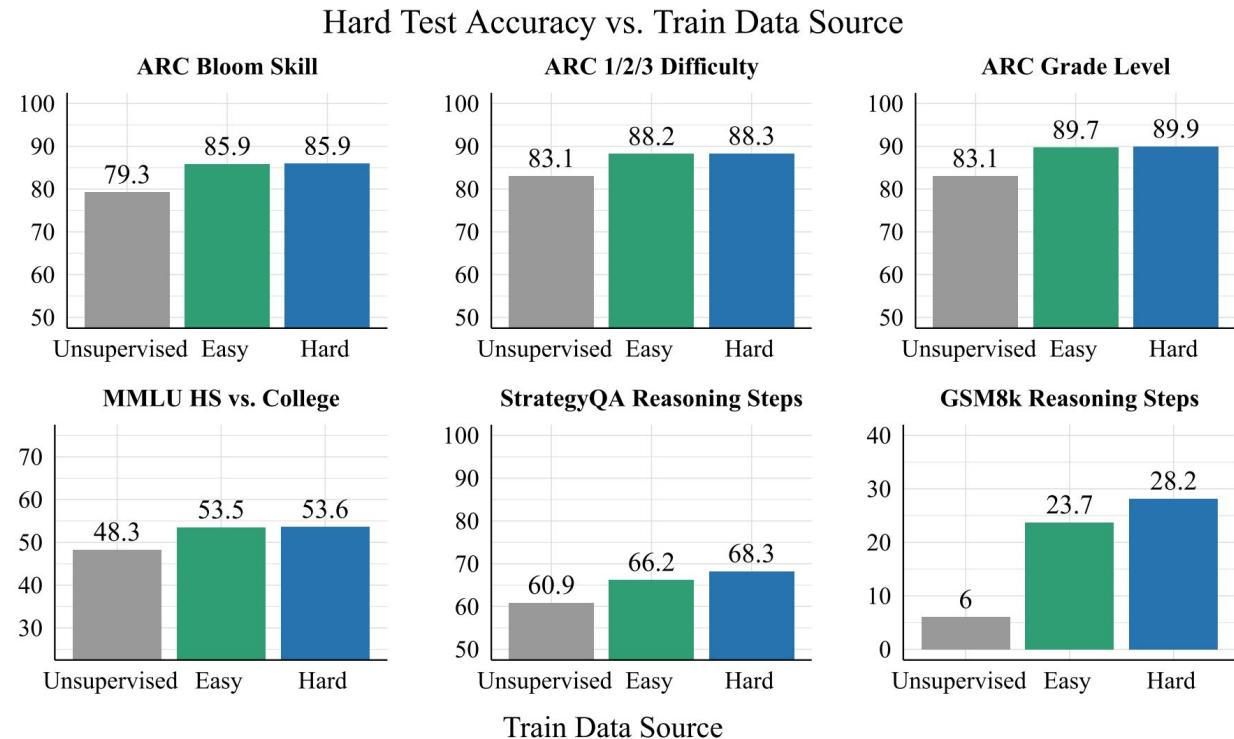
## Experiment Setup

- Models
  - Llama-2 models (7b, 13b, 70b)
  - Mixtral-8x7b, Llama-2 70b chat, Qwen-72b
- Training Methods
  - ICL,  $n \leq 10$
  - Linear probing,  $n = 160$
  - QLoRA,  $n = 160$
- Unsupervised Baseline
  - Zero-shot prompted model
- Results averaged over 5 random seeds

# Easy-to-Hard Generalization

**The Supervision Gap  
Recovered is 70-100%  
across hardness measures**

Llama-2-70b  
ICL with  $k \leq 10$



# Easy-to-Hard Generalization

---

- Previous experiments used equal amounts of cleanly labeled easy and hard data
- This is actually unrealistic
- Hard data is *more expensive* and *labels are noisier*
- What if hard data is  $2x$  as costly to collect?
- What if hard data is  $2x$  as noisy as easy data?
  - $2x$  as much high school data as college data in MMLU
  - Expert error rate in GPQA (grad questions) more than  $2x$  expert error rate in MMLU (undergrad questions)

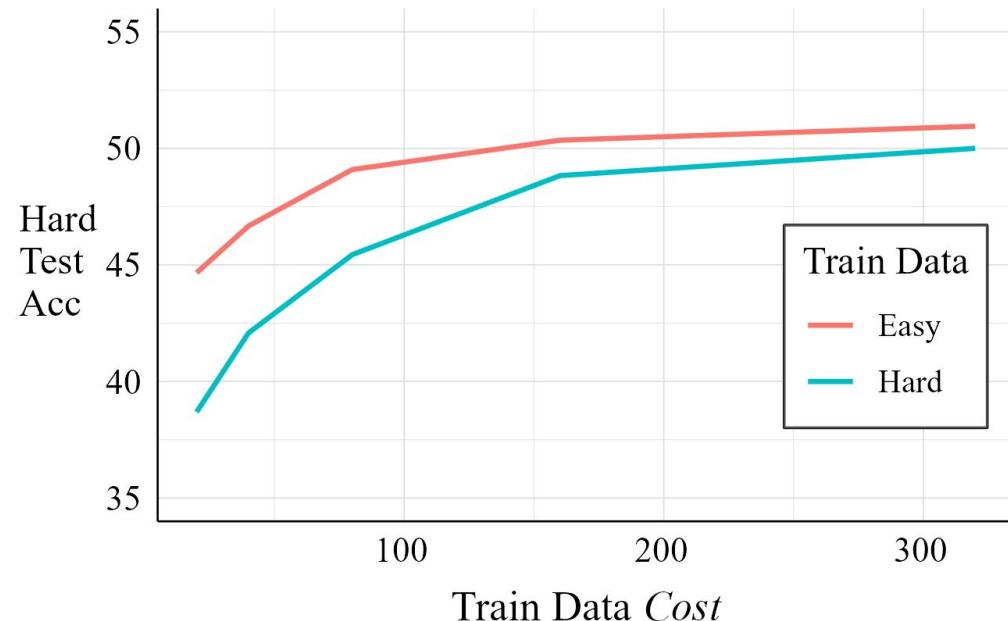
# Easy-to-Hard Generalization

Easy training data can be better than hard data

Llama-2-70b with linear probe

Testing on MMLU-STEM-5

What if Hard Data Is 2x Costlier to Label?

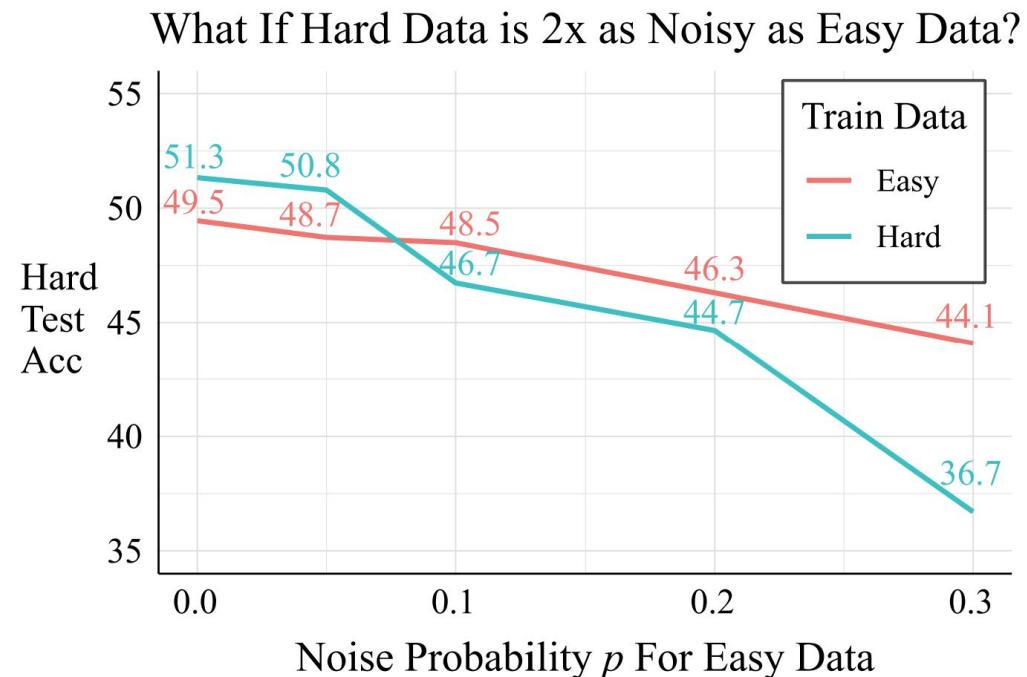


# Easy-to-Hard Generalization

Easy training data can be better than hard data

Llama-2-70b with linear probe

Testing on MMLU-STEM-5



# Easy-to-Hard Generalization

---

## Conclusions

- Easy supervision is **70-100% as good** as hard supervision
- Collecting easy data **can be better than hard data** due to cost/noise

# Final Summary

---

- Fine-grained Control
  - Motivation: **We want to fix individual model errors** (both factual and moral) over time
  - Result: **Model editing is increasingly useful for fine-grained control**
  - Paper: “Do Language Models Have Beliefs? Methods for Detecting, Updating, and Visualizing Model Beliefs” ([Hase et al., 2021](#))
- Editing for Unlearning
  - Motivation: **We want to delete ethically sensitive information from LLMs**
  - Result: **Truly deleting information from LLMs is a tractable but difficult problem**
  - Paper: “Can Sensitive Information Be Deleted From LLMs? Objectives for Defending Against Extraction Attacks” ([Patil + Hase et al., 2023](#))
- Scalable Oversight
  - Motivation: **Want models to be knowledgeable in domains where labeling data is hard**
  - Result: **Easy-to-hard generalization is surprisingly good in LLMs**
  - Paper: “The Unreasonable Effectiveness of Easy Training Data for Hard Tasks” ([Hase et al., 2024](#))

# Thank You!

---

PDFs + code: <https://peterbhase.github.io/research/>

Contact Info:

Peter Hase, UNC Chapel Hill

[peter@cs.unc.edu](mailto:peter@cs.unc.edu)

<https://peterbhase.github.io>

# Other Work

---

2020

Evaluating Explainable AI: Which Algorithmic Explanations Help Users Predict Model Behavior?

Leakage-Adjusted Simulability: Can Models Generate Non-Trivial Explanations of Their Behavior in Natural Language?

2021

When Can Models Learn From Explanations? A Formal Framework for Understanding the Roles of Explanation Data

The Out-of-Distribution Problem in Explainability and Search Methods for Feature Importance Explanations

Do Language Models Have Beliefs? Methods for Detecting, Updating, and Visualizing Model Beliefs

2022

VisFIS: Visual Feature Importance Supervision with Right-for-the-Right-Reason Objectives

2023

Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models

Can Sensitive Information Be Deleted From LLMs? Objectives for Defending Against Extraction Attacks

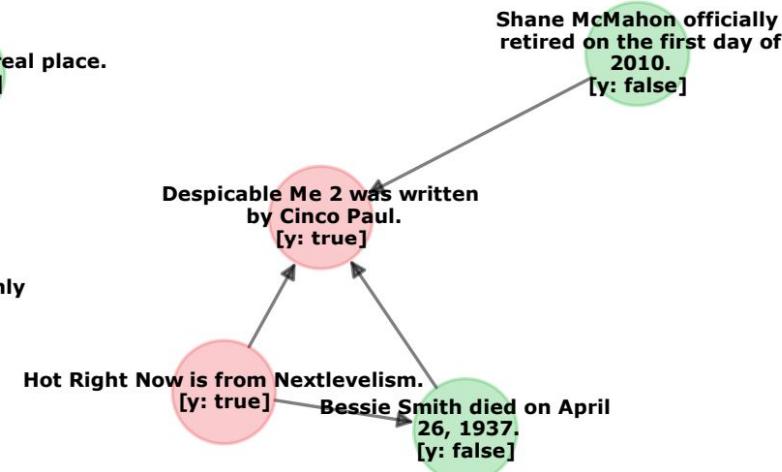
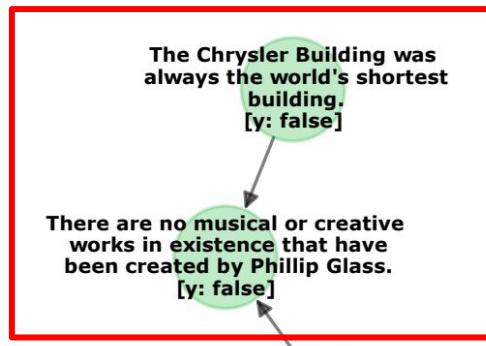
**Interpretability + Model Control**

# Model Editing

---

- What else can we do with model editing?
- Let's look at *connections* between model beliefs
- Beliefs are connected when changing one leads the other to change
  - Update belief A → observe a change in belief B

# Model Editing

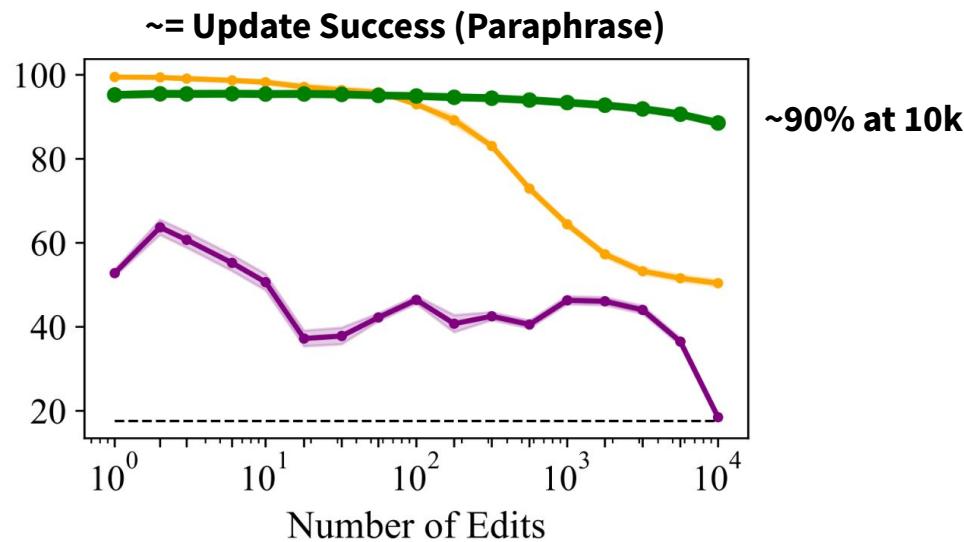


Editing not very precise...

...or t5-base knowledge not structured very logically

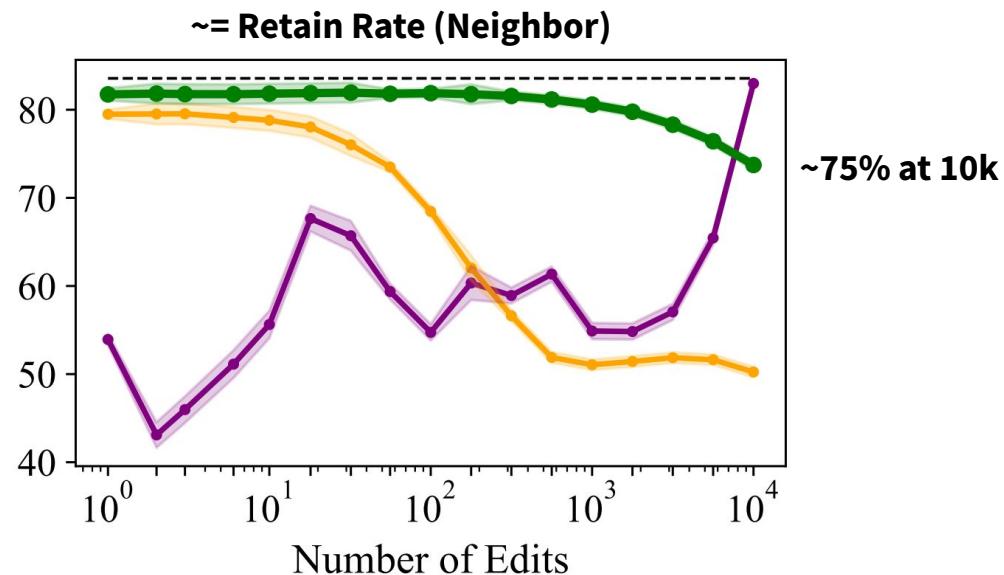
# Model Editing - Recent Work

- Continual belief updating - **MEMIT weight editing** on **GPT-J** ([Meng et al., 2022](#))



# Model Editing - Recent Work

- Continual belief updating - **MEMIT weight editing** on **GPT-J** ([Meng et al., 2022](#))



# Non-idealized Belief Revision

---

- So this is an old problem (Doyle, 1979), but LMs might require new treatment
  - Do LMs have a **single set of beliefs**?
  - Want **complete corrigibility** (i.e. complete deference to updates)
  - Models can express uncertainty in language or via probabilities
- *Non-idealized* belief revision
  - LMs **not logically omniscient**
  - **Limited compute** applied to belief updates
- Outstanding problems
  - **Problem of priors** in Bayesianism (Raven paradox)
  - Problems in **counterfactual semantics** (semantic puzzles)
  - What is entailed when a new fact is adopted?

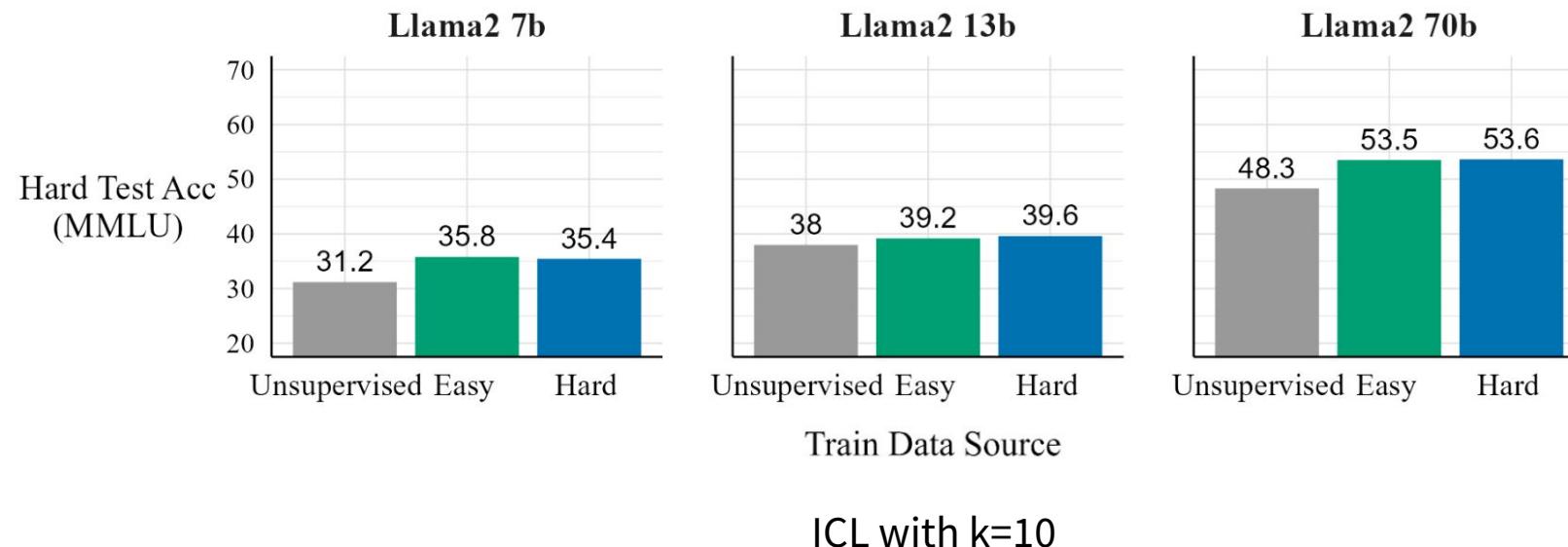
# RQ4: Scaling Model Size & Train/Test Hardness

---

- What happens as models get better?
- What happens as the train-test hardness gap grows?

# RQ4: Scaling Model Size & Train/Test Hardness

The Supervision Gap Recovered Is Similar Across Model Size

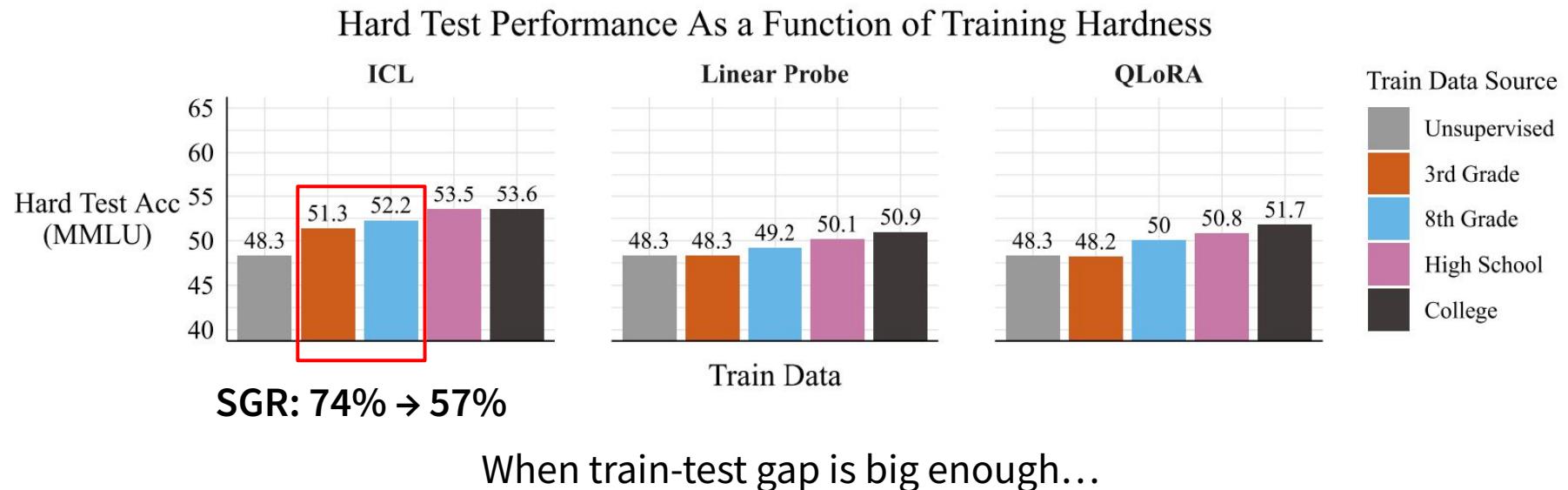


# RQ4: Scaling Model Size & Train/Test Hardness

---

- We want to increase the gap between train and test hardness
- We know that accuracy declines with test hardness  
→ fix test hardness, vary train hardness

# RQ4: Scaling Model Size & Train/Test Hardness



# RQ4: Scaling Model Size & Train/Test Hardness

---

The supervision gap recovered is robust across model scale

Easy-to-hard generalization may decline with very large train-test gaps

# Discussion

---

- Are our tasks hard enough to provide generalizable results?
  - We personally couldn't annotate MMLU
  - We consider 3rd grade to college generalization
- How are the LMs actually doing this?
  - Training elicits some latent knowledge/skill *that is hardness-invariant*
  - Not merely learning the task format
- Why not use test questions that aren't taught by the train data?
  - Wouldn't that be *true* generalization?
  - Our aim is to elicit knowledge we suspect the model may know, without knowing it ourselves – not teach something new

# Conclusion

---

## 1. How Can We Measure Hardness?

Diverse human and model-based measurements

## 2. How Good Is Easy-to-Hard Generalization?

Easy supervision is 70-100% as good as hard supervision

## 3. Cost-Benefit Tradeoffs of Easy vs. Hard Data

Collecting easy data can be better than hard data

## 4. Scaling Model Size & Train/Test Hardness

Results robust across model size

Huge train-test gaps could be an issue

# Examples

---

## MMLU College-level Computer Science Example

Question:

Assume that any assignment statement can be executed in unit time.  
If as many identical processors as needed are used, what is the minimum number of time units needed to execute the assignments

A := B + C

B := A - B

C := A \* E

D := A/F

E := B - C

F := A + B

with the same result as if the assignments were executed in the order shown?

- A) 1 - 32.3% ----- 26.3%
- B) 2 - 23.4% ----- 26.7%
- C) 3 - 22.2% ----- 27.1% **Correct!**
- D) 4 - 21.9% ----- 19.9%

**Zero-shot  
Distribution**

**vs.**      **4-shot High School  
Examples Distribution**

**High-school supervision  
improves college performance**

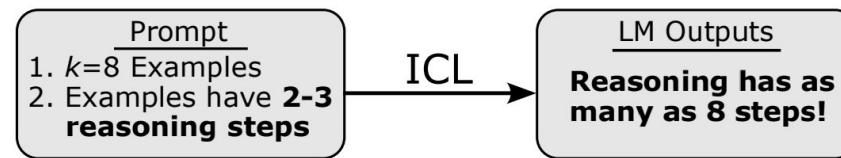
# Examples

---

## GSM8k

Test Input

LM Generated Answer



Q: John hires a driving service to get him to work each day.

His work is 30 miles away and he has to go there and back each day.

He goes to work 5 days a week for 50 weeks a year.

He gets charged \$2 per mile driven and he also gives his driver a \$150 bonus per month.

How much does he pay a year for driving?

A: John goes to work 5 days a week for 50 weeks a year.

John goes to work  $5 \times 50 = <<5*50=250>>$  250 times a year.

John pays \$2 per mile driven.

John pays  $\$2 \times 30 \times 2 = <<2*30*2=120>>$  120 for each trip.

John pays  $\$120 \times 250 = <<120*250=30000>>$  30000 for each year.

John pays \$150 bonus per month.

John pays  $\$150 \times 12 = <<150*12=1800>>$  1800 for each year.

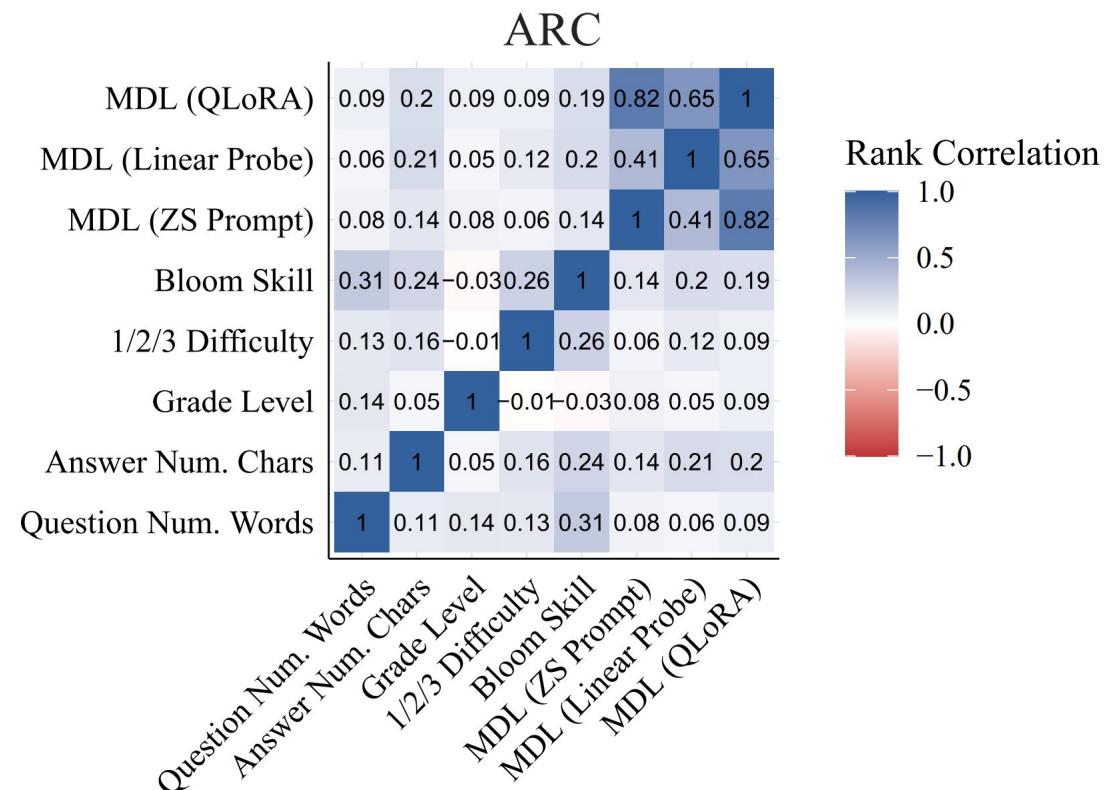
John pays  $\$30000 + \$1800 = <<30000+1800=31800>>$  31800 for each year.

So the answer is 31800

**Correct on hard problem given only easy data in prompt**

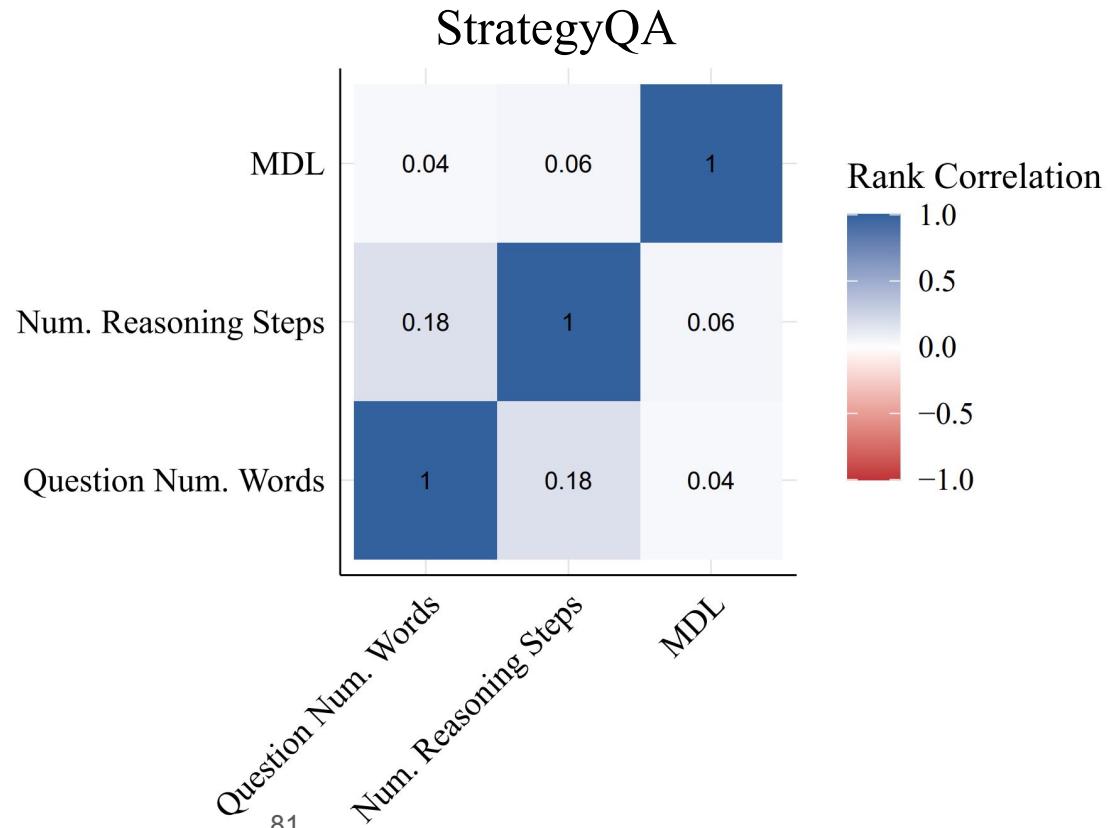
# RQ1: How Can We Measure Hardness?

**Hardness measures do not correlate strongly**

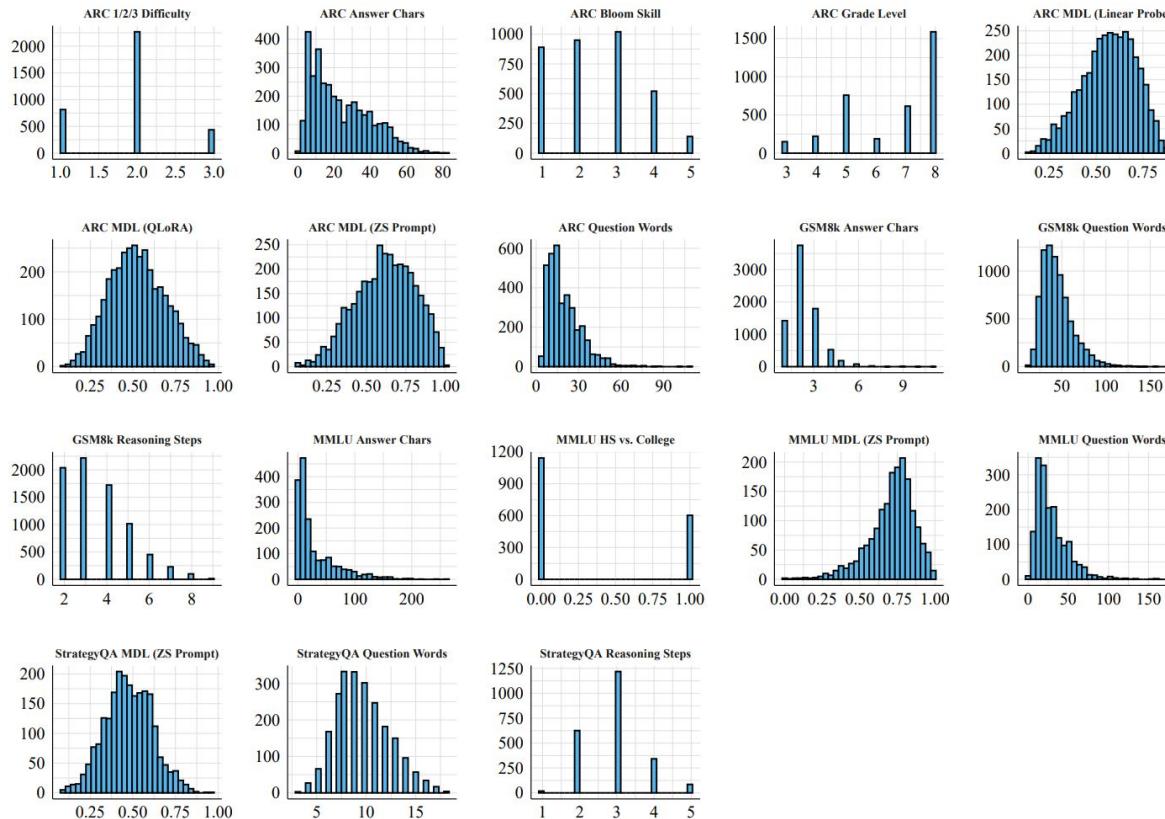


# RQ1: How Can We Measure Hardness?

**Hardness measures do not correlate strongly**



# RQ1: How Can We Measure Hardness?



# RQ1: How Can We Measure Hardness?

---

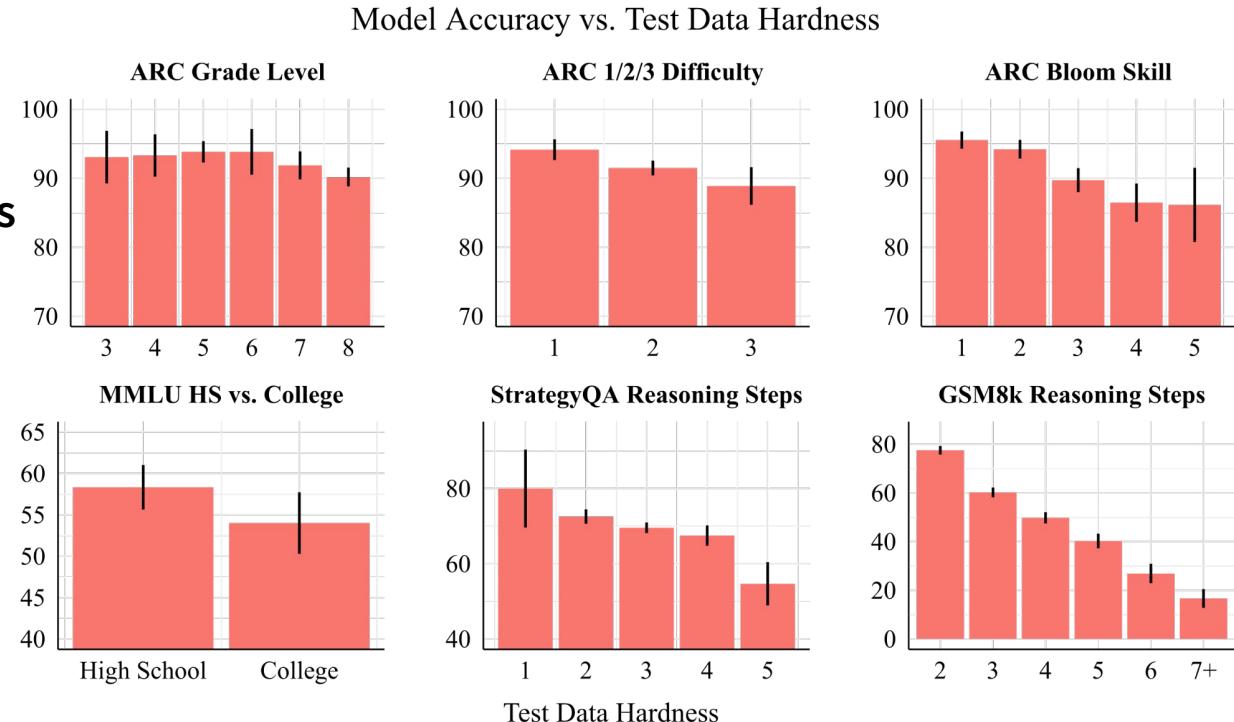
Model-based hardness: *Minimum description length* (MDL)

- (Voita and Titov, 2020)
- **How “long” does it take a model to learn the datapoint?**
- Average loss
  - Avg across  $n = \{5, 20, 80, 340, 900\}$  training points
- Training
  - Linear classifier
  - QLoRA
  - Zero-shot “MDL” with  $n = \{0\}$
- Avg over some “weaker” models
  - Falcon-7b, Mistral-7b, Persimmon-8b, Llama-1-7b

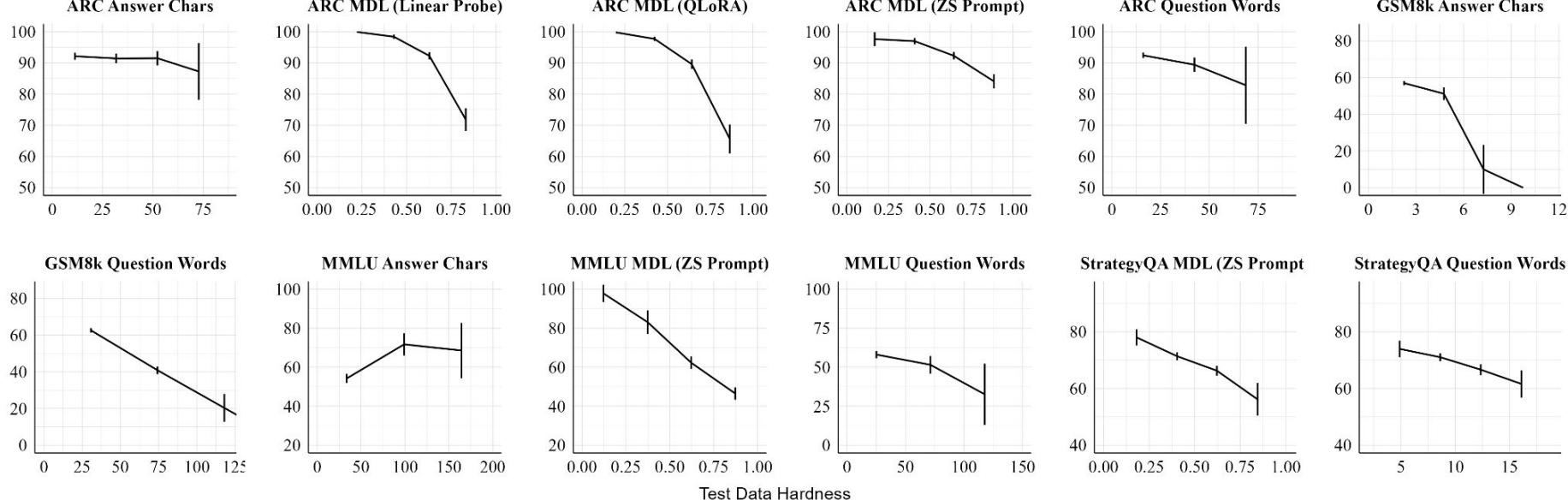
# RQ1: How Can We Measure Hardness?

**Model performance declines  
w.r.t. hardness measures**

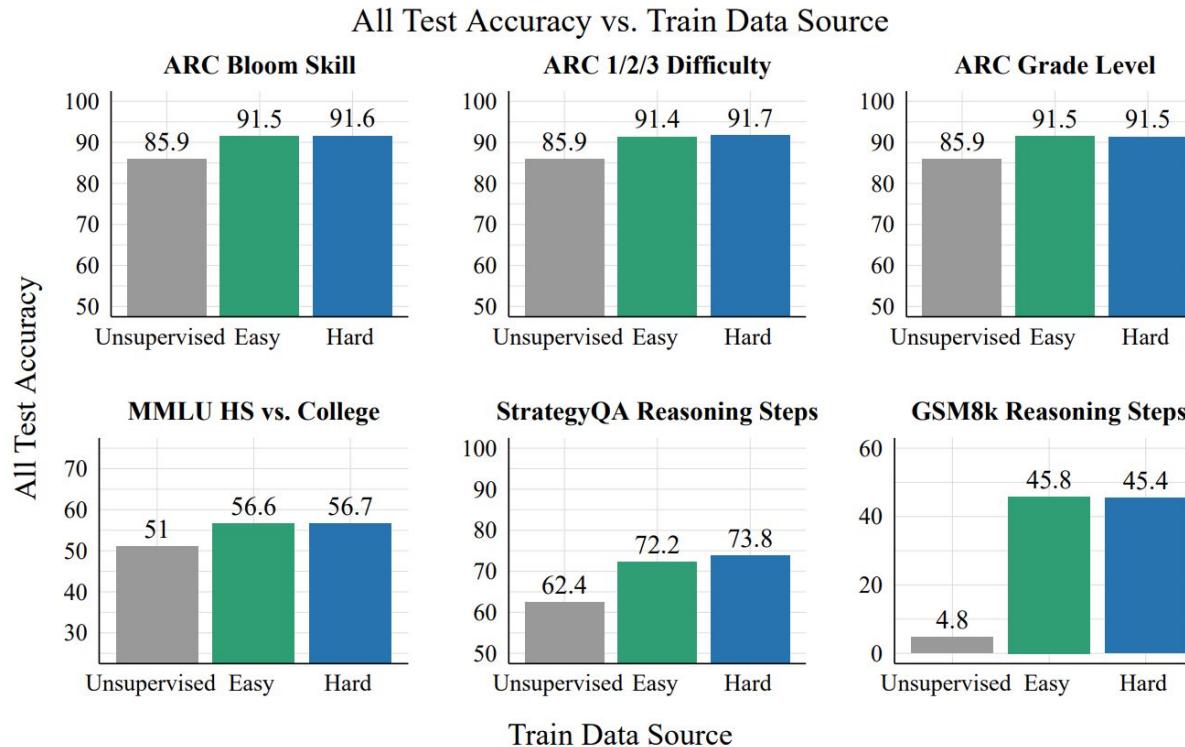
Llama-2-70b  
ICL with  $k \leq 10$



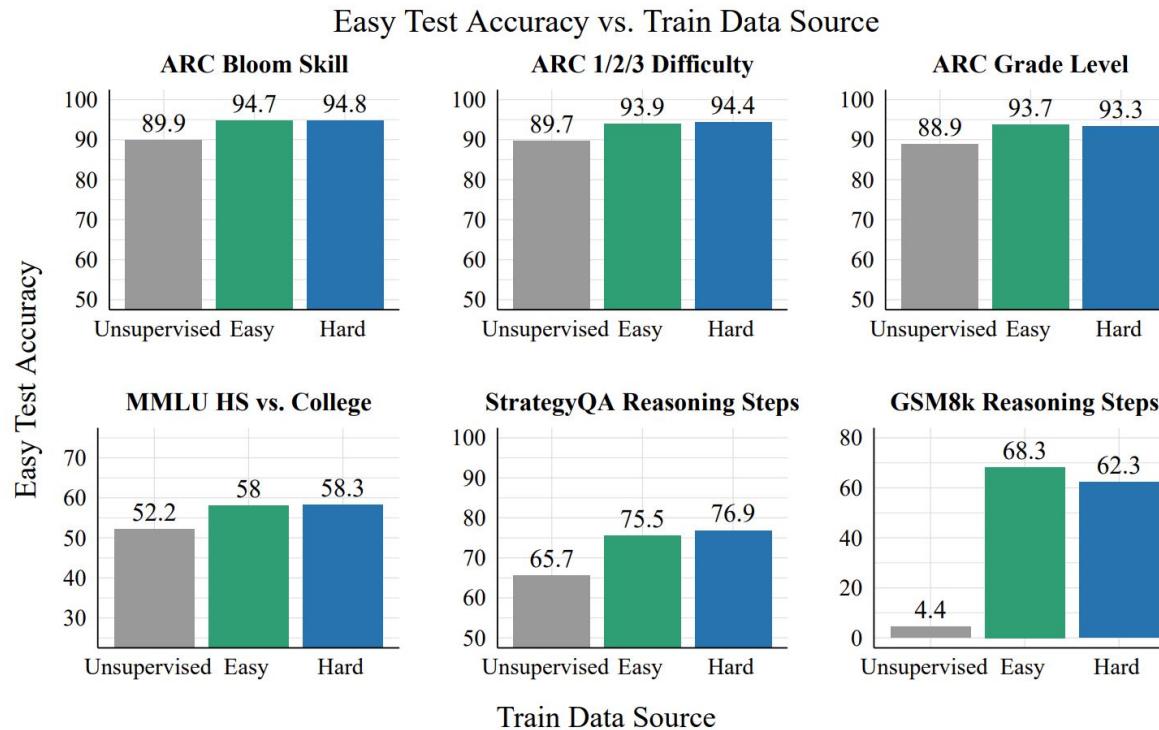
# RQ1: How Can We Measure Hardness?



# RQ2: How Good Is Easy-to-Hard Generalization?

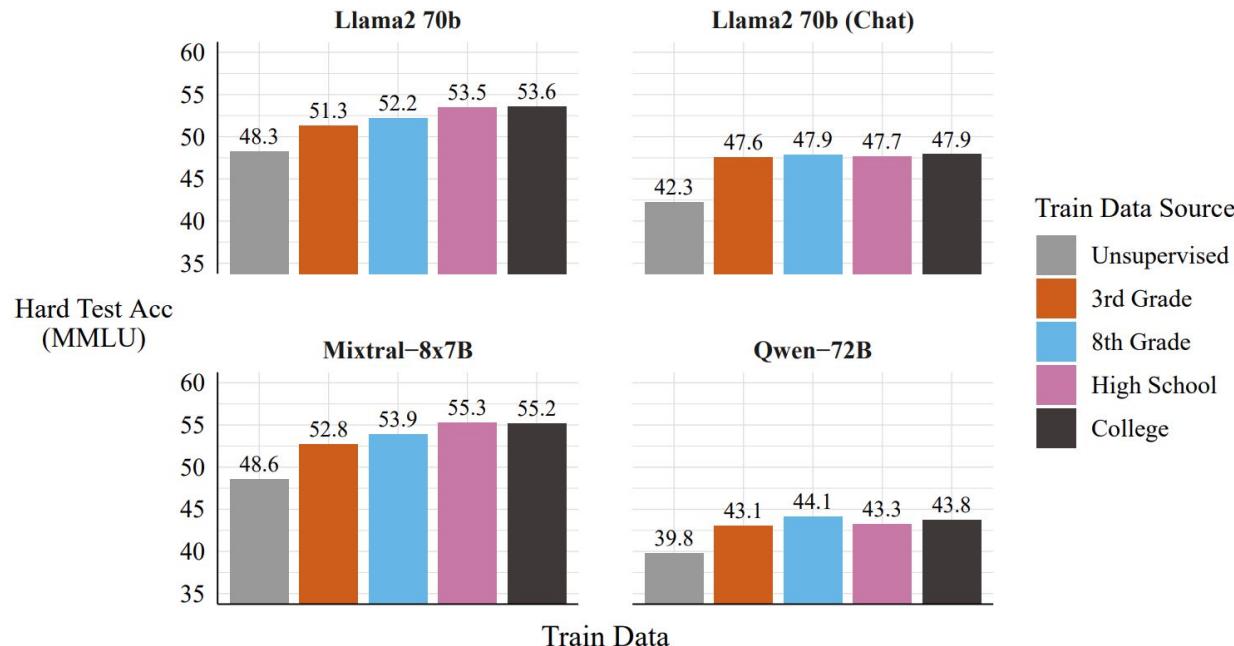


# RQ2: How Good Is Easy-to-Hard Generalization?

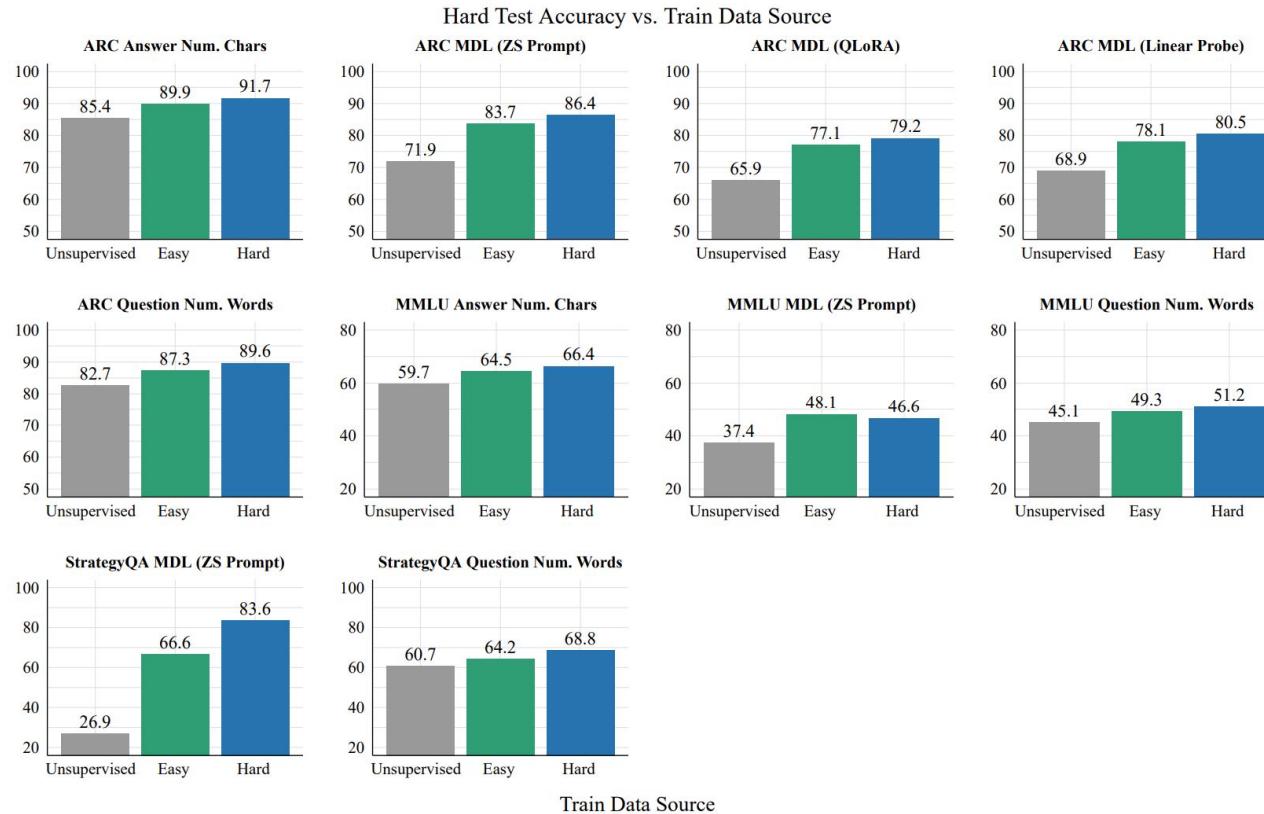


# RQ2: How Good Is Easy-to-Hard Generalization?

Hard Test Performance As a Function of Training Hardness (Across Models)



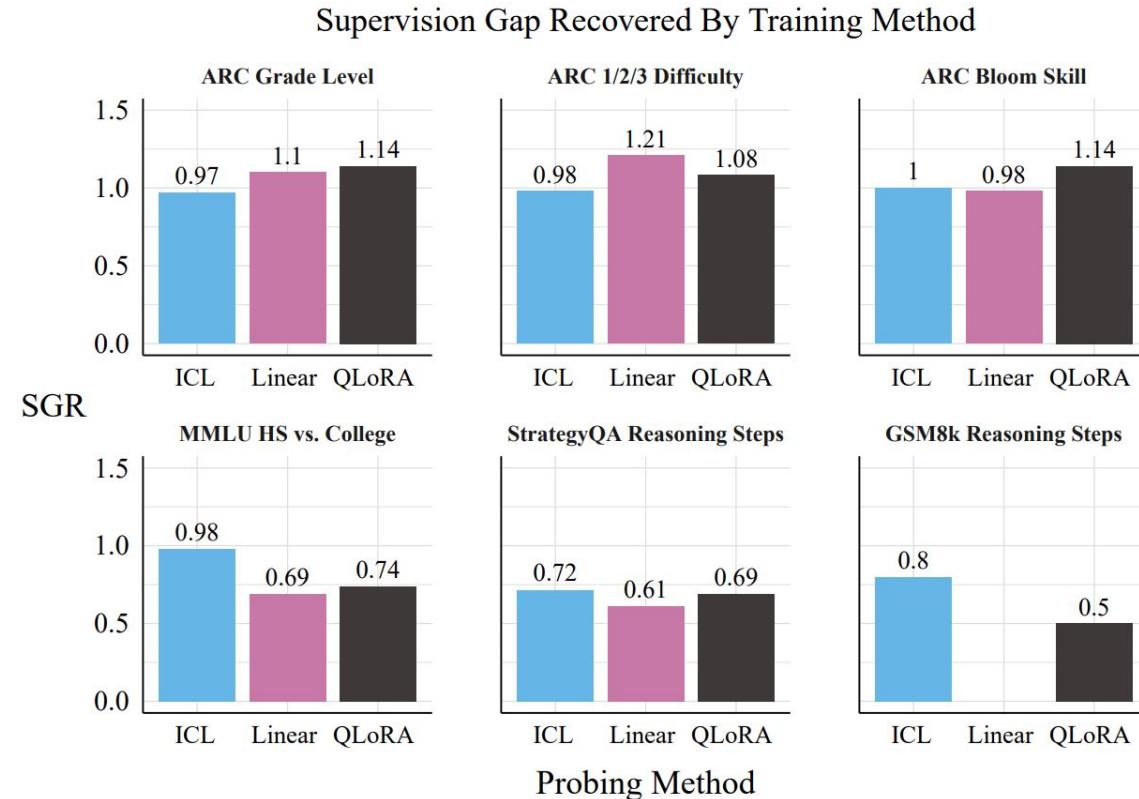
# RQ2: How Good Is Easy-to-Hard Generalization?



# RQ2: How Good Is Easy-to-Hard Generalization?

Results robust across  
training methods

Easy – Unsupervised  
Hard – Unsupervised



# Easy-to-Hard Generalization

---

Dataname	Hardness Measure	SGR Estimate	Test Hardness	<i>n</i>
ARC	Grade Level	$0.96 \pm 0.10 (p < 1e-4)$	Hard	1588
ARC	1/2/3 Difficulty	$0.98 \pm 0.36 (p = 0.0033)$	Hard	1588
ARC	Bloom Skill	$1.00 \pm 0.20 (p < 1e-4)$	Hard	1588
MMLU	HS vs. College	$0.97 \pm 0.59 (p = 0.0158)$	Hard	603
StrategyQA	Num Reasoning Steps	$0.72 \pm 0.93 (p = 0.0788)$	Hard	427
GSM8k	Num Reasoning Steps	$0.79 \pm 0.60 (p = 0.0125)$	Hard	333

We just saw these SGR values

# Easy-to-Hard Generalization

---

Dataname	Hardness Measure	SGR Estimate	Test Hardness	<i>n</i>
ARC	Grade Level	$0.96 \pm 0.10 (p < 1e-4)$	Hard	1588
ARC	1/2/3 Difficulty	$0.98 \pm 0.36 (p = 0.0033)$	Hard	1588
ARC	Bloom Skill	$1.00 \pm 0.20 (p < 1e-4)$	Hard	1588
MMLU	HS vs. College	$0.97 \pm 0.59 (p = 0.0158)$	Hard	603
StrategyQA	Num Reasoning Steps	$0.72 \pm 0.93 (p = 0.0788)$	Hard	427
GSM8k	Num Reasoning Steps	$0.79 \pm 0.60 (p = 0.0125)$	Hard	333
ARC	Grade Level	$1.00 \pm 0.09 (p < 1e-4)$	All	3521
ARC	1/2/3 Difficulty	$0.96 \pm 0.08 (p < 1e-4)$	All	3521
ARC	Bloom Skill	$0.98 \pm 0.08 (p < 1e-4)$	All	3521
MMLU	HS vs. College	$1.00 \pm 0.27 (p = 0.0001)$	All	1746
StrategyQA	Num Reasoning Steps	$0.87 \pm 0.32 (p < 1e-4)$	All	2290
GSM8k	Num Reasoning Steps	$0.98 \pm 0.39 (p = 0.0003)$	All	2065

**SGR values even higher when  
testing on “all” data**

Llama-2-70b ICL with  $k \leq 10$

# RQ4: Scaling Model Size & Train/Test Hardness

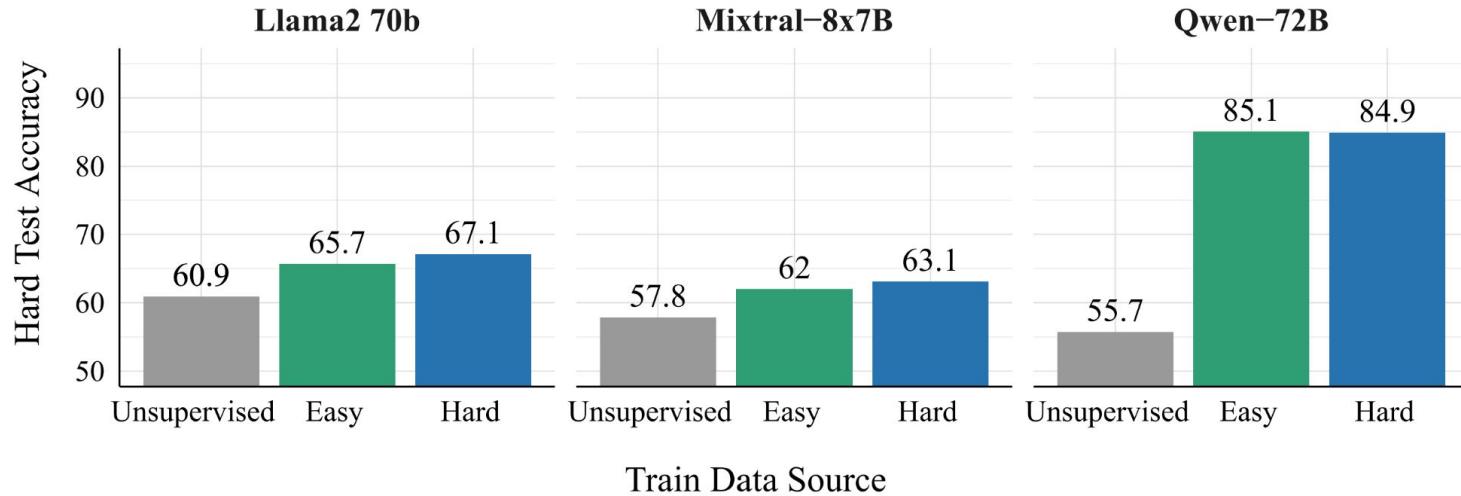
**Easy is barely worse than Medium**

Llama-2-70b  
ICL with  $k \leq 10$



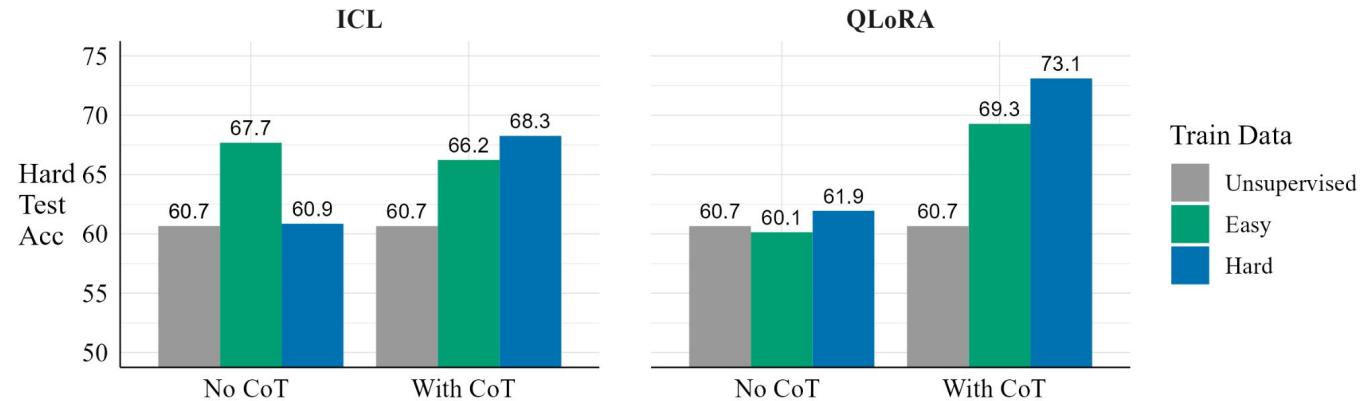
# Test Data Leakage?

Hard Test Accuracy vs. Train Data Source  
StrategyQA Reasoning Steps



# Effect of Reasoning

Does CoT Reasoning Influence Easy-to-Hard Generalization?



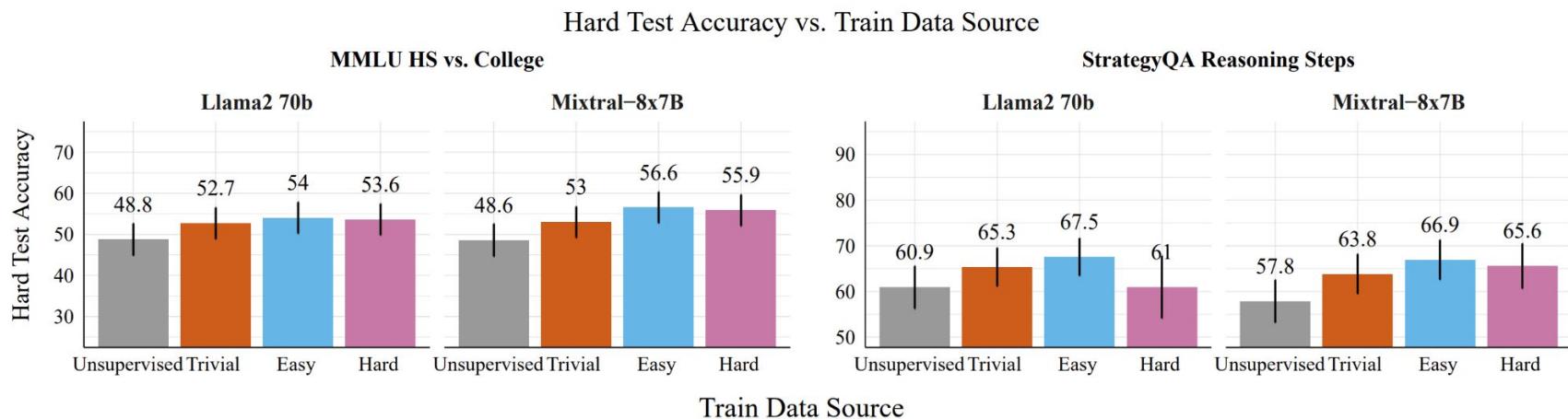
# Differences with Weak-to-Strong Paper

---

1. The baseline in SGR vs. PGR
2. We train on easy or hard data, not both
3. Human hardness variables in addition to model-based
4. All experiments with publicly available data and models (up to 70b params)
5. No early stopping
6. No new methods in our paper

# Task Format Prompts - Hard Test Data

---



# Task Format Prompts - All Test Data

---

