ACL 2025 VIENNA

Synthetic Data *in the Era of LLMs*

- **Vijay Viswanathan** (Carnegie Mellon)
- **Xiang Yue** (Carnegie Mellon)
- **Alisa Liu** (Washington)
- **Yizhong Wang** (Washington)
- **Graham Neubig** (Carnegie Mellon)

Carnegie Mellon University

Motivation

Data → NLP Progress

Language models are built on data

Pre-training

Raw text x → $P(x)$

Supervised Fine Tuning

Input x, output y → $P(y \mid x)$

Reasoning Training

Input x, output y, latent reasoning z → $P(y \mid z, x)\, P(z \mid x)$

Where do we get data?

- Scraping the internet
- Labeling manually
- Collecting from system users
- Creative curation

Why is this not enough?

- Scraping the internet — Too noisy, too massive
- Labeling manually — Too expensive, annotators not available
- Collecting from system users — Chicken and egg, privacy implications
- Creative curation — Limited applicability

Synthetic data to the rescue!

Create data order-made that is

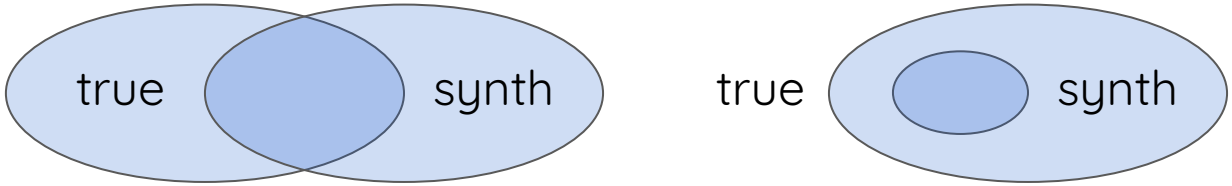
- Relatively clean
- Appropriately sized (not too big/small)
- Tailored to individual tasks
- Flexible

But getting all of these at once is hard!

But generating good synthetic data is hard…

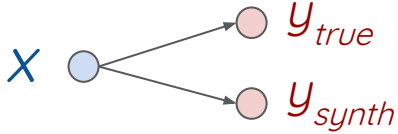
The input distribution may be **off**, or **not diverse enough**

$P_{true}(x) \neq P_{synth}(x)$



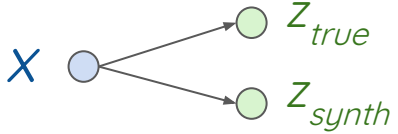
The labels may be **wrong**

$P_{true}(y \mid x) \neq P_{synth}(y \mid x)$



The reasoning may be **flawed**

$P_{true}(z \mid x) \neq P_{synth}(z \mid x)$



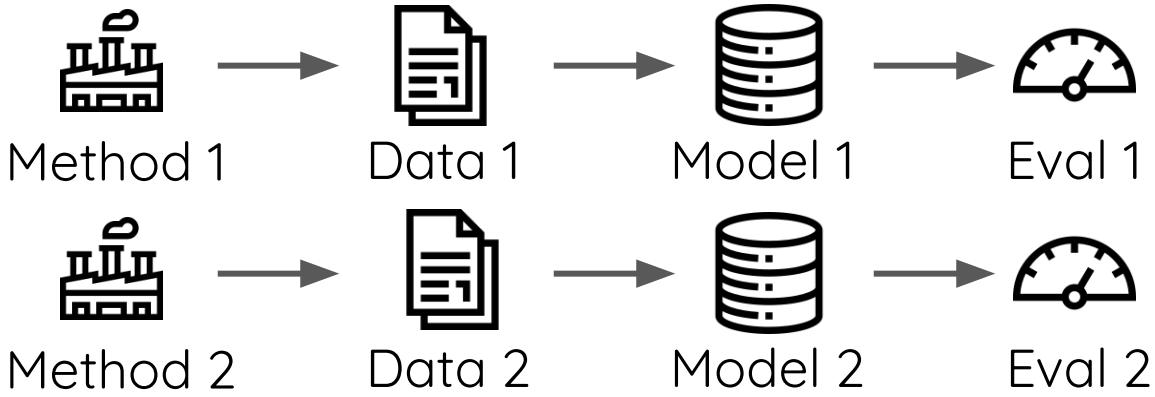
In this tutorial we will cover

- 2pm: How do we **evaluate** data quality? (15 minutes)
- 2:20pm: How do we **create** high-quality synthetic data? (35 minutes + QA)
- 3:05pm: How do we **use** synthetic data (Pt 1)? (25 minutes)
- 3:30pm: 30 minute-break
- 4pm: How do we **use** synthetic data (Pt 2)? (20 minutes + QA)
- 4:25pm: Scenario-specific **applications** (35 minutes + QA)
- 5:00pm: Limitations and open questions (25 minutes + QA)
- 5:30pm: End

What is “high-quality” synthetic data?

Evaluation of synthetic data

- **Extrinsic:** Does it help in a downstream task?



- **Intrinsic:** What are the characteristics of the data or generation process?

Intrinsic Eval: Data Correctness

- Questions regarding whether the data is correct, judged by manual or automatic methods
- E.g. Self-Instruct manually annotates:

Quality Review Question	Yes %
Does the instruction describe a valid task?	92%
Is the input appropriate for the instruction?	79%
Is the output a correct and acceptable response to the instruction and input?	58%
All fields are valid	54%

Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)

Intrinsic Eval: Data Diversity/Coverage

- How well does the generated data cover the plausible data distribution?
- E.g. DataTune evaluates bigram diversity

Dataset	Unique Bigrams Per Example	Total Tokens Per Example
Code Line Description		
Gold	13.2	32.3
Synthetic	2.5	35.0
Transformed	14.9	86.9
Elementary Math		
Gold	10.8	48.6
Synthetic	3.3	34.4
Transformed	11.6	43.8
Implicatures		
Gold	9.9	24.1
Synthetic	2.7	27.7
Transformed	17.8	39.8

Better Synthetic Data by Retrieving and Transforming Existing Datasets (Gandhi et al., 2024)

Intrinsic Eval: Other Metrics

- Many other dimensions, e.g. privacy, fairness, distributional similarity
- E.g. SynthTextEval toolkit

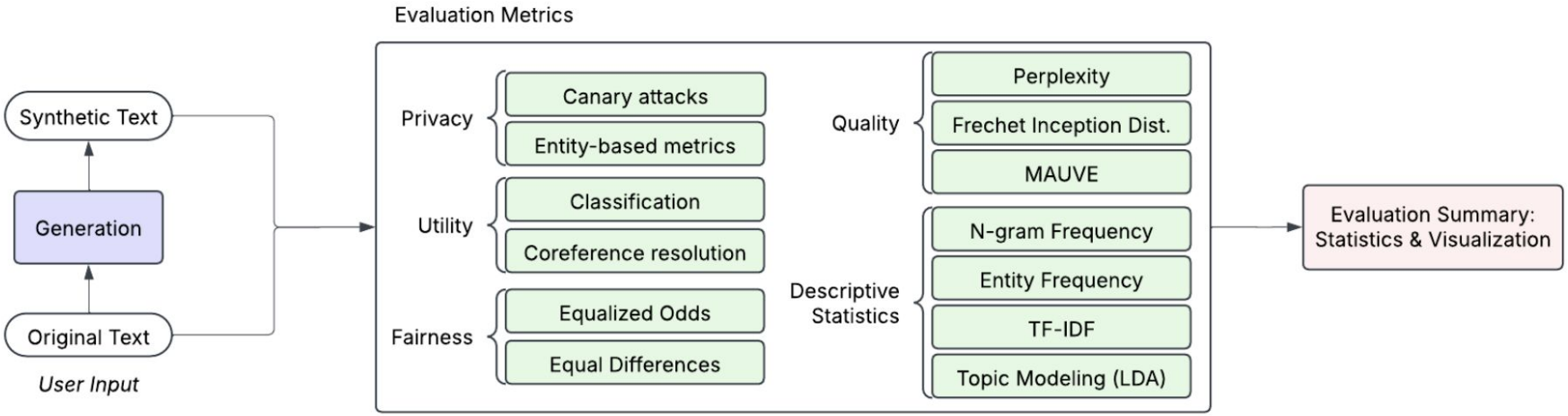
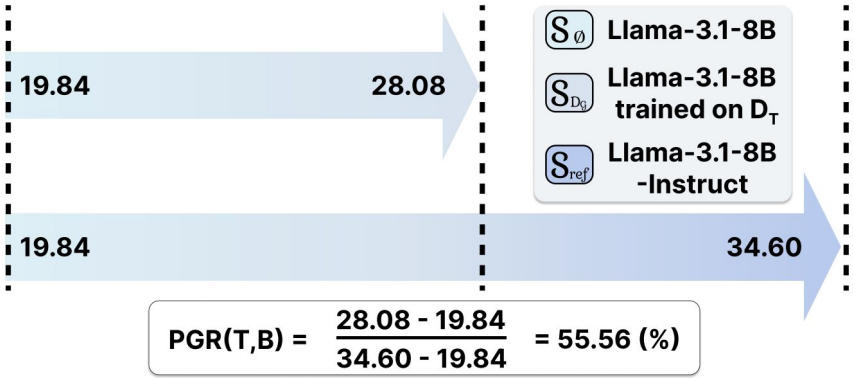


Figure 1: Architecture overview of SYNTHTEXTEVAL.

SynthTextEval: Synthetic Text Data Generation and Evaluation for High-Stakes Domains (Ramesh et al. 2025)

Evaluating Language Models as Data Generators

- We can also evaluate language models based on their ability to generate synthetic data
- E.g. AgoraBench, which measures synthetic data by different LMs based on its ability to match manually created data (at what cost)



Data Generator	API Cost		Prob. Solv.	Data Gen.
	Input	Output	Avg	Agora Bench
GPT-4o	$2.50	$10.00	80.9	29.5%
GPT-4o-mini	$0.15	$0.60	75.4	19.2%
Claude-3.5-Sonnet	$3.00	$15.00	80.5	23.6%
Llama-3.1-405B	$1.79	$1.79	75.0	11.3%
Llama-3.1-70B	$0.35	$0.40	69.6	14.1%
Llama-3.1-8B	$0.055	$0.055	50.2	15.9%

Evaluating Language Models as Synthetic Data Generators (Kim et al. 2024)

How do we create synthetic data?

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Approaches to synthetic data creation

Sampling-based data generation

Back-translation

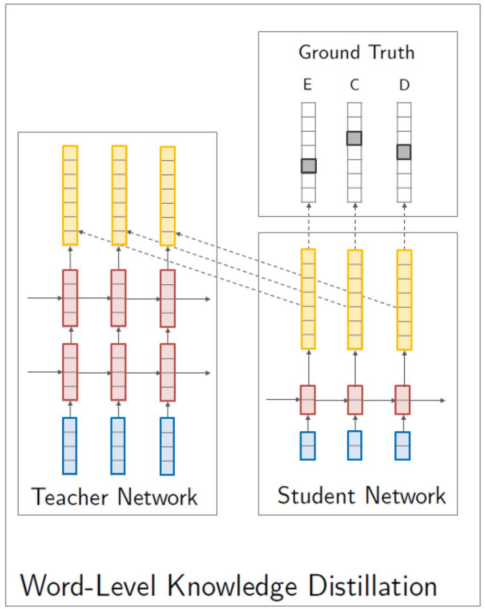
Transformation of existing data

Human-AI collaboration

Symbolic generation

Background: knowledge distillation

Train student model to mimic the teacher's predicted probability distribution (e.g., over words)

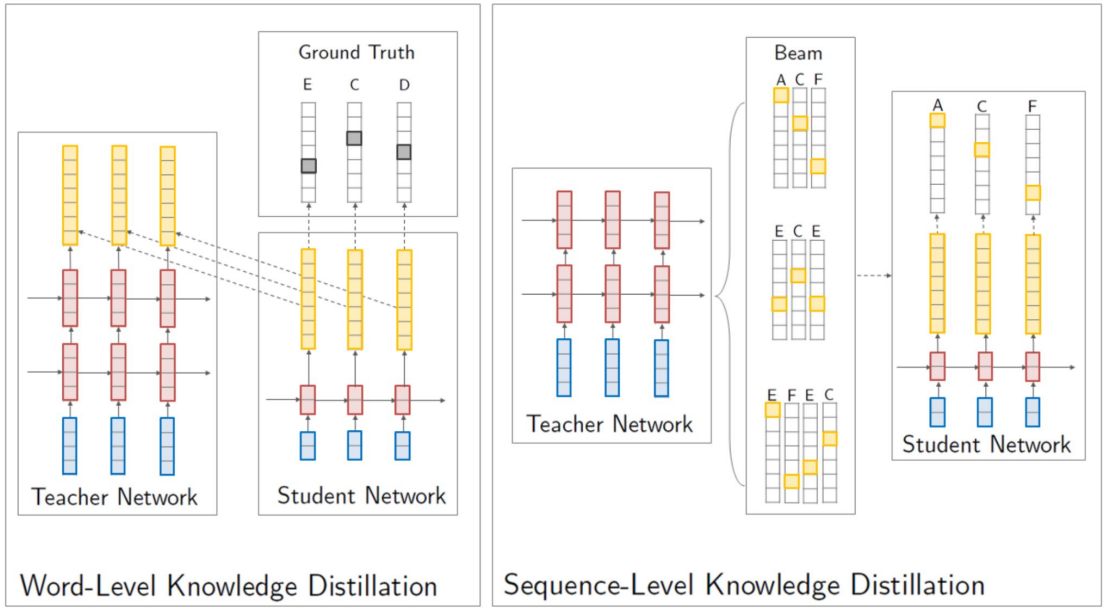


"Once the cumbersome model has been trained, we can then use a different kind of training, which we call "distillation" to transfer the knowledge to a small model"

[Distilling the Knowledge in a Neural Network (Hinton et al., 2015)]

Sequence **-level knowledge distillation**

Train student on complete generations (i.e., *sequences* of words) from the teacher



[Sequence-Level Knowledge Distillation (Kim & Rush, 2016)]

Generating task data from LMs

Use GPT-3's in-context learning ability to generate new examples of arbitrary tasks

```
Task: Write two sentences that mean
the same thing

Sentence 1: A man is playing the flute
Sentence 2: He's playing the flute
```

Create sentence-similarity examples by prompting the model to write similar (or dissimilar) sentences!

Generating Datasets with Pretrained Language Models (Schick & Schütze, 2021)

Generating instruction data from scratch

Instead of generating more examples under a given task, generate completely new tasks

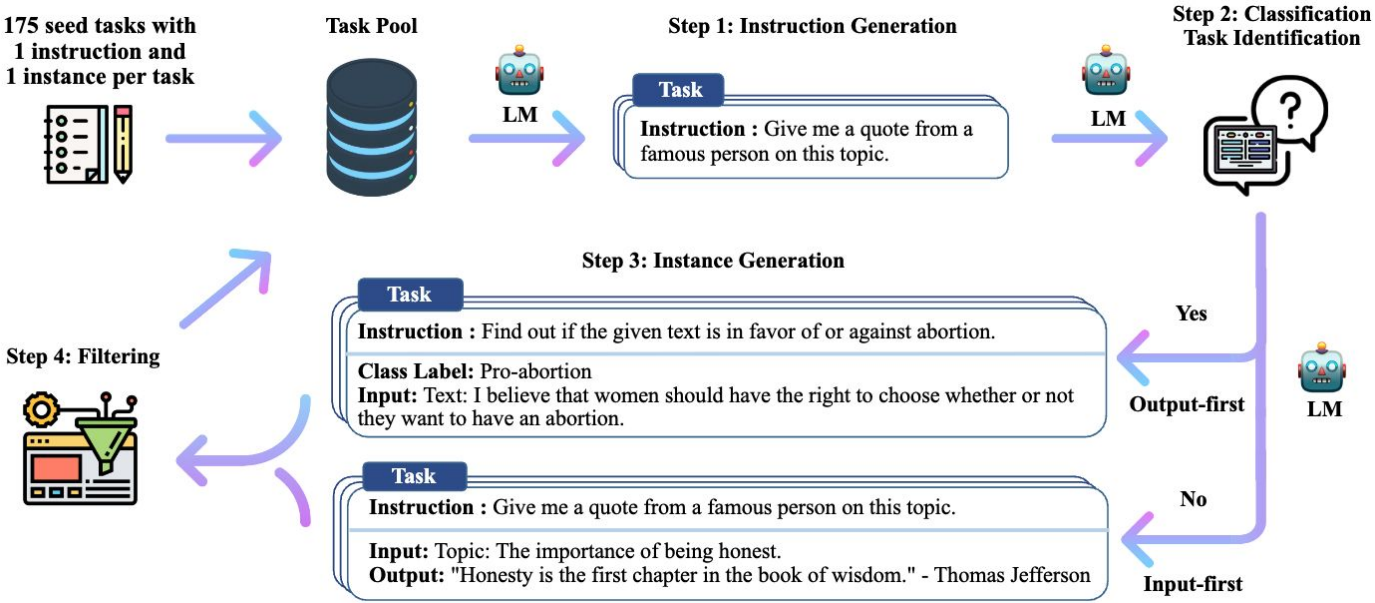
```
Come up with a series of
tasks.

{in-context examples}

Task: Given an address and
city, come up with the zip
code.
```

```
Come up with examples for the
following tasks.

{in-context examples}

Task: Given an address and city, come
up with the zip code.
Input: 123 Main Street, San Francisco
Output: 94105
```

Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)
Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor (Honovich et al., 2022)

Generating instruction data from scratch

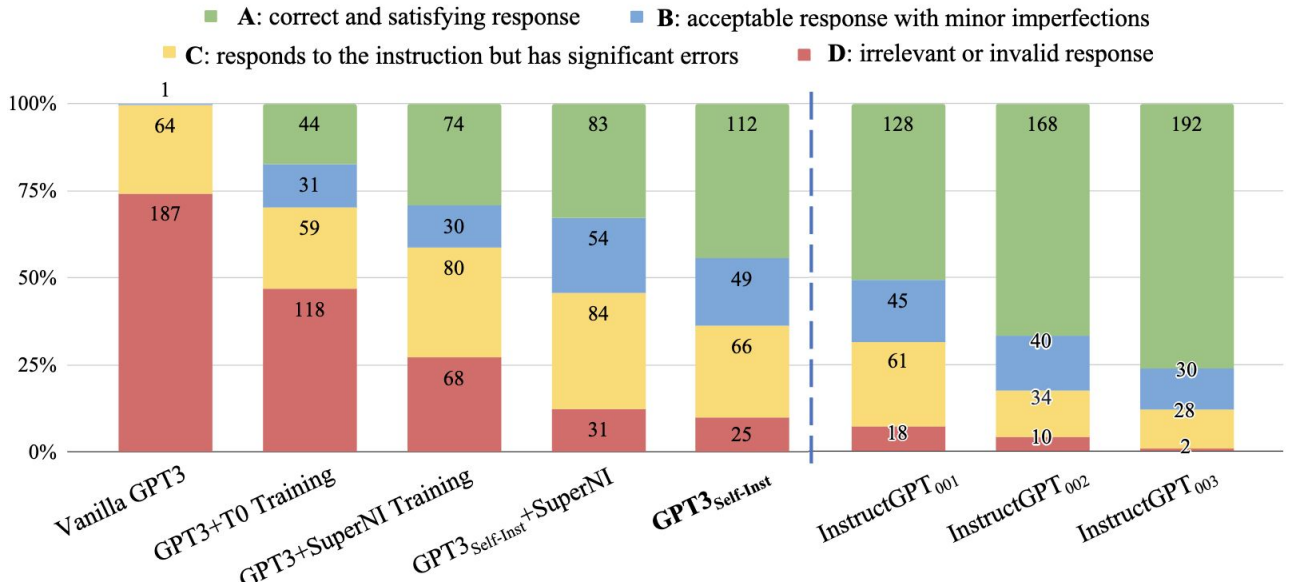
From just 175 seed examples → ~100K new examples



Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)
Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor (Honovich et al., 2022)

Generating instruction data from scratch

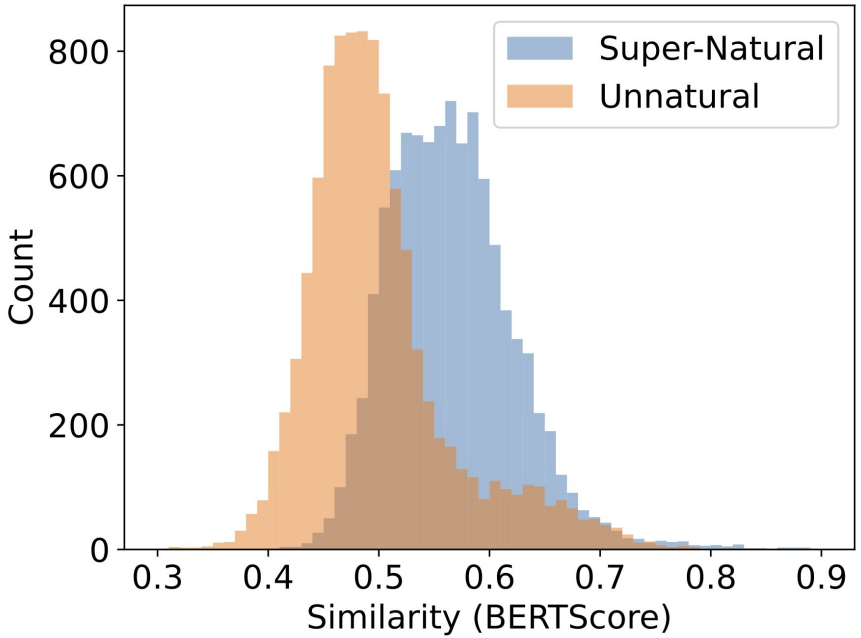
Finetuning GPT-3 on self-generated data improves over existing instruction datasets



Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)

Generating instruction data from scratch

Generated data is more diverse than human-written data



Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor (Honovich et al., 2022)

Generating instruction data from scratch

In both Self-Instruct data and Unnatural Instructions, only half of the examples are actually correct (!!)

Quality Review Question	Yes %
Does the instruction describe a valid task?	92%
Is the input appropriate for the instruction?	79%
Is the output a correct and acceptable response to the instruction and input?	58%
All fields are valid	54%

Table 2: Data quality review for the instruction, input, and output of the generated data.

113 of the 200 analyzed examples (56.5%) are correct. Of the 87 incorrect examples, 9 (4.5%) had incomprehensible instructions, 35 (17.5%) had an input that did not match the task description, and 43 (21.5%) had incorrect outputs.

Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)
Unnatural Instructions: Tuning Language Models with (Almost) No Human Labor (Honovich et al., 2022)

Takeaways from early efforts

Synthetic data can reflect creativity & diversity difficult to elicit from crowdworkers

Diversity can be more valuable than *correctness*!

Synthetic data can sometimes enable self-improvement

Data creation becomes a complex pipeline

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

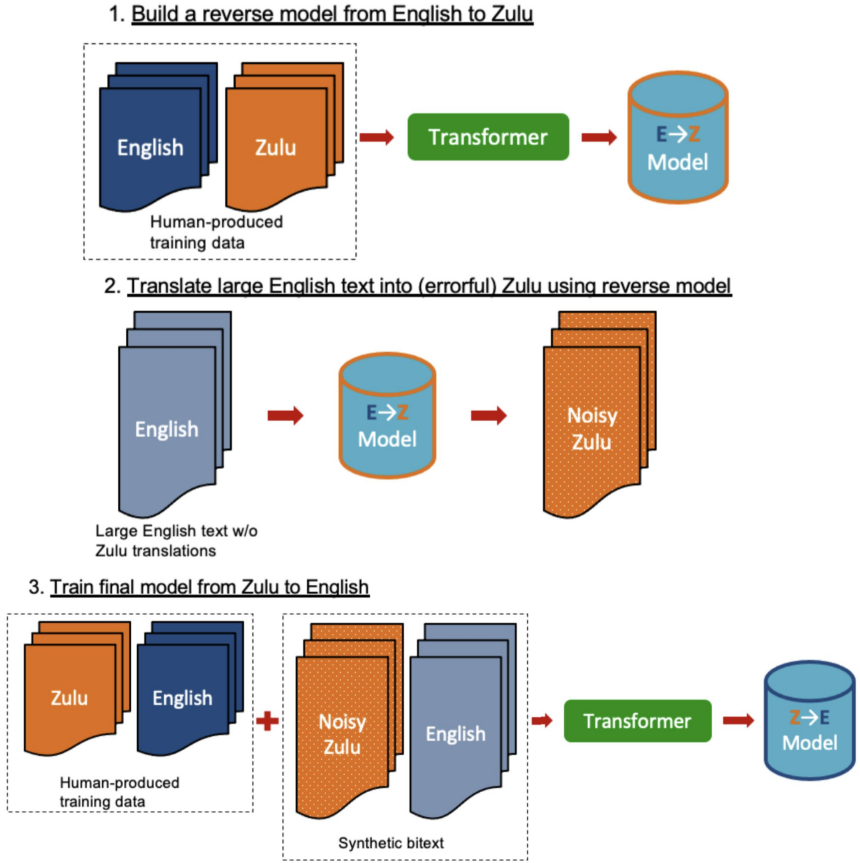
Human-AI collaboration

Symbolic generation

Background: back-translation in MT

Given an output, generate a corresponding input

Since models are trained to produce outputs, we want those to be natural (inputs can be unnatural)



[Improving Neural Machine Translation Models with Monolingual Data (Sennrich et al., 2016)]

Instruction back-translation

Given web text y, generate instruction x for which y would be a good response

y → x can be easier than x → y!

```
Output: It doesn't matter where you are in the world, how old
you are, or how much you know about meditation, it's for
everyone. The benefits of meditation are endless. Medication
can be as simple as sitting quietly for five minutes…

What kind of instruction could this be the answer to?
Instruction: Write an essay about the benefits of meditation.
```

LongForm: Effective Instruction Tuning with Reverse Instructions (Köksal et al., 2023)
Self-Alignment with Instruction Backtranslation (Li et al., 2023)

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

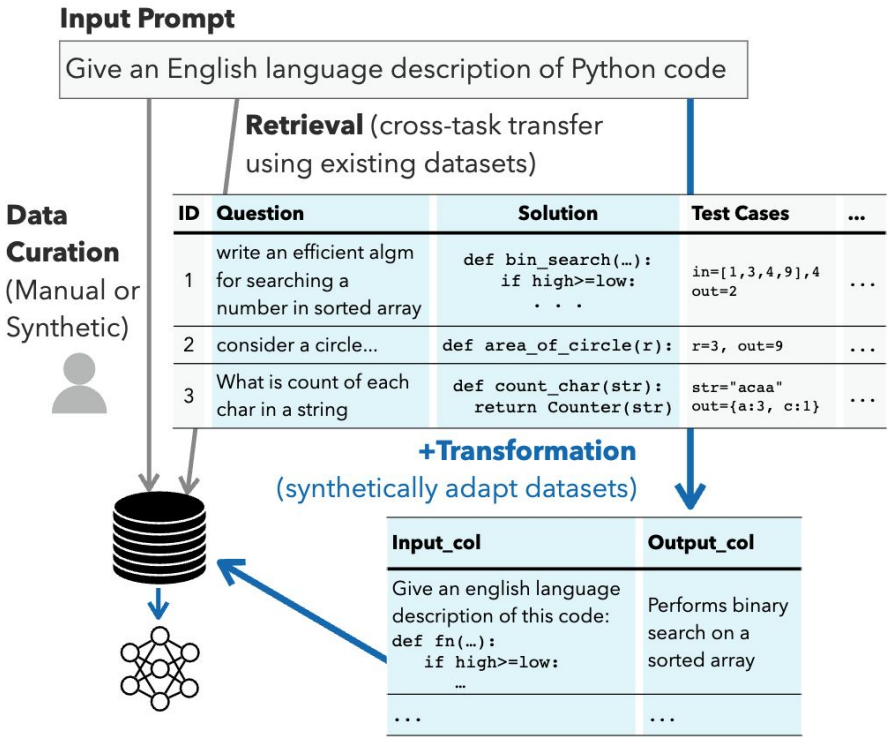
Transformation of existing data

Use or retrieve existing data, then transform it into an example under the desired task

Transform existing data into task examples

Given a task,

1. Retrieve relevant datasets or documents
2. Transform them into data under the desired task



[Better Synthetic Data by Retrieving and Transforming Existing Datasets (Gandhi et al., 2024)]
[Synthetic Dataset Generation Through Corpus Retrieval and Augmentation (Ziegler et al., 2024)]

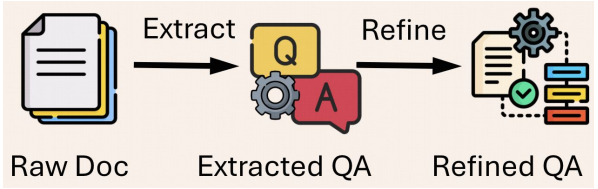
Ground generation in knowledge graphs

Condition on data from a large commonsense knowledge graph to produce diverse dialogues

SODA: Million-scale Dialogue Distillation with Social Commonsense Contextualization (Kim et al., 2023)

Extract instruction data from the web

Identify pages that may contain questions & answers, then extract and refine them!



Raw Docs *Unformatted Text, Site Information, Ads*

Topics Science\nAnatomy&Physiology\nAstronomy\nAstrophysics \nBiology\nChemistry \n...Socratic Meta...Featured Answers
How do you simplify #((u^4v^3)/(u^2v^-1)^4)^0# and write it using only positive exponents?
Answer by NickTheTurtle (Apr 1, 2017)
Explanation:\nAnything raised to the \(0^{th}\) power is simply 1. \n\nRelated Questions\n What is the quotient of powers property? \n\n\n\n How do you simplify expressions using the quotient rule?...\nImpact of this question\n1274 views around the world #Apps\niOS\nAndroid\nLinks\n[Privacy](#)\n[Terms](#)\n[Help](#)

Extract

Extracted QA *Formatted QA but lacking detailed solutions*

Question: How do you simplify $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$ and write it using only positive exponents?
Answer: Explanation: Anything to the $0th$ power is just simply 1.

Rewrite

Rewritten QA *Formatted QA augmented with detailed solutions*

Question:
How do you simplify $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$and write it using only positive exponents?

Answer:
To simplify the expression $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$ and rewrite it using only positive exponents, we start by evaluating the expression from the innermost operation outward.

First, consider the exponent of zero on the entire fraction:
Any expression raised to the power of zero is equal to 1 (provided the base is not zero). *Therefore*, $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0 = 1$

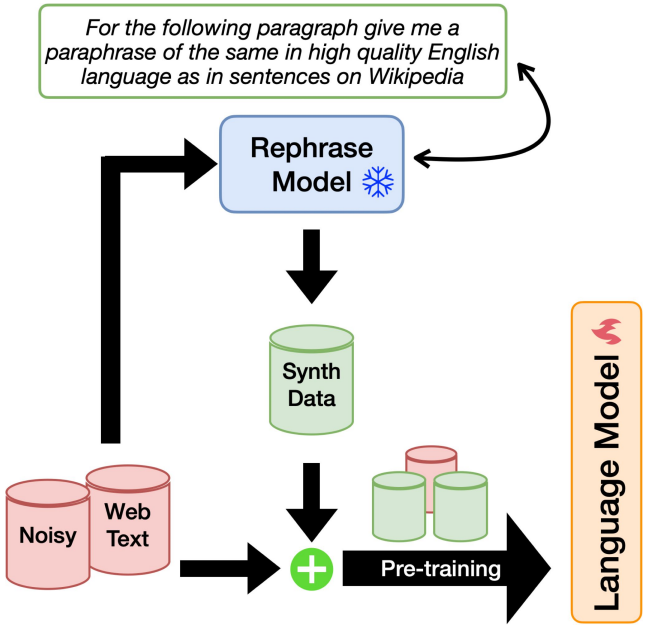
This simplification makes the other calculations unnecessary because raising any non-zero expression to the power of zero will always result in 1.

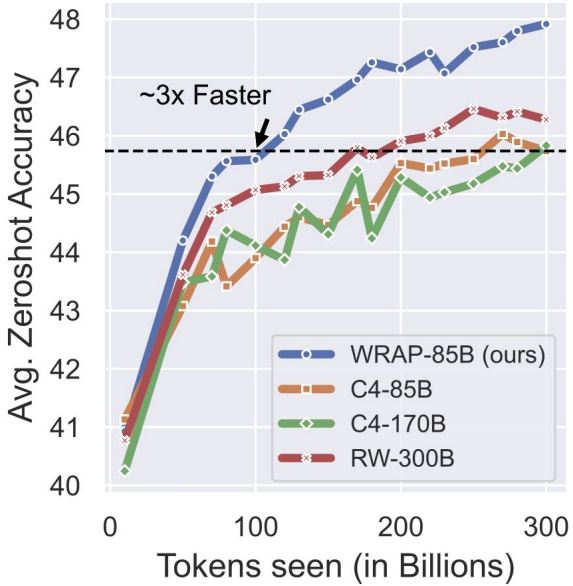
So, the simplified expression is simply: 1

[MAmmoTH2: Scaling Instructions from the Web (Yue et al., 2024)]

Rephrasing documents for pretraining

Use LMs to paraphrase noisy web text to create new data!





[Rephrasing the Web: A Recipe for Compute and Data-Efficient Language Modeling (Maini et al., 2024)]

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Human-AI collaboration

LMs are creative & diverse, but not reliably correct

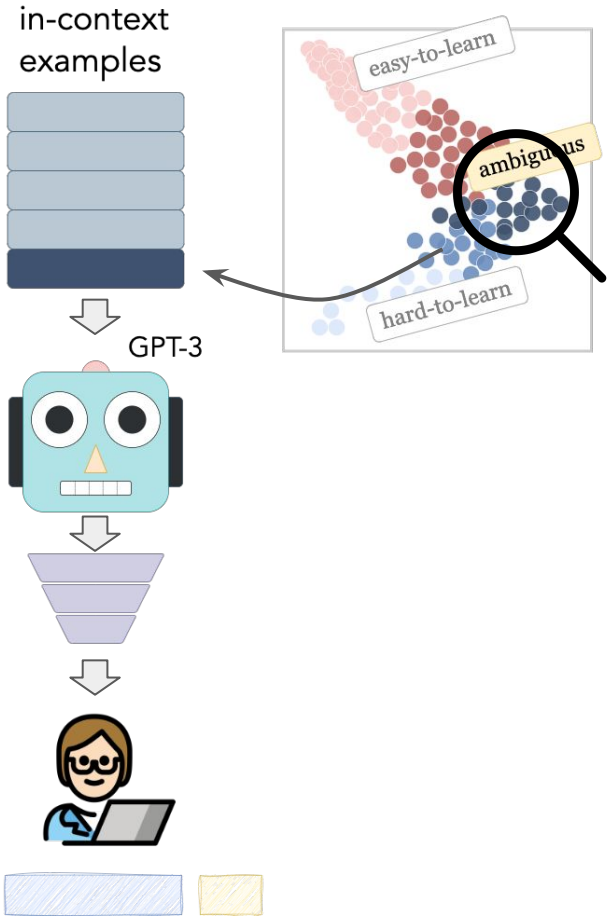
Humans can verify & improve correctness, but are not good at enumerating what they know

Combine the best of both worlds for data creation!

Human-AI collaboration for NLI

Crowdworkers revise & label generated data

Turns writing task into editing task!

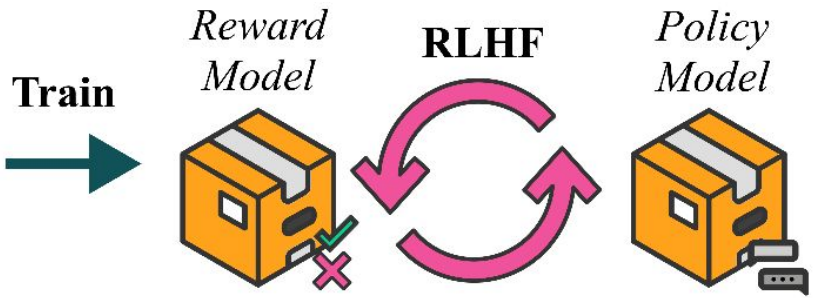


WANLI: Worker and AI Collaboration for Natural Language Inference Dataset Creation (Liu et al., 2022)
SynthBio: A Case Study in Faster Curation of Text Datasets (Yuan et al., 2021)

Route instances for human vs. AI feedback

Use router to solicit preference annotation from either human or LM

Prompt	A	B	Annotator
$x^{(1)}$	$y_1^{(1)}$	$y_2^{(1)}$	
$x^{(2)}$	$y_1^{(2)}$	$y_2^{(2)}$	
⋮	⋮		⋮
$x^{(n)}$	$y_1^{(n)}$	$y_2^{(n)}$	



[Hybrid Preferences: Learning to Route Instances for Human vs. AI Feedback (Miranda et al., 2024)]

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

Human-AI collaboration

Symbolic generation

Approaches to synthetic data creation

Sampling-based generation

Back-translation

Transformation of existing data

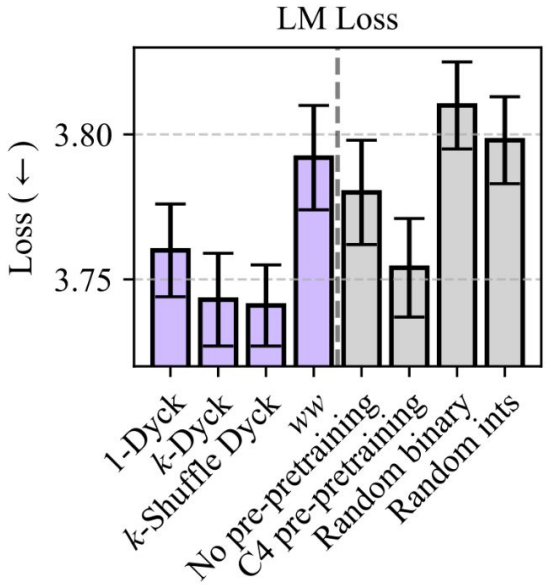
Human-AI collaboration

Symbolic generation

Symbolic generation

Doing initial pretraining on *formal languages* can lead to faster LM training and better generalization

Language	Example
1-Dyck	((( )))
k-Dyck	([{ }])
k-Shuffle Dyck	([{]) }
ww	1 2 3 1 2 3



Between Circuits and Chomsky: Pre-pretraining on Formal Languages Imparts Linguistic Biases (Hu et al., 2024)

Summary

Sampling-based generation: Generate examples from scratch from LMs

Back-translation: Given an output, generate an input

Transformation of existing data: Transform existing data into examples of the desired task

Human-AI collaboration: Mix LM generation & human annotation

Symbolic generation: Rule-based generation

Approaches to data filtering

Diversity filtering

Quality filtering

Correctness filtering

Diversity filtering: surface-level heuristics

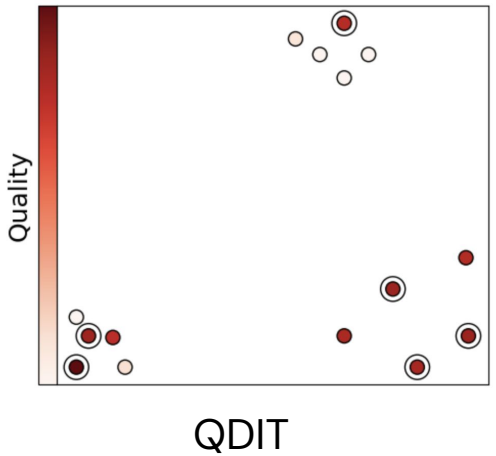
Filter similar examples as defined by

- Rouge-L (Self-Instruct; Impossible Distillation)

Diversity filtering: surface-level heuristics

Filter similar examples as defined by

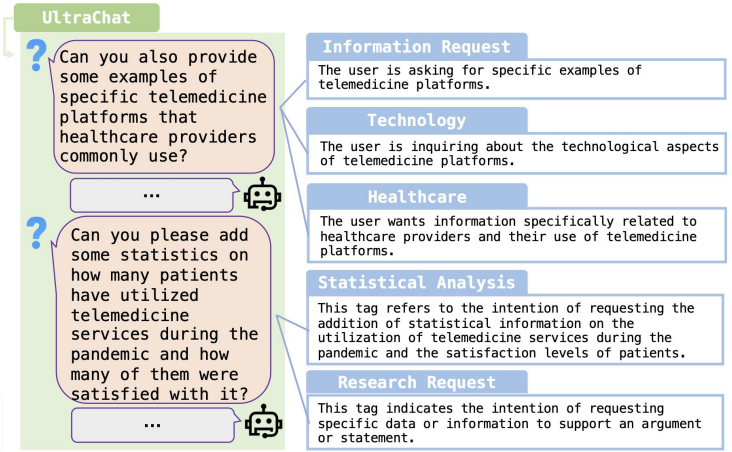
- Rouge-L (Self-Instruct; Impossible Distillation)
- Embedding similarity (QDIT, DiverseEvol, DEITA)



Diversity filtering: surface-level heuristics

Filter similar examples as defined by

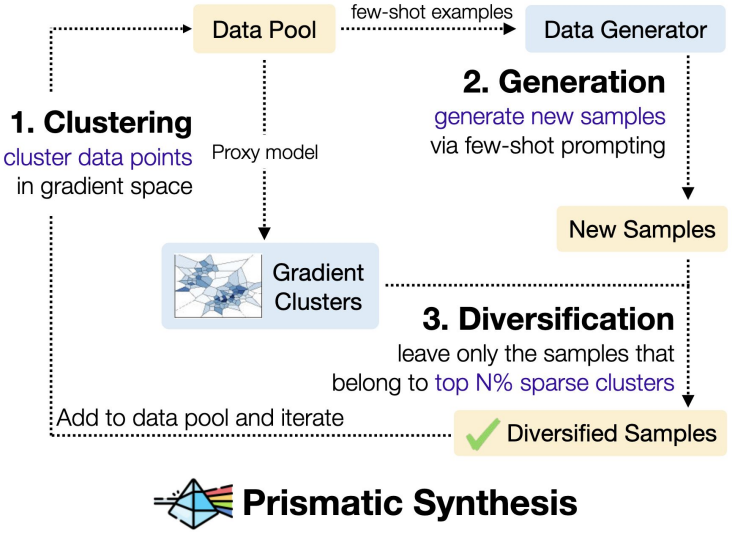
- Rouge-L (Self-Instruct; Impossible Distillation)
- Embedding similarity (QDIT, DiverseEvol, DEITA)
- Semantic tags (#InsTag)

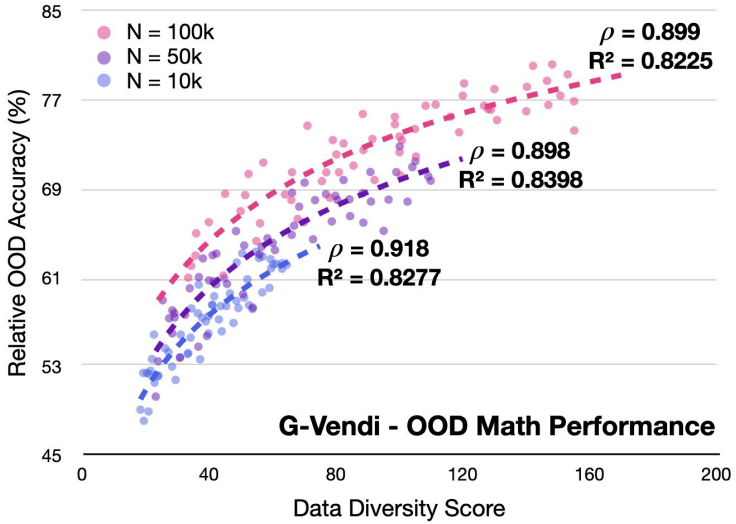


Diversity filtering: gradients

Measure diversity of data in *loss gradients*

Higher data diversity ⇒ more robust models

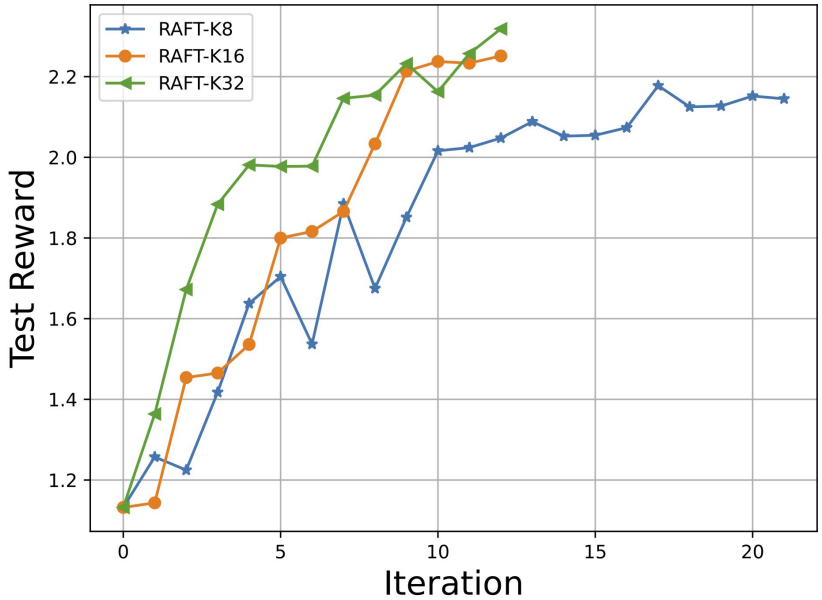




[Prismatic Synthesis: Gradient-based Data Diversification Boosts Generalization in LLM Reasoning (Jung et al., 2025)]

Quality filtering: reward models

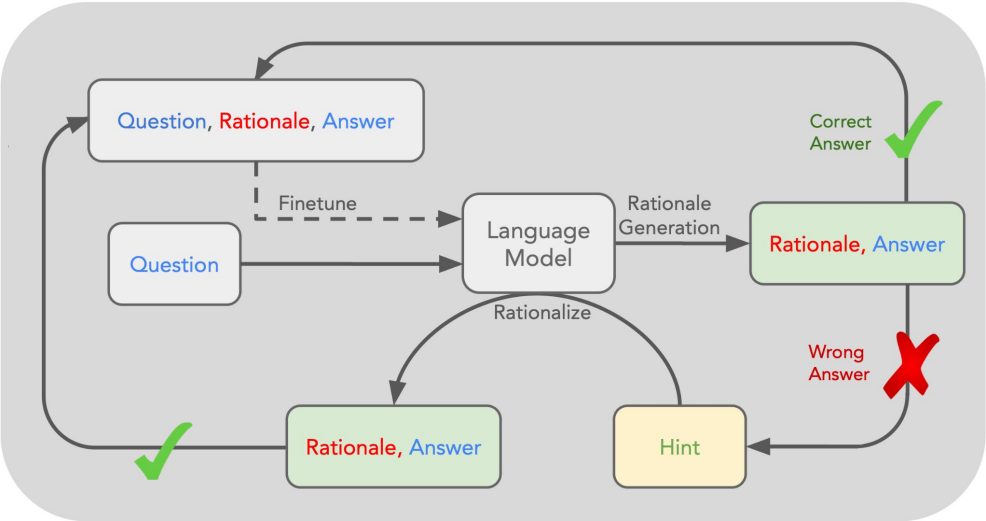
Sample K responses and take the one with the highest reward, then SFT on the best-of-K responses



RAFT: Reward rAnked FineTuning for Generative Foundation Model Alignment (Dong et al., 2023)

Correctness filtering: final answer verification

When generating synthetic reasoning traces, only keep generations whose final answers are correct




```
Q: What can be used
to carry a small dog?
Answer Choices:
(a) swimming pool
(b) basket
(c) dog show
(d) backyard
(e) own home
A: The answer must be
something that can be
used to carry a small
dog. Baskets are
designed to hold things.
Therefore, the answer
is basket (b).
```

STaR: Bootstrapping Reasoning With Reasoning (Zelikman et al., 2022)
DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via RL (Deepseek-AI, 2025)

How can we use synthetic data: Algorithms

How is synthetic data used?

Supporting fundamental language modeling algorithms

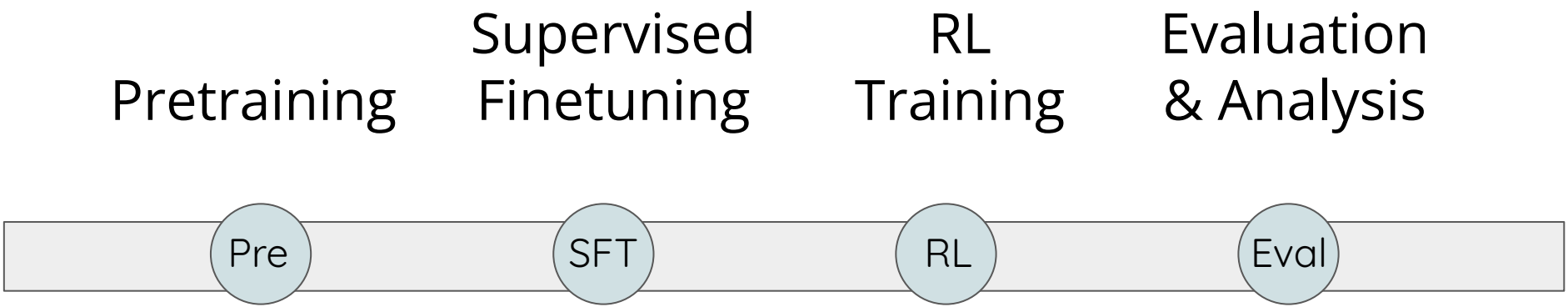
Supporting scenario-specific, end-user applications

How is synthetic data used?

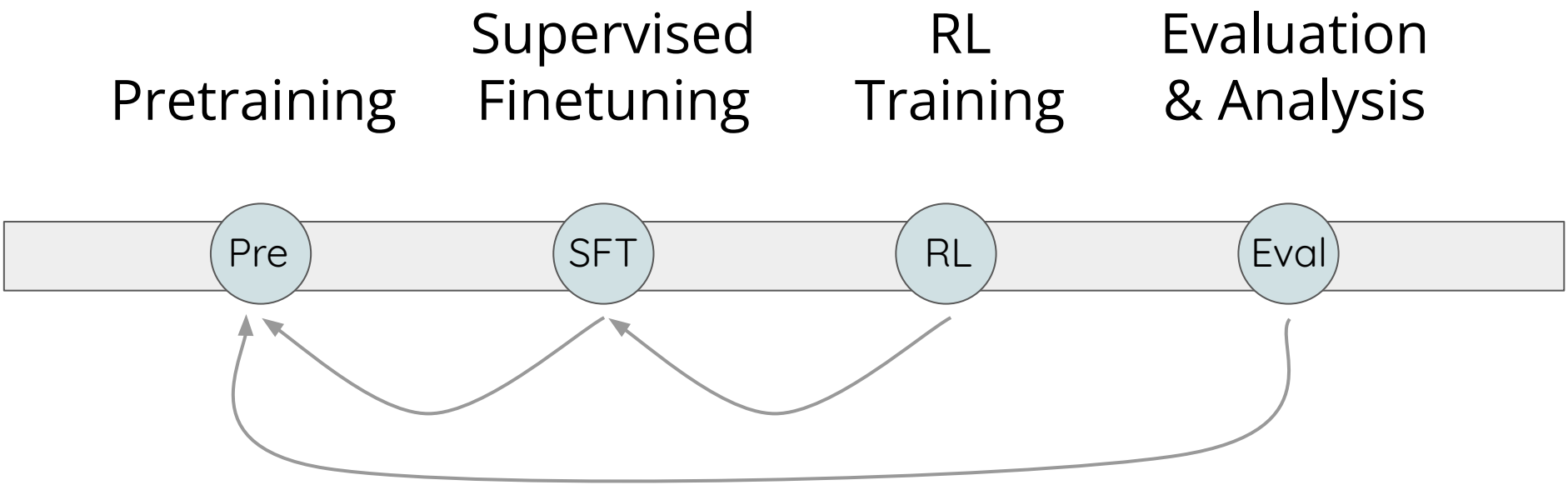
Supporting fundamental language modeling algorithms

Supporting scenario-specific, end-user applications

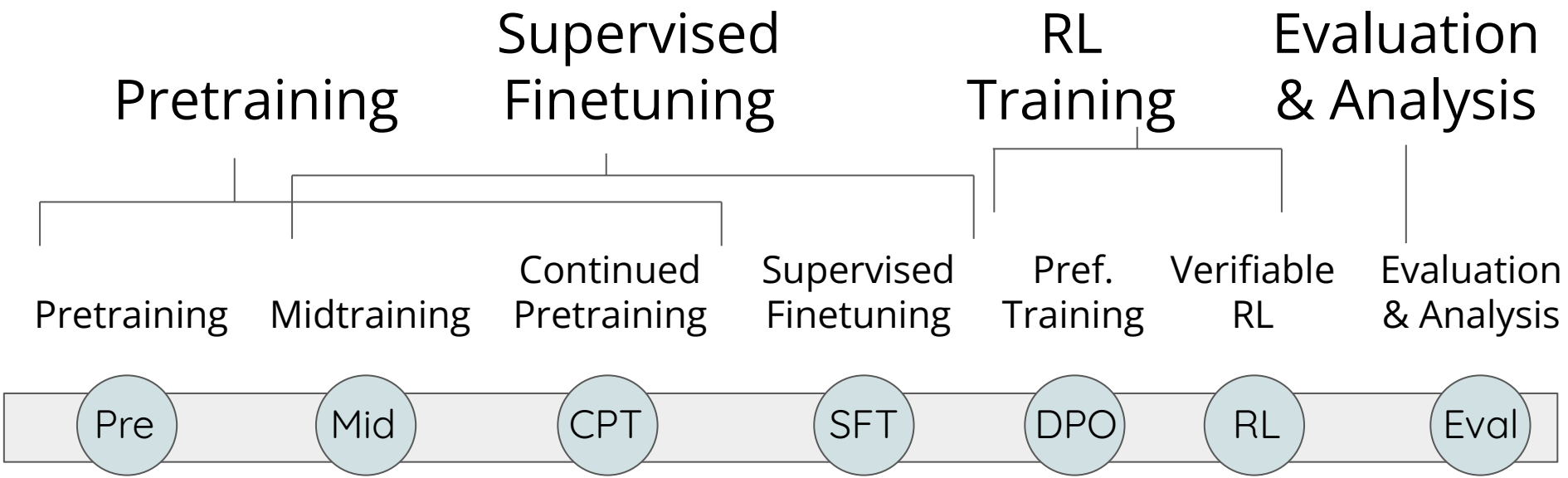
The language modeling pipeline (simplified)



The language modeling pipeline (complex)



The language modeling pipeline (complex)



Synthetic Data for Pretraining

From Ilya Sustkever's talk at NeurIPS 2024

Pre-training as we know it will end

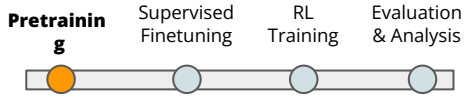
Compute is growing:

- Better hardware
- Better algorithms
- Larger clusters

Data is not growing:

- We have but one internet
- **The fossil fuel of AI**

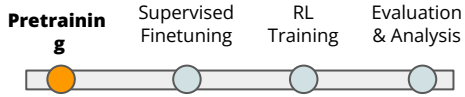
Does synthetic pretraining make sense?



Problem:

- Pretraining lets models learn linguistic patterns and facts
- Synthetic data generators can't invent *new linguistic patterns* or *real facts* they weren't trained on

Does synthetic pretraining make sense?

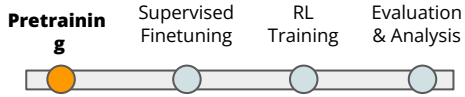


Problem:

- Pretraining lets models learn linguistic patterns and facts
- Synthetic data generators can't invent *new linguistic patterns* or *real facts* they weren't trained on

Solutions:

- Rephrase existing text
- Verbalize knowledge bases using LMs
- Generate text without using LMs (e.g. formal languages)



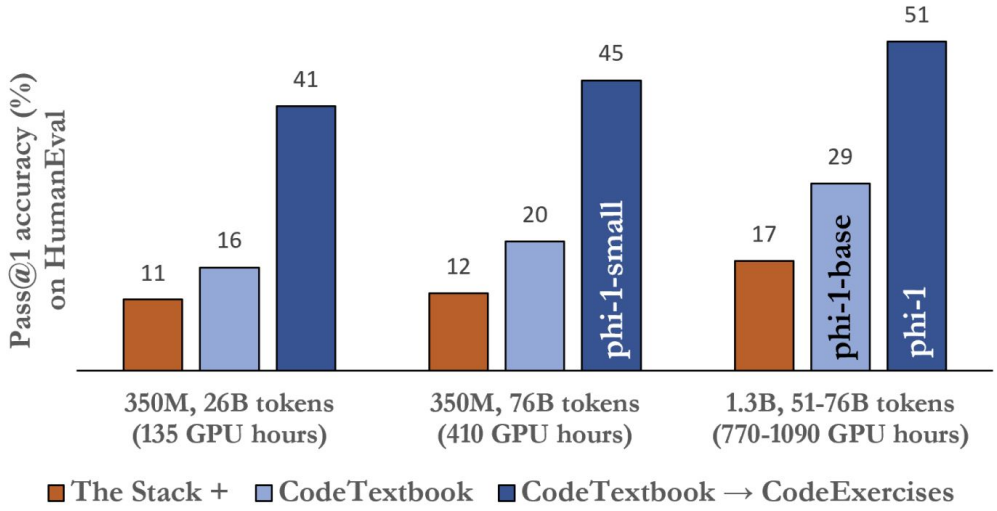
Synthetic Data for Pretraining: formatting

- Generated pretraining data by obtaining task-relevant documents from the internet, and convert them into training examples for SFT
  - example of *transformation of existing data*

Textbooks Are All You Need (Gunasekar et al., 2023)

Synthetic Data for Pretraining: formatting

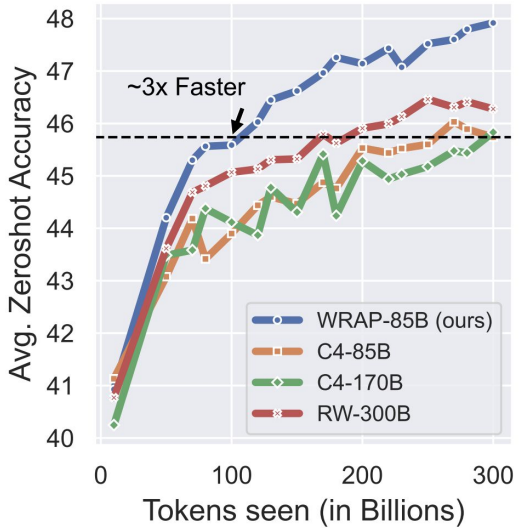
- Synthetically generated pretraining data (1B) can improve over a large amount of scraped data (6B)



Textbooks Are All You Need (Gunasekar et al., 2023)

Synthetic Data for Pretraining: formatting

- It seems to be easier to learn from scraped data reworded and cleaned by an LLM than on the same scraped data



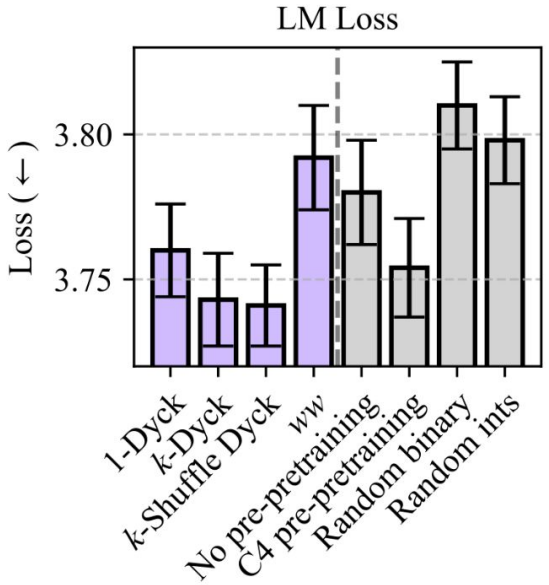
Rephrasing the Web: A Recipe for Compute and Data-Efficient Language Modeling (Maini et al., 2024)

Symbolic generation: inductive bias for pretraining

Pretraining on *formal languages*

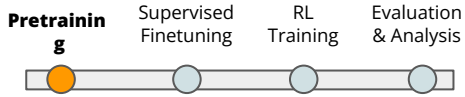
- Learning language without knowledge

Language	Example
1-Dyck	((( )))
k-Dyck	([{ }])
k-Shuffle Dyck	([{]) }
ww	1 2 3 1 2 3



Between Circuits and Chomsky: Pre-pretraining on Formal Languages Imparts Linguistic Biases (Hu et al., 2024)

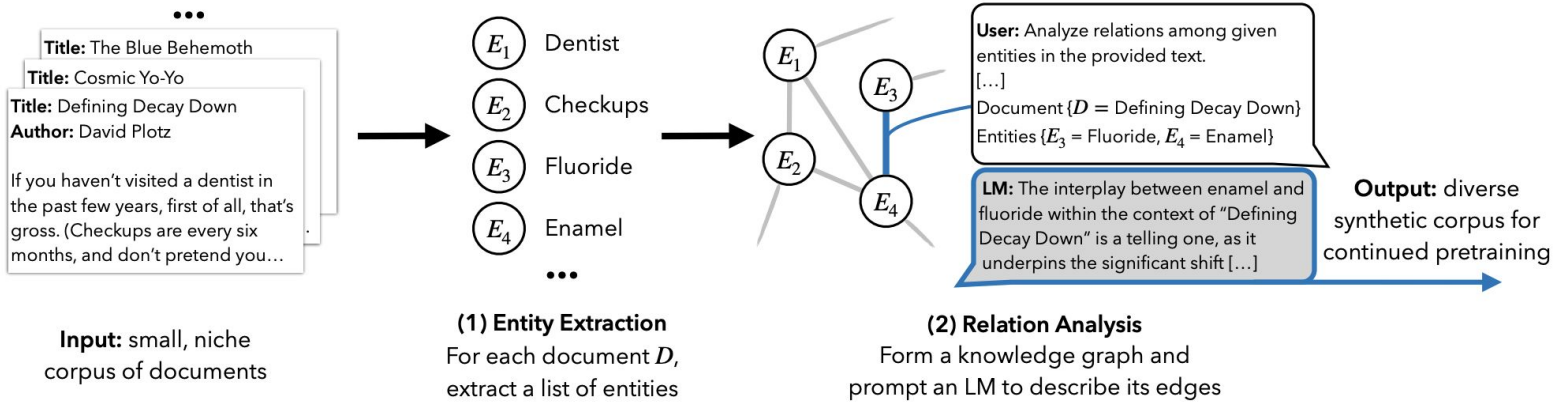
Data for Pretraining: domain adaptation



- Continuing pretraining of an LM on in-domain data is known to improve performance in the target domain
- This requires having abundant in-domain data

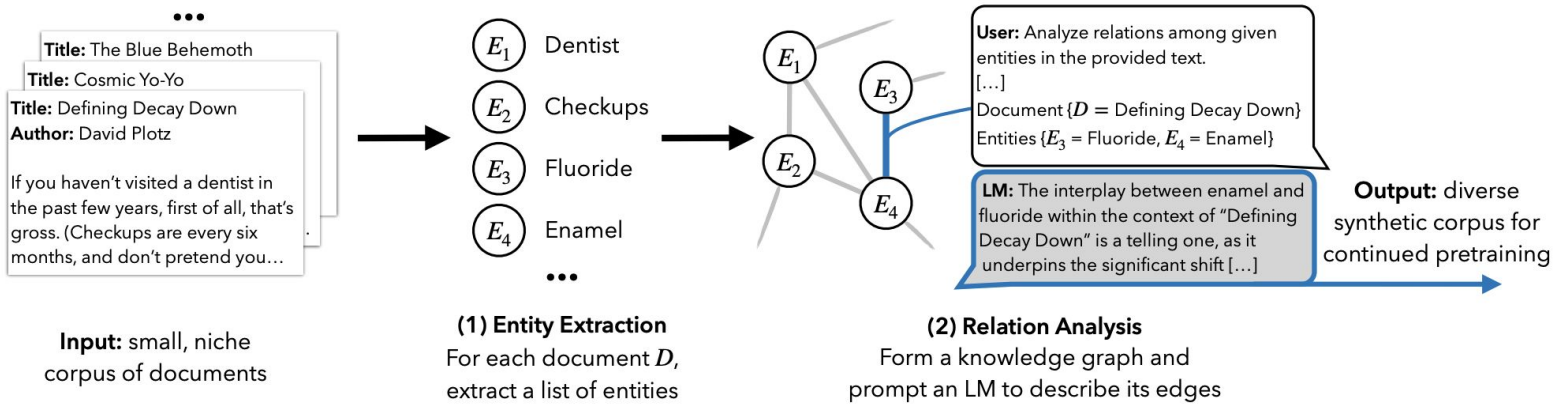
Don't Stop Pretraining: Adapt Language Models to Domains and Tasks (Gururangan et al. 2020)

Synthetic Data for Pretraining: domain adaptation



Synthetic continued pretraining (Yang et al., 2024)

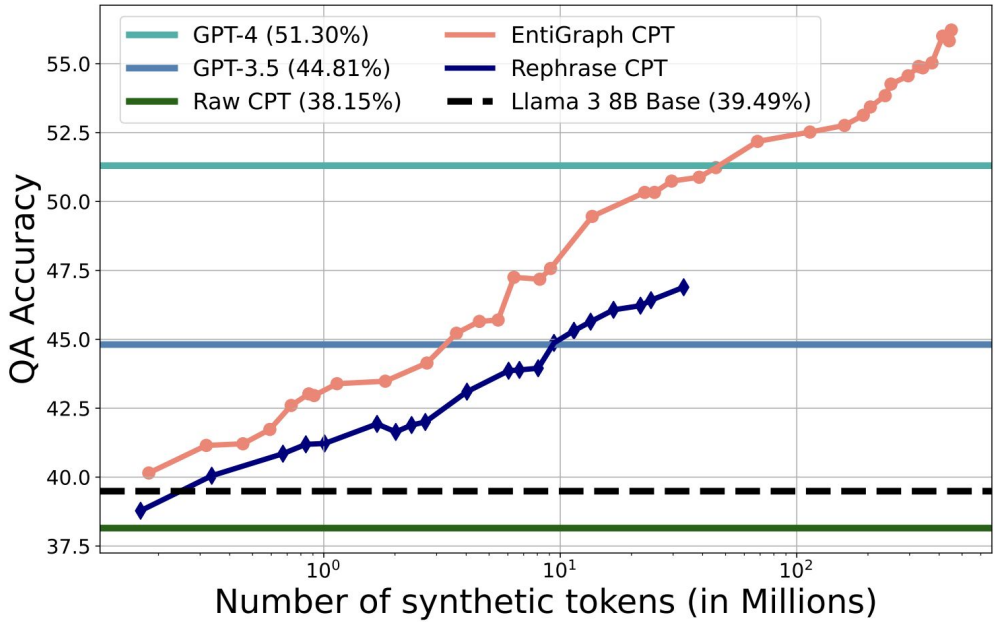
Synthetic Data for Pretraining: domain adaptation



- example of *transformation of existing data*

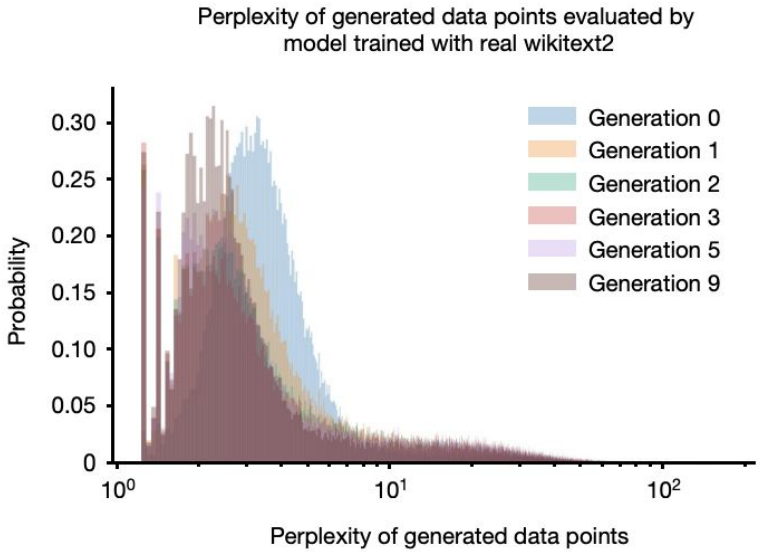
Synthetic continued pretraining (Yang et al., 2024)

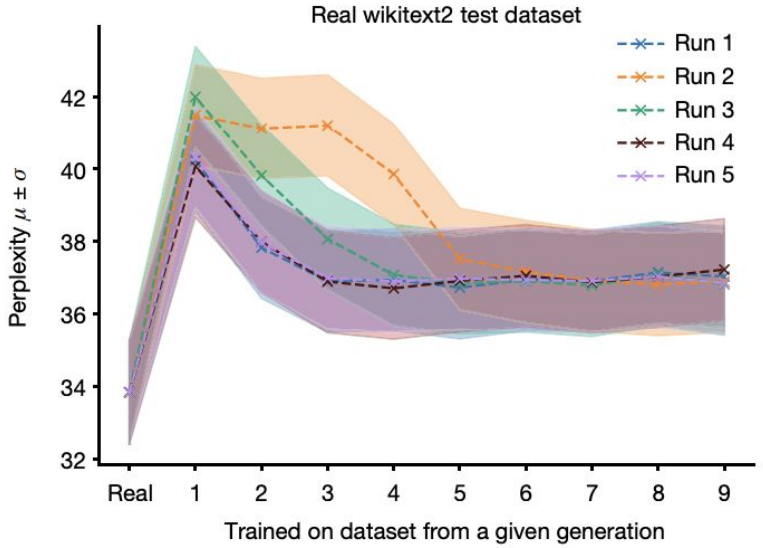
Synthetic Data for Pretraining: domain adaptation



Synthetic continued pretraining (Yang et al., 2024)

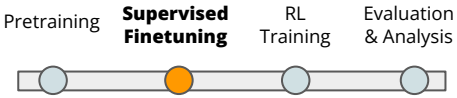
Synthetic Data for Pretraining: risks





[AI models collapse when trained on recursively generated data (Shumailov et al., 2024)]

Supervised Finetuning

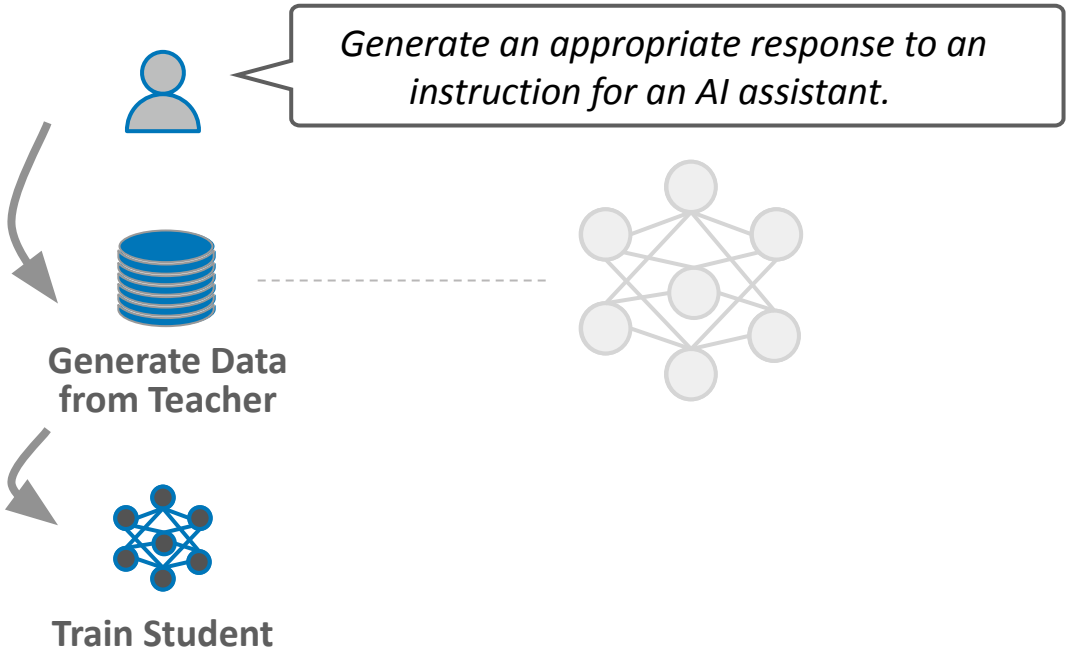


Goals of SFT:

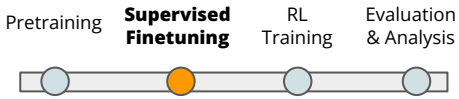
- Control the style of the model's output
- Specialize behavior for a particular use-case
- Feed new information to the model

Supervised Finetuning: Distillation

If a better model is available, you can train a *student* model to imitate the *teacher*



Supervised Finetuning: Distillation



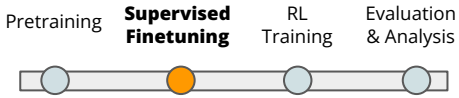
If a better model is available, you can train a *student* model to imitate the *teacher*

Pros

- Requires less human effort than manual annotation

Cons

- The student's performance is bounded by the teacher
- Legal issues, e.g. no-distillation clauses in terms of service



Supervised Finetuning: Limitations of Distillation

Goals of SFT:

- Control the style of the model's output
- Specialize behavior for a particular use-case
- Feed new information to the model

Observation: much of the benefit of distillation comes from adopting the teacher's style

- This does not require teaching the model new information

[The False Promise of Imitating Proprietary LLMs (Gudibande et al., 2023)]

Supervised Finetuning: Self-Guide

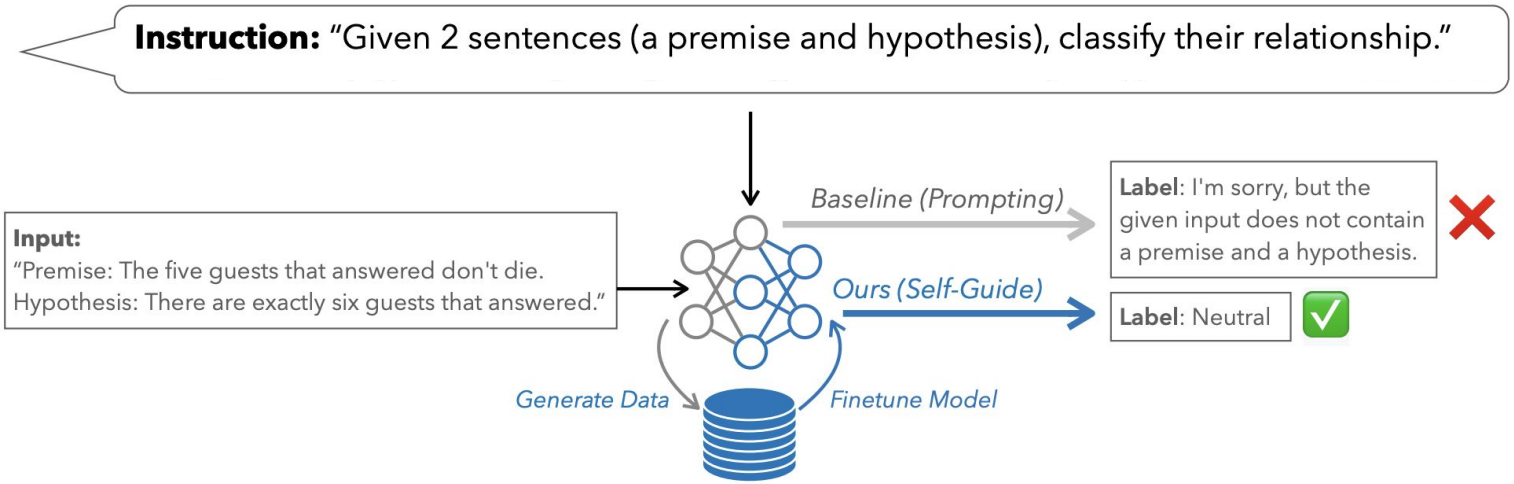


Figure 1: SELF-GUIDE uses a model's ability to generate synthetic data as a vehicle to improve the model's ability to execute a task as specified by an instruction.

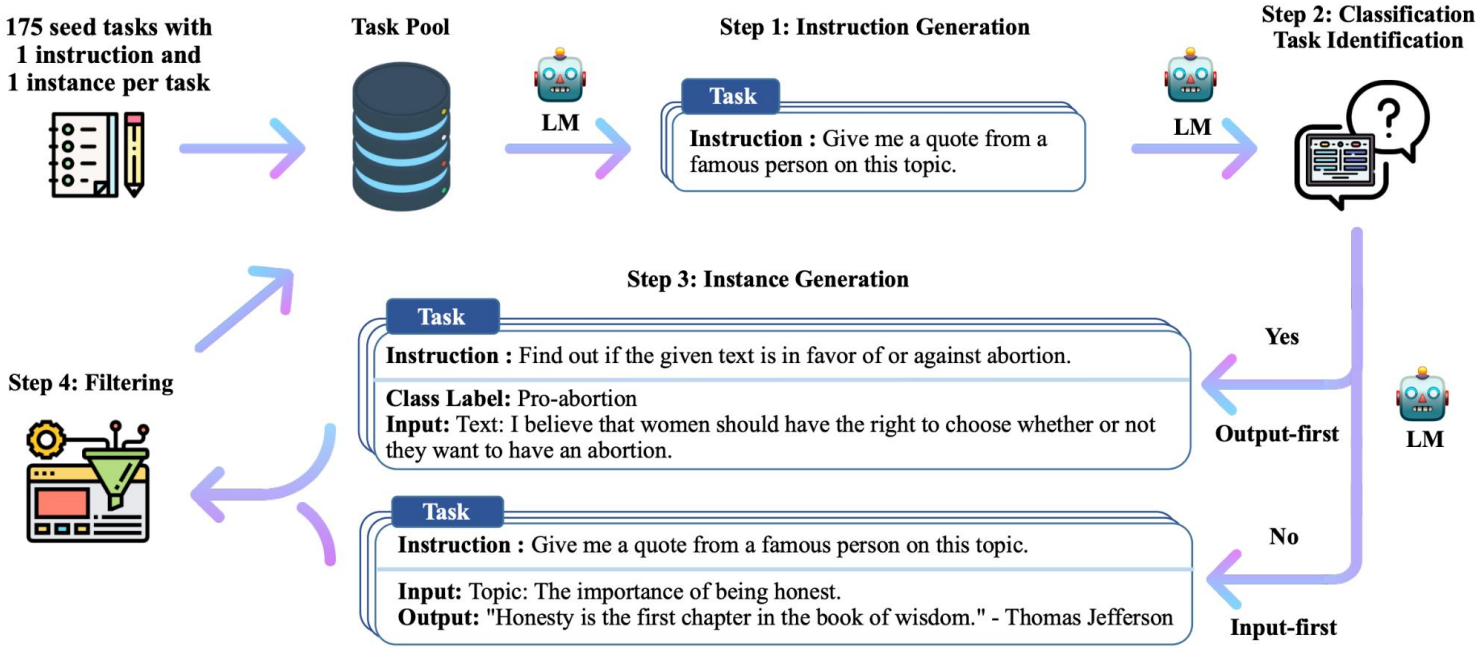
SELF-GUIDE: Better Task-Specific Instruction Following via Self-Synthetic Finetuning (Zhao et al., 2024)

Supervised Finetuning: Self-Guide

Task ID	Task Description	Prompting *(Baseline)*	Few-Shot Finetuning	SELF-GUIDE *(Ours)*	Δ
	Classification Tasks				
task1516	NLI (IMPPRES)	17.6	32.2	35.2	17.6
task1529	NLI (SciTail)	8.5	48.9	54.5	46.0
task1612	Sentiment Class. (SICK)	51.3	33.3	33.3	-18.0
task1615	NLI (SICK)	0.5	33.3	33.1	32.6
task284	Sentiment Class. (IMDB)	90.0	71.9	82.2	-7.8
task329	Coreferent Class.	29.1	45.4	44.7	15.6
task346	Word POS Class.	35.1	49.9	50.7	15.6
Avg	**Metric: Exact Match**	**33.2**	**45.0**	**47.7**	**14.5**
	Generation Tasks				
task1345	Question Paraphrasing	40.7	36.0	50.5	9.8
task281	Find Common Entity	46.8	40.7	49.3	2.5
task1562	Question Paraphrasing	29.5	48.6	59.3	29.8
task1622	Fluency Correction	49.2	86.2	78.5	29.3
Avg	**Metric: ROUGE-L**	**41.6**	**52.9**	**59.4**	**17.9**

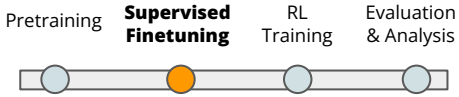
SELF-GUIDE: Better Task-Specific Instruction Following via Self-Synthetic Finetuning (Zhao et al., 2024)

Supervised Finetuning: Self-Instruct



Self-Instruct: Aligning Language Models with Self-Generated Instructions (Wang et al., 2022)

Supervised Finetuning: Self-Instruct

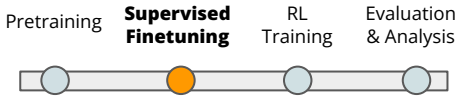


- Originally developed to instruction-tune GPT-3 using itself, to nearly match InstructGPT

Model	# Params	ROUGE-L
GPT3$_{\text{SELF-INST}}$ (Ours)	175B	39.9
InstructGPT$_{001}$	175B	**40.8**

Alpaca: A Strong, Replicable Instruction-Following Model (Taori et al., 2023)

Supervised Finetuning: ~~Self-~~Instruct

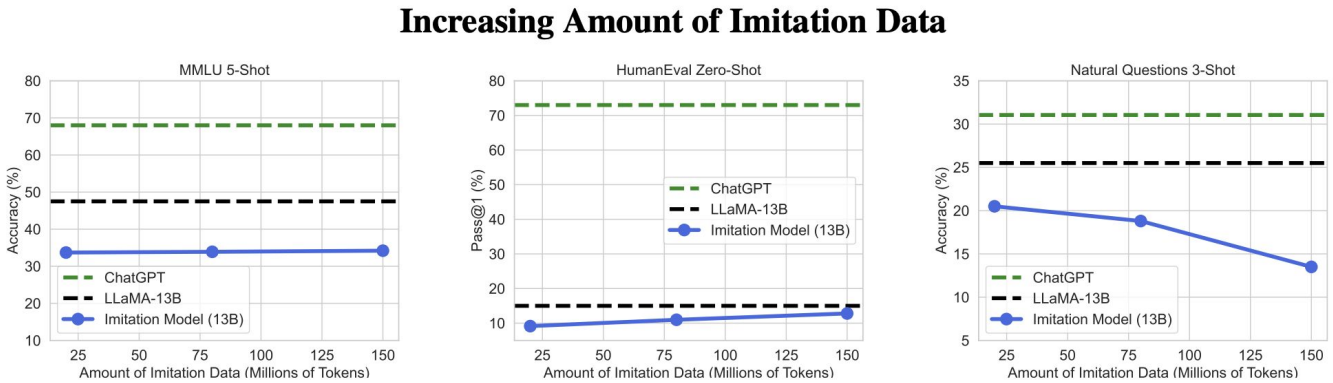


- Unsurprisingly, the same algorithm can also benefit from using a bigger teacher when available
  - *Alpaca* used this algorithm with GPT-3 as the data generator

Alpaca: A Strong, Replicable Instruction-Following Model (Taori et al., 2023)

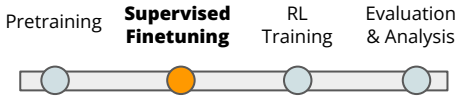
Supervised Finetuning: ~~Self-~~Instruct

- With sampling-based distillation, students mostly learn only surface-level behaviors (style, toxicity, refusals)



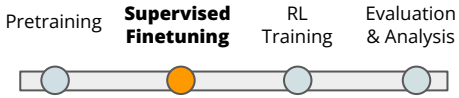
[The False Promise of Imitating Proprietary LLMs (Gudibande et al., 2023)]

Supervised Finetuning: Distillation



- To learn nontrivial capabilities via *sampling-based generation*, complex sampling strategies are needed

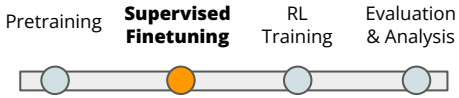
Supervised Finetuning: Distillation



- To learn nontrivial capabilities via *sampling-based generation*, complex sampling strategies are needed
- Example 1: *Evol-Instruct*
  - Start with *Alpaca* data
  - Randomly sample constraint types to add these instructions to add complexity
  - For each base instruction, generate analogies in other domains
  - Filter out problematic instructions

WizardLM: Empowering large pre-trained language models to follow complex instructions (Xu et al., 2023)

Supervised Finetuning: Distillation

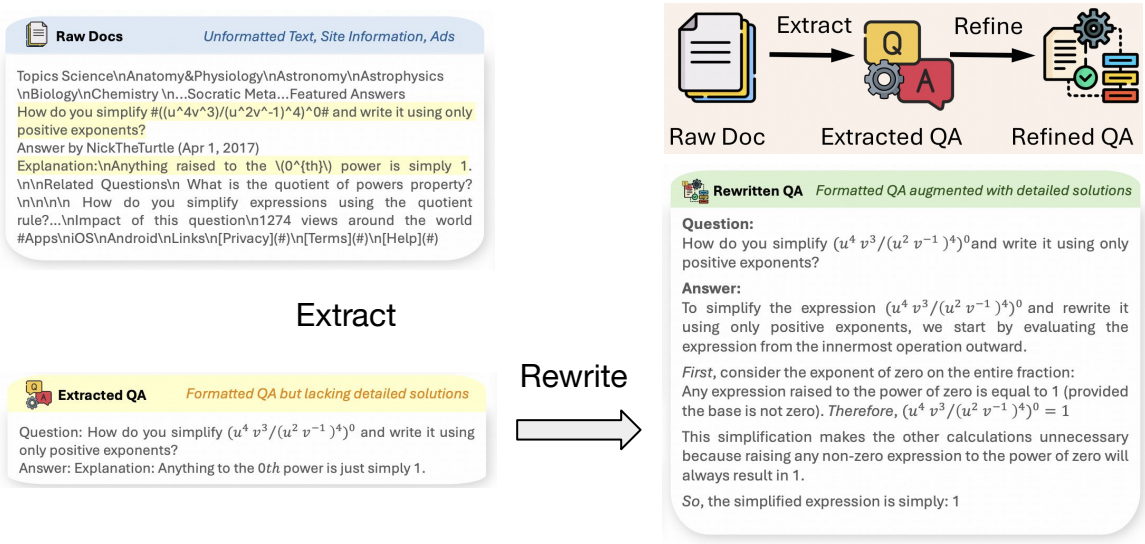


- To learn nontrivial capabilities via *sampling-based generation*, complex sampling strategies are needed
- Example 2: *Orca*
  - Generate 5 million SFT examples from GPT 3.5 and GPT 4 with synthetic reasoning traces
  - Use curriculum learning
  - Randomly sample different system prompts

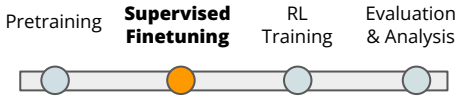
Orca: Progressive Learning from Complex Explanation Traces of GPT-4 (Mukherjee et al., 2023)

Supervised Finetuning: Beyond Distillation

- Simpler approach: *transformation of existing data* (e.g. “MAmmoTH2”)



MAmmoTH2: Scaling Instructions from the Web (Yue et al., 2023)

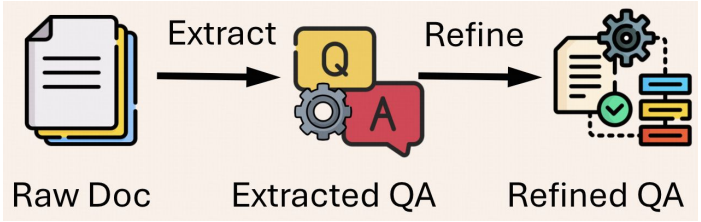


Supervised Finetuning: Beyond Distillation

- Simpler approach: *transformation of existing data* (e.g. “MAmmoTH2”)

MAmmoTH2: Scaling Instructions from the Web (Yue et al., 2023)

Supervised Finetuning: Beyond Distillation



Raw Docs *Unformatted Text, Site Information, Ads*

Topics Science\nAnatomy&Physiology\nAstronomy\nAstrophysics \nBiology\nChemistry \n...Socratic Meta...Featured Answers
How do you simplify #((u^4v^3)/(u^2v^-1)^4)^0# and write it using only positive exponents?
Answer by NickTheTurtle (Apr 1, 2017)
Explanation:\nAnything raised to the \(0^{th}\) power is simply 1. \n\nRelated Questions\n What is the quotient of powers property? \n\n\n\n How do you simplify expressions using the quotient rule?...\nImpact of this question\n1274 views around the world #Apps\niOS\nAndroid\nLinks\n[Privacy](#)\n[Terms](#)\n[Help](#)

Extract

Extracted QA *Formatted QA but lacking detailed solutions*

Question: How do you simplify $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$ and write it using only positive exponents?
Answer: Explanation: Anything to the $0th$ power is just simply 1.

Rewrite

Rewritten QA *Formatted QA augmented with detailed solutions*

Question:
How do you simplify $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$and write it using only positive exponents?

Answer:
To simplify the expression $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0$ and rewrite it using only positive exponents, we start by evaluating the expression from the innermost operation outward.

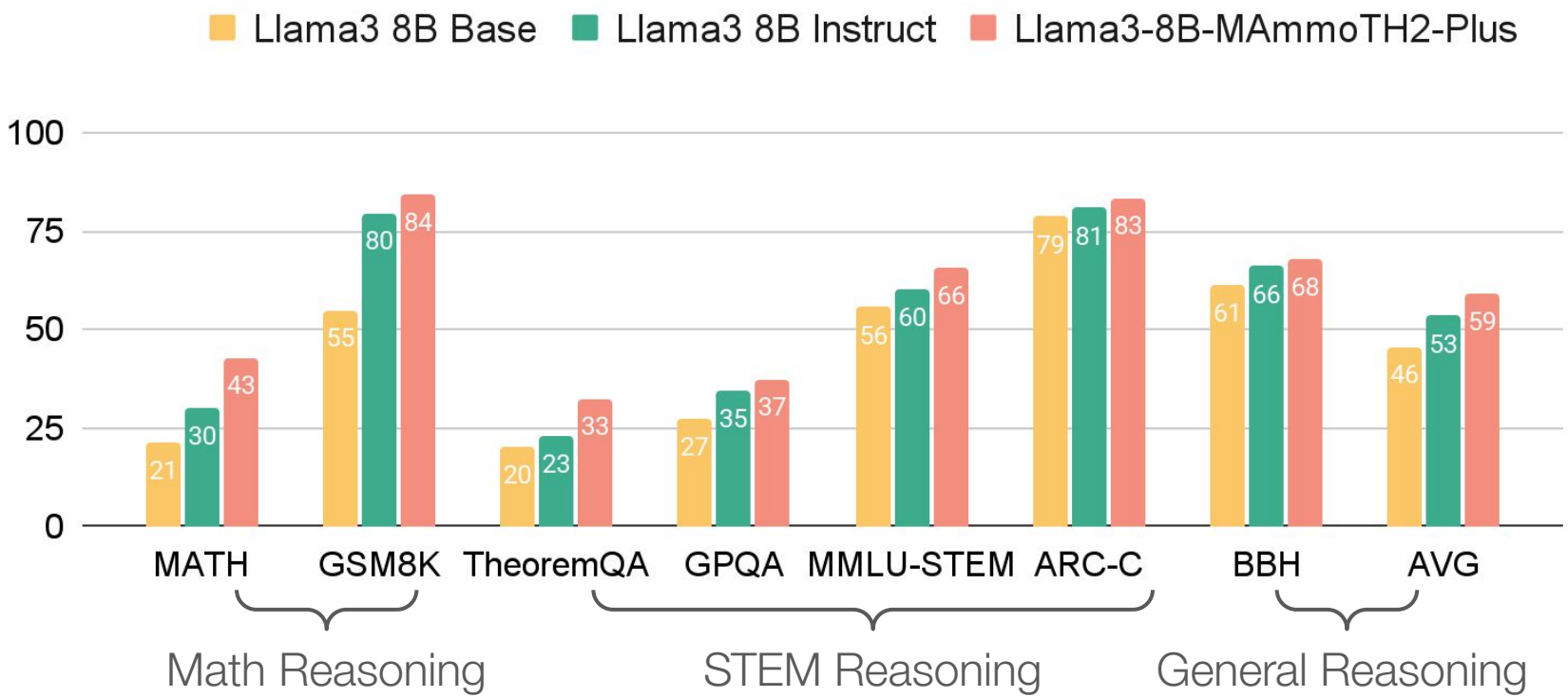
First, consider the exponent of zero on the entire fraction:
Any expression raised to the power of zero is equal to 1 (provided the base is not zero). *Therefore*, $(u^4\,v^3/(u^2\,v^{-1}\,)^4)^0 = 1$

This simplification makes the other calculations unnecessary because raising any non-zero expression to the power of zero will always result in 1.

So, the simplified expression is simply: 1

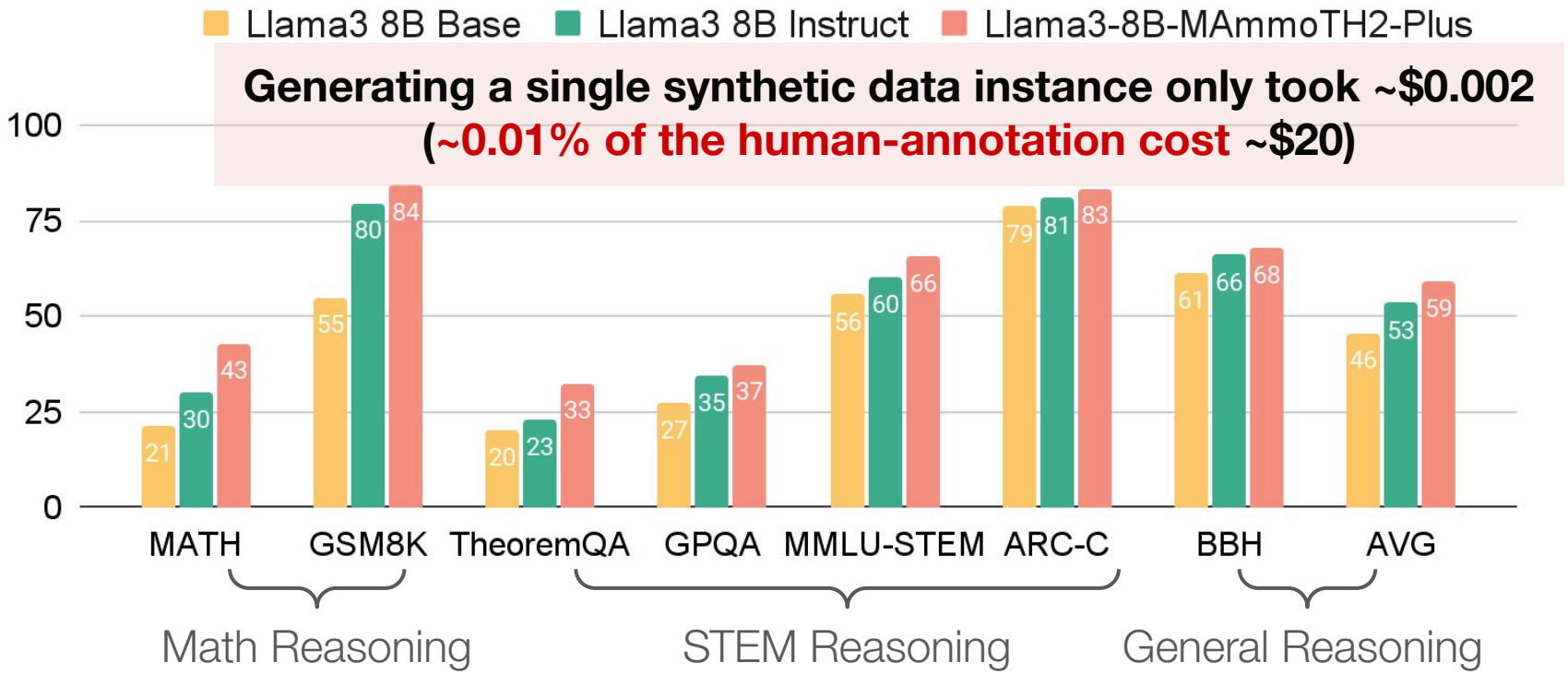
Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

MAmmoTH2 outperforms Llama3-Instruct on science/math/reasoning



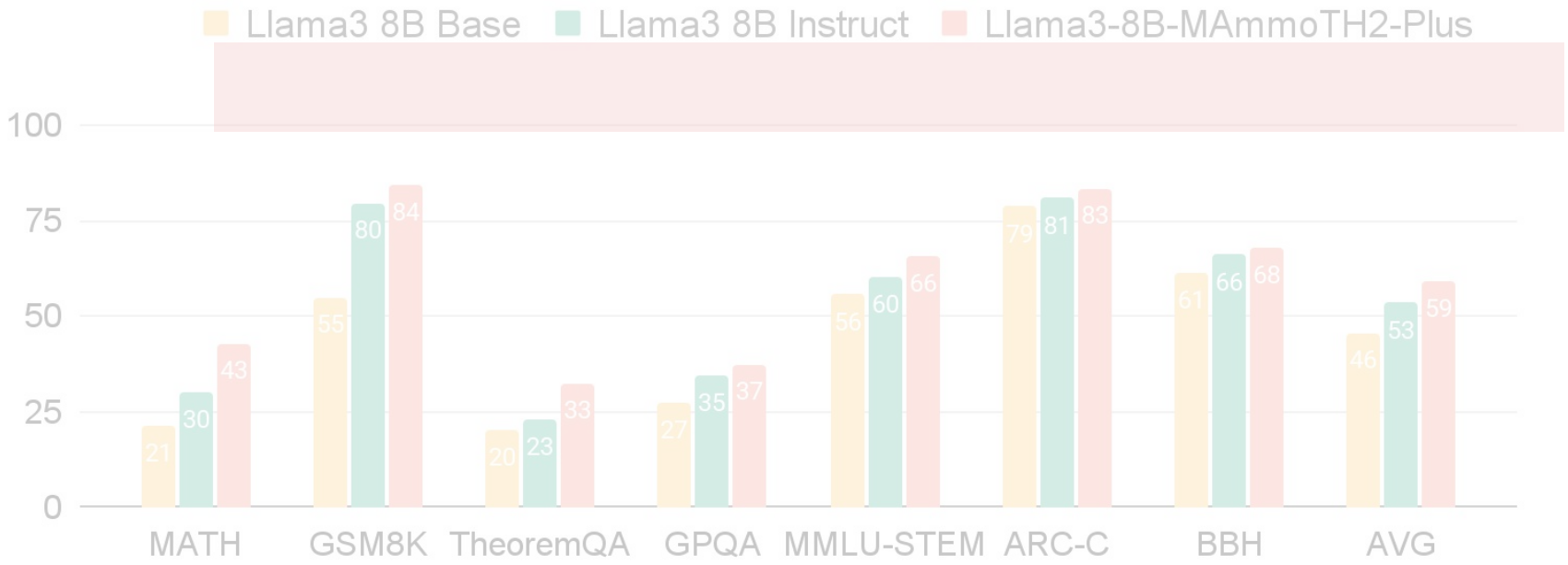
Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

MAmmoTH2 outperforms Llama3-Instruct on science/math/reasoning



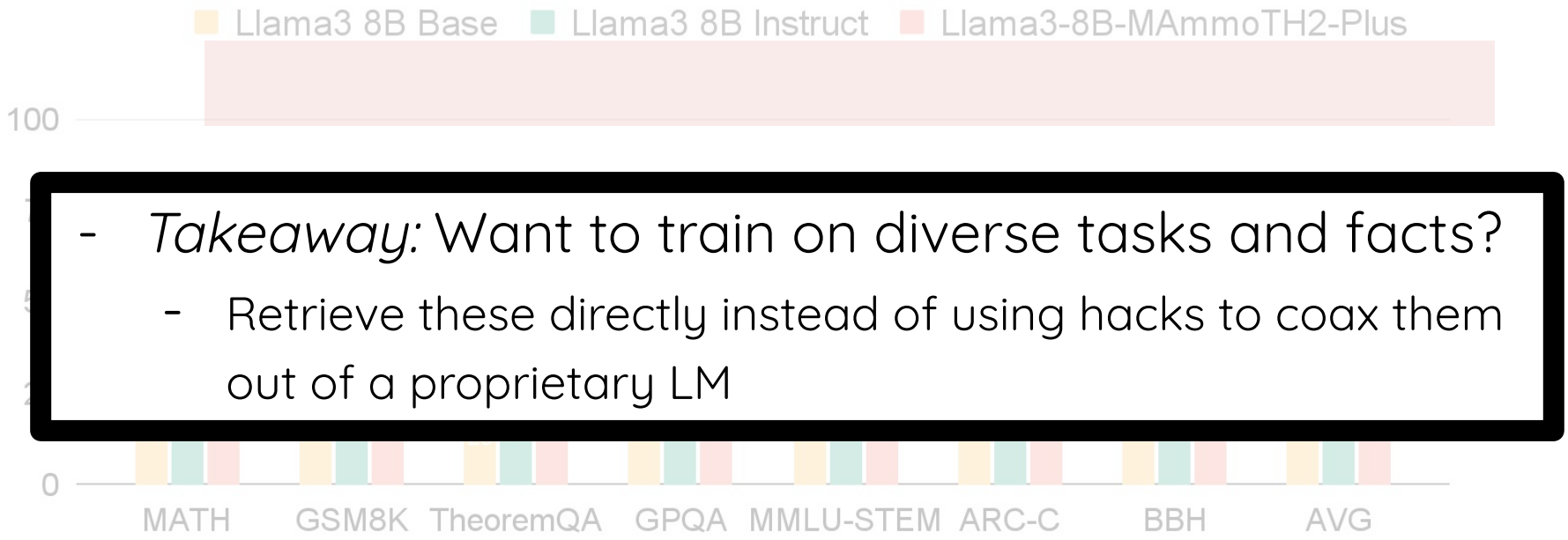
Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

MAmmoTH2 outperforms Llama3-Instruct on science/math/reasoning



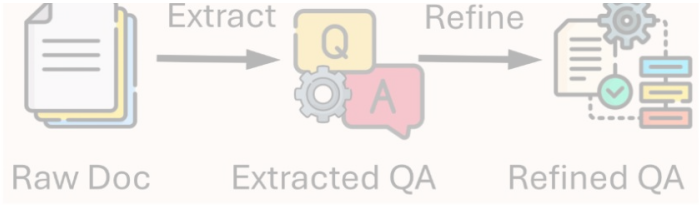
Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

Supervised Finetuning: Beyond Distillation



- *Takeaway:* Want to train on diverse tasks and facts?
  - Retrieve these directly instead of using hacks to coax them out of a proprietary LM

Supervised Finetuning: Beyond Distillation



- *Takeaway:* Want to train on diverse tasks and facts?
  - Retrieve these directly instead of using hacks to coax them out of a proprietary LM

Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

Reinforcement Learning

Goals of RL (non-exhaustive):

- Learn from minimal supervision
- Learn from negative examples (e.g. harmful behavior)
- Adapt models to their own token distribution rather than text written by others (“exposure bias”)

Magpie: Alignment Data Synthesis from Scratch by Prompting Aligned LLMs with Nothing (Xu et al., 2024)

RL from Synthetic Feedback

Intuition: it is often easier to *verify* that an utterance is good than to generate that *utterance*

- "The Generator-Verifier Gap"

All Roads Lead to Likelihood: The Value of Reinforcement Learning in Fine-Tuning (Swamy et al., 2025)

RL from Synthetic Feedback

- Synthetic feedback is standard in RL
- For example, *reward models* are literally generators of synthetic rewards

Scalable agent alignment via reward modeling: a research direction (Leike et al., 2018)

RL from Synthetic Feedback

- Synthetic feedback is standard in RL
- For example, *reward models* are literally generators of synthetic rewards
    - Train a model to imitate human preferences
    - Then, use the model to grade sampled responses during RL

Scalable agent alignment via reward modeling: a research direction (Leike et al., 2018)

RL from Synthetic Feedback

- Synthetic feedback is standard in RL
- For example, *reward models* are literally generators of synthetic rewards

    - Train a model to imitate human preferences
        - Typically using the *Bradley-Terry model* to learn to generate continuous scores from prefrence pairs
    - Then, use the model to grade sampled responses during RL

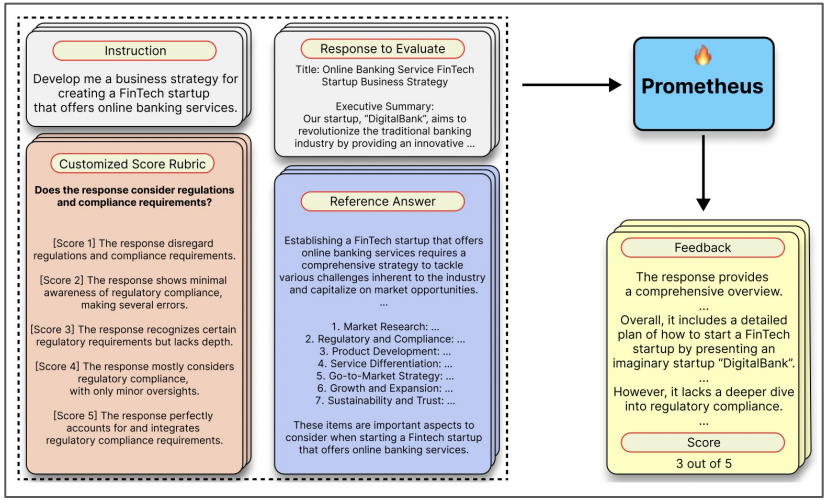
Scalable agent alignment via reward modeling: a research direction (Leike et al., 2018)

RL from Synthetic Feedback

- Synthetic feedback for RL need not come from Bradley-Terry models

RL from Synthetic Feedback

- Synthetic feedback for RL need not come from Bradley-Terry models, e.g. LLM-as-a-judge
  - A trained judge that, given a rubric, outputs scores and critiques



[Prometheus: Inducing Fine-grained Evaluation Capability in Language Models (Kim et al., 2024)]

RL from Synthetic Feedback

- Synthetic feedback doesn't require training data at all
  - You can just prompt a judge model to rate a response from 1 to 5 on a criterion (e.g. UltraFeedback)

UltraFeedback: Boosting Language Models with Scaled AI Feedback (Cui et al., 2023)

RL from Synthetic Feedback

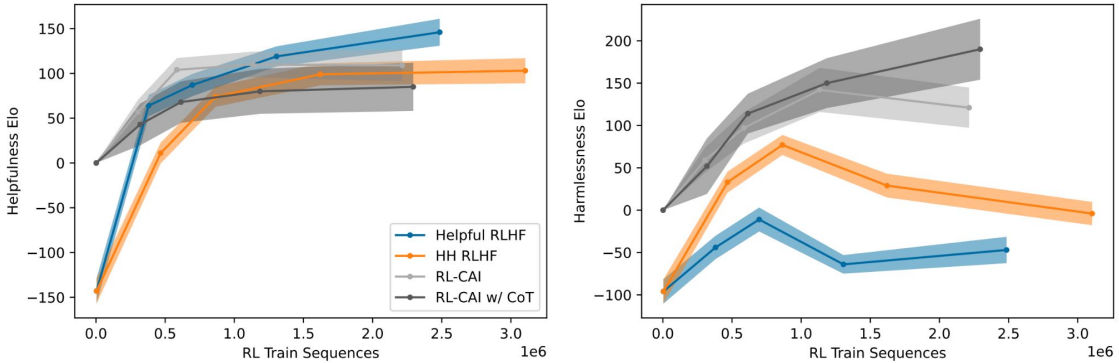
- Synthetic feedback doesn't require training data at all
  - Or, with preference optimization, ask the judge to choose which of a pair of responses is better, according to a criterion

```
Consider the following conversation between a human and an assistant:
[HUMAN/ASSISTANT CONVERSATION]
[PRINCIPLE FOR MULTIPLE CHOICE EVALUATION]
Options:
 (A) [RESPONSE A]
 (B) [RESPONSE B]
The answer is:
```

Constitutional AI: Harmlessness from AI Feedback (Bai et al., 2022)

RL from Synthetic Feedback

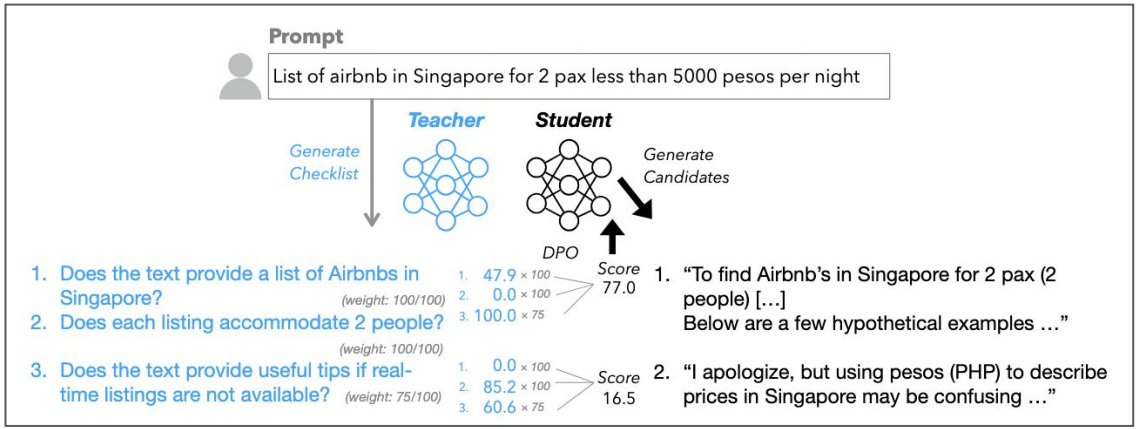
- Synthetic feedback doesn't require training data at all
  - Or, with preference optimization, ask the judge to choose which of a pair of responses is better, according to a criterion
  - The set of criteria can become complex (e.g. 16 criteria in "Constitutional AI")



[Constitutional AI: Harmlessness from AI Feedback (Bai et al., 2022)]

RL with Feedback from Flexible Criteria

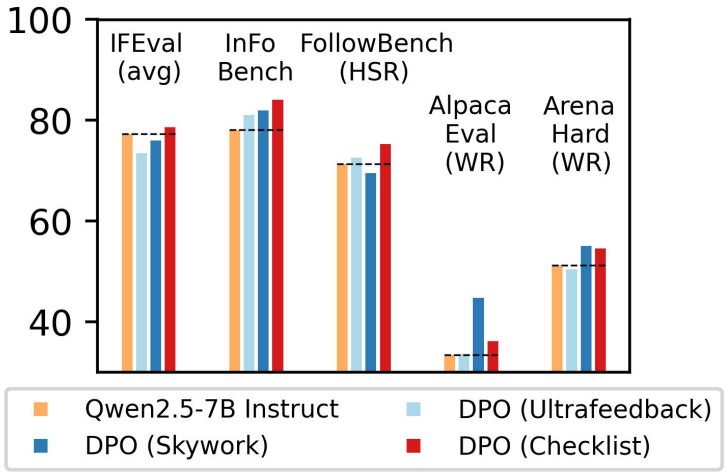
- Can we list all possible aspects of response quality?
- Solution: generate *checklists* for each prompt



Checklists Are Better Than Reward Models For Aligning Language Models (Viswanathan et al., 2025)

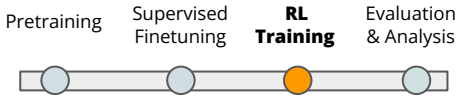
RL with Feedback from Flexible Criteria

- Solution: generate *checklists* for each prompt
- Better than reward models or judges at teaching instruction-following



Checklists Are Better Than Reward Models For Aligning Language Models (Viswanathan et al., 2025)

Evaluation of Synthetic Feedback



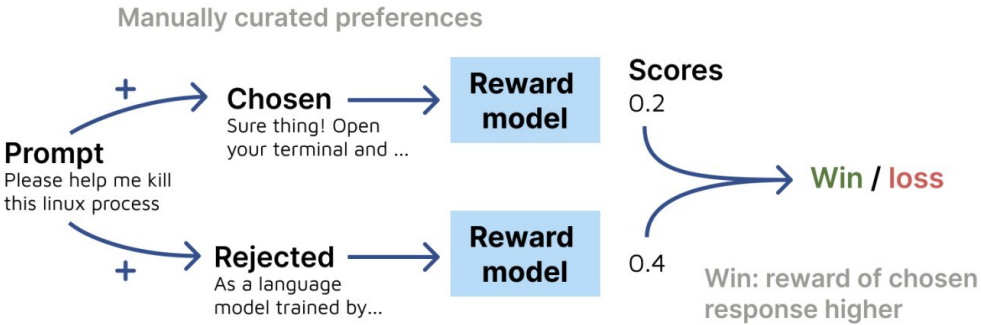
How do you know what kind of judge to use?

Does it even matter?

Evaluation of Synthetic Feedback

Three ways to evaluate a “judge”:

1. Agreement with human preferences (e.g. RewardBench)
2.
3.



[Evaluating Reward Models for Language Modeling (Lambert et al 2024)]

Evaluation of Synthetic Feedback

Three ways to evaluate a “judge”:

1. Agreement with human preferences (e.g. RewardBench)
2. **Agreement with generic benchmarks (via reranking)**
3.

Evaluation of Synthetic Feedback

Three ways to evaluate a “judge”:

1. Agreement with human preferences (e.g. RewardBench)
2. **Agreement with generic benchmarks (via reranking)**
3.
   - Choose a benchmark you care about
   - For every question, sample 16 responses
   - Use your judge to choose the best response
   - This should improve your score on the benchmark

Evaluation of Synthetic Feedback

Three ways to evaluate a “judge”:

1. Agreement with human preferences (e.g. RewardBench)
2. Agreement with generic benchmarks (via reranking)
3. **Effectiveness in RL pipelines**

Evaluation of Synthetic Feedback

Three ways to evaluate a “judge”:

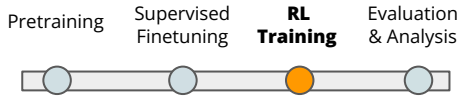
1. Agreement with human preferences (e.g. RewardBench)
2. Agreement with generic benchmarks (via reranking)
3. **Effectiveness in RL pipelines**

   - Choose an RL algorithm, a model, and some prompts
   - Train the model to maximize your reward
   - Evaluate it on benchmarks of your choosing

Evaluation of Synthetic Feedback

Problem: these things are not always correlated

1. Agreement with human preferences (e.g. RewardBench)
2. Agreement with generic benchmarks (via reranking)
3. Effectiveness in RL pipelines

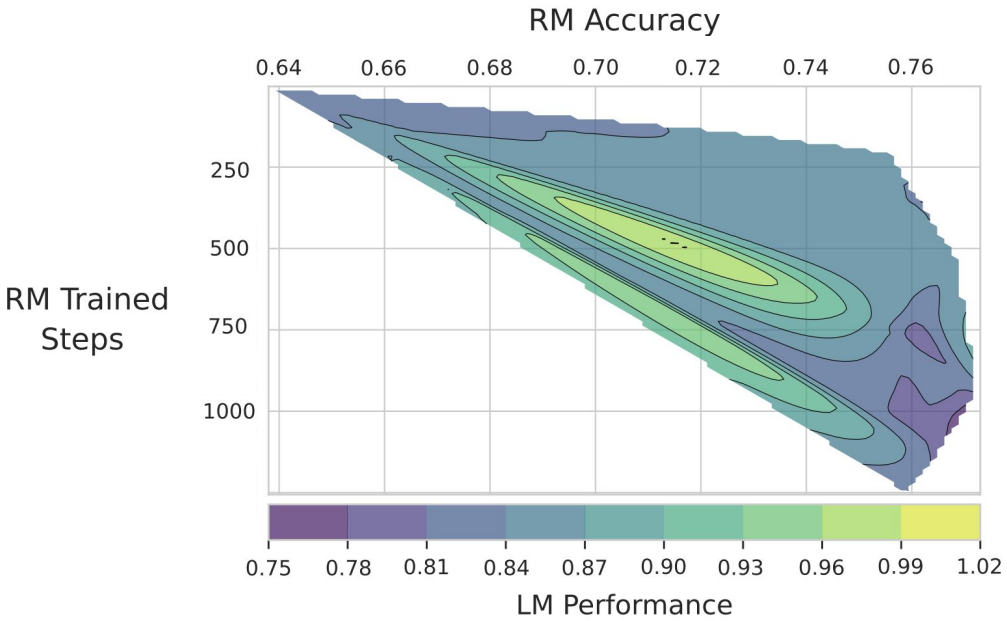


Open Problem in RL from Synthetic Feedback

- What makes a judge a good teacher for RL?

Open Problem in RL from Synthetic Feedback

- What makes a judge a good teacher for RL? *Accuracy?*

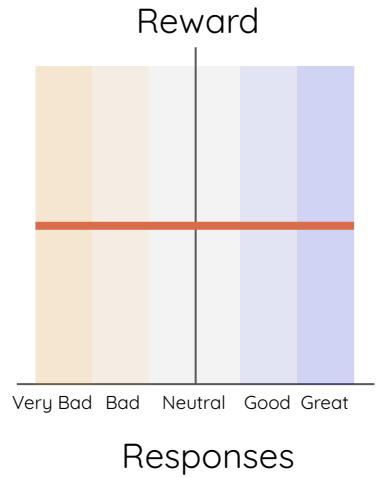


[The Accuracy Paradox in RLHF: When Better Reward Models Don't Yield Better Language Models (Chen et al., 2024)]

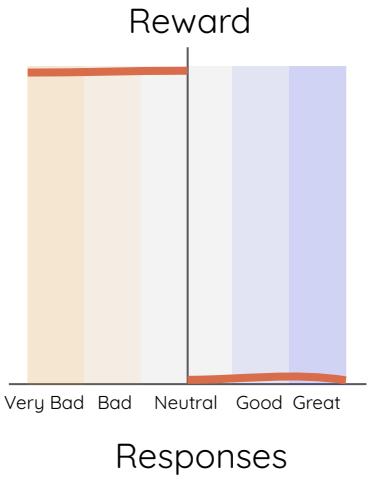
Open Problem in RL from Synthetic Feedback

- What makes a judge a good teacher for RL? *Variance?*

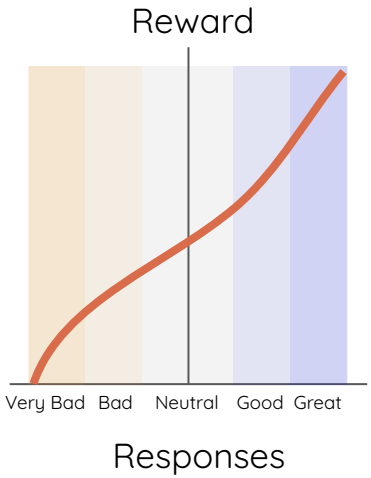
Low Acc, Low Var



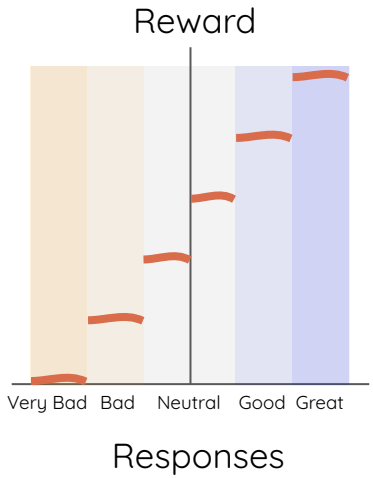
Low Acc, High Var



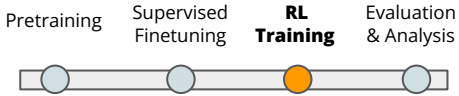
High Acc, Low Var



High Acc, High Var



[What Makes a Reward Model a Good Teacher? An Optimization Perspective (Razin et al., 2025)]

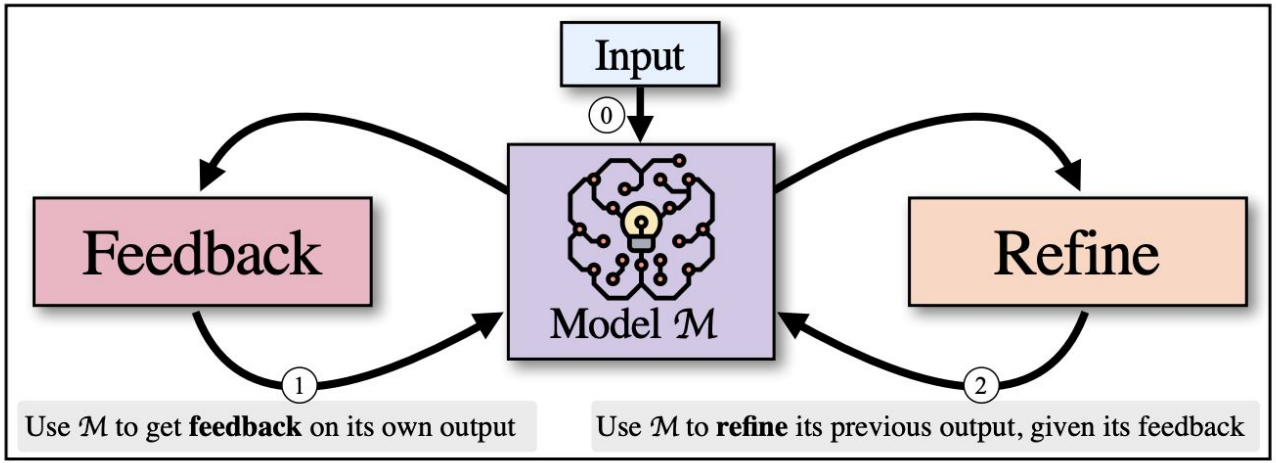


Open Problem in RL from Synthetic Feedback

- What makes a judge a good teacher for RL? ***Unclear***

Non-RL Feedback (Critiques)

- Treating synthetic feedback as a turn in a conversation



Self-Refine: Iterative Refinement with Self-Feedback (Madaan et al., 2023)

Non-RL Feedback (Critiques)

- Treating synthetic feedback as a turn in a conversation
  - This can be effective when prompting with big models

	GPT-3.5		GPT-4	
Task	Base	+SELF-REFINE	Base	+SELF-REFINE
Sentiment Reversal	8.8	**30.4** (↑21.6)	3.8	**36.2** (↑32.4)
Dialogue Response	36.4	**63.6** (↑27.2)	25.4	**74.6** (↑49.2)
Code Optimization	14.8	**23.0** (↑8.2)	27.3	**36.0** (↑8.7)
Code Readability	37.4	**51.3** (↑13.9)	27.4	**56.2** (↑28.8)
Math Reasoning	**64.1**	**64.1** (0)	92.9	**93.1** (↑0.2)
Acronym Generation	41.6	**56.4** (↑14.8)	30.4	**56.0** (↑25.6)
Constrained Generation	16.0	**39.7** (↑23.7)	4.4	**61.3** (↑56.9)

Self-Refine: Iterative Refinement with Self-Feedback (Madaan et al., 2023)

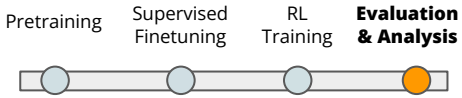
Non-RL Feedback (Critiques)

- Treating synthetic feedback as a turn in a conversation
  - This can be effective when prompting with **big** models

	GPT-3.5		GPT-4	
Task	Base	+SELF-REFINE	Base	+SELF-REFINE
Sentiment Reversal	8.8	**30.4** (↑21.6)	3.8	**36.2** (↑32.4)
Dialogue Response	36.4	**63.6** (↑27.2)	25.4	**74.6** (↑49.2)
Code Optimization	14.8	**23.0** (↑8.2)	27.3	**36.0** (↑8.7)
Code Readability	37.4	**51.3** (↑13.9)	27.4	**56.2** (↑28.8)
Math Reasoning	**64.1**	**64.1** (0)	92.9	**93.1** (↑0.2)
Acronym Generation	41.6	**56.4** (↑14.8)	30.4	**56.0** (↑25.6)
Constrained Generation	16.0	**39.7** (↑23.7)	4.4	**61.3** (↑56.9)

Self-Refine: Iterative Refinement with Self-Feedback (Madaan et al., 2023)

Evaluation and Analysis

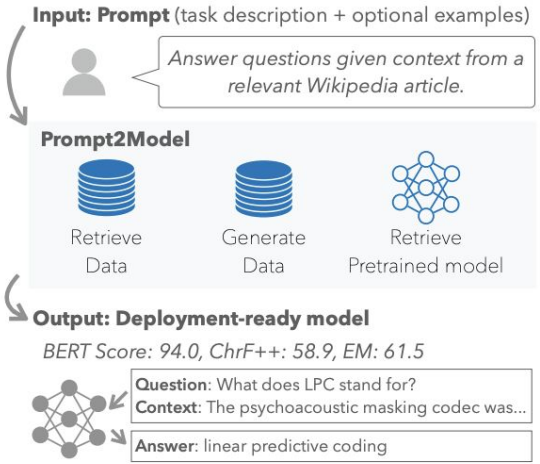


Goals of Evaluation:

- Test readiness of a model for deployment
- Discover areas of improvement to make a model better
- Learn fundamental insights about language or machine learning

Generating Synthetic Eval Data

- The Prompt2Model library synthesizes training and *evaluation* data

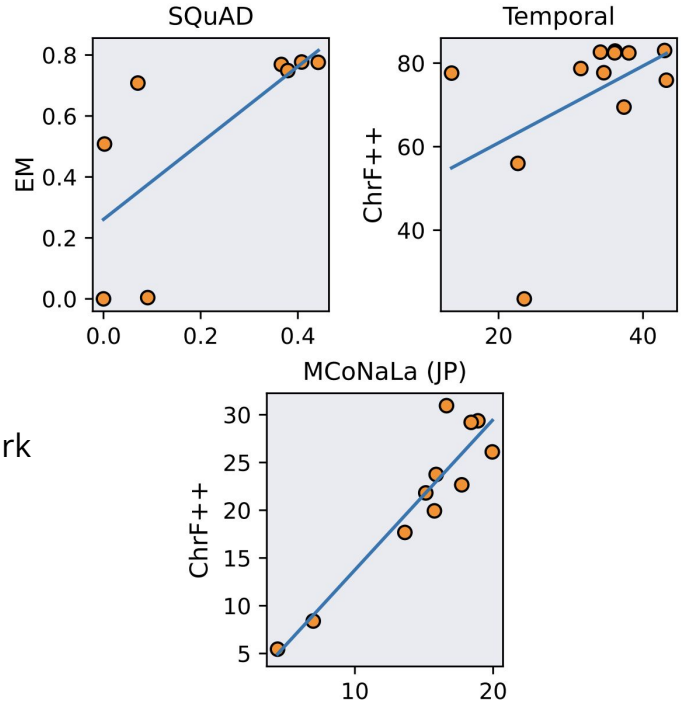


https://github.com/neulab/prompt2model

Prompt2Model: Generating Deployable Models from Natural Language Instructions (Viswanathan et al., 2023)

Generating Synthetic Eval Data

- Synthetic evaluation data consistently overestimates true performance



Y-axis: Synthetic Benchmark Accuracy

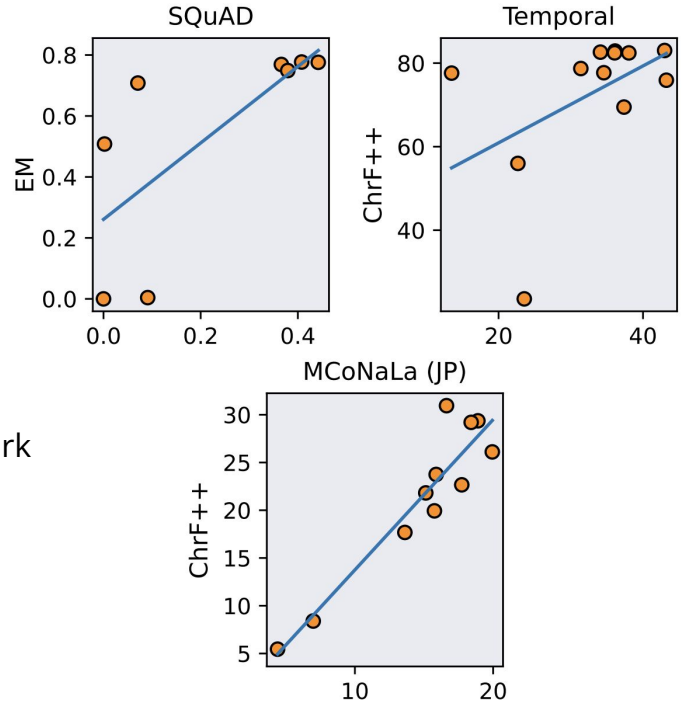
X-axis: Real benchmark accuracy

https://github.com/neulab/prompt2model

Prompt2Model: Generating Deployable Models from Natural Language Instructions (Viswanathan et al., 2023)

Generating Synthetic Eval Data

- Synthetic evaluation data consistently overestimates true performance



Y-axis: Synthetic Benchmark Accuracy

X-axis: Real benchmark accuracy

https://github.com/neulab/prompt2model

Prompt2Model: Generating Deployable Models from Natural Language Instructions (Viswanathan et al., 2023)

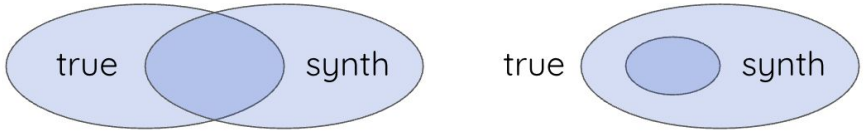
Generating Synthetic Eval Data

- Synthetic evaluation data consistently overestimates true performance

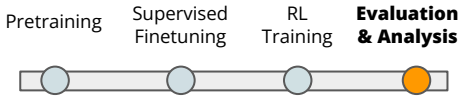
Recall from earlier slides

The input distribution may be **off**, or **not diverse enough**

$P_{true}(x) \neq P_{synth}(x)$



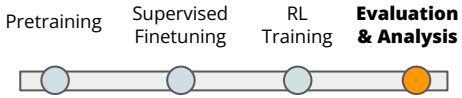
Generating Adversarial Eval Data



- *If you have an accurate mistake classifier:*
  1. Train a generator to generate prompts to elicit mistakes via RL

Red Teaming Language Models with Language Models (Perez et al., 2022)

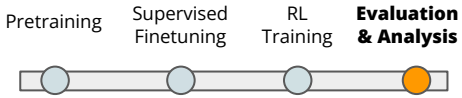
Generating Adversarial Eval Data



- *If you have an accurate mistake classifier:*
  1. Train a generator to generate prompts to elicit mistakes via RL
  2. Find inputs that lead your *target model* to make mistakes by rejection sampling via the *red model*

Red Teaming Language Models with Language Models (Perez et al., 2022)

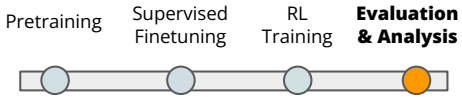
Generating Adversarial Eval Data



- *If you have an accurate mistake classifier:*
  1. Train a generator to generate prompts to elicit mistakes via RL
  2. Find inputs that lead your *target model* to make mistakes by rejection sampling via the *red model*
- Synthetic data generation helps cheaply obtain adversarial inputs in non-critical situations

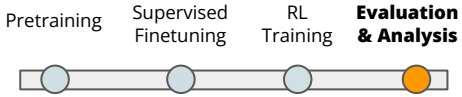
Red Teaming Language Models with Language Models (Perez et al., 2022)

Synthetic Data for Analyzing LMs



- Rule-based synthetic data has been used for probing and understanding models since the advent of NLP

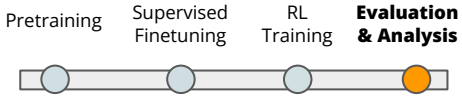
Synthetic Data for Analyzing LMs



- Modern example: Kassner, Krojer, and Schutze (2020) generated a synthetic pretraining corpus generated by *synthetic facts* (e.g. “jupiter is big”) and logical rules (e.g. “jupiter is not small”)

Are Pretrained Language Models Symbolic Reasoners Over Knowledge? (Kassner et al., 2020)

Synthetic Data for Analyzing LMs



- Modern example: Kassner, Krojer, and Schutze (2020) generated a synthetic pretraining corpus generated by *synthetic facts* (e.g. “jupiter is big”) and logical rules (e.g. “jupiter is not small”)
- Pretrained BERT from scratch on this corpus
- Discovered that BERT struggles with two-hop reasoning

Are Pretrained Language Models Symbolic Reasoners Over Knowledge? (Kassner et al., 2020)

How is synthetic data used?

Supporting fundamental language modeling algorithms

Supporting scenario-specific, end-user applications

How is synthetic data used?

Supporting fundamental language modeling algorithms

Supporting scenario-specific, end-user applications

How can we use synthetic data: Applications

Part IV

Scenario-specific Applications

Xiang Yue

Postdoc Researcher

Carnegie Mellon University

https://xiangyue9607.github.io/

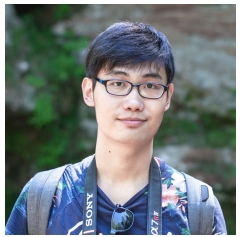


Table of Content

Scenario-specific Applications

- Reasoning
- Code Generation
- Tool use and Agents
- Multilingual and Multimodal
- Conclusion and Future Directions

Reasoning

Solving Complex Reasoning Problems with LLMs

OPENAI / ARTIFICIAL INTELLIGENCE / TECH

OpenAI releases o1, its first 'reas

OpenAI o1 Model Sets New Math and

Understanding R1 and DeepSeek

R1 belongs to a new category of AI models known as "reasoning models," with OpenAI's o1 being the most well-known example. What makes reasoning models special is their approach to problem-solving. Rather than generating immediate responses, they employ an internal reasoning process that mirrors human trains of thought.

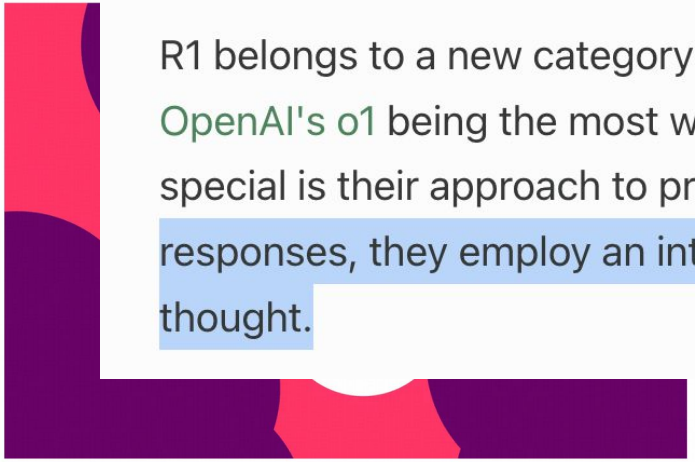


Image: The Verge

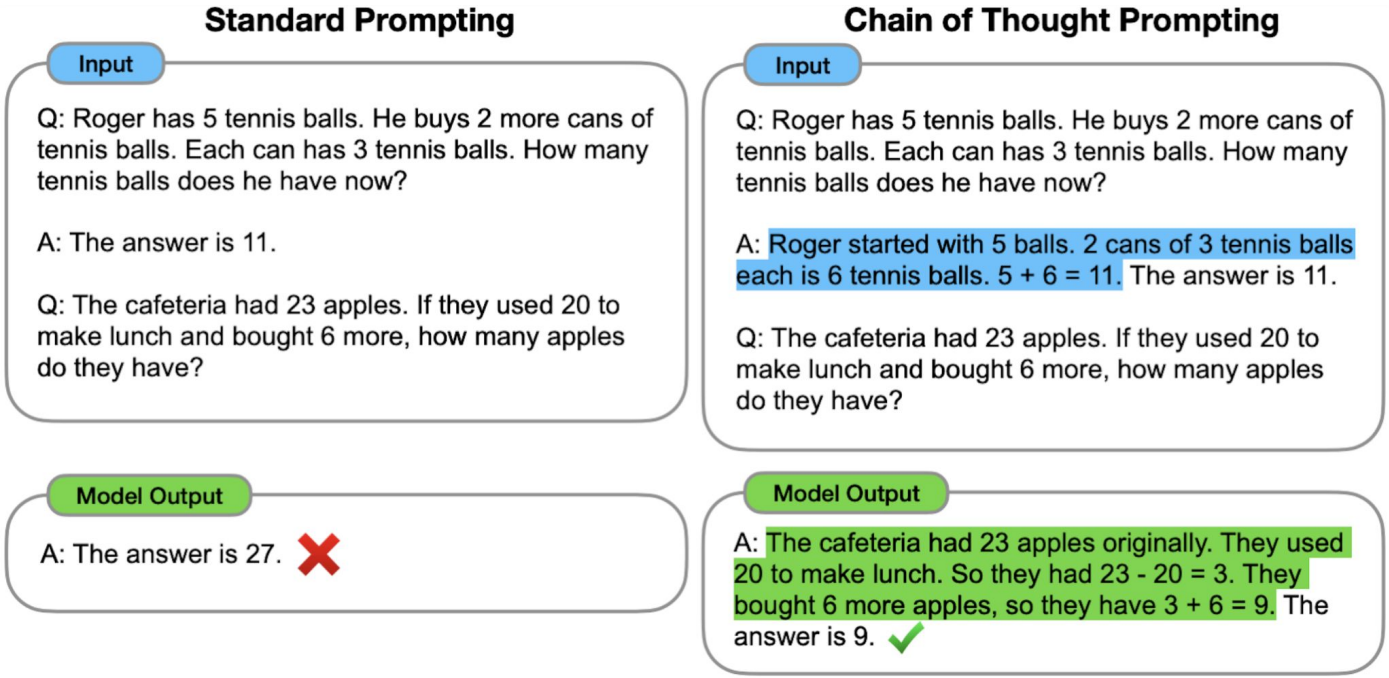
OpenAI is releasing a new model called o1, the first in a planne
of "reasoning" models that have been trained to answer more c
questions, faster than a human can. It's being released alongsid
mini, a smaller, cheaper version. And yes, if you're steeped in AI rumors:

IMAGE CREDITS: PERESMEH / GETTY IMAGES

benchmarks

Kyle Wiggers 2:27 PM PST · January 27, 2025

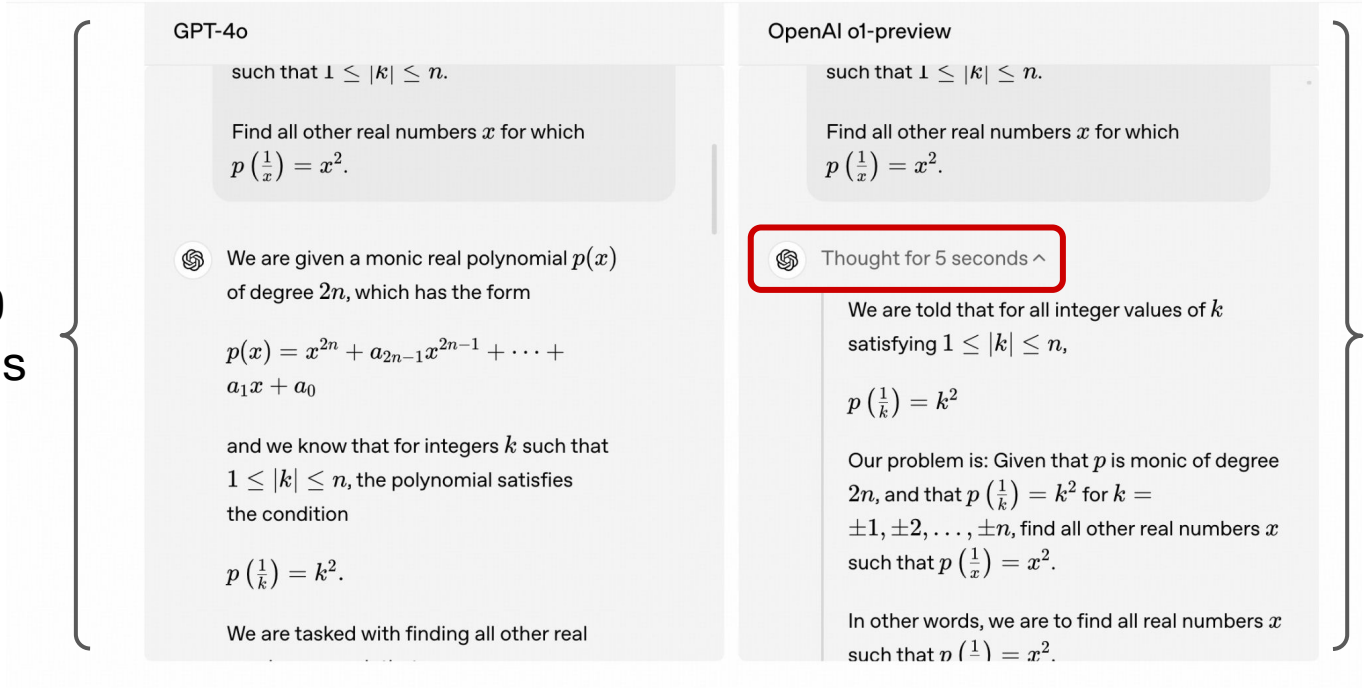
Reason with Rationales: Chain-of-thought (CoT)



Wei, J. (2022). Chain-of-thought prompting elicits reasoning in large language models. NeurIPS 2022

Scaling up *Inference* Compute with Long CoT

~200 Tokens

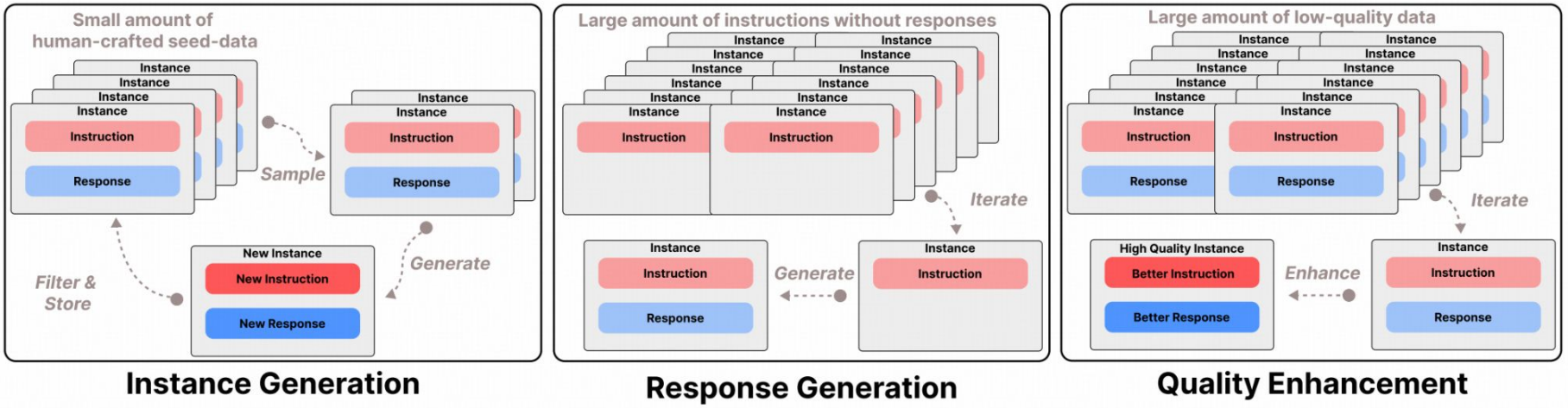


~2000 Tokens

OpenAI o1 generates longer CoTs for reasoning

How do we generate synthetic reasoning data?

A Unified View of Synthetic Text Generation

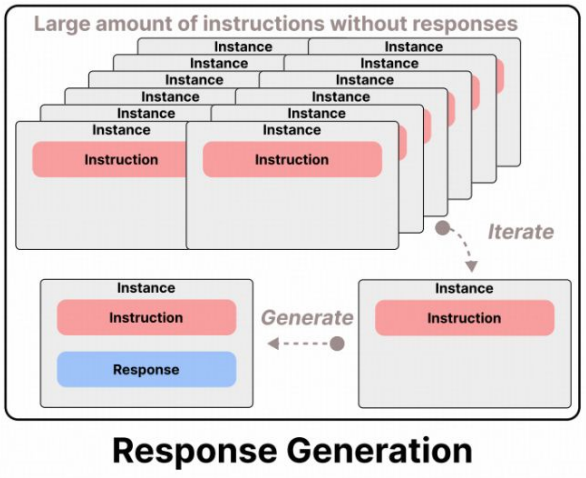


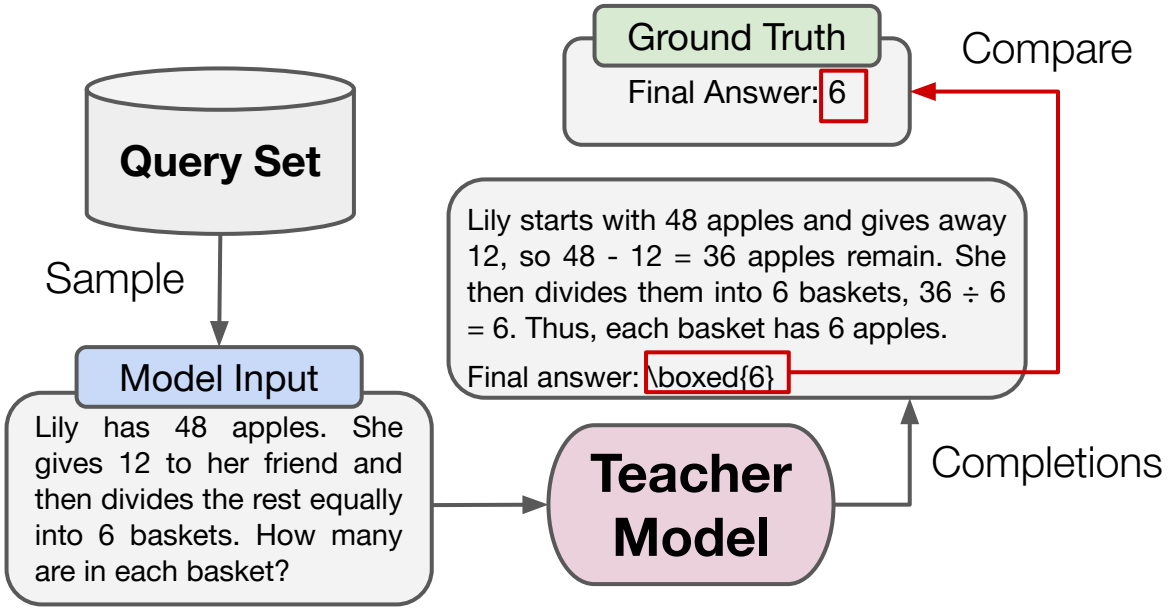
Apply an existing LLM θ to the prompt and seed data D to produce text

$$D_{\mathrm{s}} = \underbrace{\mathcal{O}\left(\mathrm{Prompt}_{s}\left(D\right);\theta\right)}_{\text{LLM output}}$$

Kim, S., Suk, J., Yue, X.,... & Neubig, G. (2024). Evaluating Language Models as Synthetic Data Generators. *ACL 2025*

Reasoning Data: Response Generation





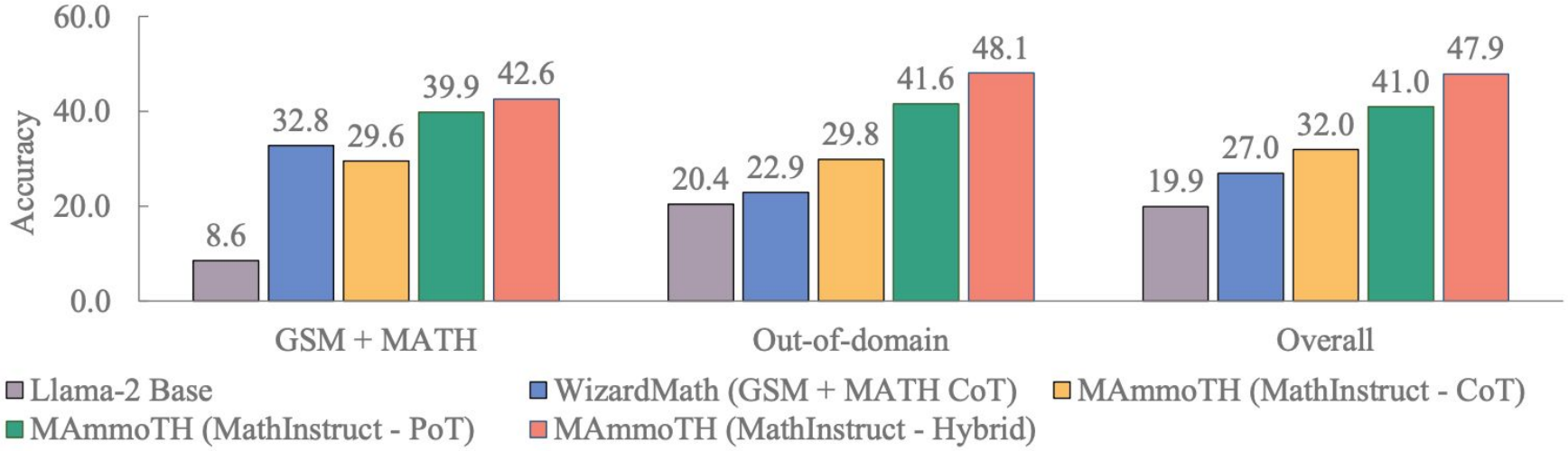
Kim, S., Suk, J., Yue, X.,... & Neubig, G. (2024). Evaluating Language Models as Synthetic Data Generators. *ACL 2025*

MAmmoTH: MathInstruct

Training Dataset	Type	Annotation	# Samples	Characteristics	Fields
GSM8K (Cobbe et al., 2021)	CoT	Human	7K	Grade Schol Exam	
GSM8K-RFT (Yuan et al., 2023)	CoT	Llama	28K	Llama + Validated	
AQuA-RAT (Ling et al., 2017)	CoT	Human	90K	GRE/GMAT Exam	
MATH (Hendrycks et al., 2021b)	CoT	Human	7K	Math Competition	
TheoremQA (Chen et al., 2023) ☆	CoT	GPT-4	600	GPT4 + Validated	
Camel-Math (Li et al., 2023a)	CoT	GPT-4	50K	GPT4 (Unvalidated)	
College-Math ☆	CoT	GPT-4	1.8K	GPT4 (Unvalidated)	
GSM8K ☆	PoT	GPT4	14K	GPT4 + Validated	
AQuA-RAT ☆	PoT	GPT4	9.7K	GPT4 + Validated	
MATH ☆	PoT	GPT4	7K	GPT4 + Validated	
TheoremQA ☆	PoT	GPT4	700	GPT4 + Validated	
MathQA (Amini et al., 2019)	PoT	Human	25K	AQuA-RAT Subset	
NumGLUE (Mishra et al., 2022a)	PoT	Human	13K	Lila Annotated	
`MathInstruct`			260K		

Yue, X*., ... & Chen, W*. **MAmmoTH**: Building math generalist models through hybrid instruction tuning. **ICLR 2024**

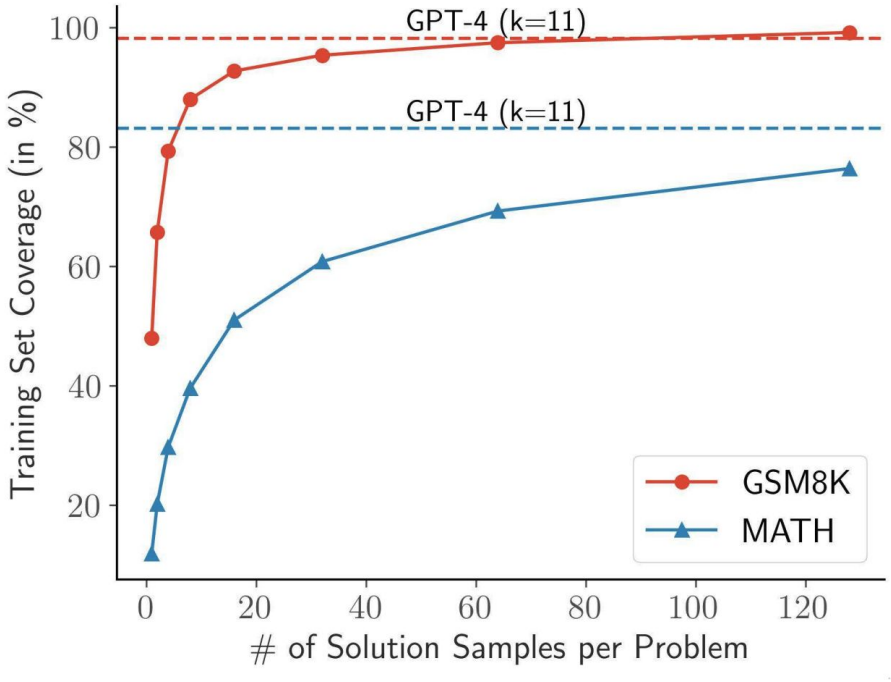
MAmmoTH: MathInstruct



Hybrid of thoughts improves the reasoning performance

Yue, X*., ... & Chen, W*. **MAmmoTH**: Building math generalist models through hybrid instruction tuning. **ICLR 2024**

OpenMathInstruct-1

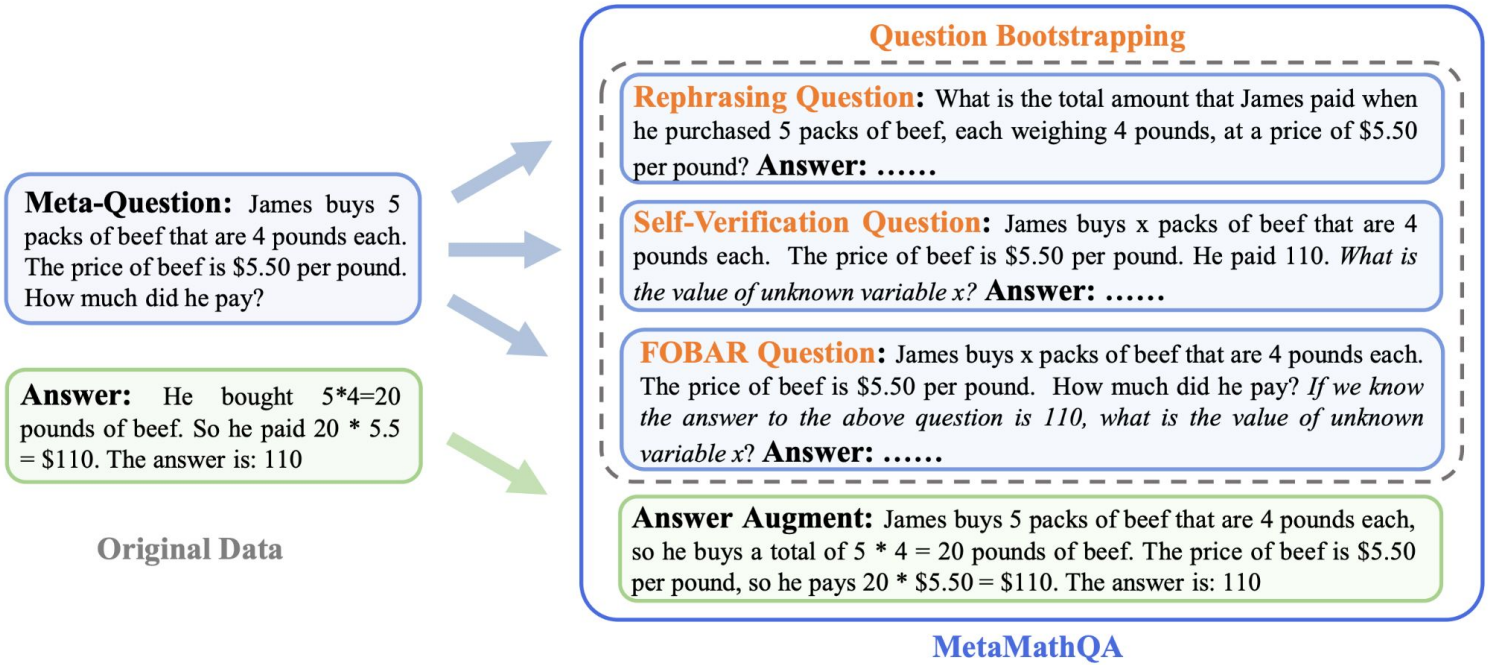


Prompt	MATH # Samples	MATH # Unique Solns.	MATH Coverage (in %)
Default	224	177K	80.1
+ Subj	224	191K	80.1
Mask-Text	224	192K	85.9
+ Subj	224	227K	87.5
Total	896	787K	93.0

Prompt	GSM8K # Samples	GSM8K # Unique Solns.	GSM8K Coverage (in %)
Default	128	434K	99.1
+ Subj	-	-	-
Mask-Text	128	602K	99.9
+ Subj	-	-	-
Total	256	1036K	99.9

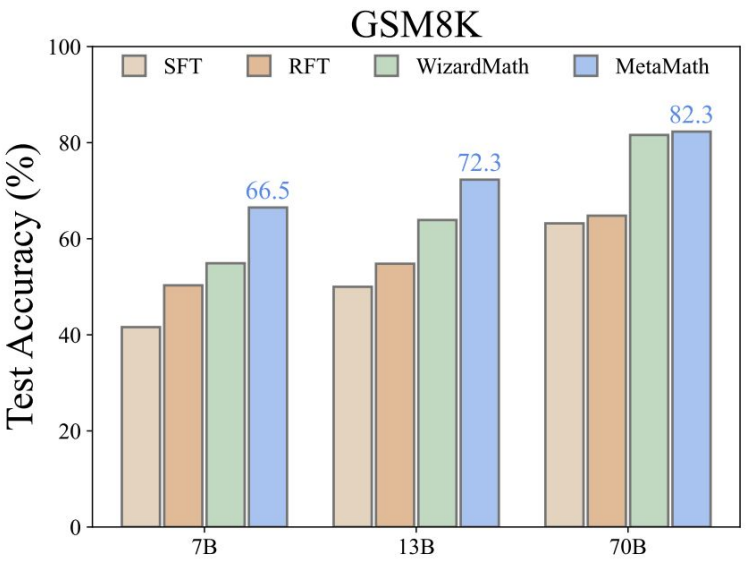
Toshniwal, Shubham, et al. "Openmathinstruct-1: A 1.8 million math instruction tuning dataset." *NeurIPS 2024*

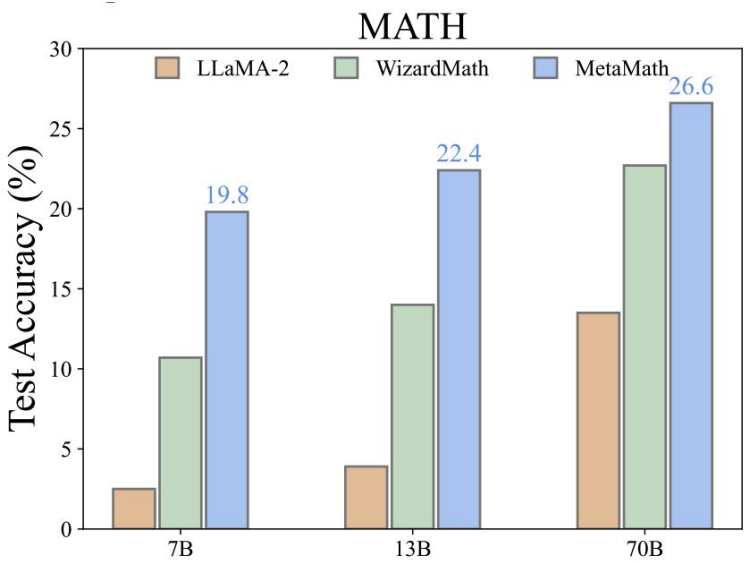
MetaMath



Yu, Longhui, et al. "Metamath: Bootstrap your own mathematical questions for large language models." *ICLR 2024*

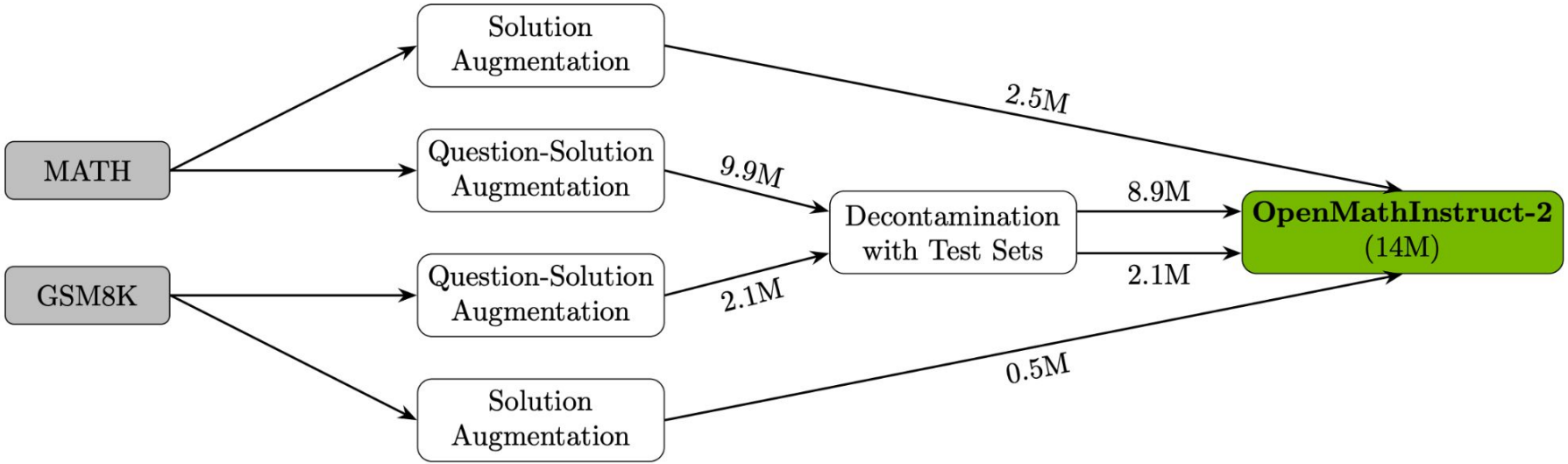
MetaMath





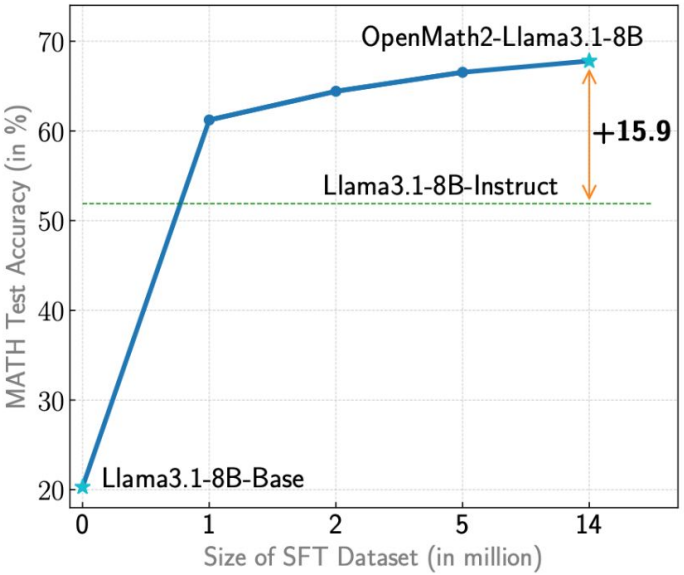
Yu, Longhui, et al. "Metamath: Bootstrap your own mathematical questions for large language models." *ICLR 2024*

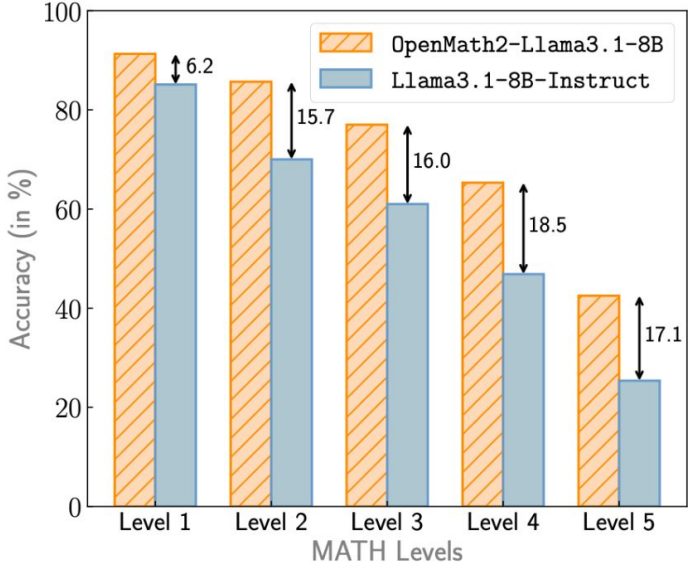
OpenMathInstruct-2



Toshniwal, Shubham, et al. "OpenMathInstruct-2: Accelerating AI for Math with Massive Open-Source Instruction Data." *ICLR 2025*

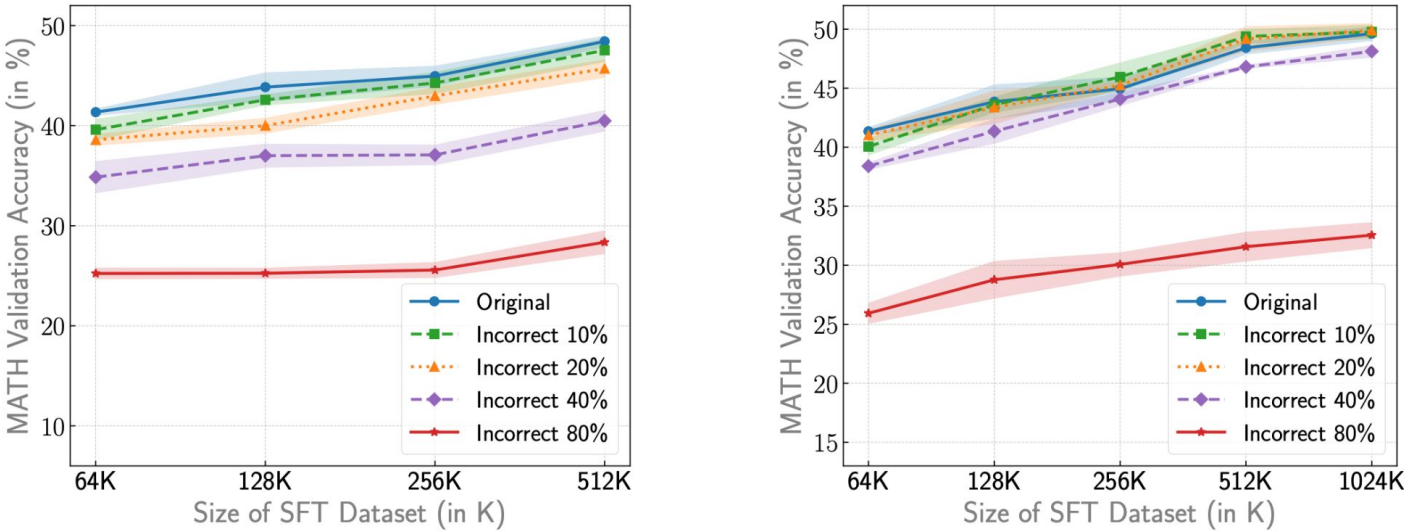
OpenMathInstruct-2





Toshniwal, Shubham, et al. "OpenMathInstruct-2: Accelerating AI for Math with Massive Open-Source Instruction Data." *ICLR 2025*

OpenMathInstruct-2



(a) Adding wrong-answer solutions.

(b) Correct solutions mismatched with questions

Figure 5: Impact of low-quality solutions on the SFT performance.

Toshniwal, Shubham, et al. "OpenMathInstruct-2: Accelerating AI for Math with Massive Open-Source Instruction Data." *ICLR 2025*

OpenMathInstruct-2

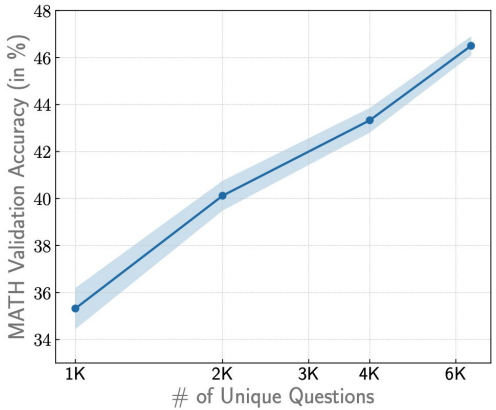
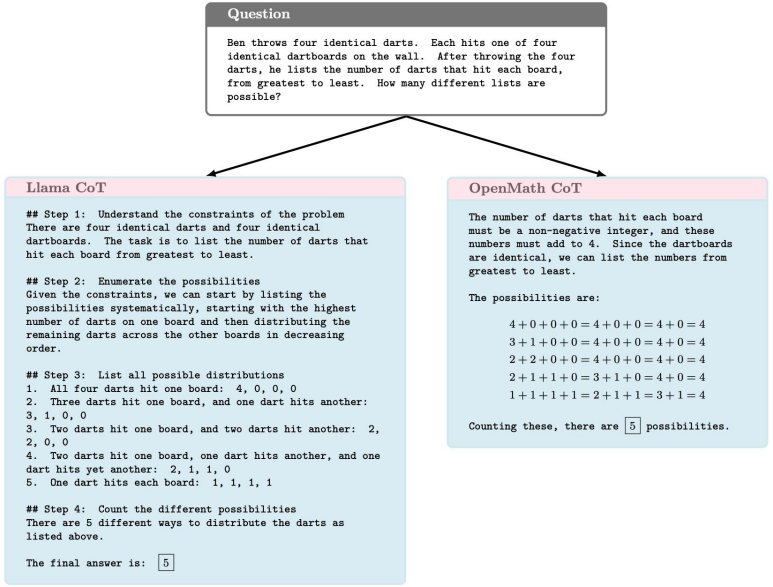


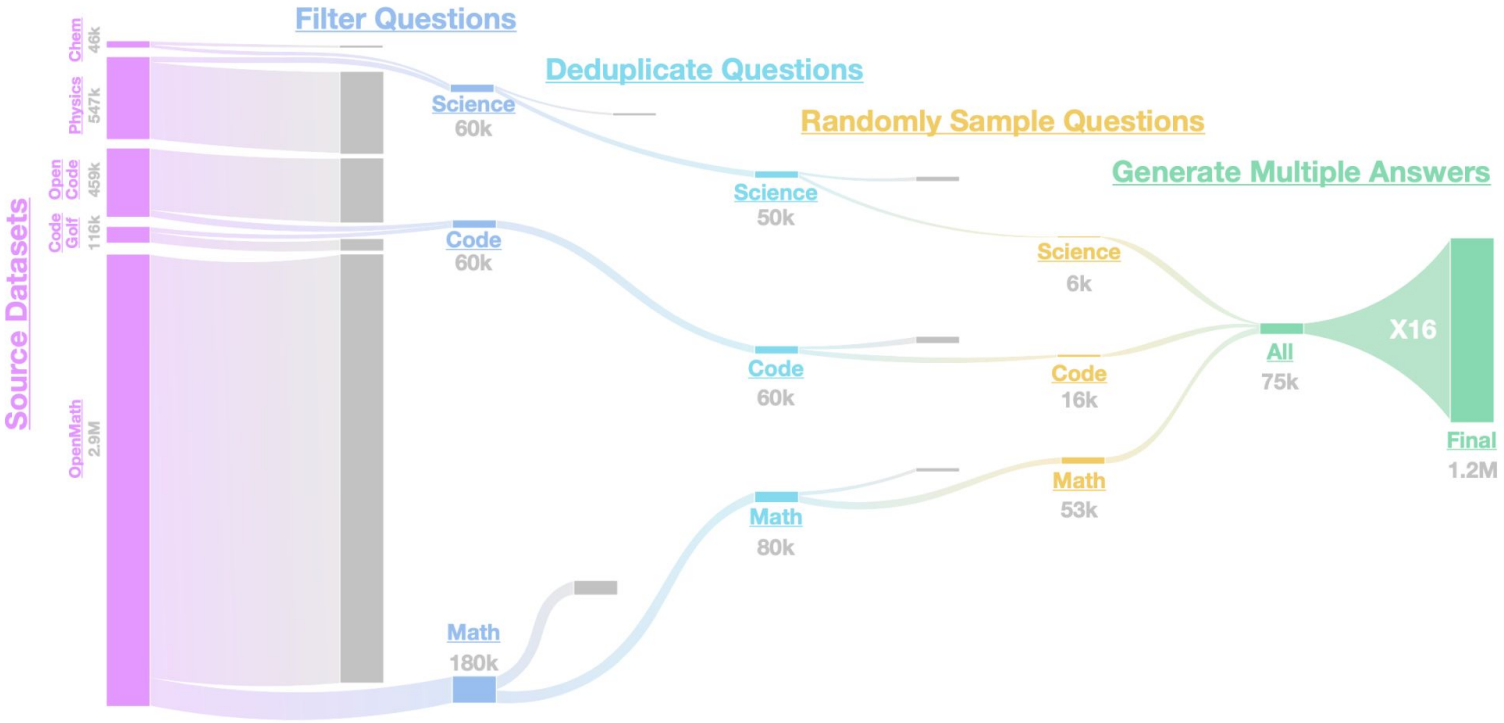
Figure 6: Impact of question diversity on MATH validation accuracy.



	MATH Validation Accuracy	Mean Solution Length
Llama CoT	40.6 ± 0.6	331.3
OpenMath CoT	44.5 ± 0.8	237.0

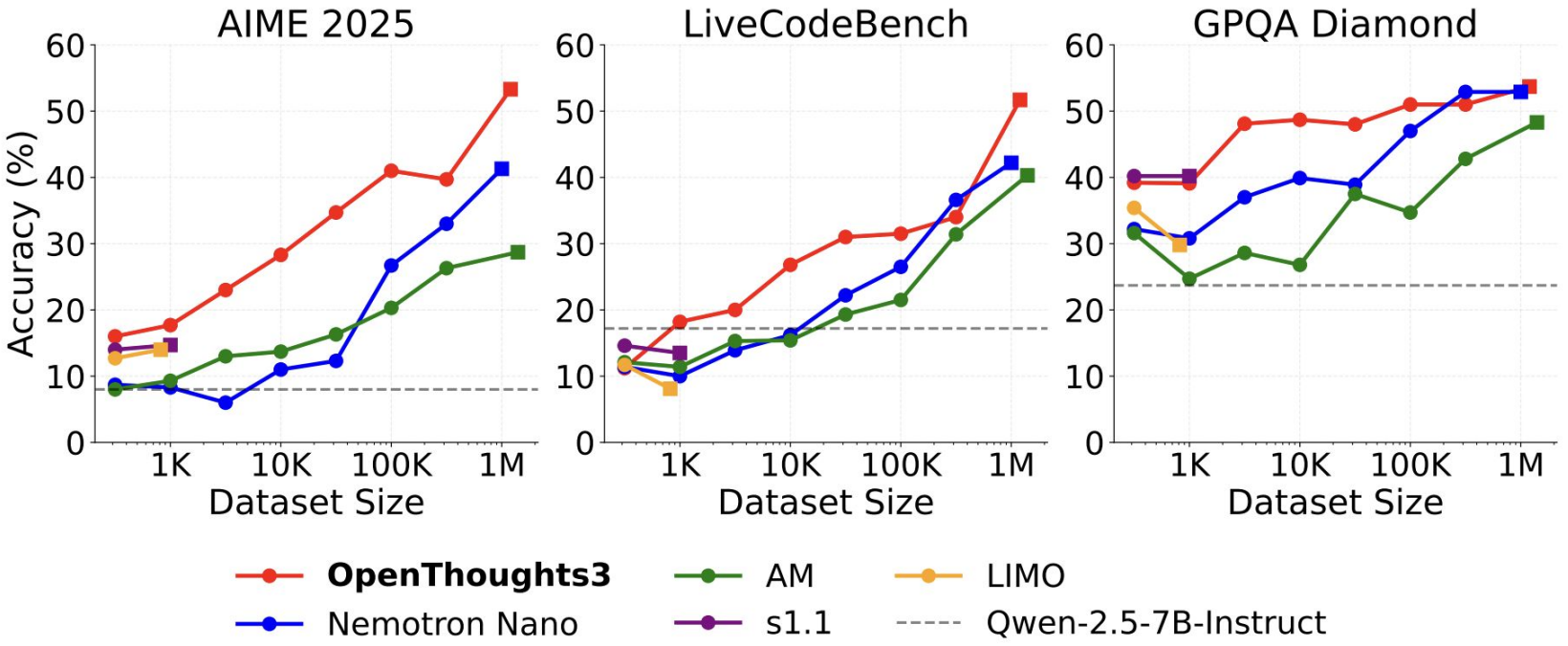
Toshniwal, Shubham, et al. "OpenMathInstruct-2: Accelerating AI for Math with Massive Open-Source Instruction Data." *ICLR 2025*

OpenThoughts



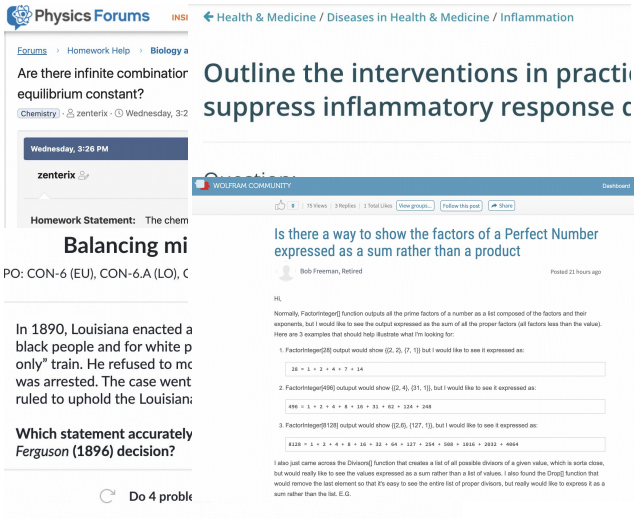
Guha, Etash, et al. "OpenThoughts: Data Recipes for Reasoning Models." *arXiv preprint arXiv:2506.04178* (2025).

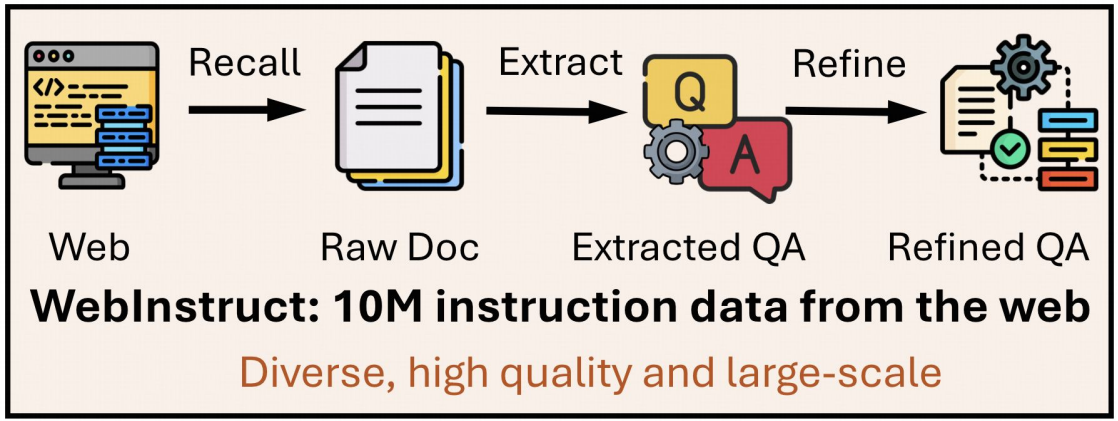
OpenThoughts



Guha, Etash, et al. "OpenThoughts: Data Recipes for Reasoning Models." *arXiv preprint arXiv:2506.04178* (2025).

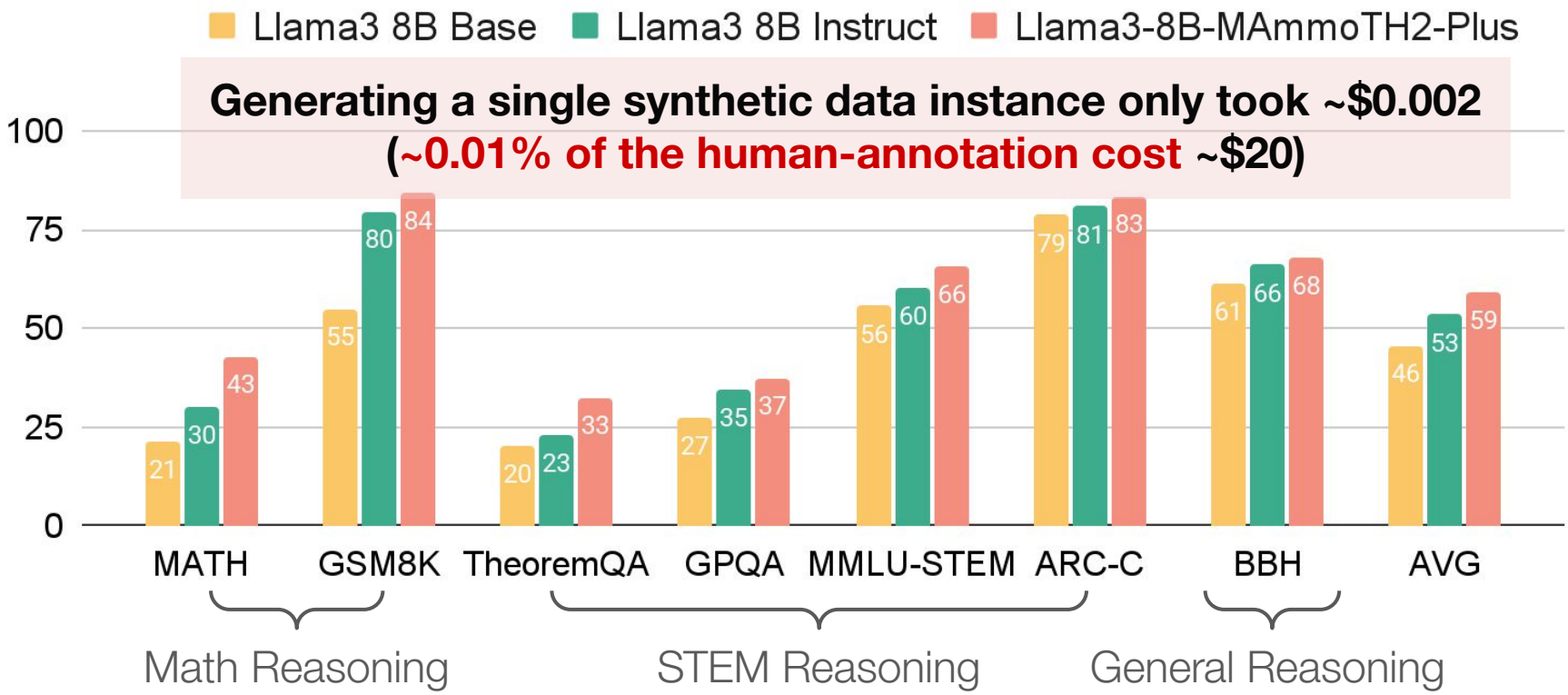
WebInstruct:
10M Synthetic Instruction Data from the Web





Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

10M Synthetic Data vs 10M Human Annotated Data



Yue, X., Zheng, T., Zhang, G., & Chen, W. (2024). MAmmoTH2: Scaling instructions from the web. **NeurIPS 2024**

Code Generation

Code Instructions

WizardCoder: Empowering Code Large Language Models with Evol-Instruct

Ziyang Luo[2*] Can Xu[1*] Pu Zhao[1] Qingfeng Sun[1] Xiubo Geng[1]
Wenxiang Hu[1] Chongyang Tao[1] Jing Ma[2] Qingwei Lin[1] Daxin Jiang[1†]
[1]Microsoft
[2]Hong Kong Baptist University
{caxu,puzhao,qins,xigeng,wenxh,chongyang.tao,qlin,djiang}@microsoft.com
{cszyluo, majing}@comp.hkbu.edu.hk

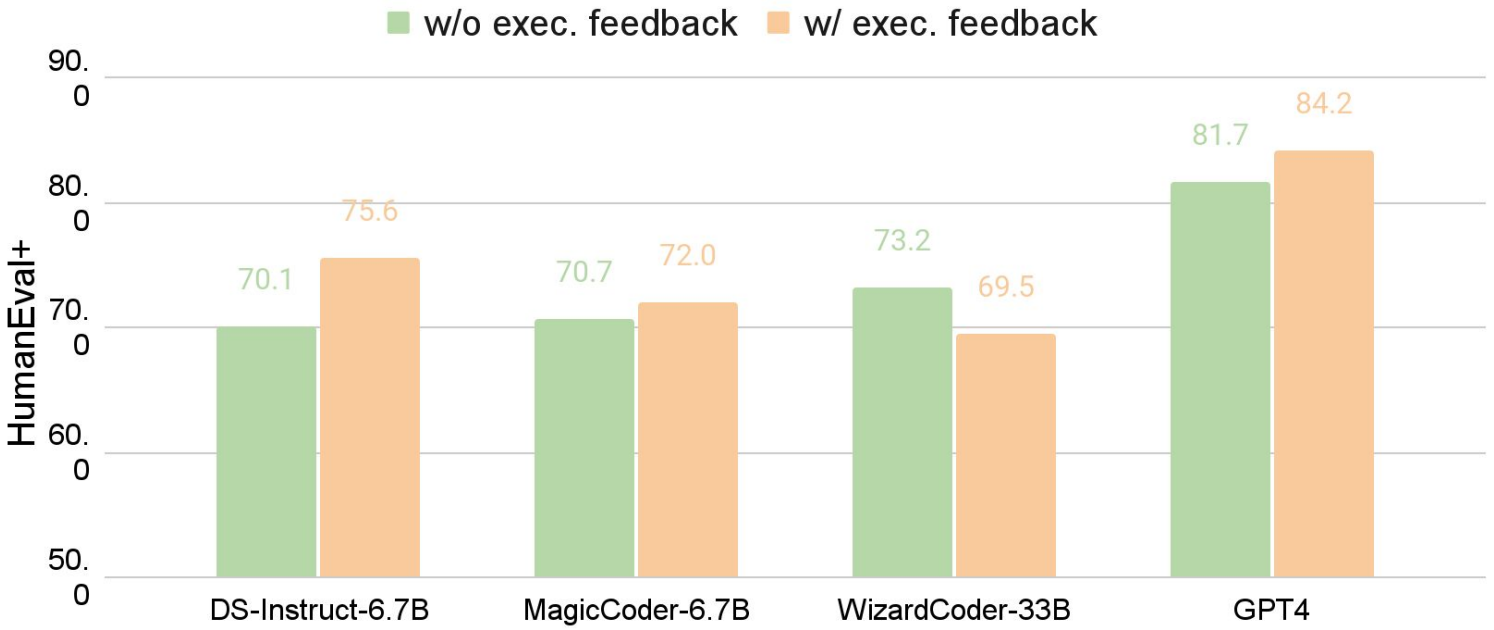
🌊 WaveCoder: Widespread And Versatile Enhanced Instruction Tuning with Refined Data Generation.

Zhaojian Yu, Xin Zhang, Ning Shang, Yangyu Huang, Can Xu, Yishujie Zhao, Wenxiang Hu, Qiufeng Yin
Microsoft
{v-zhaojianyu,xinzhang3,nishang,yanghuan,caxu,v-yiszhao,wenxh,qfyin}@microsoft.com

🎩 Magicoder: Source Code Is All You Need

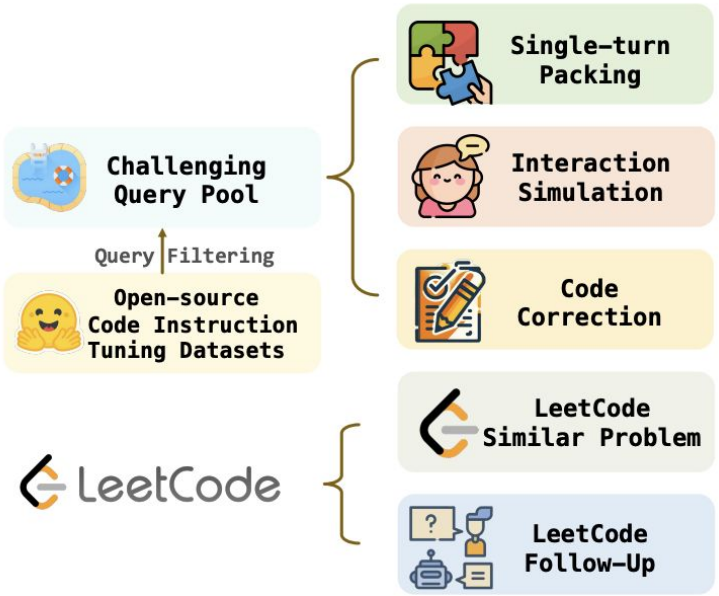
Yuxiang Wei[1] Zhe Wang[2] Jiawei Liu[1] Yifeng Ding[1] Lingming Zhang[1]
[1]University of Illinois at Urbana-Champaign [2]Tsinghua University
{ywei40,jiawei6,yifeng6,lingming}@illinois.edu zhewang20@mails.tsinghua.edu.cn
https://github.com/ise-uiuc/magicoder

Can models understand the execution feedback?



Zheng, Tianyu, et al. "Opencodeinterpreter: Integrating code generation with execution and refinement." *ACL 2024 Findings*

Code-Feedback

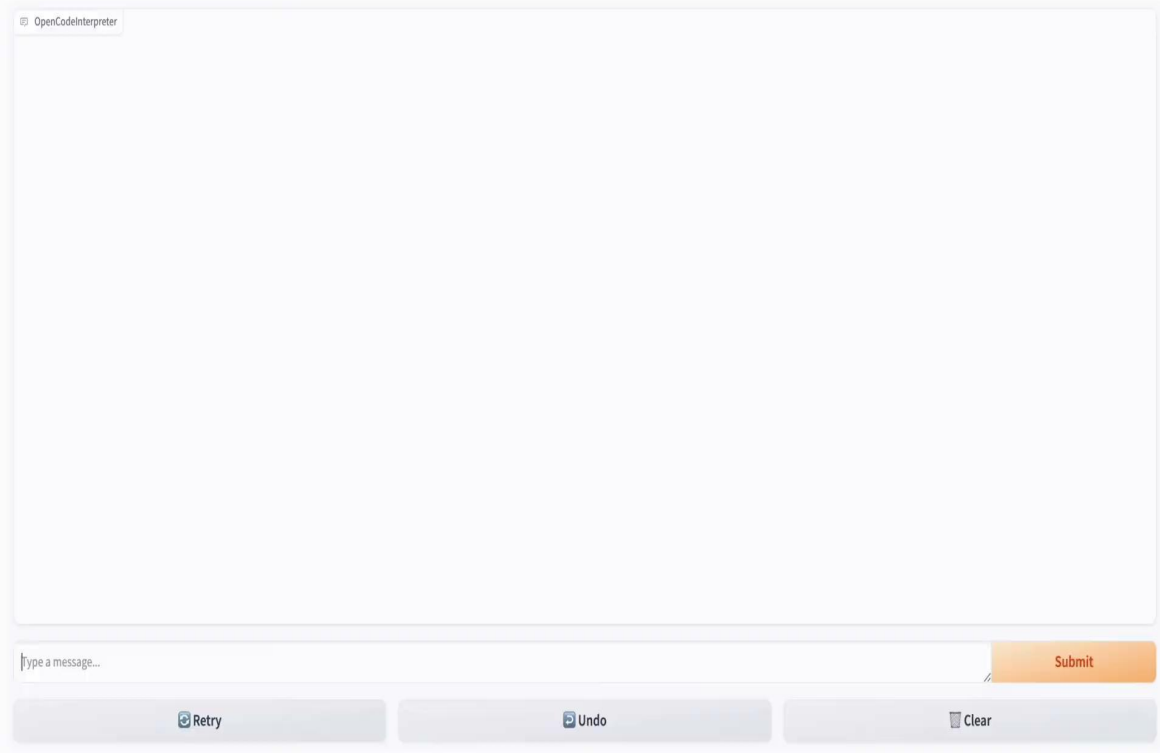


Dataset	#Sample	#Turn	M.T	E.F	H.F
CodeAlpaca[0]	20k	20K	✗	✗	✗
Magicoder-OSS-Instruct[1]	75K	75K	✗	✗	✗
Python-Code-ShareGPT[2]	23K	23K	✗	✗	✗
Magicoder-Evol-Instruct[3]	111K	111K	✗	✗	✗
EvolInstruct-Code[4]	80k	80K	✗	✗	✗
Code-Feedback (Ours)	68K	192K	✔	✔	✔
Single-turn Packing	16K	33.5K	✔	✗	✔
Interaction Simulation	51K	155.5K	✔	✔	✔
Code Correction	0.5K	1.2K	✔	✔	✗
LeetCode Similar Problem	0.3K	0.65K	✔	✗	✔
LeetCode Follow-Up	0.2K	0.76K	✔	✗	✔

Figure 2: Summary of our proposed dataset Code-Feedback construction and comparison with existing code instruction tuning datasets. M.T: Multi Turn, E.F: Execute Feedback, H.F: Human Feedback.

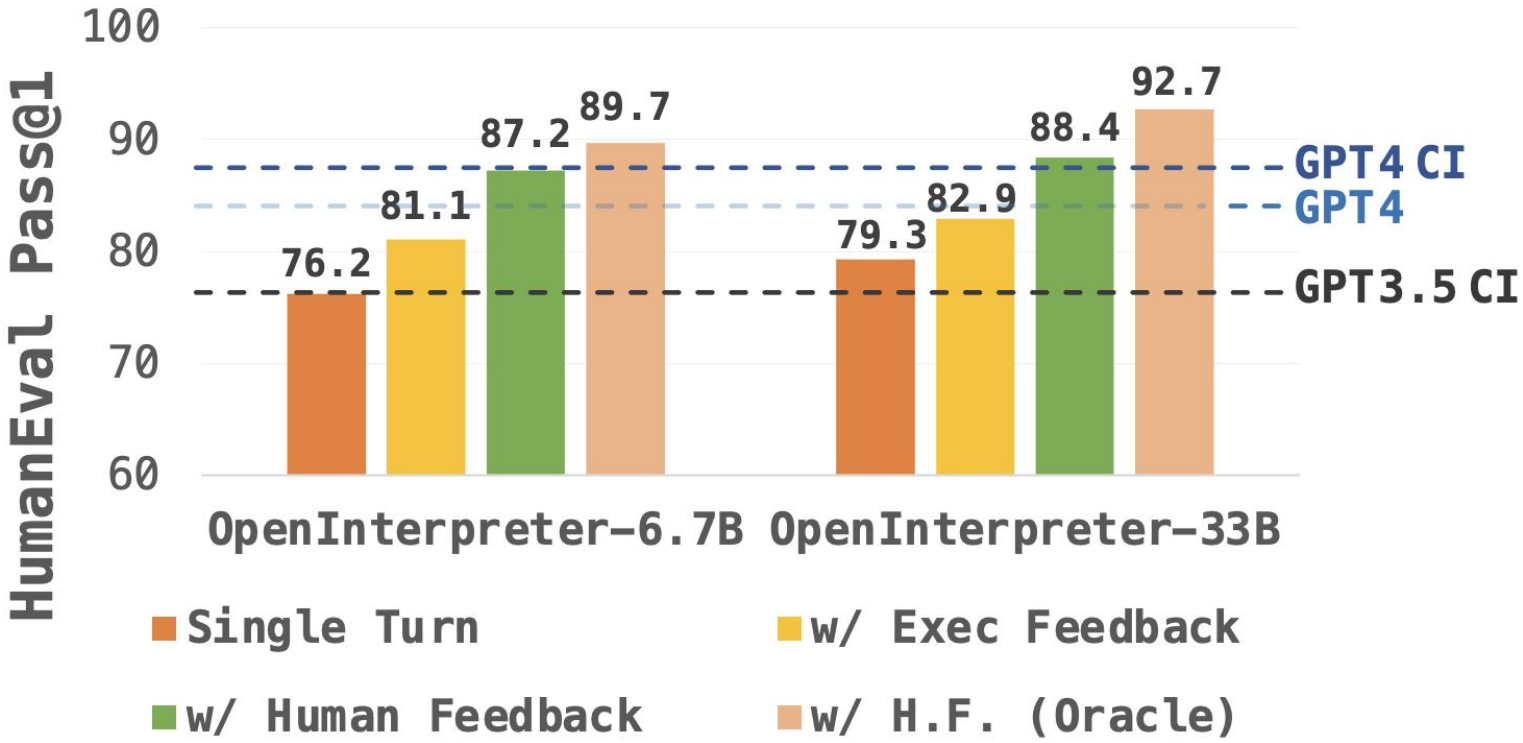
Zheng, Tianyu, et al. "Opencodeinterpreter: Integrating code generation with execution and refinement." *ACL 2024 Findings*

OpenCodeInterpreter



Zheng, Tianyu, et al. "Opencodeinterpreter: Integrating code generation with execution and refinement." *ACL 2024 Findings*

OpenCodeInterpreter: Results



Zheng, Tianyu, et al. "Opencodeinterpreter: Integrating code generation with execution and refinement." *ACL 2024 Findings*

Tool use and Agents

Toolformer

The New England Journal of Medicine is a registered trademark of [QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society] the MMS.

Out of 1400 participants, 400 (or [Calculator(400 / 1400) → 0.29] 29%) passed the test.

The name derives from "la tortuga", the Spanish word for [MT("tortuga") → turtle] turtle.

The Brown Act is California's law [WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.] that requires legislative bodies, like city councils, to hold their meetings open to the public.

Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where "question" is the question you want to ask. Here are some examples of API calls:

Input: Joe Biden was born in Scranton, Pennsylvania.

Output: Joe Biden was born in [QA("Where was Joe Biden born?")] Scranton, [QA("In which state is Scranton?")] Pennsylvania.

Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

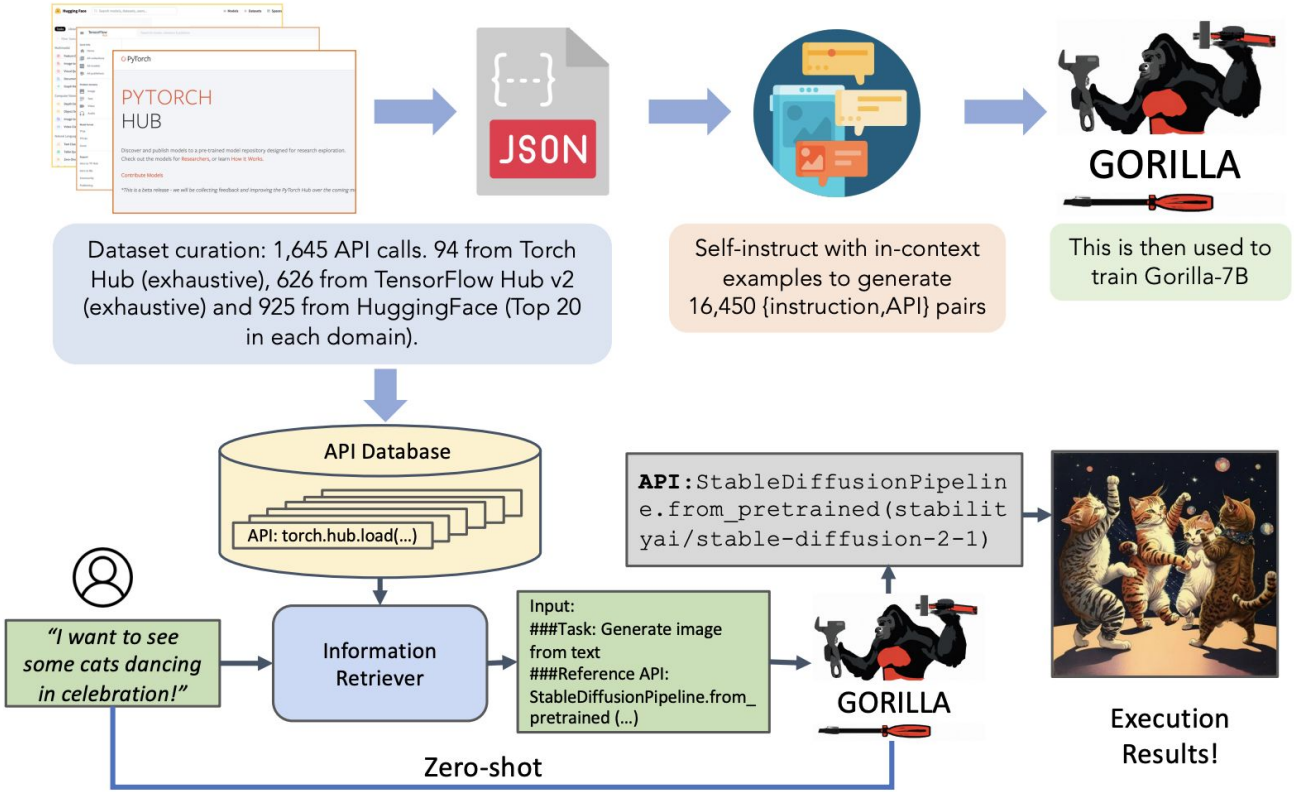
Output: Coca-Cola, or [QA("What other name is Coca-Cola known by?")] Coke, is a carbonated soft drink manufactured by [QA("Who manufactures Coca-Cola?")] the Coca-Cola Company.

Input: x

Output:

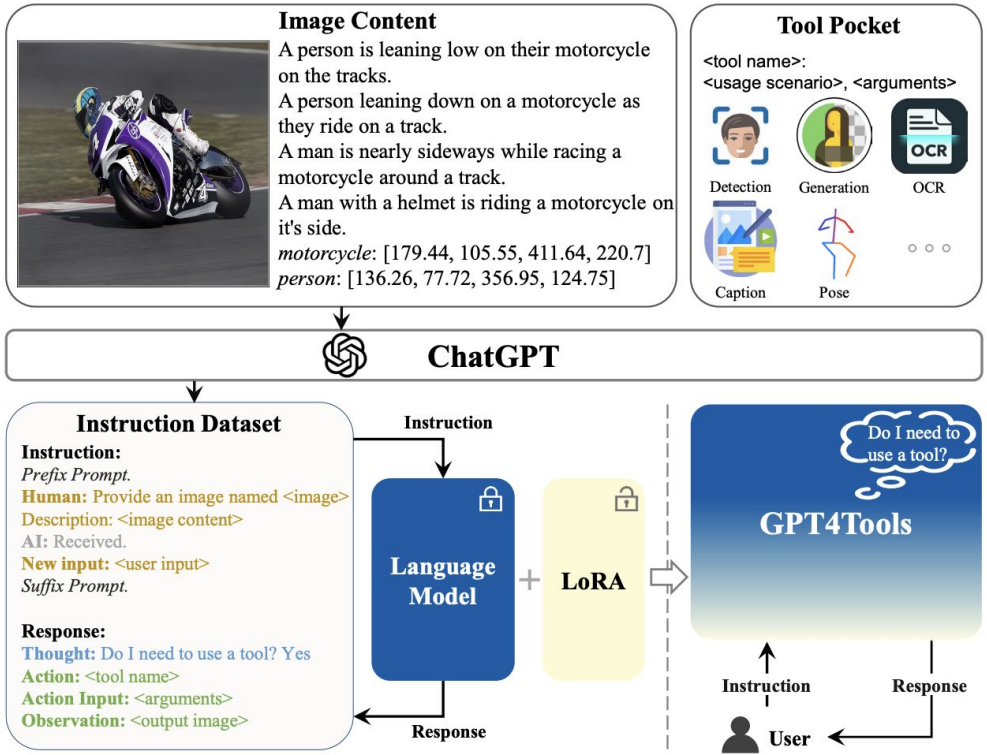
Schick, Timo, et al. "Toolformer: Language models can teach themselves to use tools." *Advances in Neural Information Processing Systems* 36 (2023): 68539-68551.

Gorilla



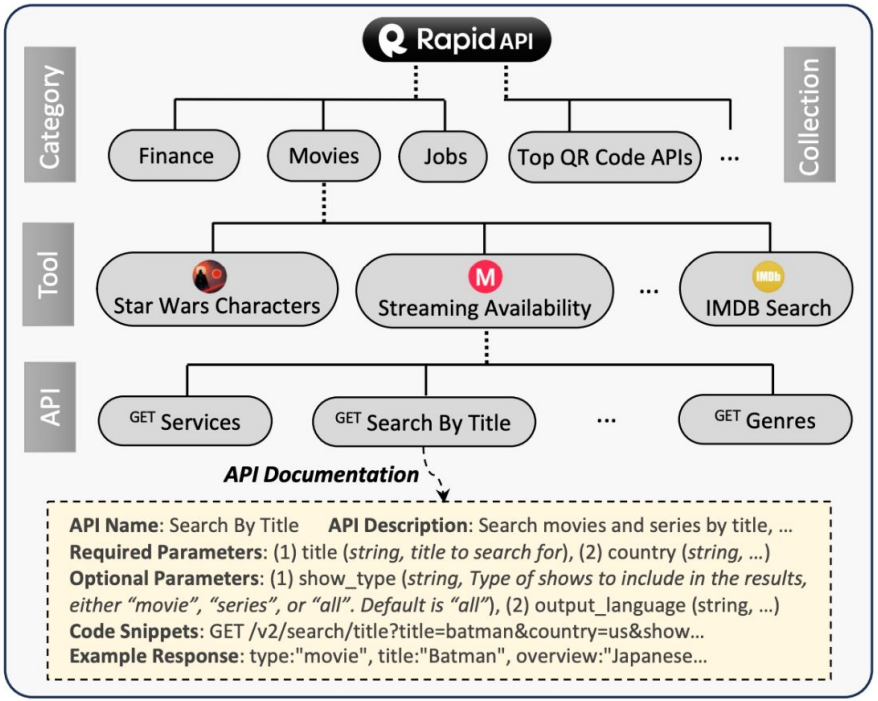
Patil, Shishir G., et al. "Gorilla: Large language model connected with massive apis." *Advances in Neural Information Processing Systems* 37 (2024)

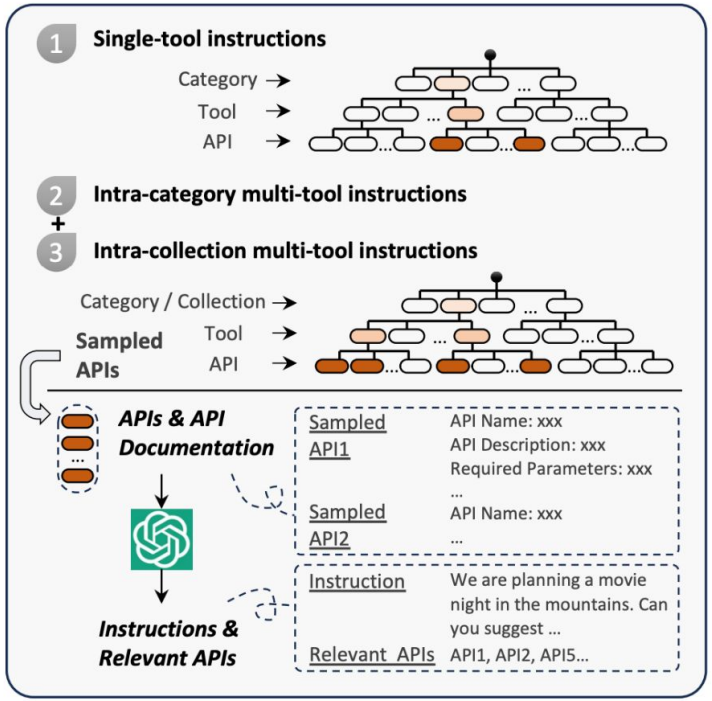
GPT4Tools



Yang, Rui, et al. "Gpt4tools: Teaching large language model to use tools via self-instruction." Advances in Neural Information Processing Systems 36 (2023): 71995-72007.

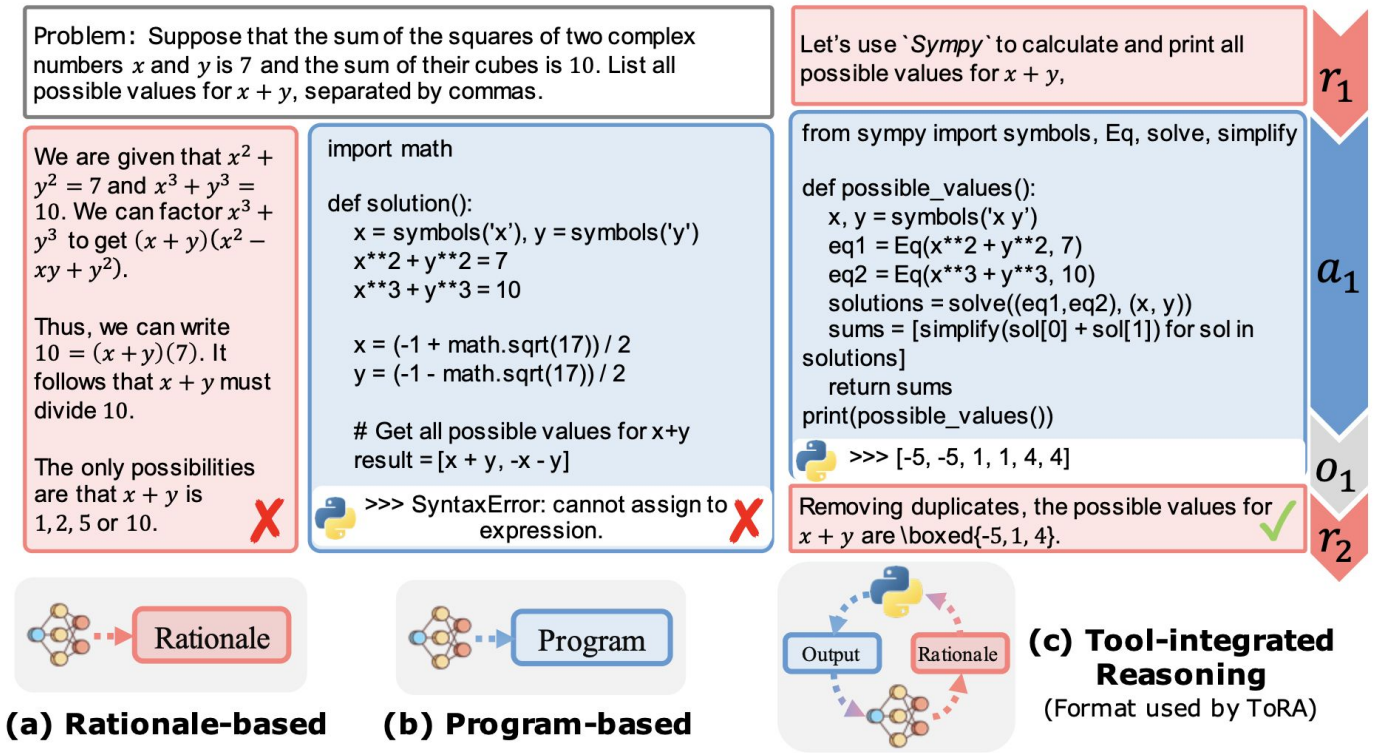
TooLLM





Qin, Yujia, et al. "ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs." The Twelfth International Conference on Learning Representations.

ToRA: Tool-Integrated Reasoning



Gou, Zhibin, et al. "ToRA: A Tool-Integrated Reasoning Agent for Mathematical Problem Solving." The Twelfth International Conference on Learning Representations.

ToRA: Tool-Integrated Reasoning

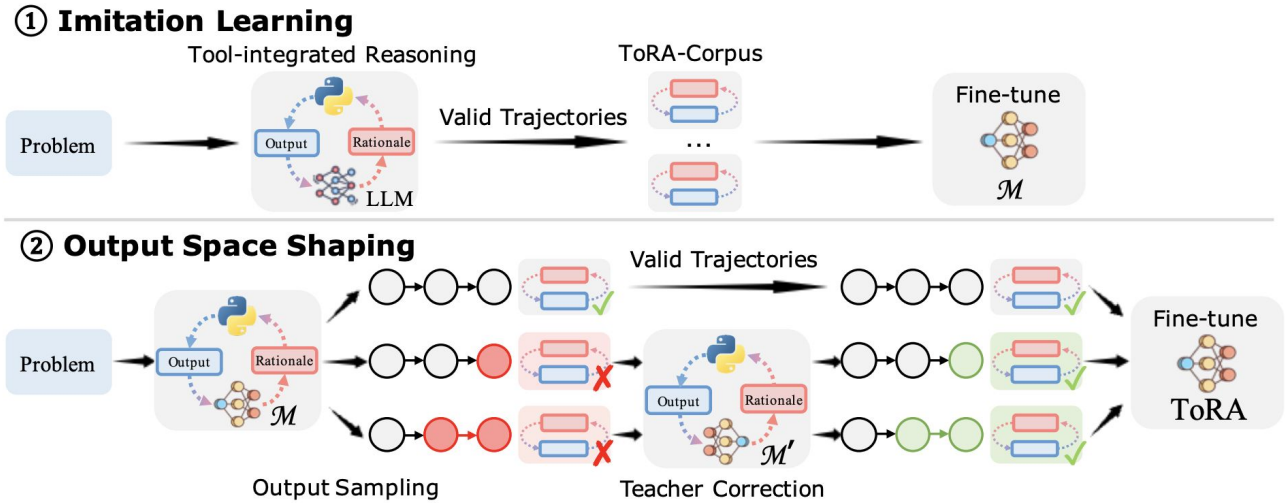
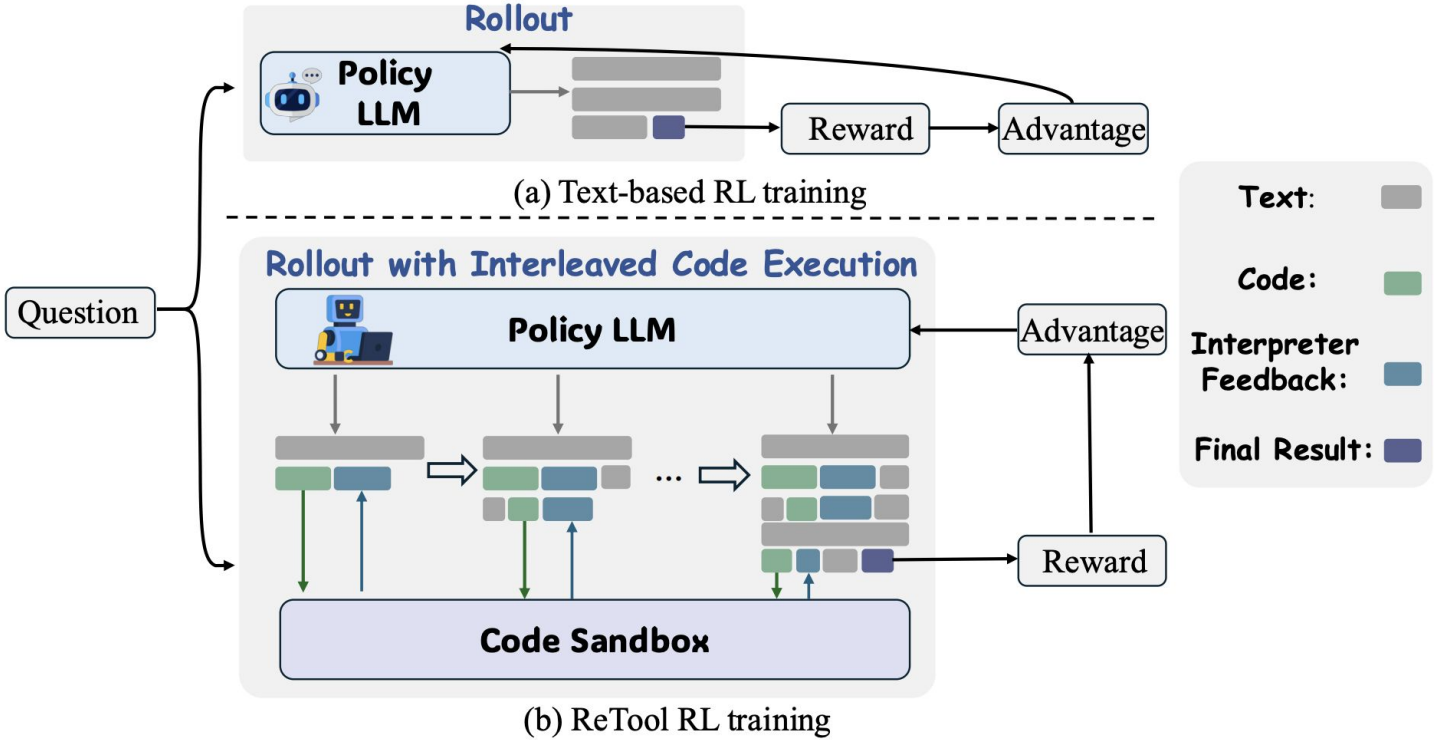


Figure 3: Training ToRA contains two steps. ① **Imitation Learning**: Prompt LLMs like GPT-4 to generate Tool-integrated Reasoning trajectories (ToRA-Corpus) and use this corpus to fine-tune a model $\mathcal{M}$; ② **Output Space Shaping**: Sample diverse tool-use trajectories with $\mathcal{M}$, keep the valid ones, correct the invalid ones with a teacher model $\mathcal{M}'$, and retrain $\mathcal{M}$ on the union of sampled valid trajectories, corrected ones, and the initial ToRA-Corpus to obtain ToRA.

Gou, Zhibin, et al. "ToRA: A Tool-Integrated Reasoning Agent for Mathematical Problem Solving." The Twelfth International Conference on Learning Representations.

ReTool: RL for Strategic Tool Use



Feng, Jiazhan, et al. "Retool: Reinforcement learning for strategic tool use in llms." arXiv preprint arXiv:2504.11536 (2025).

ReTool: RL for Strategic Tool Use

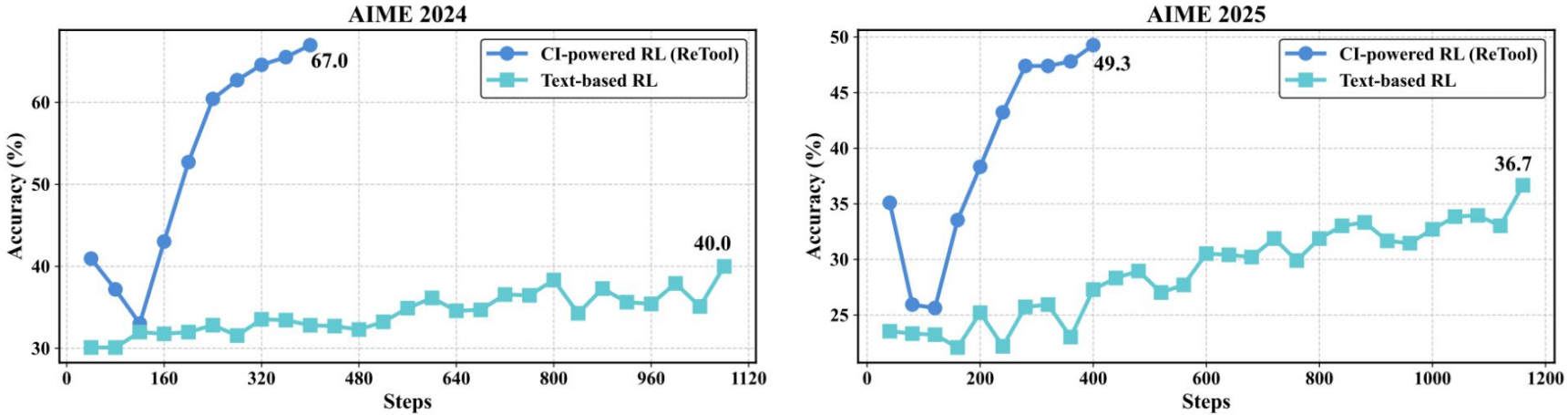
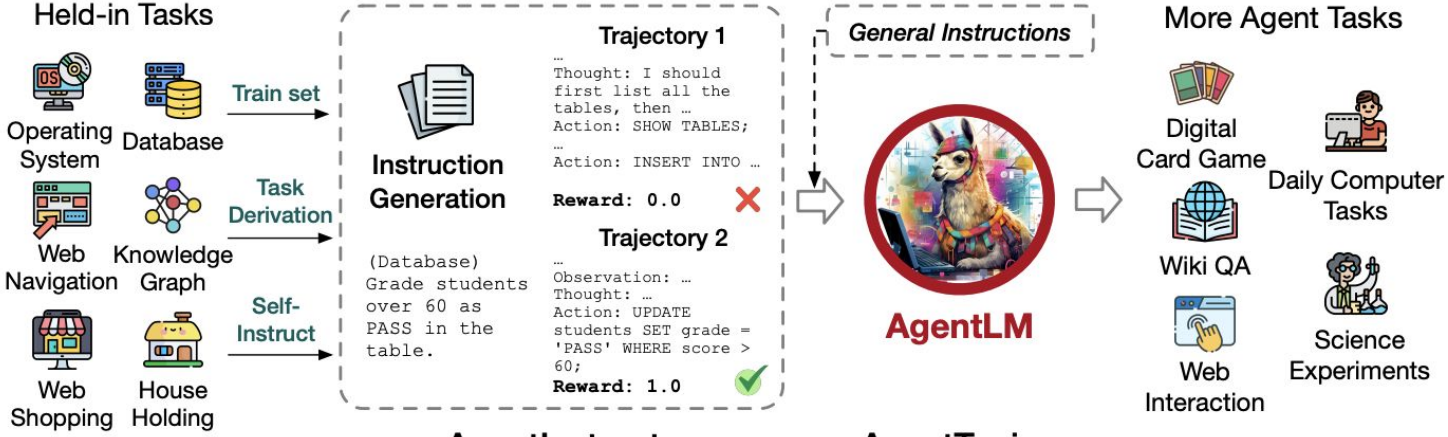
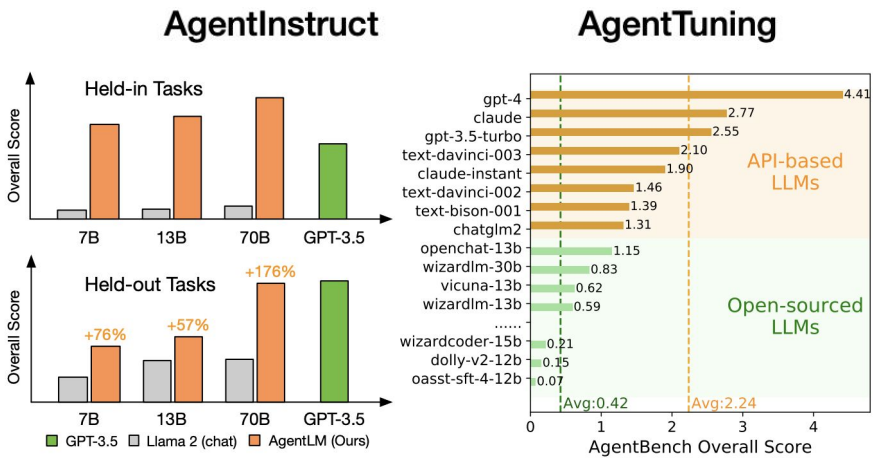


Figure 1 AIME 2024 & 2025 scores of ReTool and text-based RL baseline on the Qwen2.5-32B-Instruct model.

Feng, Jiazhan, et al. "Retool: Reinforcement learning for strategic tool use in llms." arXiv preprint arXiv:2504.11536 (2025).

AgentTuning





Zeng, Aohan, et al. "AgentTuning: Enabling Generalized Agent Abilities for LLMs." *Findings of ACL 2024*.

CodeActInstruct

Example

User:

① Find the sum of the reciprocals of the roots of $x^2-13x+4=0$.

Assistant:

② To find the sum of the reciprocals of the roots of the quadratic equation $x^2-13x+4=0$ [...]

③
```
<execute>
import sympy
x = sympy.Symbol('x')
roots = sympy.solve(x**2 - 13*x + 4)
print(1/roots[0] + 1/roots[1])
</execute>
```

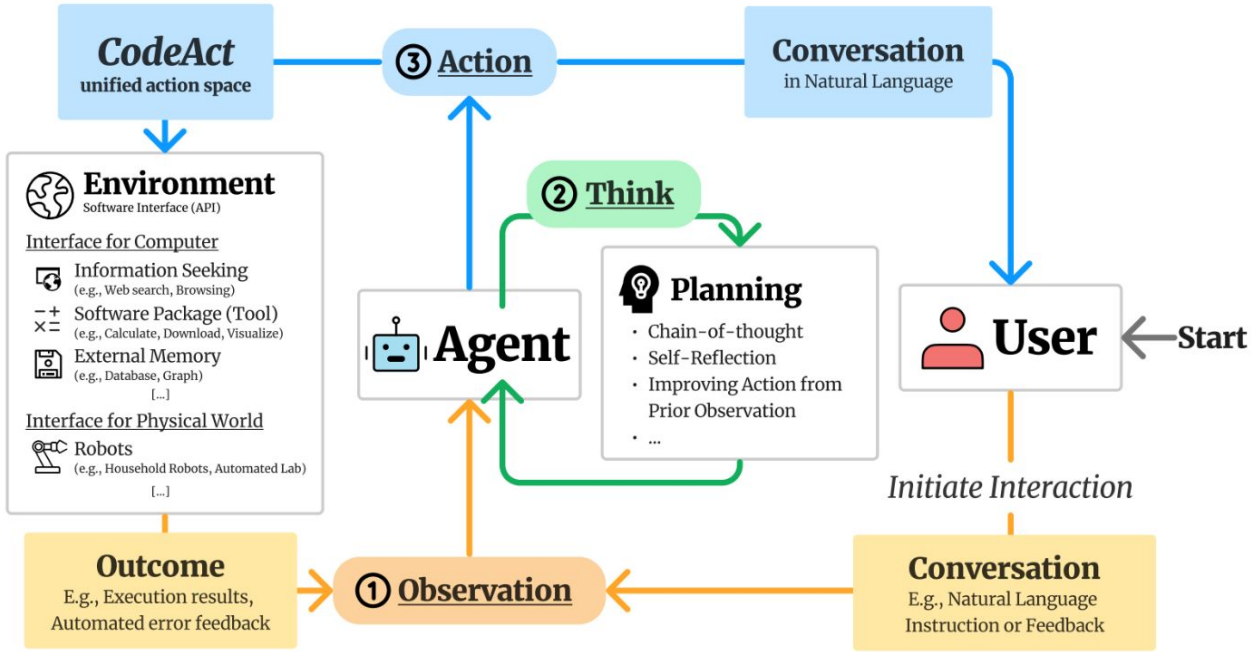
Env:

① 1/(3*sqrt(17)/2 + 13/2) + 1/(13/2 - 3*sqrt(17)/2)

Assistant:

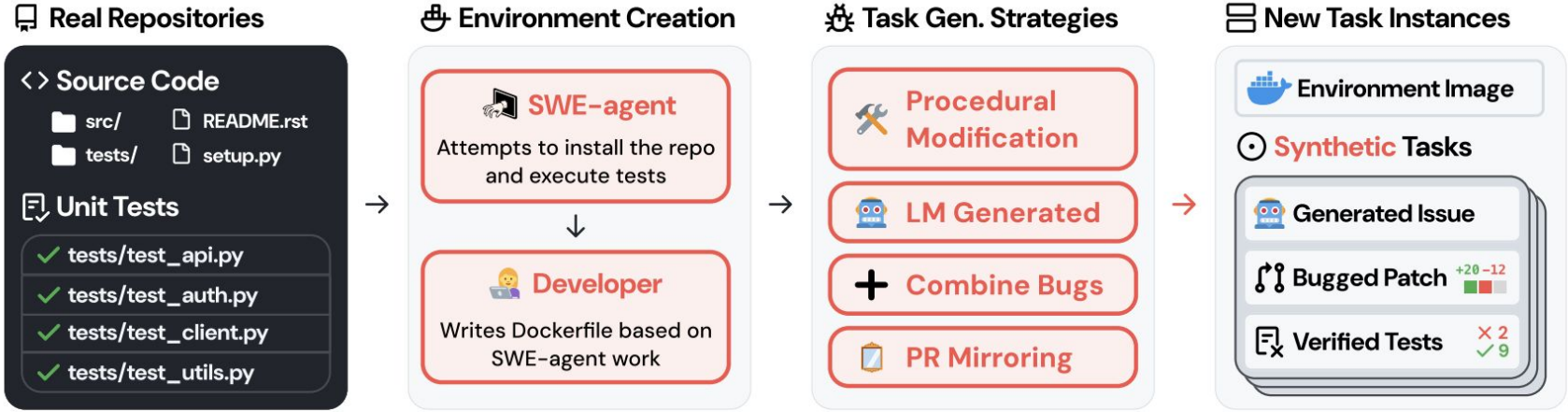
② The roots obtained are in symbolic form, which makes it difficult to evaluate the sum directly. Let's use the quadratic formula to explicitly find the numerical values of the roots first.

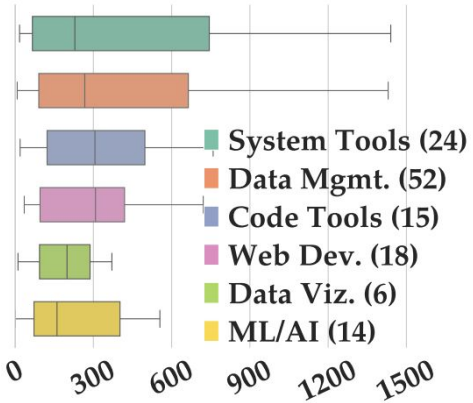
[...]



Wang, Xingyao, et al. "Executable code actions elicit better llm agents." *Forty-first International Conference on Machine Learning*. 2024.

SWE-Smith

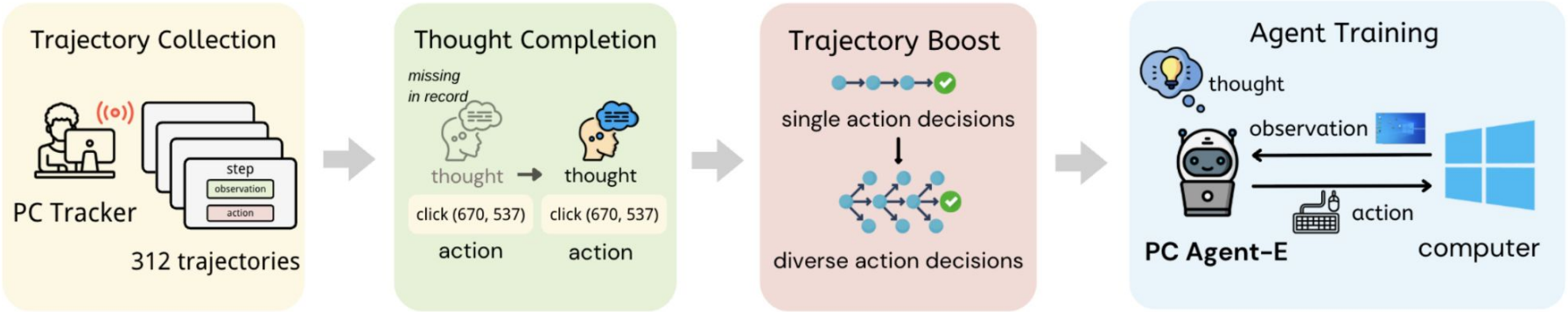


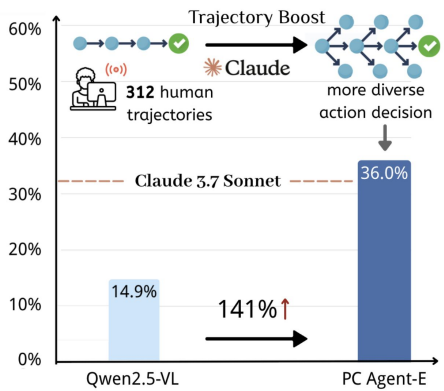


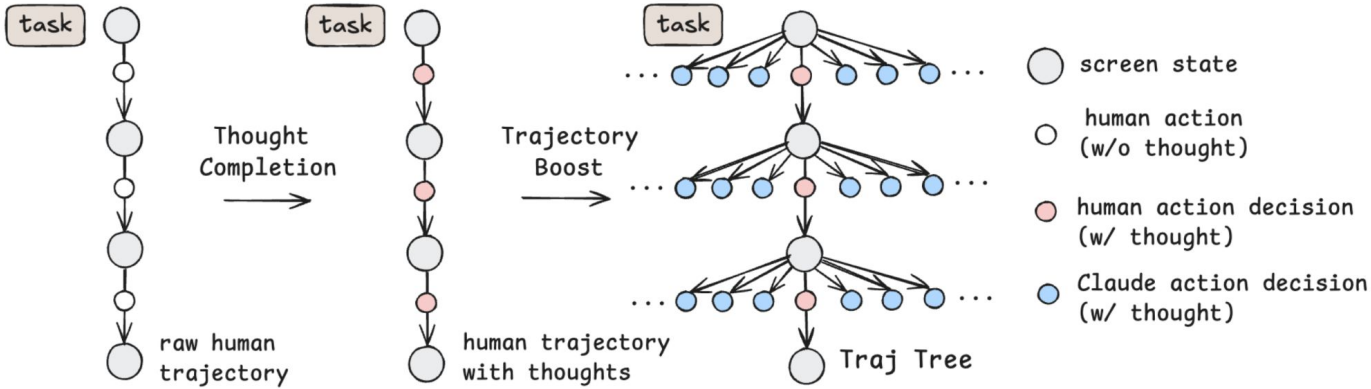
Bug Type	Yield %	# Insts	Cost	F2P	Lines
Combine	96.9%	10,092	0.00¢	15	11
LM Modify	56.0%	17,887	0.38¢	4	3
LM Rewrite	35.0%	4,173	3.93¢	4	24
PR Mirror	33.8%	2,344	5.53¢	3	14
Procedural	40.2%	15,641	0.00¢	7	5
Total	50.1	50,137	2.32¢	6	5

Yang, John, et al. "Swe-smith: Scaling data for software engineering agents." *arXiv preprint arXiv:2504.21798* (2025).

Agent-E



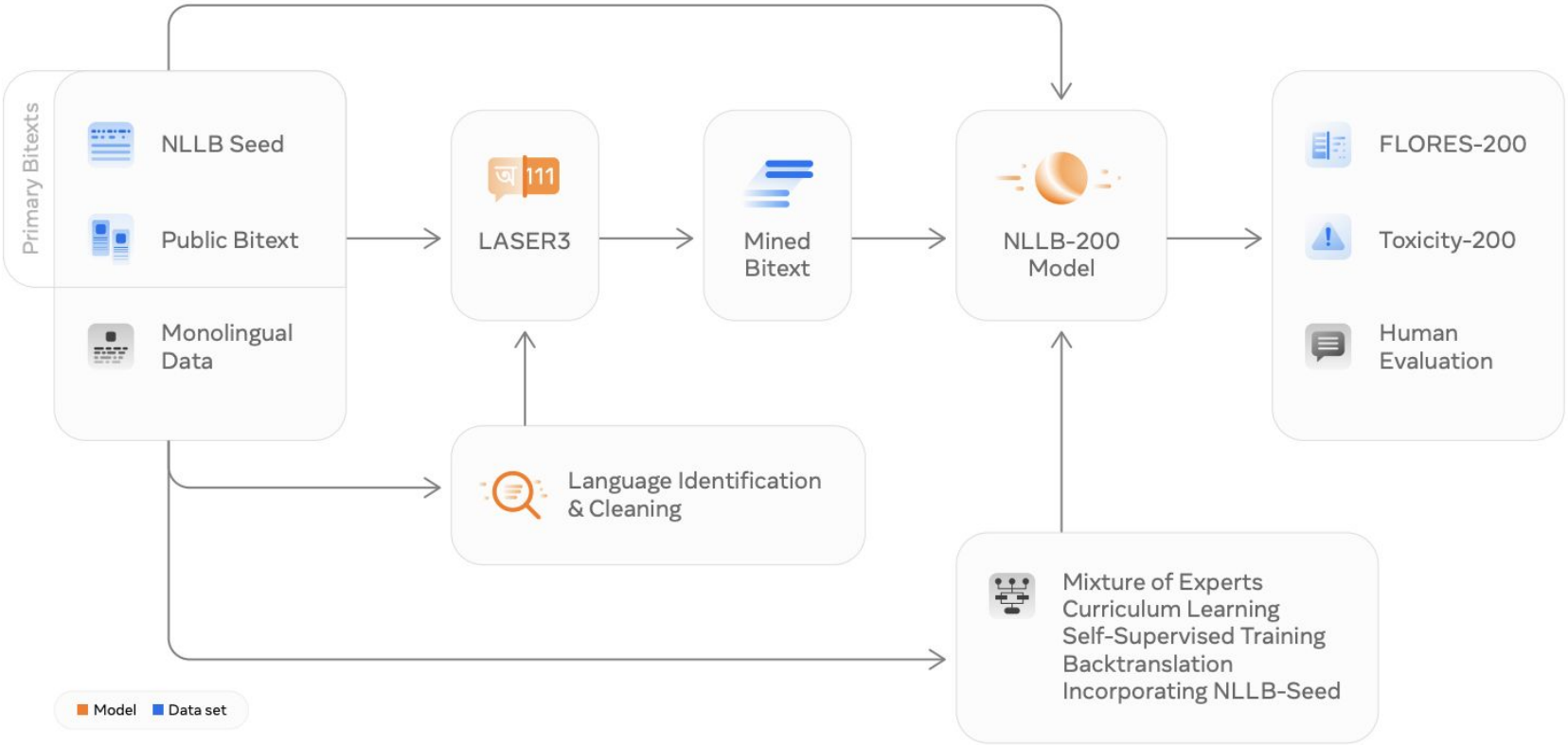




He, Yanheng, Jiahe Jin, and Pengfei Liu. "Efficient Agent Training for Computer Use." *arXiv preprint arXiv:2505.13909* (2025).

Multilingual and Multimodal

NLLB: Scaling Machine Translation



NLLB Team. Scaling neural machine translation to 200 languages. *Nature* 630, 841–846 (2024)

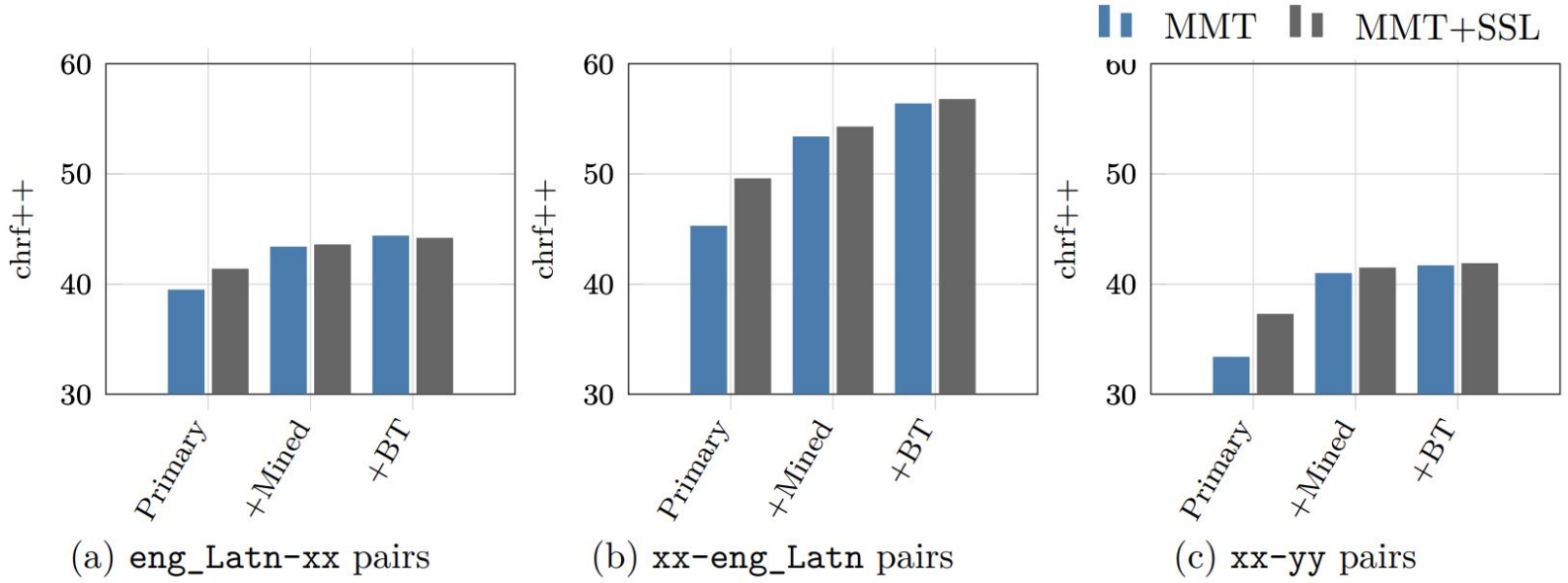
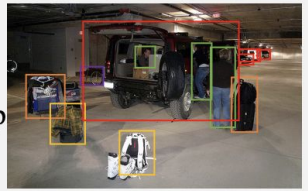


Figure 36: **Comparison of only MMT vs MMT+SSL Multitask Performance** on aggregate over Into English and Out of English directions, when trained on different sources of data. We compare on chrF++ scores. We observe that as we add more and more data from mining and/or backtranslation, we start to see diminishing improvements from the SSL task.

NLLB Team. Scaling neural machine translation to 200 languages. *Nature* 630, 841–846 (2024)

LLaVA: Visual Instruction Tuning

Context type 1: Captions



A group of people standing outside of a black vehicle with various luggage.
Luggage surrounds a vehicle in an underground parking area
People try to fit all of their luggage in an SUV.
The sport utility vehicle is parked in the public garage, being packed for a trip
Some people with luggage near a van that is transporting it.

Context type 2: Boxes

person: [0.681, 0.242, 0.774, 0.694], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

Response type 1: conversation

Question: What type of vehicle is featured in the image?
Answer: The image features a black sport utility vehicle (SUV) ...<omitted>

Response type 2: detailed description

The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>

Response type 3: complex reasoning

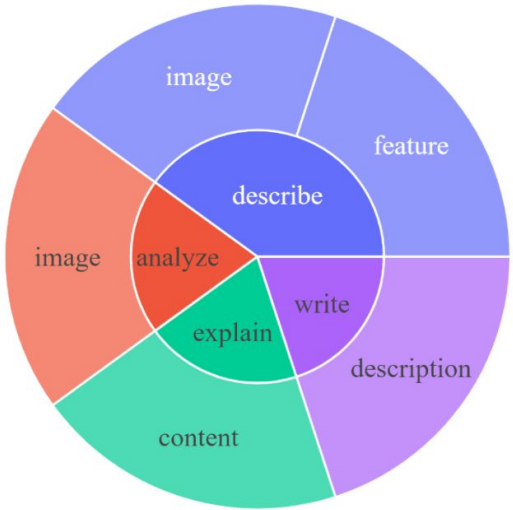
Question: What challenges do these people face?
Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Liu, Haotian, et al. "Visual instruction tuning." *Advances in neural information processing systems* 36 (2023): 34892-34916.

Data file name	File Size	Sample Size
conversation_58k.json	126 MB	58K
detail_23k.json	20.5 MB	23K
complex_reasoning_77k.json	79.6 MB	77K

For each subset, we visualize the root noun-verb pairs for the instruction and response. For each chart, please click the link for the interactive page to check out the noun-verb pairs whose frequency is higher the given number.

Instruction: Conversation [0, 20, 50]

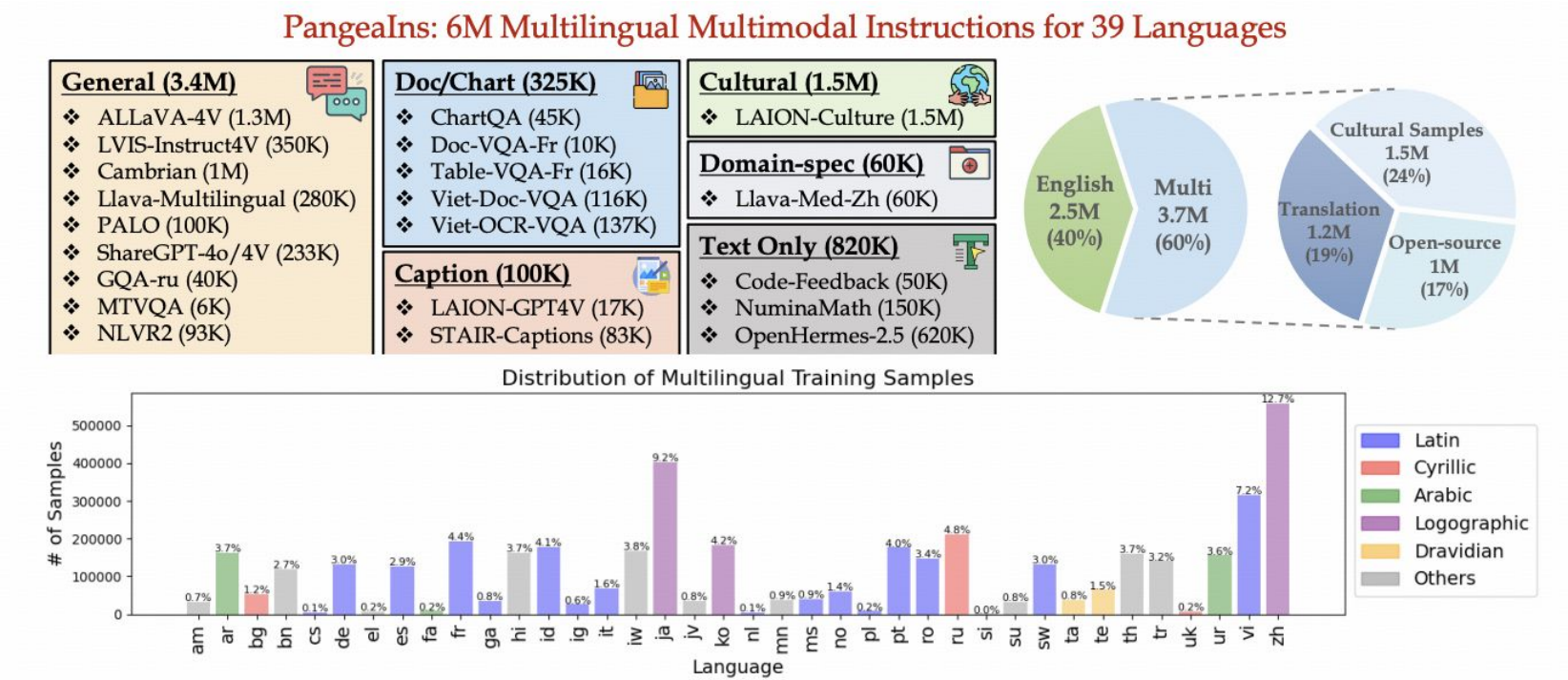


Instruction: Detailed Description [0]

Instruction: Complex Reasoning [0, 20, 50]

Liu, Haotian, et al. "Visual instruction tuning." *Advances in neural information processing systems* 36 (2023): 34892-34916.

Training Data: 6M Synthetic Instructions in 39 Languages



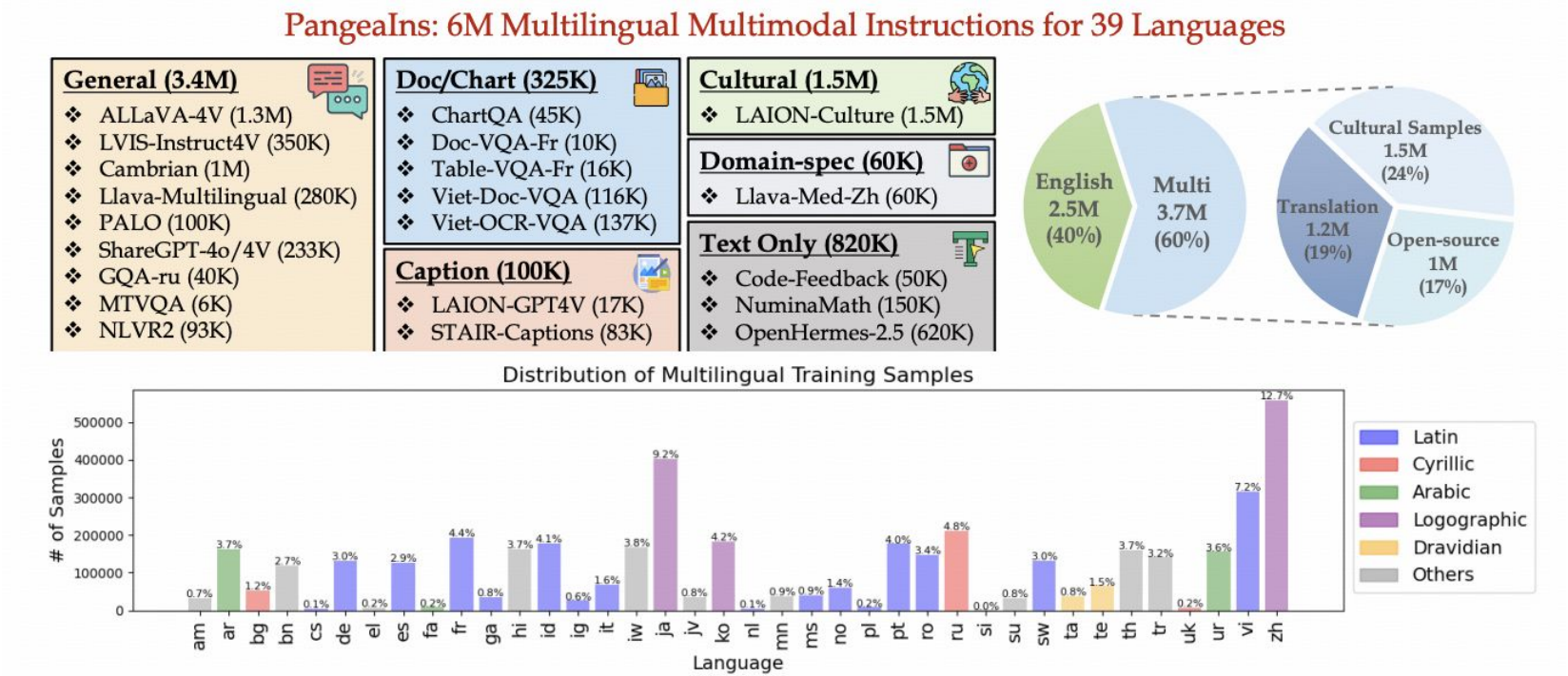
Translation

Culture-aware Synthetic Data

Open Data

Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Training Data: 6M Synthetic Instructions in 39 Languages

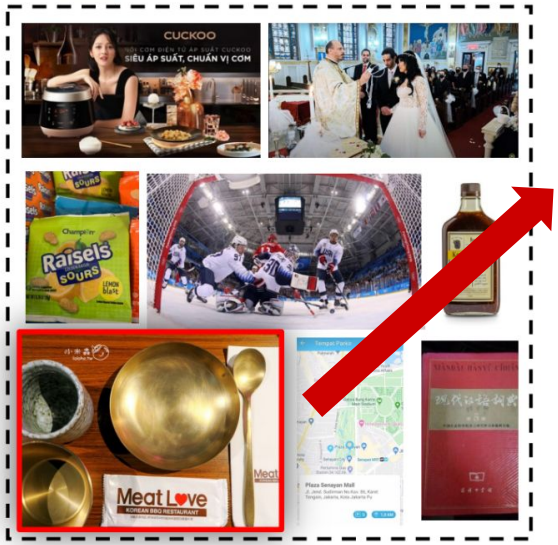


Translation | **Culture-aware Synthetic Data** | Open Data

Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Step 1: Selecting Culturally Diverse Images with LLMs

LAION-Multi



Alt Text: 台北信義燒肉｜Meat Love橡木炭火燒肉 韓國烤肉最高7折優惠!小菜無限吃到飽! *(Taipei Xinyi BBQ | Meat Love Oak Charcoal BBQ – Korean BBQ up to 30% off! Unlimited free side dishes!)*

LLM Scoring

❑ Informativeness

Rate the following alt text on a scale from 1 to 5 based on its quality in describing the image...

❑ Topic Classify

Assign a category to the alt text based on its content. Choose from the following categories...

❑ Country Classify

Decide if the alt text is related to a specific country's culture...

Informativeness: 4

Select images with informative text

Topic: Cooking and Food

Balance domains and topics of images

Country: Korea

Select culturally relevant images

Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Step 2: Recaptioning Multicultural Images

Alt Text: 上海ディズニーランドの模型のそばにいるウォルト・ディズニー・カンパニーの社長兼CEO

(President and CEO of The Walt Disney Company by a model of Shanghai Disneyland)

Caption w/o Alt Text: 画像には、暗いスーツに淡い青色のシャツを着て、ネクタイをしていない男性が、大きな城の模型の前に立っている様子が映っています。その城は… *(The image features a man in a dark suit, light blue shirt, and no tie, standing in front of a large model of a castle. The castle …)*

Recaption with Alt Text: 画像には、*ウォルト・ディズニー・カンパニーの社長兼CEO* が *上海ディズニーランド* の模型の前に立っている様子が映っています。背景には色鮮やかなフラワーアレンジメントが広がっています。*(The image features the President and CEO of The Walt Disney Company standing in front of a model of Shanghai Disneyland. In the background, vibrant floral arrangements …)*

Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Step 3: Generating Multilingual Instructions

Recaption with Alt Text: 画像には、*ウォルト・ディズニー・カンパニーの社長兼CEO* が *上海ディズニーランド* の模型の前に立っている様子が映っています。背景には色鮮やかなフラワーアレンジメントが広がっています。*(The image features the President and CEO of The Walt Disney Company standing in front of a model of Shanghai Disneyland. In the background, vibrant floral arrangements …)*

$$D_s = F_s\left(\mathcal{O}\left(\text{Prompt}_s(D); \theta\right)\right)$$

Prompt a LLM to generate data

Prompt$_s$: Generate two **instruction-response pairs** based on the visual content of an image. The language should be in **[Japanese]**, the same as the caption text…

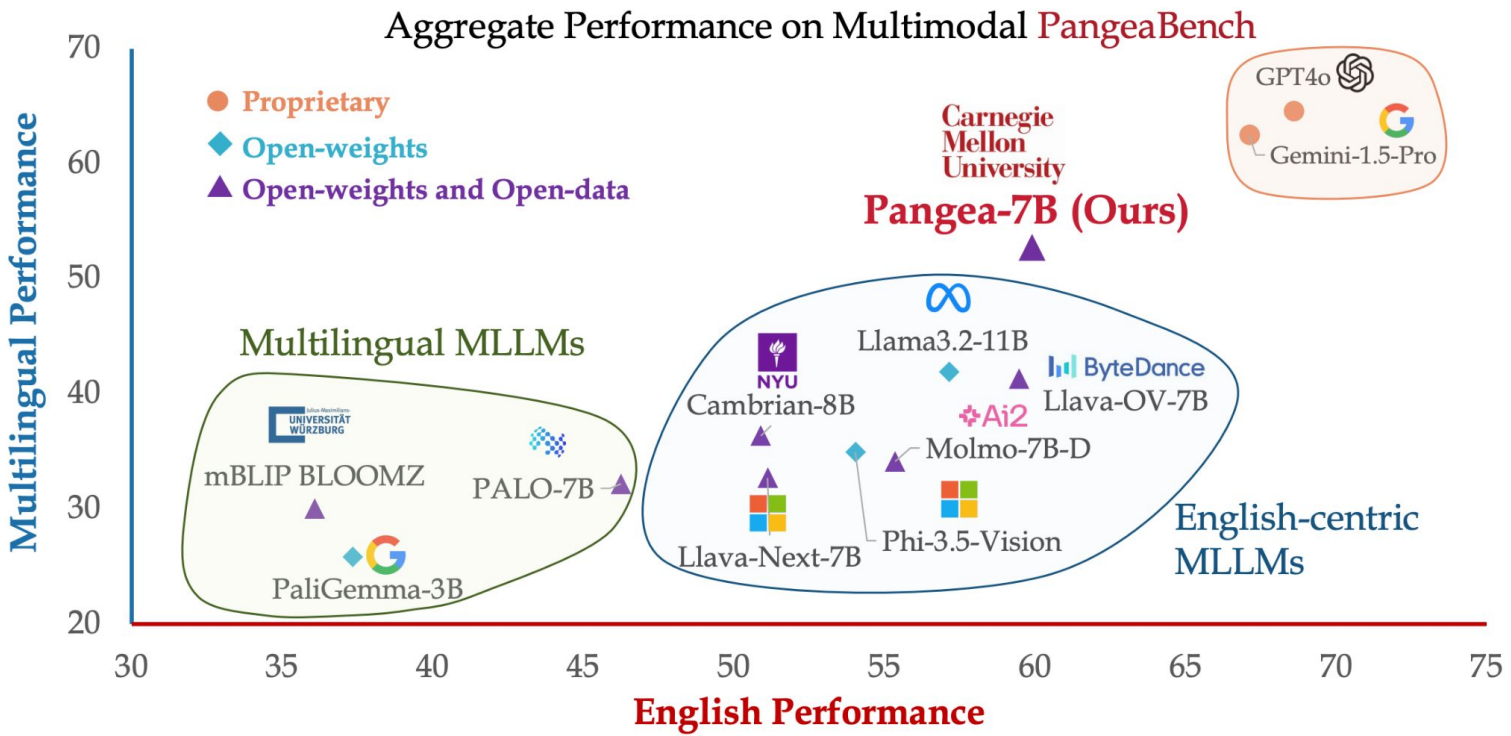
Instruction: 画像の中の男性がイベント中に行っているかもしれないスピーチを想像してください。

(Imagine a speech the man in the image might be giving during the event.)

Response: ここに立ち、上海ディズニーランドの素晴らしいモデルの前で皆さんとこの特別な瞬間を共有できることを、大変光栄に思います。... *(I am truly honored to stand here and share this special moment with you in front of this magnificent model of Shanghai Disneyland…)*

Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Pangea: SOTA fully-open **multilingual multimodal** LLM



Yue, X*, Song, Y* et al., Pangea: A fully open multilingual multimodal llm for 39 languages. *ICLR 2025*

Conclusion and Future Directions

Conclusion and Future Directions

- Quality vs. Quantity
- How to incorporate domain-specific experience?
- More diverse modalities & tasks
- Combining techniques
- Community-driven efforts

Limitations and Open Questions

Tutorial Summary

- Motivation
  - Why do we need synthetic data?
  - What is “high-quality” synthetic data?
- How do we get the data?
  - Sampling-based generation
  - Back-translation
  - Transformation of existing data
  - Human-AI collaboration
  - Symbolic generation

Tutorial Summary (cont'd)

- How do we use the synthetic data?
  - Pretraining
  - Supervised finetuning
  - RL training
  - Evaluation & Analysis
- Scenario-specific applications
  - Reasoning
  - Code generation
  - Tool use & agents
  - Multilingual & multimodal

Limitations and open challenges

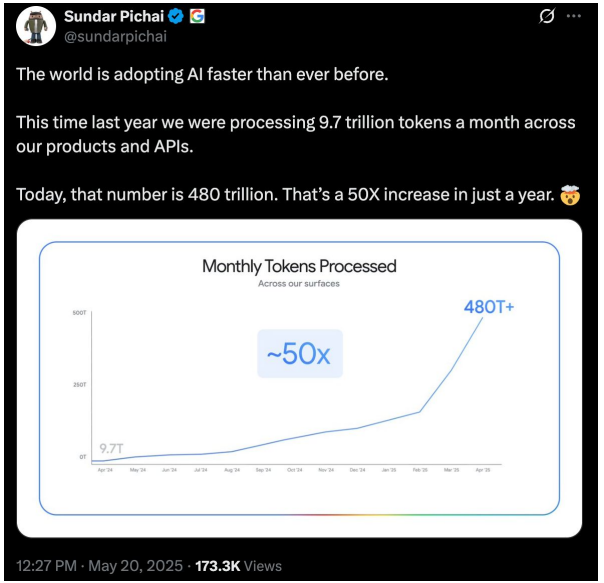
- Synthetic data vs real-world data
  - Distribution gap
  - Quality comparison
- The scaling of synthetic data
  - Model collapse?
  - Self-improving?
  - The information perspective
- Licensing & copyright

Synthetic vs Real-world data

- Real-world data refers to data produced by **real users, when serving their actual demands**.

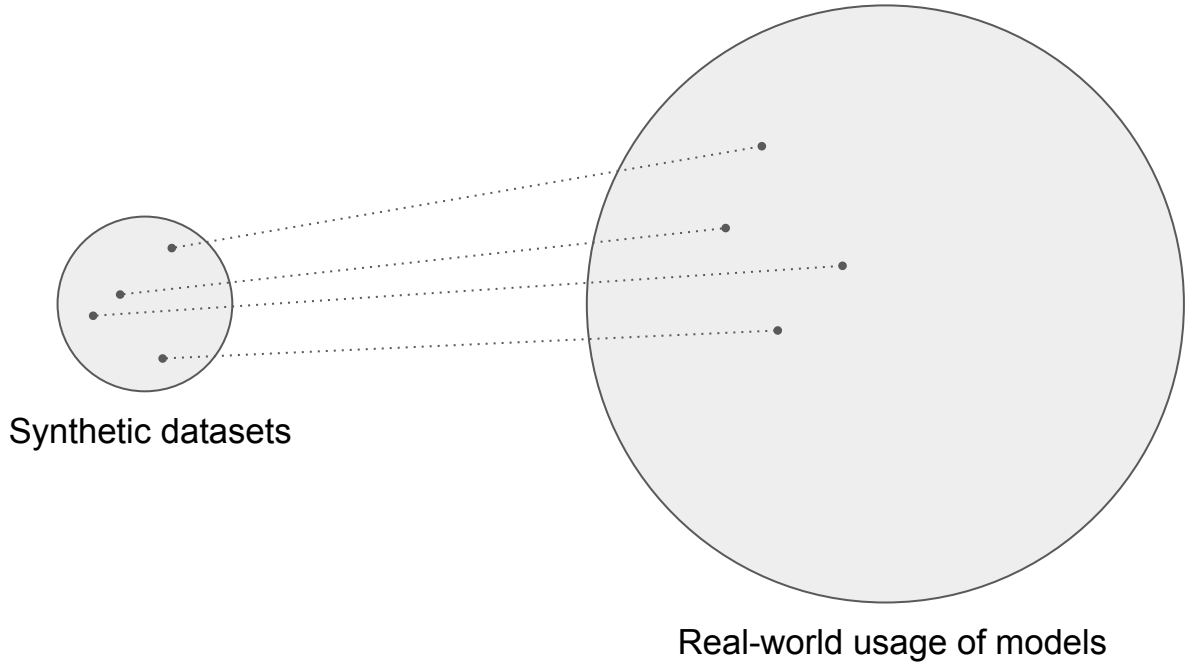
Synthetic vs Real-world data

- Real-world data refers to data produced by **real users, when serving their actual demands**.



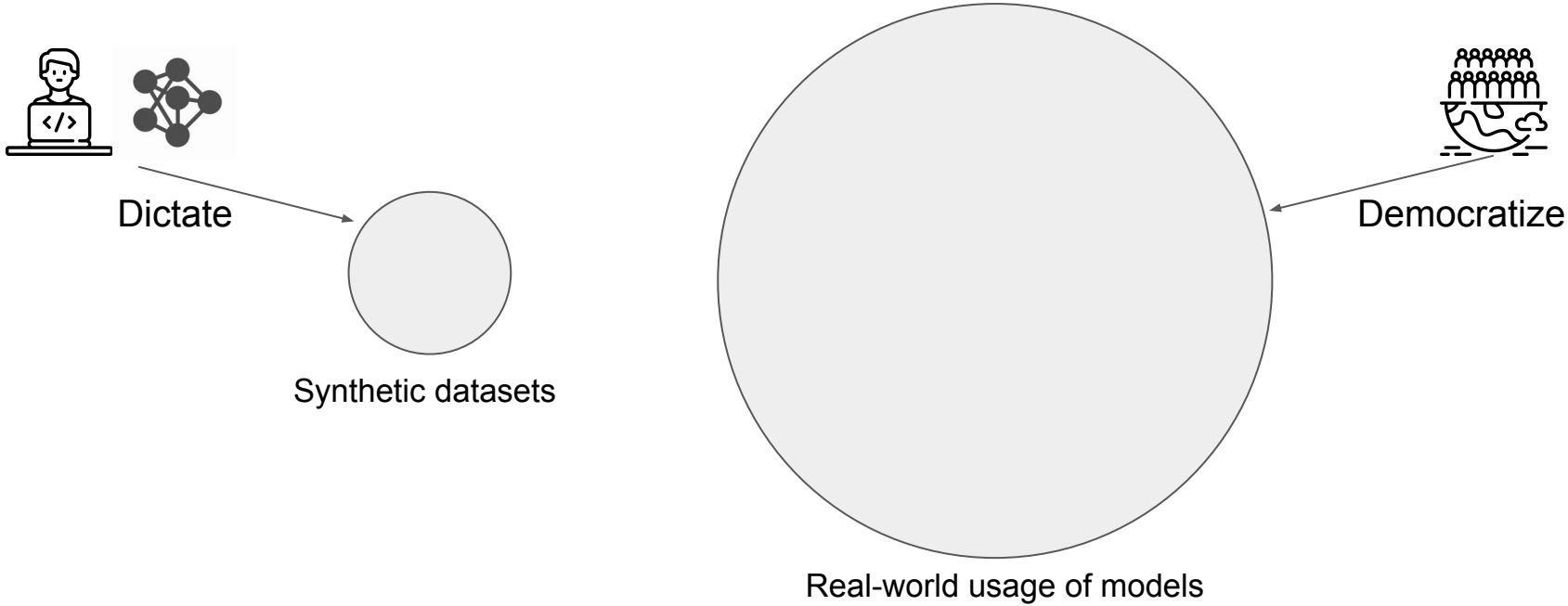
Synthetic vs Real-world data

- There is still a significant **gap in size, diversity, & distribution**!



Synthetic vs Real-world data

- Fundamentally, they are produced differently.



Understanding real-world usage

- Anthropic Clio [Tamkin et al., 2024]

Understanding real-world usage

- Anthropic Clio [Tamkin et al., 2024]
- **Open** data: WildChat [Zhao et al., 2024]
- **Open** data: LMSys-Chat-1M [Zheng et al., 2024]
- **Open-source tool** for analysis: EvalTree [Zeng et al., 2025]
- It's very hard to capture the nuances and richfulness of real-world data.

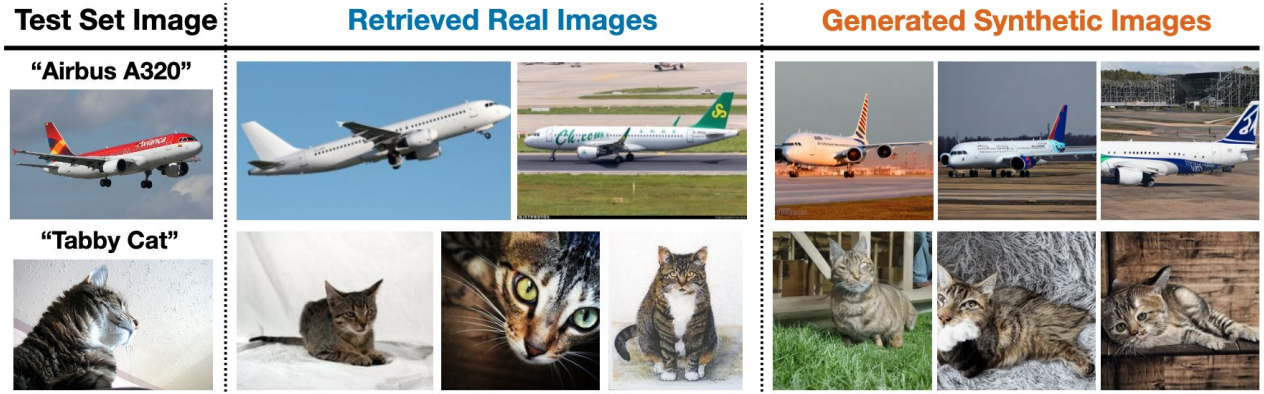
How does synthetic data compare to real data in terms of quality?

How does synthetic data compare to real data in terms of quality?

- In a controlled setup, synthetic data still underperform real data (if available) [Geng et al., 2024].

How does synthetic data compare to real data in terms of quality?

- In a controlled setup, synthetic data still underperform real data (if available) [Geng et al., 2024].
- Analysis shows synthetic data often contains generator artifacts and distort class-level visual content.



How can we measure the quality of data?

- **Quality control** is a critical step in classic data pipeline [Daniel et al., 2018]
- Most synthetic datasets from the community ***do not*** **implement quality checks**.

How can we measure the quality of data?

- There are two common proxies for synthetic data quality:
  - Better **downstream performance** of the trained model implies better quality of the data.
  - Distilling from **stronger generation models** (e.g., GPT4) produces data of better quality.

How can we measure the quality of data?

- There are two common proxies for synthetic data quality:
  - Better **downstream performance** of the trained model implies better quality of the data.
  - Distilling from **stronger generation models** (e.g., GPT4) produces data of better quality.
- Both of them are not always true, as there are many moving factors in the generator, data, and downstream tasks [Kim et al. 2025].

Ideally, we also want to measure instance-level quality

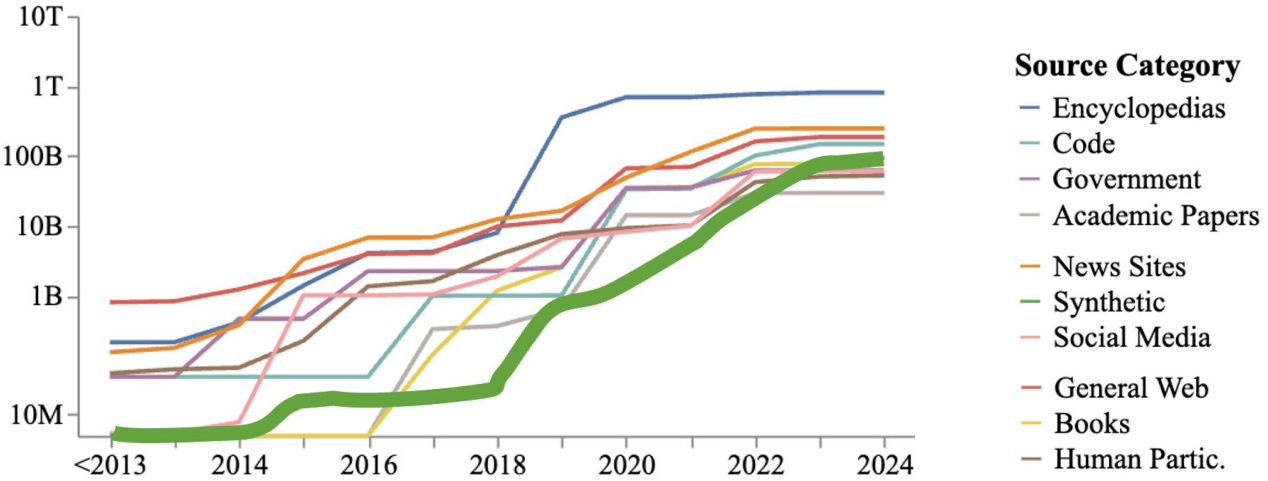
- Reward models
- LLM as a judge
- Rule-based verification
- Decomposition [Min et al., 2023; Li et al., 2024]
- However, building a **generalist verifier** is hard! [Sutton, 2001]

What if synthetic data scales up?

- AI models will be trained on **increasing** amount of model-generated data, inevitably.
  - Model builders intentionally add them.
  - Model-generated content populates on the Internet

What if synthetic data scales up?

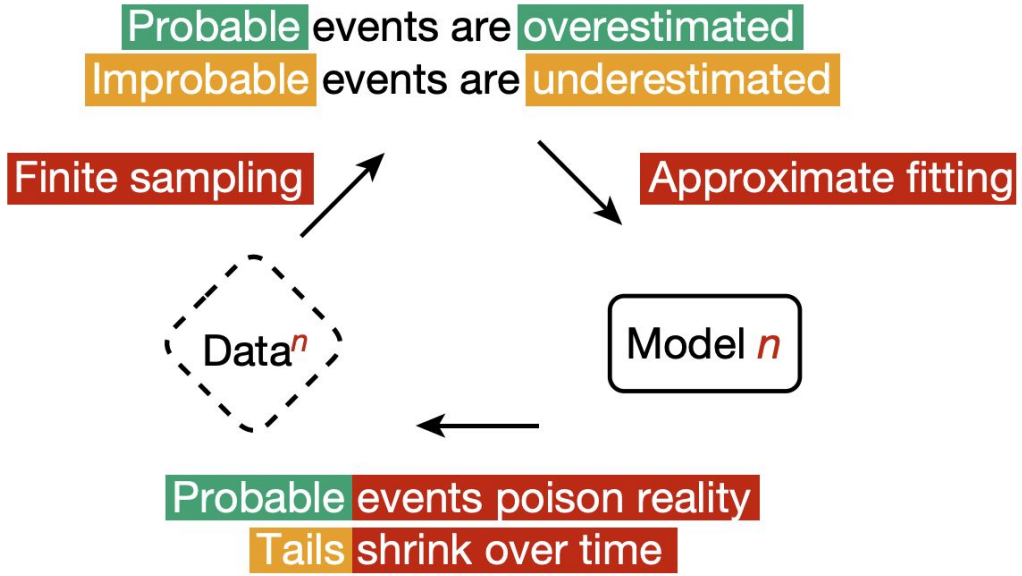
- AI models will be trained on **increasing** amount of model-generated data, inevitably.



The cumulative size of text data from different sources for post-training LMs [Longpre et al., 2024]

Model collapse or self-improving

- AI models collapse when trained on recursively generated data [Shumailov et al., 2023].



Model collapse or self-improving

- AI models collapse when trained on recursively generated data [Shumailov et al., 2023].
- AI models can **self-**instruct [Wang et al., 2022], **self-**improve [Huang et al., 2022], **self-**refine [Madaan et al., 2023], **self-**reward [Yuan et al., 2024], etc.

Model collapse or self-improving

- AI models collapse when trained on recursively generated data [Shumailov et al., 2023].

What makes the difference?

- AI models can self-instruct [Wang et al., 2022], self-improve [Huang et al., 2022], self-refine [Madaan et al., 2023], self-reward [Yuan et al., 2024], etc.

Bringing in additional information

- Human selection, editing, & supervision.
- Human prior in the generation pipeline design.
  - E.g., prompts, principles for filtering, etc.
- Grounded documents, retrieved information, tool results.
- Rewards from interacting with environments.
- …

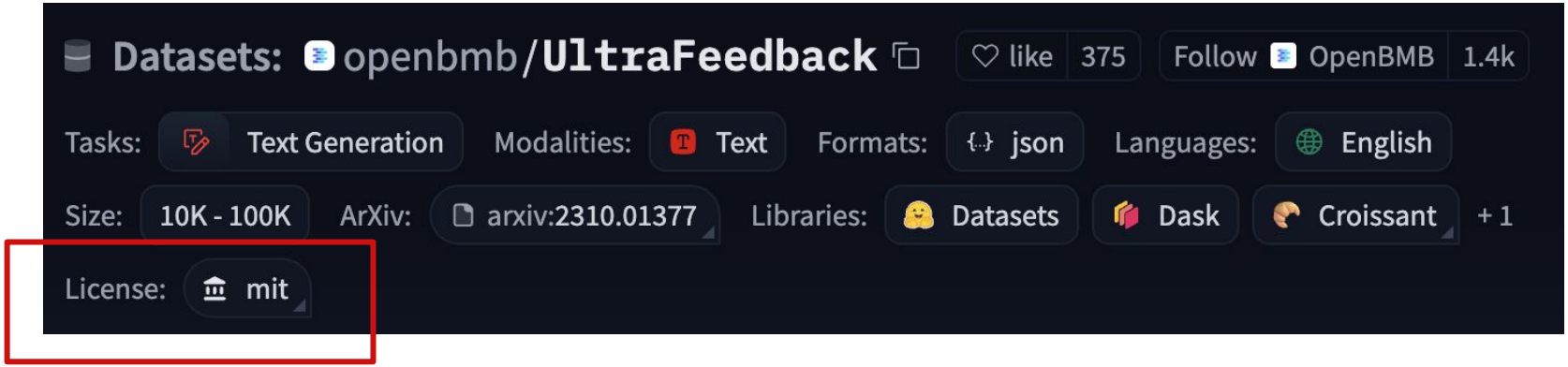
Bringing in additional information

- Human selection, editing, & supervision.
- Human prior in the generation pipeline design.
  - E.g., prompts, principles for filtering, etc.
- Grounded documents, retrieved information, tool results.
- Rewards from interacting with environments.
- …
- Synthetic data **accelerates the combination & transformation** of useful information.

What licensing is proper for synthetic data?

What licensing is proper for synthetic data?

- Ultrafeedback as an example: A widely-used synthetic preference dataset with MIT license [Cui et al., 2024]



What licensing is proper for synthetic data?

- The licensing of synthetic datasets is tricky:
  - The dataset might be transformed from **other less permissive datasets.**

Source datasets of Ultrafeedback	License
evol_instruct	MIT
false_qa	**Unclear**
flan	**Apache 2.0 only for generation code**
sharegpt	**Unclear**
truthful_qa	Apache 2.0
ultrachat	MIT

What licensing is proper for synthetic data?

- The licensing of synthetic datasets is tricky:
  - The dataset might be transformed from other less permissive datasets.
  - A lot of generations rely on **distillation-unfriendly models**.

What licensing is proper for synthetic data?

DeepSeek ToS

“ You may apply the Inputs and Outputs of the Services to a wide range of use cases, including personal use, academic research, derivative product development, **training other models (such as model distillation)**, etc. ”

Quoted as of 07/22/2025

What licensing is proper for synthetic data?

OpenAI ToS

“ You may not use our Services for any illegal, harmful, or abusive activity. For example, you may not:

…

Use output to develop models that compete with OpenAI. ”

Anthropic ToS

“ You may not access or use our services

…

To develop any products or services that compete with our Services, including to develop or train any AI or ML algorithms or models or resell the Services. ”

Gemini API ToS

“ You may not use the Services **to develop models that compete with the Services** (e.g., Gemini API or Google AI Studio).”

Quoted as of 07/22/2025

What licensing is proper for synthetic data?

- It's not clear about the ownership/copyright of AI-generated content [more guidelines from US Copyright Office].
- Technically, how to detect and enforce whether one model's outputs are used to train another model?

Summary of open problems

- **Diversity & distribution**: understand the richfulness of real-world data and close the gap.
- **Quality:** develop quality checks/validation methods to promote high-quality data.
- **New information**: bring in additional information in model self-improving, and avoid model collapse.
- **Licensing & copyright:** call for lawful guidance on the use of synthetic data, and the community for technical innovation and ethical practice.
- **Many more** …