Exploring Trustworthy Foundation Models: Benchmarking, Finetuning, and Reasoning

Prof. Bo Han
HKBU TMLR Group / RIKEN AIP Team
Associate Professor / BAIHO Visiting Scientist
https://bhanml.github.io

Trustworthy Foundation Models

Benchmarking

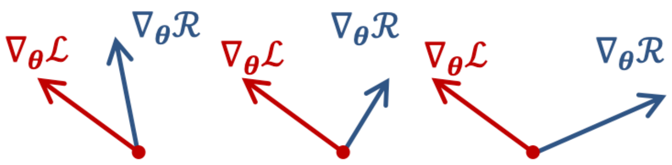
Existing datasets are NOT proper to assess if **VLMs** are robust.

CounterAnimal, a reliable benchmark for assessing VLMs.

- **Scaling backbone models** and **improving data quality** improve the robustness of VLMs.
- **Scaling raw training data** does not necessary enhance reliability.

Finetuning

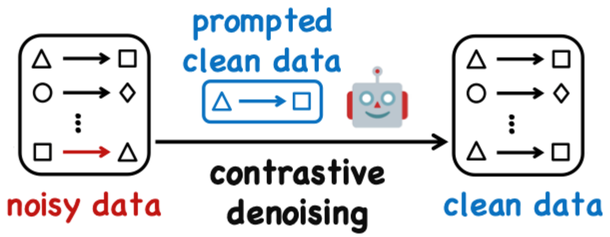
Analyzing the dynamics of **LLMs** unlearning is critical yet hard.



- **Analyzing gradients** provides insights into unlearning dynamics.
- **Wrong token reweighting** within gradients leads to failures in previous methods.

Reasoning

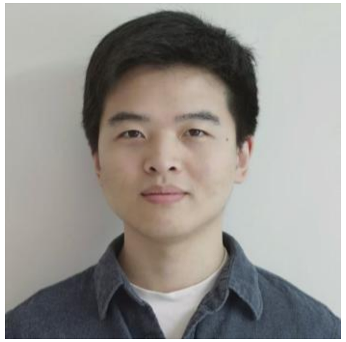
Noisy rationales within chain of thoughts mislead **LLMs** reasoning.



- It is **hard** for LLMs to denoise noisy rationales without guidance.
- It is **easier** for LLMs to denoise by contrasting noisy and clean data.

Part I: Benchmarking

Qizhou Wang

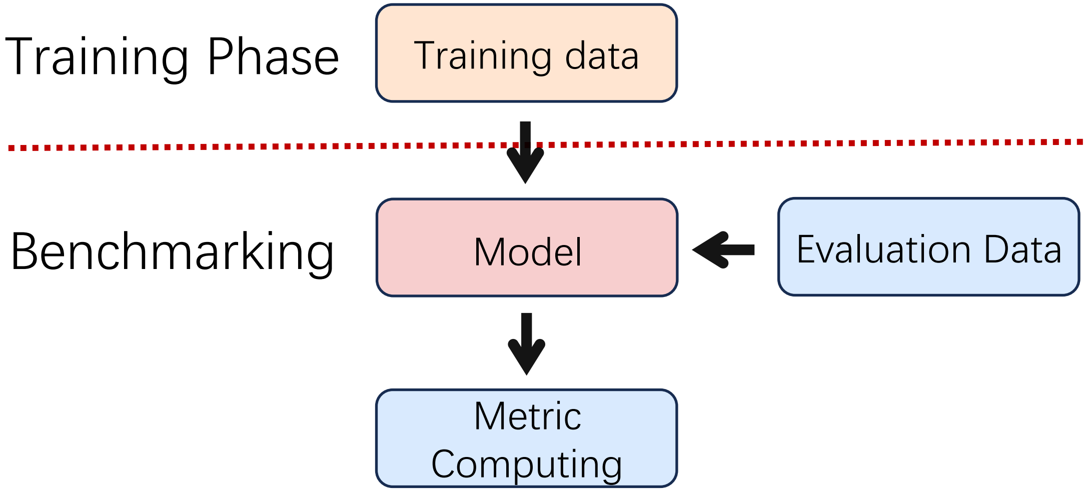


Yongqiang Chen

Benchmarking is critical to evaluate and compare model quality.

- Gathering **reliable evaluation data.**
- Conducting **proper metric evaluations.**

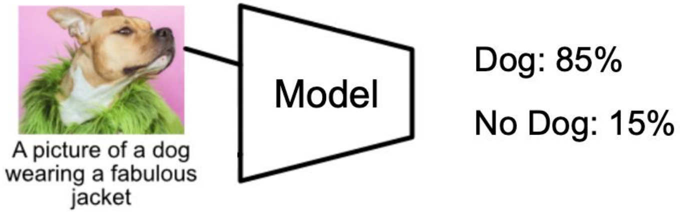
Training and evaluation data have **distribution shifts** to reflect **OOD Generalization**.



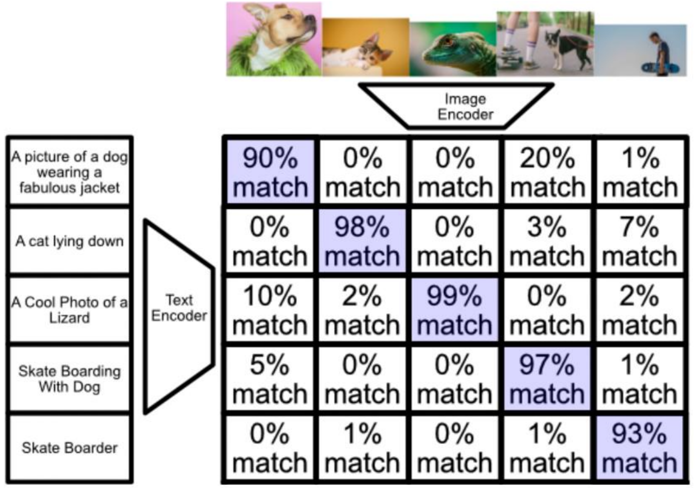
Qizhou Wang, Yong Lin, **Yongqiang Chen**, Ludwig Schmidt, **Bo Han**, and Tong Zhang.
A Sober Look at the Robustness of CLIPs to Spurious Features. In *NeurIPS*, 2024.

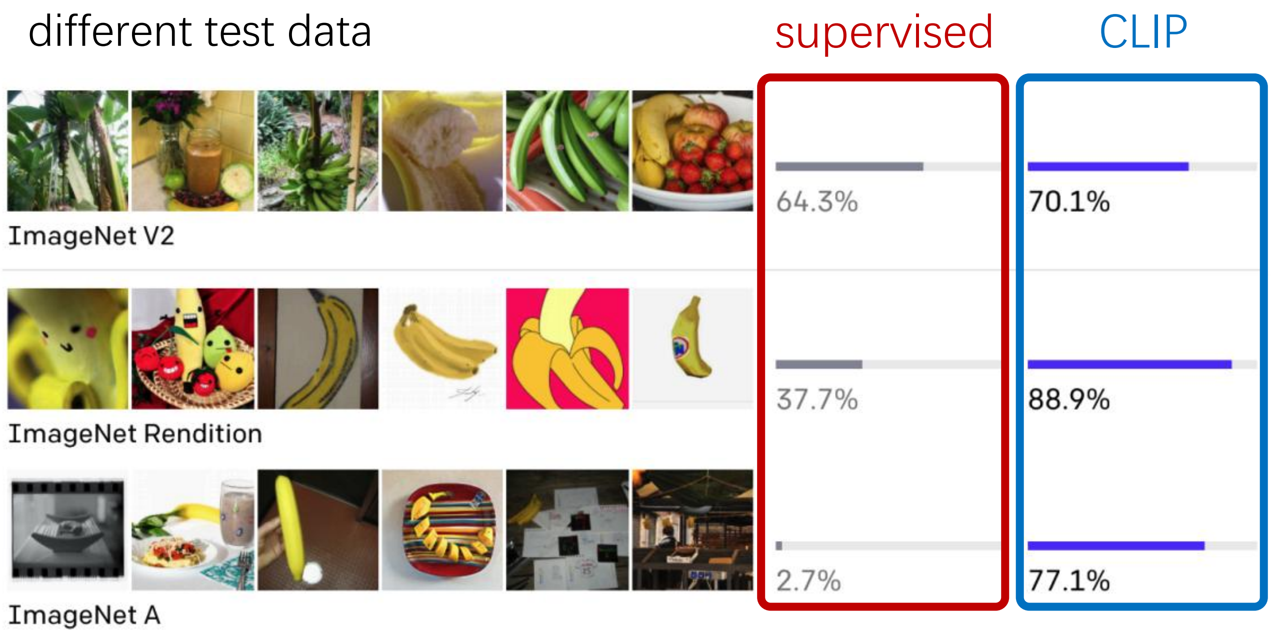
Supervised vs CLIP Training

Supervised Training *label supervision*



CLIP Training *cross-modal supervision*





*Comparison of the **OOD evaluation accuracy** between supervised and CLIP training shows that **CLIP performs better!***

Previous Belief: CLIP is **more robust to distribution shifts** than conventional supervised training.

(Radford et al., 2021)

Is the Conclusion Correct?

These OOD datasets are crafted for the distribution shifts **within ImageNet setups**, which are **NOT valid for CLIP models.**

- **Data Contamination:** Datasets considered **OOD for ImageNet-trained models** may be **ID for CLIP models.**

ImageNet V2

CLIP models may have seen ImageNet V2 during training, which is in fact ID for CLIP setups.

- **Biased Spuriousness:** Features that **mislead ImageNet-trained models** may **not mislead CLIP models** necessarily.

ImageNet A

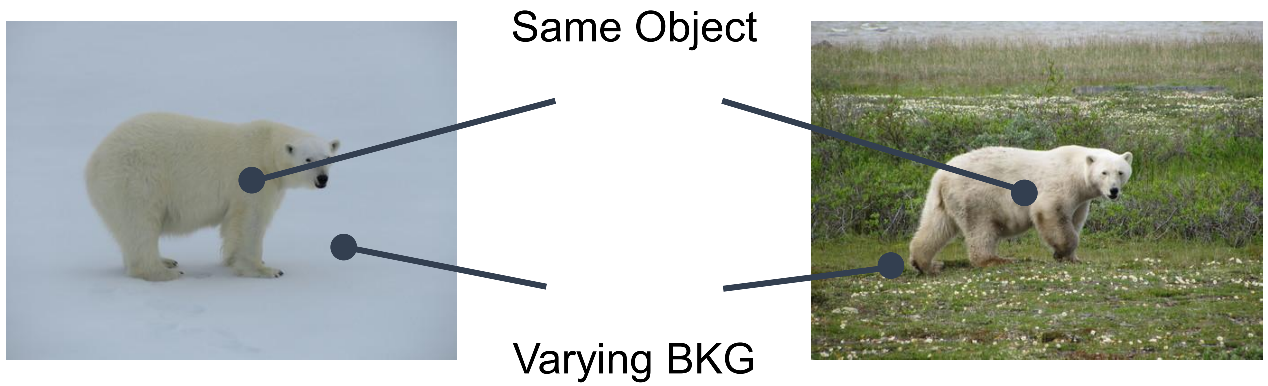
ImageNet A contains data that mislead ImageNet models, which may not make CLIP models fail.

ImageNet OOD datasets CANNOT reflect the OOD Generalization for CLIP setups!

CounterAnimal: A New Benchmark

Is there a benchmark capturing **true OOD performance** of CLIP?

- **Spuriousness:** Considering **background changes** as potential spurious features.
- **Generality:** The captured spurious features should impact **diverse CLIP configurations**.



common

uncommon

The changes of backgrounds represent the impacts of spurious features, which is a typical distribution shift.

Basic Assumption: Since "ice bears" are more commonly appear with "ice" rather than "grass" backgrounds, CLIP may rely on ice-related spurious features.

CounterAnimal Construction

Step 1. Data Collection

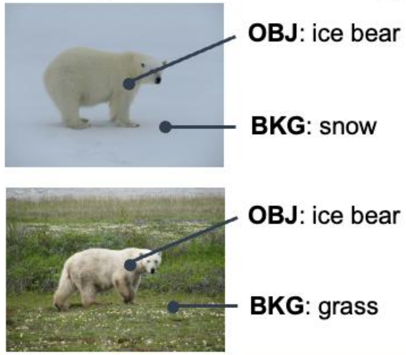
Raw data from iNaturalist (https://www.inaturalist.org)

query and crawl **animal photos** given the names

Step 2. Data Curation

Raw data are susceptible to **noise** and **ambiguities**, which should be **cleansed manually**.

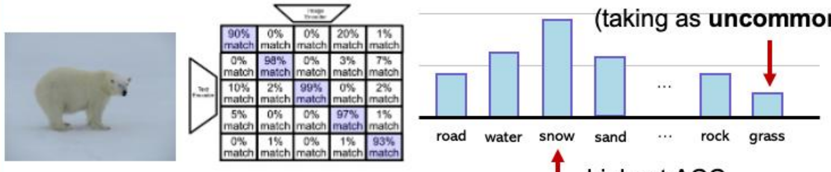
Step 3. Data Labelling



OBJ labels: *ostrich, African crocodile, water snake, ice bear*, and other totally **45 animal names**.

BKG labels: *ground, water, earth*, and other totally **16 background labels**.

Step 4. Spurious Discovery



The pair of backgrounds where the CLIP shows high-performance drops are preserved.

CounterAnimal Construction

Step 1. Data Collection

Raw data from iNaturalist (https://www.inaturalist.org)

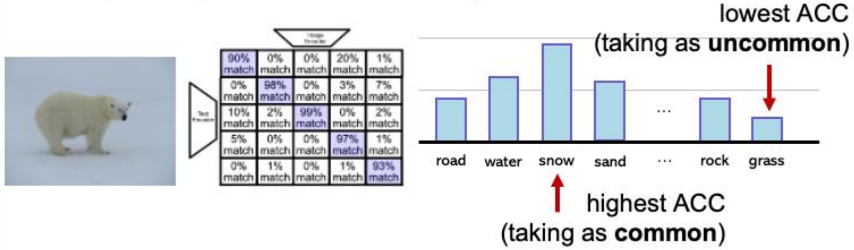
query and crawl **animal photos** given the names

Step 2. Data Curation

Raw data are susceptible to **noise** and **ambiguities**, which should be **cleansed manually**.

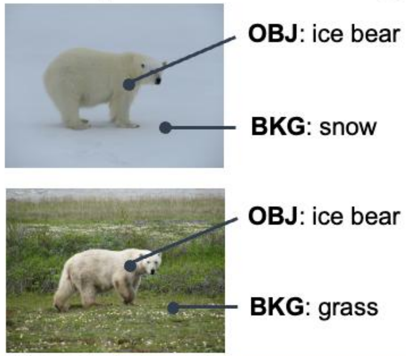
clean | noise | occlusion | obscurity

Step 4. Spurious Discovery



The pair of backgrounds where the CLIP shows high-performance drops are preserved.

Step 3. Data Labelling



OBJ labels: *ostrich*, *African crocodile*, *water snake*, *ice bear*, and other totally **45 animal names**.

BKG labels: *ground*, *water*, *earth*, and other totally **16 background labels**.

CounterAnimal Construction

Step 1. Data Collection

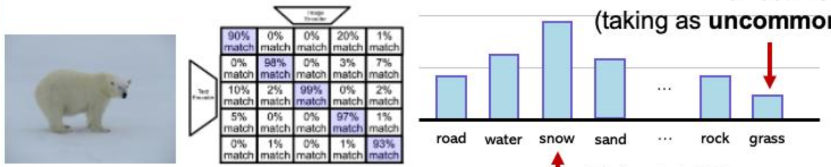
Raw data from iNaturalist (https://www.inaturalist.org)

query and crawl **animal photos** given the names

Step 2. Data Curation

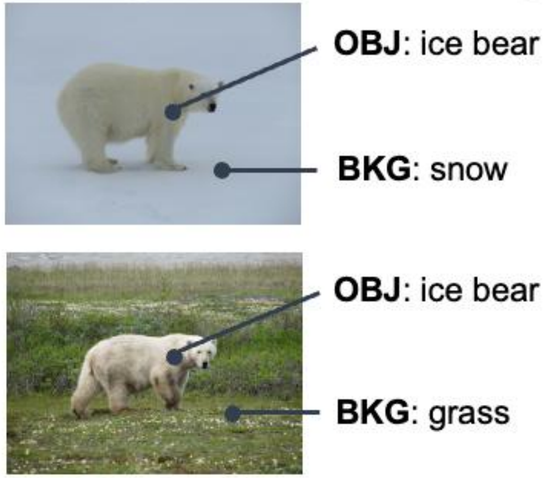
Raw data are susceptible to **noise** and **ambiguities**, which should be **cleansed manually**.

Step 4. Spurious Discovery



The pair of backgrounds where the CLIP shows high-performance drops are preserved.

Step 3. Data Labelling



OBJ labels: *ostrich*, *African crocodile*, *water snake*, *ice bear*, and other totally **45 animal names**.

BKG labels: *ground*, *water*, *earth*, and other totally **16 background labels**.

CounterAnimal Construction

Step 1. Data Collection

Raw data from iNaturalist (https://www.inaturalist.org)

query and crawl **animal photos** given the names

Step 2. Data Curation

Raw data are susceptible to **noise** and **ambiguities**, which should be **cleansed manually**.

clean noise occlusion obscurity

Step 3. Data Labelling

OBJ: ice bear

BKG: snow

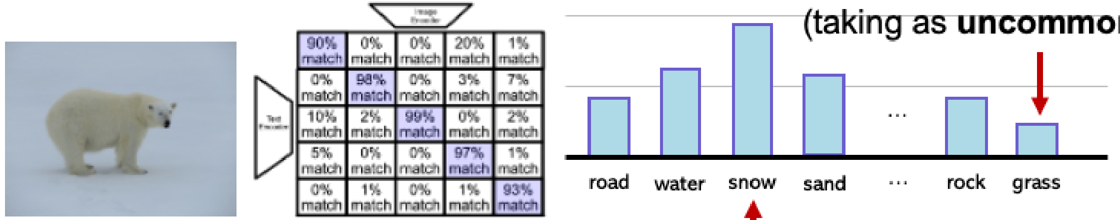
OBJ: ice bear

BKG: grass

OBJ labels: *ostrich*, *African crocodile*, *water snake*, *ice bear*, and other totally **45 animal names**.

BKG labels: *ground*, *water*, *earth*, and other totally **16 background labels**.

Step 4. Spurious Discovery



lowest ACC (taking as **uncommon**)

highest ACC (taking as **common**)

The pair of backgrounds where the CLIP shows high-performance drops are preserved.

CounterAnimal Characteristics

```
CounterAnimal
├── ice bear
│   ├── common-ice
│   │   ├── figure1.jpeg
│   │   ├── figure2.jpeg
│   │   └── ...
│   └── uncommon-grass
│       ├── figure1.jpeg
│       ├── figure2.jpeg
│       └── ...
└── brambling
    ├── common-green
    │   ├── figure1.jpeg
    │   ├── figure2.jpeg
    │   └── ...
    └── counter-sky
        ├── figure1.jpeg
        ├── figure2.jpeg
        └── ...
```

Data Structure. *Images are organized per class and each further divided into two groups: common and uncommon.*

*Photos of **ice bear** in **snow** background*

*Photos of **ice bear** in **grass** background*

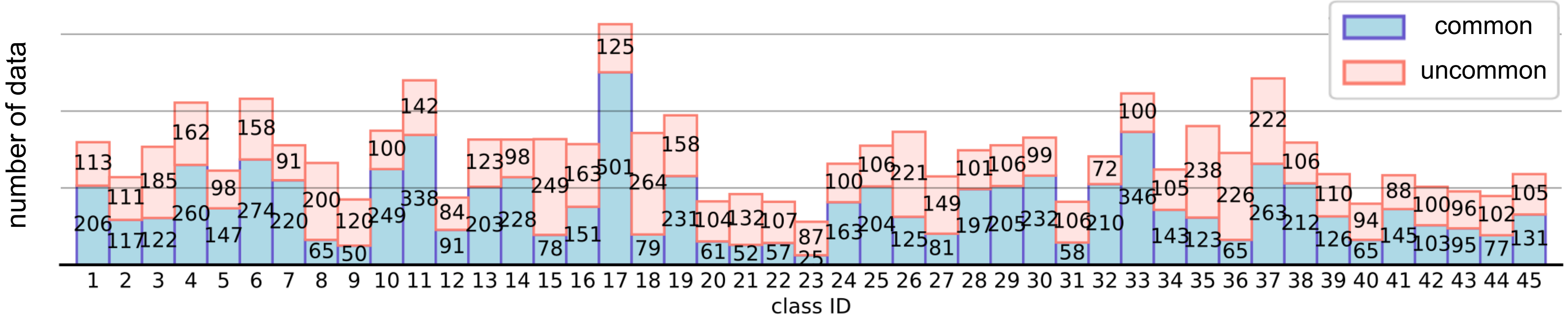
Common vs. Uncommon: Photos are grouped according to their backgrounds. For each class, we identify **group pairs** that cause **high performance drop** when evaluating with CLIP.

Assessing Robustness: The **performance drop** between common and uncommon groups indicates the robustness of evaluated models.

CounterAnimal Characteristics

```
CounterAnimal
├── ice bear
│   ├── common-ice
│   │   ├── figure1.jpeg
│   │   ├── figure2.jpeg
│   │   └── ...
│   └── uncommon-grass
│       ├── figure1.jpeg
│       ├── figure2.jpeg
│       └── ...
└── brambling
    ├── common-green
    │   ├── figure1.jpeg
    │   ├── figure2.jpeg
    │   └── ...
    └── counter-sky
        ├── figure1.jpeg
        ├── figure2.jpeg
        └── ...
```

Data Structure. *Images are organized per class and each further divided into two groups: common and uncommon.*



The ***data distributions*** *illustrate variations across different animal classes, categorized into* ***common*** *and* ***uncommon*** *groups. The horizontal axis denotes the* ***class IDs****, e.g., ID 1 to "ostrich", ID 2: to "brambling", ···, ID 8 to "box turtle", ID 9 to "common iguana",···, ID 18 to "scorpion", ID 19 to "tarantula", ···, ID 32 to "African hunting dog", ID 33 to "hyena", ···.*

We collect **45 classes** of animals with **7,000 common** and **6,000 uncommon** examples.

Experimental Results

common acc – uncommon acc

CLIP Training

CounterAnimal

backbone	pre-train dataset	common	uncommon	drop
RN-101	OpenAI	64.27	45.15	19.12
RN-50×4	OpenAI	70.02	49.07	20.95
ViT-B/16	LAION400M	73.11	52.17	20.94
ViT-B/16	OpenAI	73.08	56.56	16.52
ViT-B/16	DataComp1B*	80.36	64.24	16.12
ViT-B/16	LAION2B	73.18	53.18	20.00
ViT-B/16	DFN2B*	85.03	70.61	14.42
ViT-B/32	LAION400M	67.13	36.95	30.18
ViT-B/32	OpenAI	69.13	45.62	23.51
ViT-B/32	DataComp1B*	75.96	53.74	22.22
ViT-B/32	LAION2B	72.94	48.74	24.20
ViT-L/14	LAION400M	80.90	63.31	17.59
ViT-L/14	OpenAI	85.38	70.28	15.10
ViT-L/14	DataComp1B*	89.29	79.90	9.39
ViT-L/14	LAION2B	82.23	66.27	15.96
ViT-L/14	DFN2B*	90.77	80.55	10.22
ViT-L/14-336	OpenAI	86.36	73.14	13.21
ViT-H/14	LAION2B	85.74	73.13	12.61
ViT-H/14	DFN5B*	88.55	79.13	9.42
ViT-G/14	LAION2B	86.81	73.32	13.49
ViT-bigG/14	LAION2B	87.57	76.96	10.61

increasing model scale **diverse data source**

(ImageNet) Supervised Training

backbone	common	uncommon	drop
AlexNet	59.56	39.24	20.31
VGG-11	73.37	56.12	17.25
VGG-13	75.33	58.43	16.90
VGG-19	77.84	61.74	16.10
RN-18	74.36	56.07	18.29
RN-34	78.31	61.01	17.30
RN-50	81.44	66.07	15.37
RN-101	81.76	68.18	13.57
ViT-B/16	84.97	74.98	9.99
ViT-B/32	79.84	64.36	15.48
ViT-L/16	83.74	72.69	11.05
ViT-L/32	81.23	67.54	13.69
ConvNext-S	88.27	79.97	8.30
ConvNext-B	88.60	80.53	8.07
ConvNext-L	89.12	81.47	7.65

increasing model scale

Other LVLMs (large VLMs)

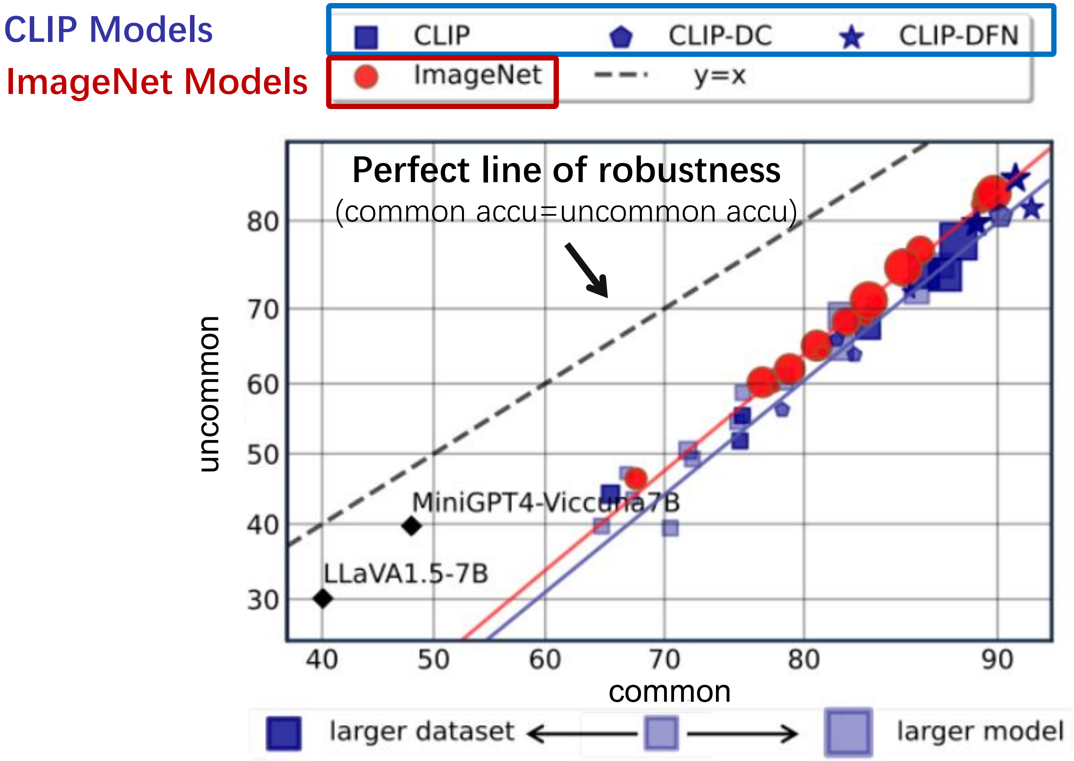
LVLMs	common	uncommon	drop
MiniGPT4-Viccuna7B	47.99	39.73	8.26
LLaVA1.5-7B	40.06	30.09	9.97
CLIP-LAION400M-ViT-L/14	80.90	63.31	17.59
CLIP-OpenAI-ViT-L/14	85.38	70.28	15.10
CLIP-DataComp1B-ViT-L/14	89.29	79.90	9.39
CLIP-LAION2B-ViT-L/14	82.23	66.27	15.96
CLIP-DFN2B-ViT-L/14	90.77	80.55	10.22

different LVLM paradigms

What observations can we draw from these results?

Observations

DataComp (DC) and Data Filtering Networks (DFN) are two **high-quality CLIP data sources.**



*The **marker size** indicates the **backbone scale**, and the **color shade** indicates **pre-train data scale**.*

Observation 1 (ImageNet Models vs. CLIPs).

ImageNet models perform better than CLIPs against spuriousness within CounterAnimal.

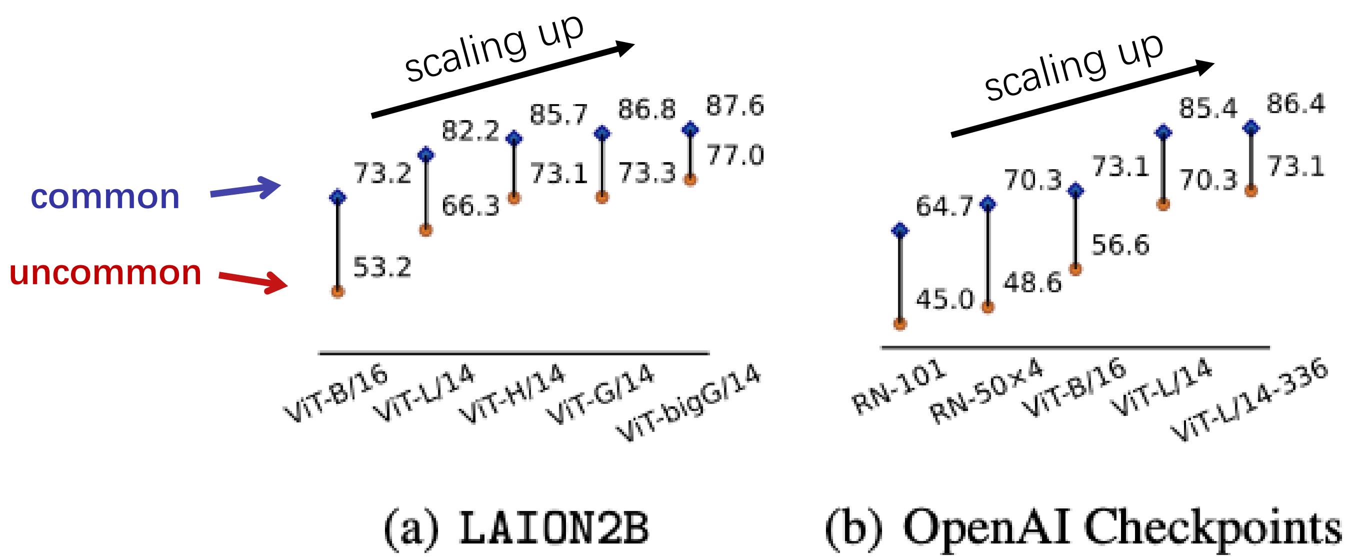
Note. CounterAnimal characterizes the spuriousness within CLIPs, thus proper for assessing CLIPs.

Observation 2 (CLIPs vs. More Advanced LVLMs).

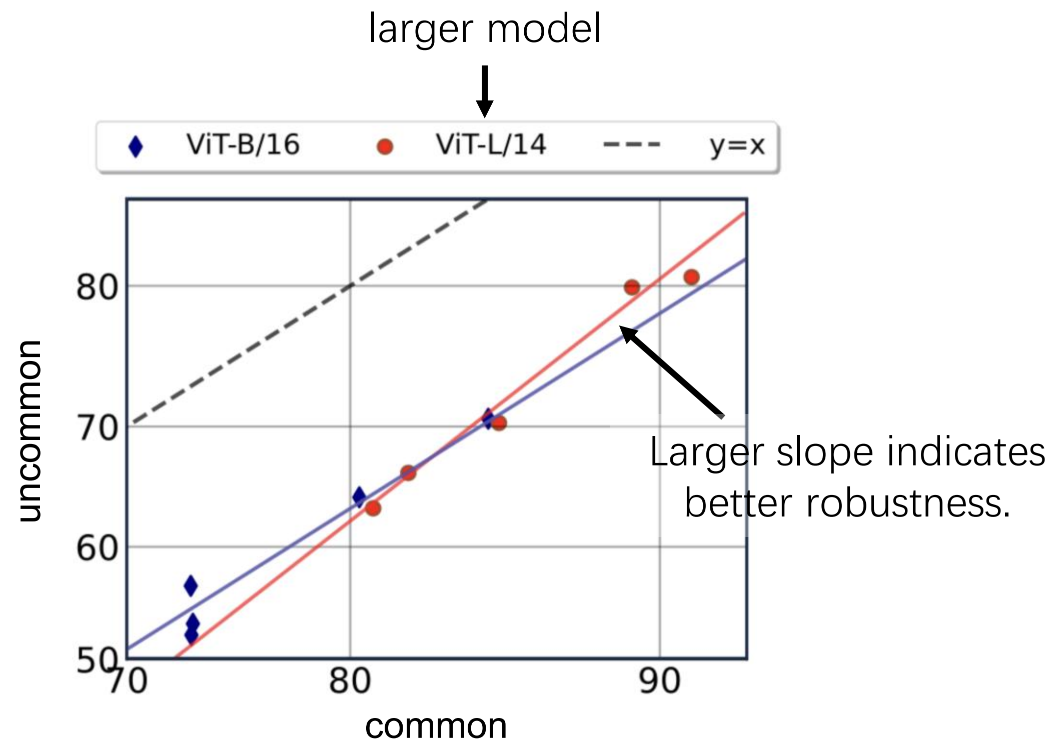
LLaVA and MinGPT4 show **stronger robustness** (closer to $y = x$) yet with **lower performance** than CLIPs.

Note. More advanced VLMs built upon CLIPs are still affected by spuriousness within CounterAnimal.

Observations



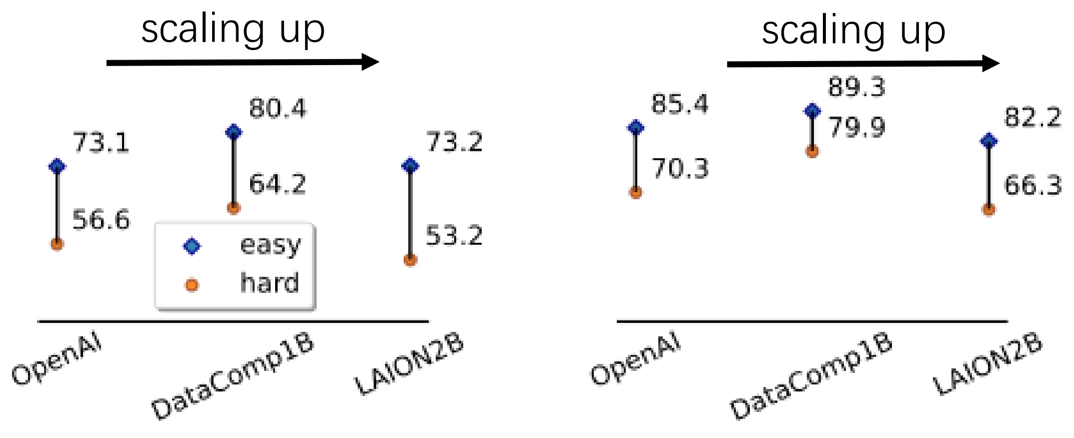
Accuracy and Performance Drop.



Effective Robustness.

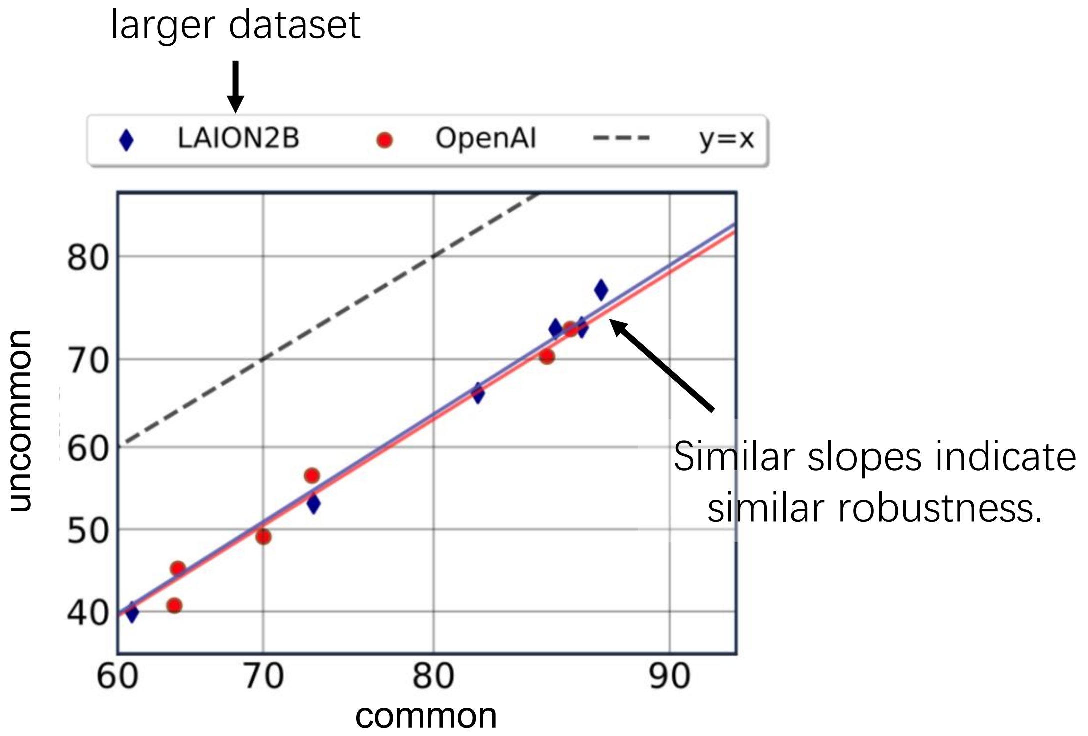
Observation 3 (Model Size). Scaling up model size CAN enhance CLIP robustness.

Observations



(a) ViT-B/16 (b) ViT-L/14

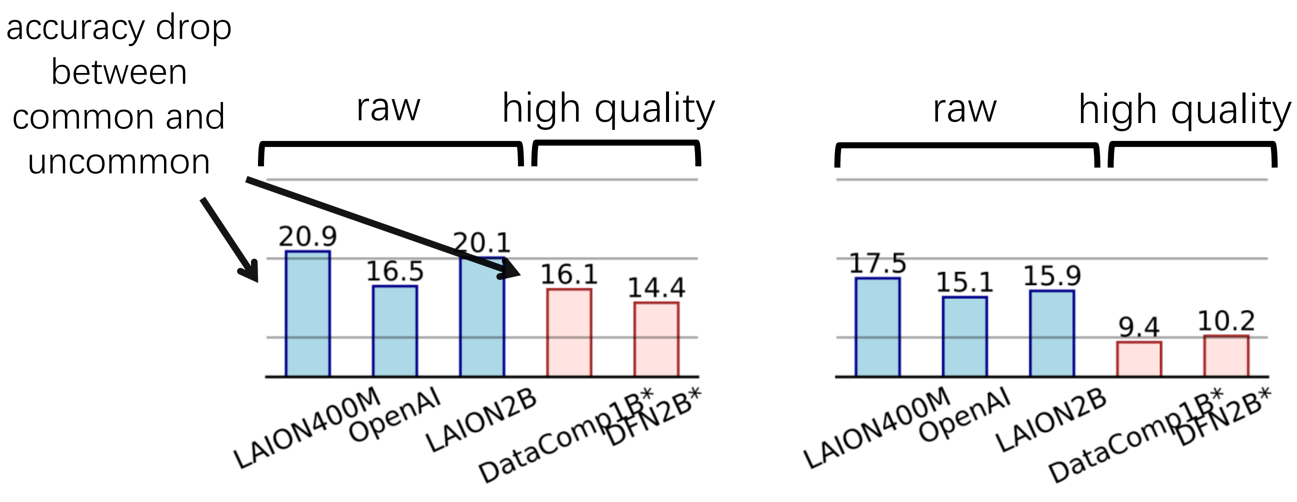
Accuracy and Performance Drop.



Effective Robustness.

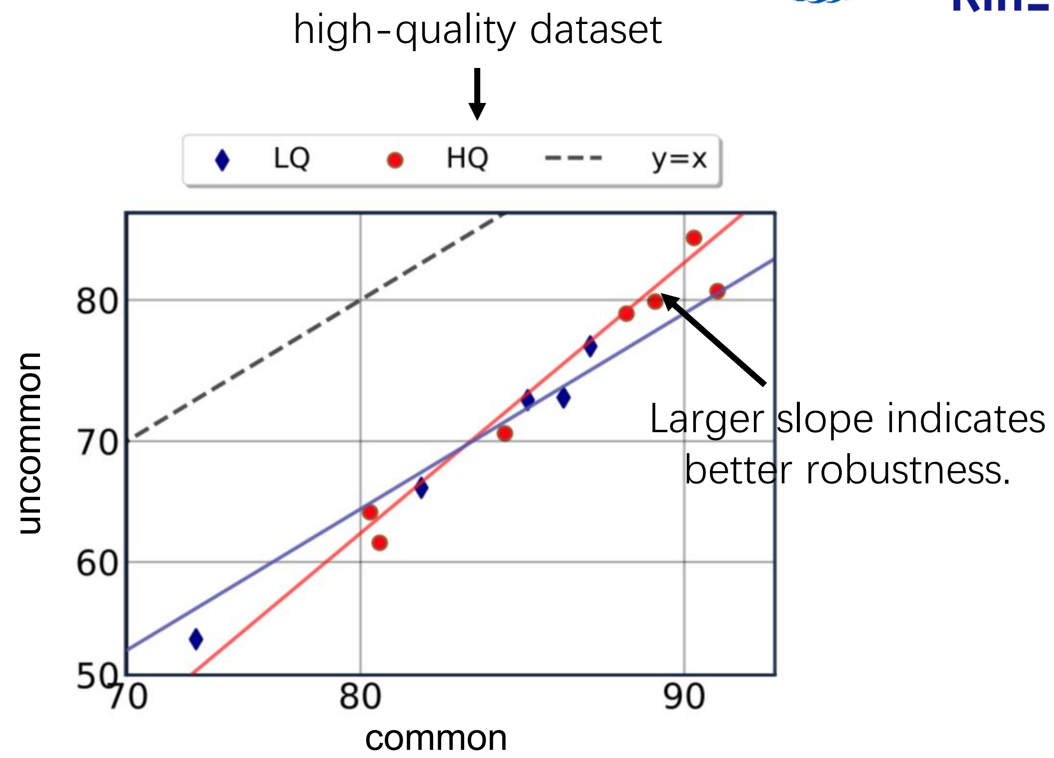
Observation 4 (Data Size). Scaling up data size CANNOT enhance CLIP robustness.

Observations



(a) ViT-B/16 (b) ViT-L/14

Performance Drop.



Effective Robustness.

Observation 5 (Data Quality). Improving data quality CAN enhance CLIP robustness.

Theoretical Understanding

Assumption (Multi-modal Dataset). Considering n image-text pairs $\{(\boldsymbol{x}_{\mathrm{I}}^{i}, \boldsymbol{x}_{\mathrm{T}}^{i})\}_{i=1}^{n}$, both $\boldsymbol{x}_{\mathrm{I}}^{i}$ and $\boldsymbol{x}_{\mathrm{T}}^{i}$ are generated from the latent factor $\boldsymbol{z}_i$, where $\boldsymbol{z} = [z_{inv}, z_{spu}] \in \mathbb{R}^2$ is composed of

- **invariant feature** $z_{inv} \sim \mathcal{N}(\mu_{inv} y, \sigma_{inv}^2)$
- **spurious feature** $z_{spu} \sim \mathcal{N}(\mu_{spu} a, \sigma_{spu}^2)$

with $\Pr(a = y) = p_{spr}$ otherwise $a = -y$. y is the label uniformly drawn from $\{-1,1\}$. The training data $\mathcal{D}^{tr}$ is drawn with $\frac{1}{2} \leq p_{spr} \leq 1$ and test data $\mathcal{D}^{*}$ is drawn with $p_{spu} = \frac{1}{2}$.

Note. The dataset is **biased** to spurious feature z_{spu} due to **different** p_{spr} between training and test.

Theorem 1. Given the multi-modal dataset with a large spurious correlation $p_{spu} = 1 - o(1)$. Then, under reasonable assumptions, w.p. at least $1 - O(1)$, the CLIP model achieves

- a **small zero-shot error** on test data where $a = y$: $\mathrm{Acc}(g_{\mathrm{I}}, g_{\mathrm{T}}) \geq 1 - \Phi(\kappa_2) - o(1)$,
- a **large zero-shot error** on test data where $a \neq y$: $\mathrm{Err}(g_{\mathrm{I}}, g_{\mathrm{T}}) \geq 1 - \Phi(\kappa_1) - o(1)$.

Therein, κ_1, κ_2 are constants that depend on $\mu_{inv}, \sigma_{inv}, \mu_{inv}$, and σ_{inv}.

Note. The model relies on whether $a = y$ (whether biased) to make right predictions.

Take Home Messages

We should be cautious about **test setups** when assessing new **training setups**.

CounterAnimal (https://counteranimal.github.io/) is a proper benchmark for assessing the robustness of CLIPs to spurious features.

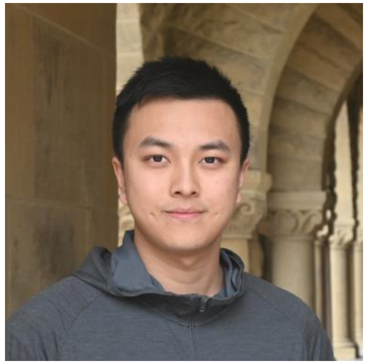
Distribution shifts remain an open question for CLIP and other VLMs.

Scaling up model size can enhance robustness, while **scaling up pre-train data** is not that effective.

Improving data quality is effective to enhance robustness.

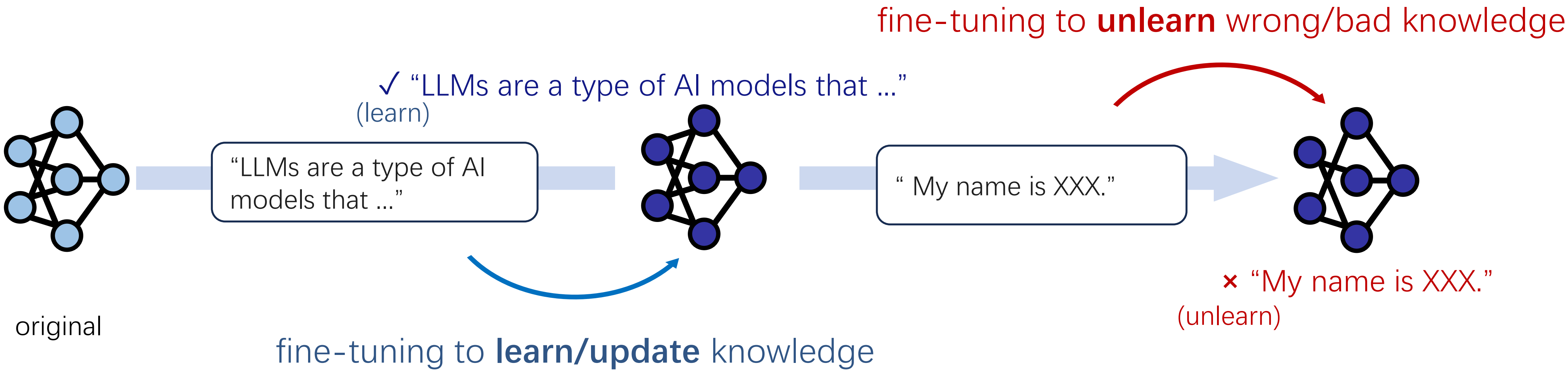
Part II: Finetuning

Qizhou Wang



Zhanke Zhou

Finetuning aims to adapts the model parameters to fit tasks or knowledge, of which the specific goals can be attributed to **learning** and **unlearning**.



Qizhou Wang, Jin Peng Zhou, **Zhanke Zhou**, Saebyeol Shin, **Bo Han**, Kilian Q. Weinberger. Rethinking LLM Unlearning Objectives: A Gradient Perspective and Go Beyond. In *ICLR*, 2025.

Right to be Forgotten

"The data subject shall have the right to obtain from the controller the erasure of personal data concerning him or her without undue delay and the controller shall have the obligation to erase personal data …"

"A consumer shall have the right to request that a business delete any personal information about the consumer which the business has collected from the consumer …"

LLM Unlearning

to be unlearned

Bi-objective Goal

- **Unlearn:** removing model capability to generate targeted data $\mathcal{D}_{\mathrm{u}} = \{s_{\mathrm{u}}\}_{n_{\mathrm{u}}}$
- **Retain:** maintain performance on other non-targeted data $\mathcal{D}_{\mathrm{r}} = \{s_{\mathrm{r}}\}_{n_{\mathrm{r}}}$

not to be unlearned

Gradient Ascent (GA)-based Method

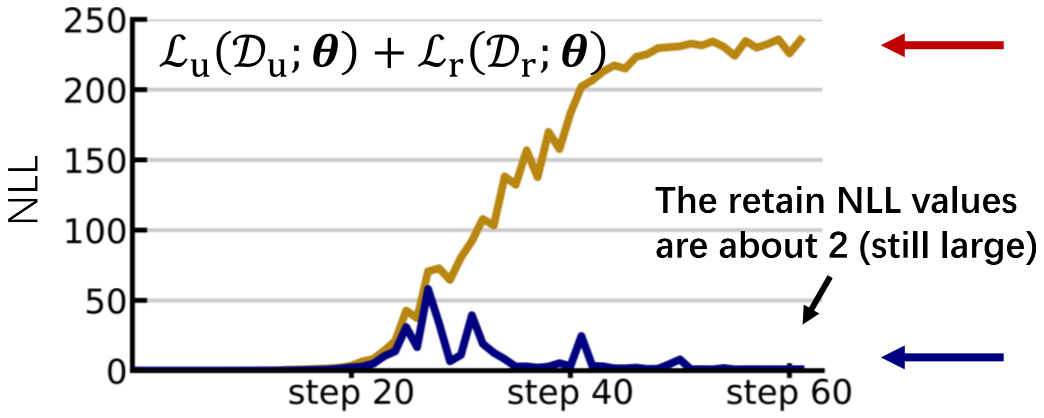
$$\min_{\boldsymbol{\theta}} \underbrace{\mathbb{E}_{\mathcal{D}_{\mathrm{u}}} \log P(s_{\mathrm{u}}; \boldsymbol{\theta})}_{\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})} + \underbrace{\mathbb{E}_{\mathcal{D}_{\mathrm{r}}} -\log P(s_{\mathrm{r}}; \boldsymbol{\theta})}_{\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}}; \boldsymbol{\theta})}$$

Unlearn Objective **Retain Objective**

Basic Assumption: If the negative log-likelihood is a proper objective for learning, then the log-likelihood should be appropriate for unlearning.

Impacts of GA

Negative log-likelihood (NLL) as the **metric** $\mathcal{R}$ to assess performance.



Performance regarding unlearning and retention.

$\mathcal{R}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$: **large unlearn NLL** indicates **strong** unlearning.

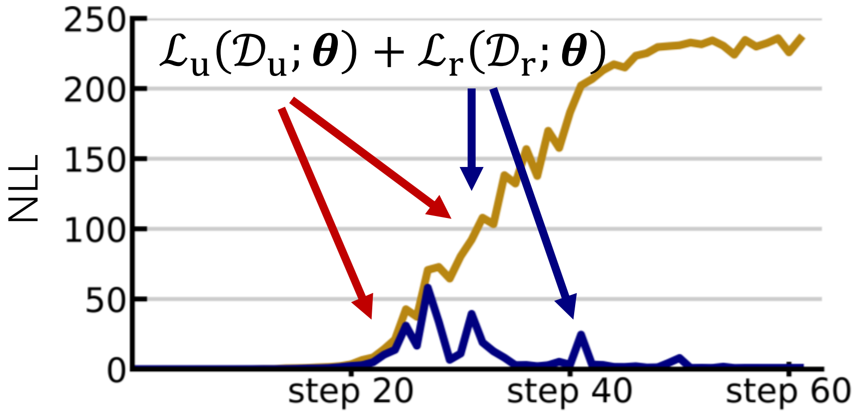
$\mathcal{R}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$: **large retain NLL** indicates **damage** in retention.

Observation 1. GA-based methods CAN achieve strong unlearning but CANNOT ensure reliable retention, thus **NOT meeting the dual-objective goal.**

Delve Deeper?

Performance metrics offer **limited** insights towards deeper understandings.

Limitation 1. We CANNOT **disentangle** the impacts of $\mathcal{L}_\mathrm{u}(\mathcal{D}_\mathrm{u};\boldsymbol{\theta})$ and $\mathcal{L}_\mathrm{r}(\mathcal{D}_\mathrm{r};\boldsymbol{\theta})$ on model performance.



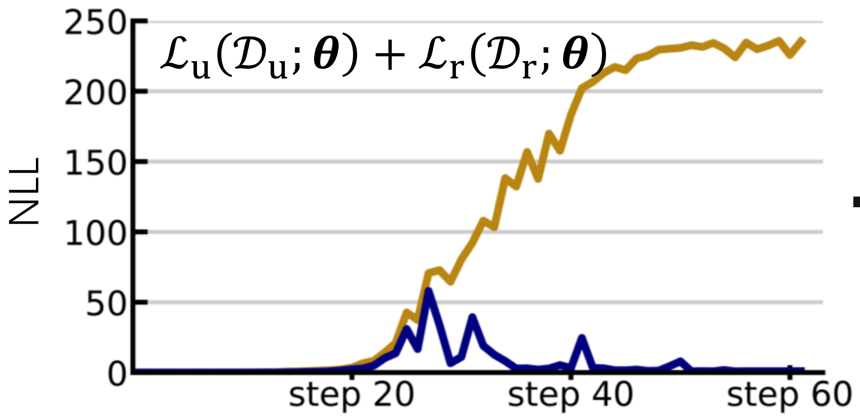
Using NLL to assess performance changes regarding unlearning and retention.

Both $\mathcal{L}_\mathrm{u}(\mathcal{D}_\mathrm{u};\boldsymbol{\theta})$ and $\mathcal{L}_\mathrm{r}(\mathcal{D}_\mathrm{r};\boldsymbol{\theta})$ have impacts on $\mathcal{R}(\mathcal{D}_\mathrm{u};\boldsymbol{\theta})$ and $\mathcal{R}(\mathcal{D}_\mathrm{r};\boldsymbol{\theta})$ in an **intertwined** manner.

Delve Deeper?

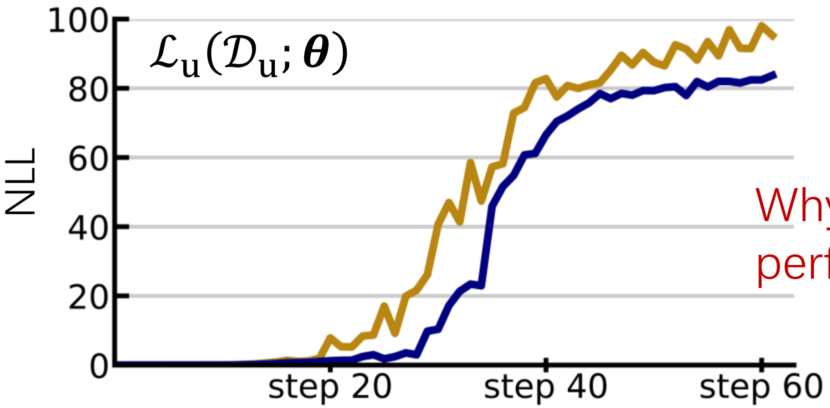
Performance metrics offer **limited** insights towards deeper understandings.

Limitation 2. Even disentangled, we CANNOT fully **understand the factors** that lead to the observed behaviors.



Unlearning with $\mathcal{L}_u(\mathcal{D}_u;\boldsymbol{\theta})+\mathcal{L}_r(\mathcal{D}_r;\boldsymbol{\theta})$.

→



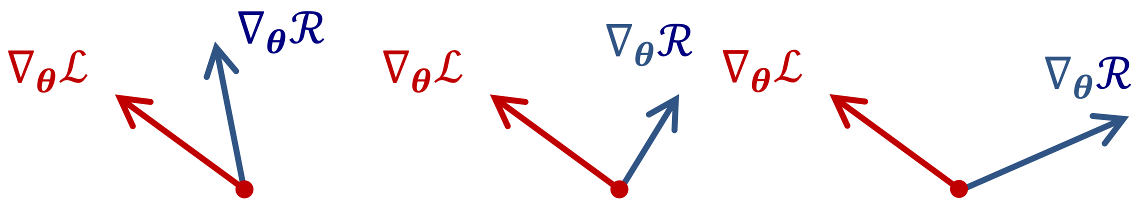
For illustration, we approximate the disentanglement by unlearning only with $\mathcal{L}_u(\mathcal{D}_u;\boldsymbol{\theta})$.

Why does the retention 🤔 performance drop so quick?

G-effect: A Gradient View

Studying the impacts of **unlearning methods** (e.g., GA) on **performance metrics** (e.g., NLL) from a gradient view.

$$e = \overbrace{\nabla_{\boldsymbol{\theta}}\mathcal{L}(\mathcal{D};\boldsymbol{\theta})}^{\text{gradients of }\textbf{objective}\text{ (unlearning method)}}{}^{\top}\underbrace{\nabla_{\boldsymbol{\theta}}\mathcal{R}(\mathcal{D};\boldsymbol{\theta})}_{\text{gradients of }\textbf{metric}}$$



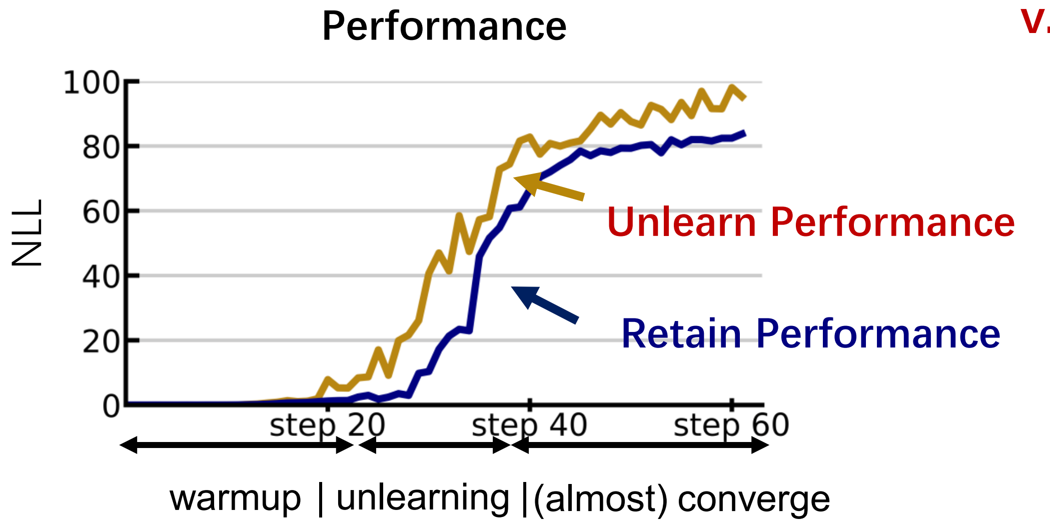
$\mathcal{L}$ benefits $\mathcal{R}$	mutual orthogonal	$\mathcal{L}$ damages $\mathcal{R}$
$\Downarrow$	$\Downarrow$	$\Downarrow$
positive e	**zero** e	**negative** e

- **Fulfill Goal 1** as the G-effect can be computed for $\mathcal{L}_{\mathrm{u}}(\mathcal{D}_{\mathrm{u}};\boldsymbol{\theta})$ and $\mathcal{L}_{\mathrm{r}}(\mathcal{D}_{\mathrm{r}};\boldsymbol{\theta})$ separately.
- **Fulfill Goal 2** as gradients provide more messages than merely CE performance.

G-effect: An Example

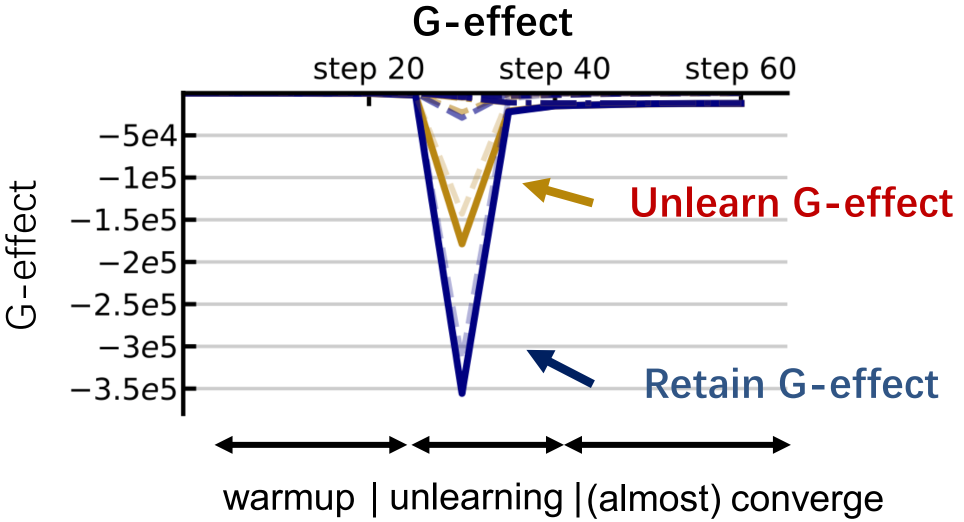
Retain G-effect: $e_{\mathrm{r}} = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})^{\top} \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D}_{\mathrm{r}}; \boldsymbol{\theta})$. A **positive** e_{r} is preferred to enhance retention.

Unlearn G-effect: $e_{\mathrm{u}} = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})^{\top} \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D}_{\mathrm{u}}; \boldsymbol{\theta})$. A **negative** e_{u} is preferred for strong unlearning.



Using NLL to assess performance.

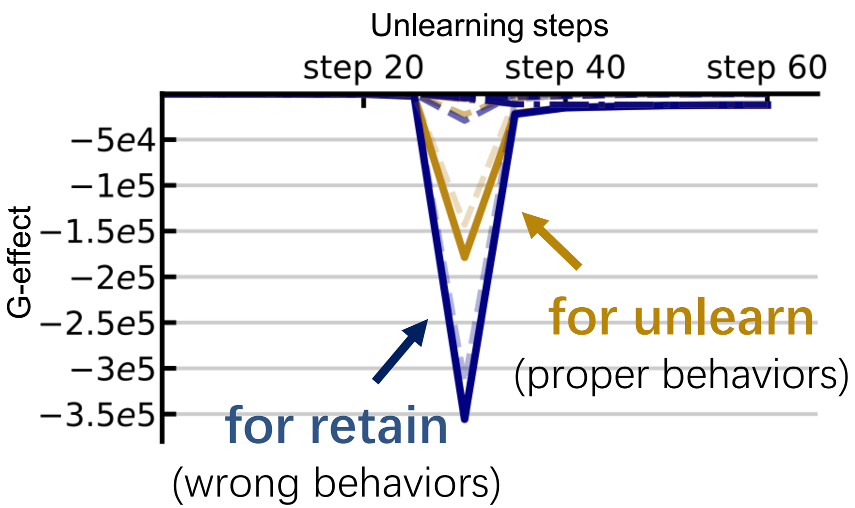
v.s.



Using G-effect to assess performance change.

Note. The G-effect quantifies the **rate of change** (increase/decrease) in performance, which can be calculated **separately** for retention and unlearning.

GA: Objective 1



The G-effects of GA.

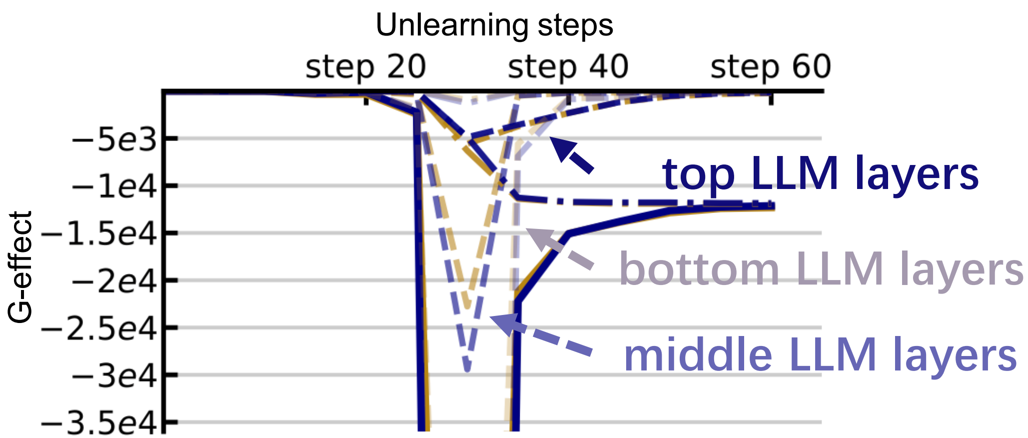
Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{1}{P(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta})} \nabla_{\boldsymbol{\theta}}}_{\text{inverse likelihood}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Observation 2. Excessive extent of removal incurs negative costs to retention.

Reason. The inverse likelihood wrongly focuses more on sufficiently unlearned tokens, leading to **over-unlearning** that negatively impacts model utility.

GA: Objective 1



The G-effects of GA (closer look).

Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{1}{P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)}}_{\text{inverse likelihood}} \nabla_{\boldsymbol{\theta}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Observation 3. Unlearning **affects on bottom layers** of LLMs more than others.

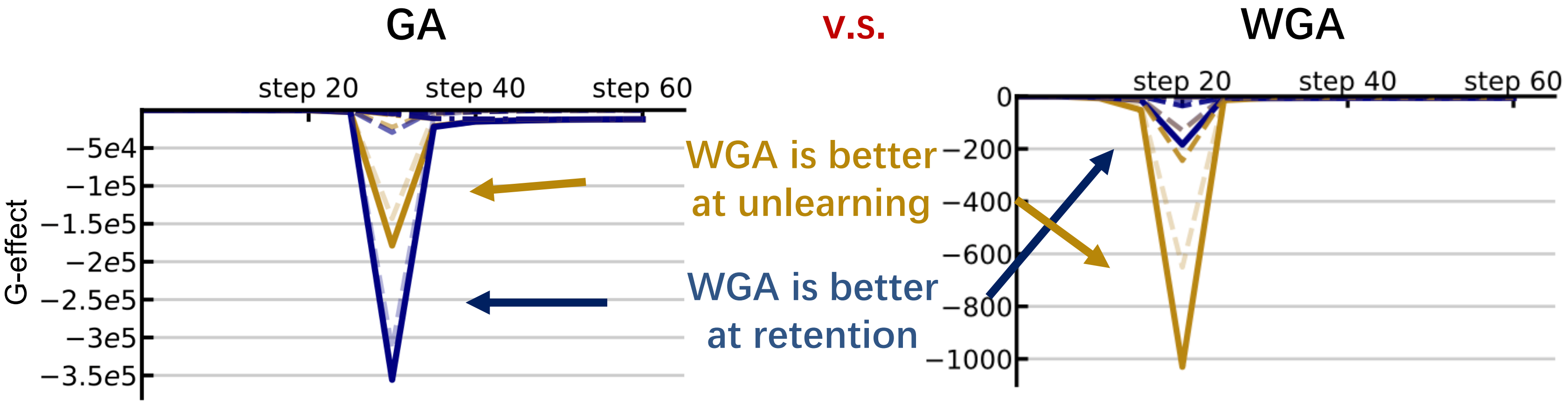
Reason. Large gradients will **accumulate** due to the chain rule, a general scenario holds for many other unlearning objectives.

WGA: Improvement 1

Motivation: Combating the inverse likelihood term via **loss reweighting.**

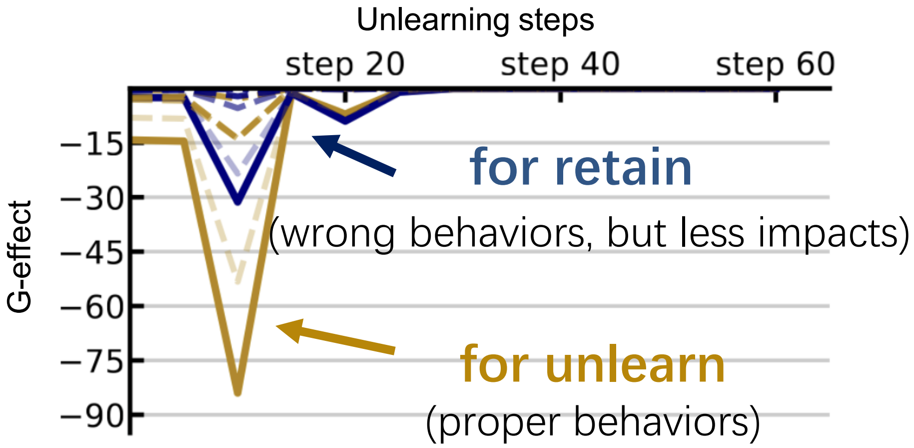
Original GA: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$ ➔ **Weighted GA:** $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)^{\alpha} \log P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$

Gradients: $\mathbb{E}_{s_\mathrm{u} \sim \mathcal{D}_\mathrm{u}} \sum_i \underbrace{P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)^{\alpha-1}}_{\text{counteract the inverse likelihood}} \nabla_{\boldsymbol{\theta}} P\left(s_\mathrm{u}^i | s_\mathrm{u}^{<i}; \boldsymbol{\theta}\right)$



Comparison of the G-effects between GA and WGA.

NPO: Objective 2



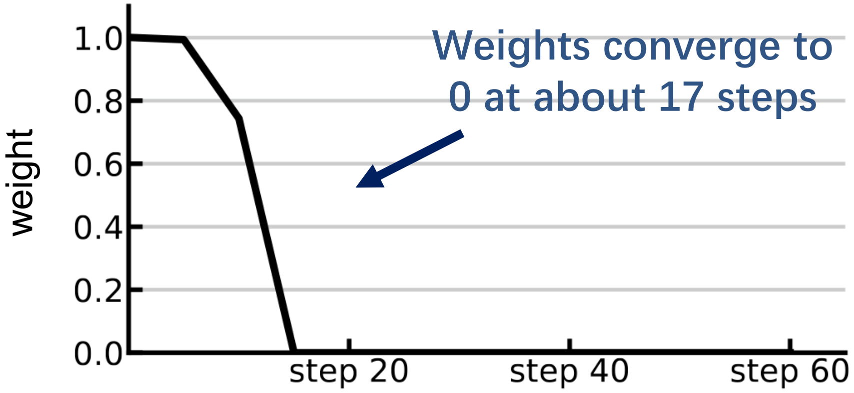
The G-effects of NPO.

Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \frac{1}{\beta} \log(1 + \left(\frac{p(s_\mathrm{u};\boldsymbol{\theta})}{p(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})}\right)^{\beta})$

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{2P(s_\mathrm{u};\boldsymbol{\theta})^{\beta}}{P(s_\mathrm{u};\boldsymbol{\theta})^{\beta}+P(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})^{\beta}}}_{w_\mathrm{npo}\text{ reweighting}} \nabla_{\boldsymbol{\theta}} \log P(s_\mathrm{u};\boldsymbol{\theta})$

Observation 4. NPO (Negative Preference Optimization) has **fewer negative impacts** on retention compared to GA.
Reason. The gradients of NPO are very similar to GA, yet further **reweighting** by w_npo, which mainly contributes to its improvements over GA.

NPO: Objective 2



The curve of w_{npo} *during unlearning.*

Objective: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \frac{1}{\beta} \log(1 + \left(\frac{p(s_\mathrm{u};\boldsymbol{\theta})}{p(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})}\right)^\beta)$

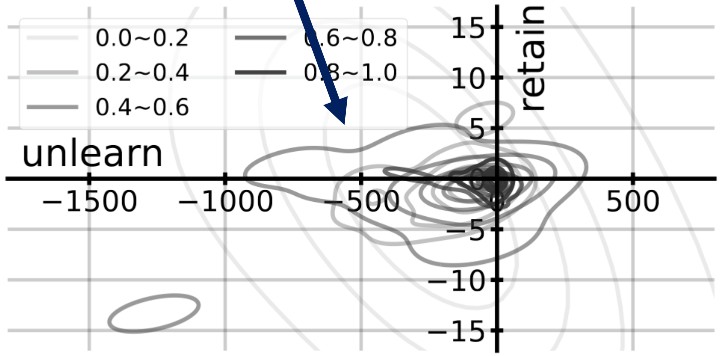
Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \underbrace{\frac{2P(s_\mathrm{u};\boldsymbol{\theta})^\beta}{P(s_\mathrm{u};\boldsymbol{\theta})^\beta + P(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})^\beta}}_{w_{\mathrm{npo}}\text{ reweighting}} \nabla_{\boldsymbol{\theta}} \log P(s_\mathrm{u};\boldsymbol{\theta})$

Observation 5. The NPO weight w_{npo} serves a role like **early stopping.**

Reason. w_{npo} approaches 0 when $P(s_\mathrm{u};\boldsymbol{\theta}) \to 0$.

NPO: Objective 2

Larger weights are assigned to those instances with larger retaining PG-effects.



The distributions of the point-wise G-effects across different range of w_{npo}.

Gradient: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \sum_i \frac{2P(s_\mathrm{u};\boldsymbol{\theta})^\beta}{P(s_\mathrm{u};\boldsymbol{\theta})^\beta + P(s_\mathrm{u};\boldsymbol{\theta}_\mathrm{o})^\beta} \nabla_{\boldsymbol{\theta}} \log P(s_\mathrm{u};\boldsymbol{\theta})$

G-effect: $\mathbb{E}_{\mathcal{D}_\mathrm{u}} \underbrace{w_\mathrm{npo}}_{\text{weights}} \underbrace{\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{s}_\mathrm{u};\boldsymbol{\theta})^\top \nabla_{\boldsymbol{\theta}} \mathcal{R}(\mathcal{D};\boldsymbol{\theta})}_{\text{point-wise G-effect (PG-effect)}}$

(The impacts of a particular data point on model performance.)

Observation 6. The NPO reweighting mechanism w_npo **prioritizes instances** that less damages retention.

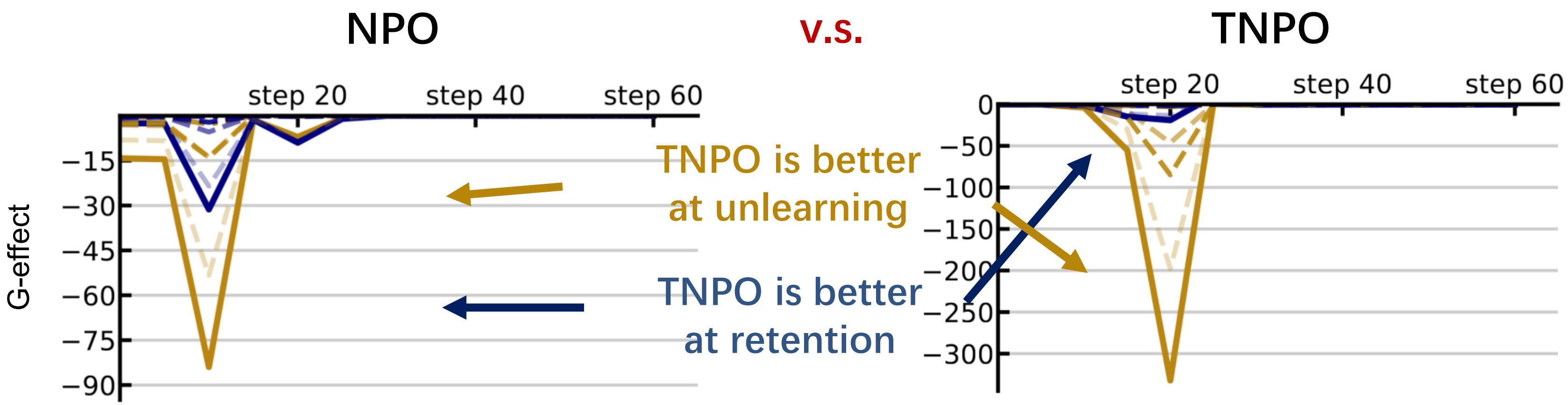
Reason. Data that have small impacts on **retention** also have small impacts on **unlearning.**

TNPO: Improvement 2

Motivation: Generalized the reweighting mechanism of NPO for tokens.

Token-wise NPO $\sum_i w_{\mathrm{tnpo}}^{i} \log P\left(s_{\mathrm{u}}^{i} | s_{\mathrm{u}}^{<i}; \boldsymbol{\theta}\right)$ with $w_{\mathrm{tnpo}}^{i} = \underbrace{\frac{2P\left(s_{\mathrm{u}}^{i} | s_{\mathrm{u}}^{<i}; \boldsymbol{\theta}\right)^{\alpha}}{P\left(s_{\mathrm{u}}^{i} | s_{\mathrm{u}}^{<i}; \boldsymbol{\theta}\right)^{\alpha} + P\left(s_{\mathrm{u}}^{i} | s_{\mathrm{u}}^{<i}; \boldsymbol{\theta}_{\mathrm{o}}\right)^{\alpha}}}$

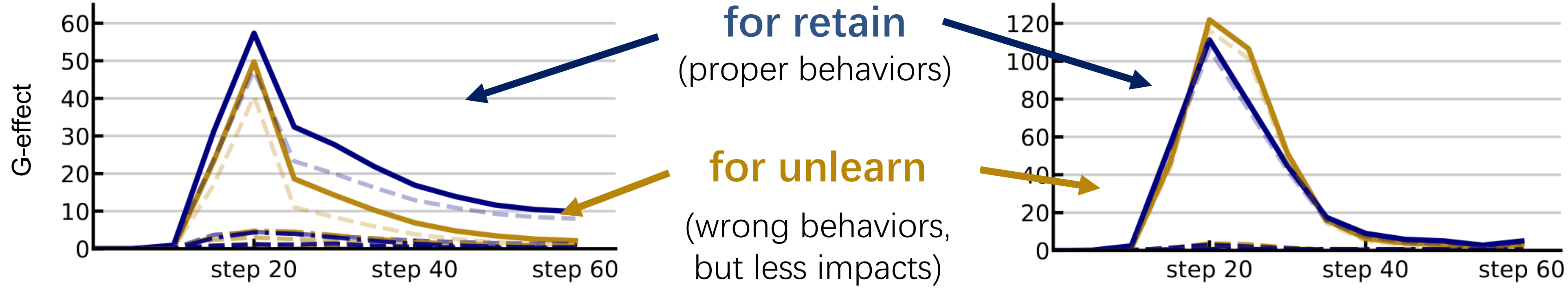
same reweighting scheme yet applied point-wise.



Comparison of the G-effects between NPO and TNPO.

Retain Objectives

NLL $\mathbb{E}_{\mathcal{D}_\mathrm{r}}[-\log P(s_\mathrm{r};\boldsymbol{\theta})]$ **v.s.** **KL** $\mathbb{E}_{\mathcal{D}_\mathrm{r}}\mathrm{KL}[P(s_\mathrm{r};\boldsymbol{\theta})||P(s_\mathrm{r};\boldsymbol{\theta}_\mathrm{o})]$



Comparison between two representative retain objectives.

Observation 7. **NLL** and **KL** are both effective for retention, while KL can lead to overall larger retain G-effect, thus preferred.

Note. The unlearn G-effect for the unlearning objective is much larger than for the retain objectives. Thus, we do not need to worry about the side effect on unlearning.

Empirical evaluations

LLM		Phi-1.5						Llama-2-7B					
		ES-exact		ES-perturb				ES-exact		ES-perturb			
setup	method	retain ↑	unlearn ↓	retain ↑	unlearn ↓	MU ↑	FQ ↑	retain ↑	unlearn ↓	retain ↑	unlearn ↓	MU ↑	FQ ↑
	before unlearning	0.44	0.59	0.21	0.16	0.52	-5.80	0.82	0.80	0.53	0.40	0.63	-7.59
	GA	0.11	0.05	0.08	0.08	0.37	-0.54	0.42	**0.05**	0.26	**0.04**	0.53	-0.54
	PO	**0.36**	0.84	**0.16**	0.36	**0.51**	-4.24	**0.75**	0.83	**0.47**	0.52	0.62	-5.80
	WGA	**0.36**	**0.03**	**0.18**	**0.02**	**0.51**	**-0.54**	**0.67**	0.08	0.38	0.06	**0.65**	**-0.08**
1%	NPO	0.27	0.09	0.11	0.07	0.48	-2.91	0.47	0.12	0.38	0.09	0.62	-1.32
	TNPO	0.33	**0.03**	0.12	**0.04**	0.49	**-0.08**	0.51	**0.03**	**0.43**	**0.03**	**0.64**	**-0.08**
	RMU	0.23	0.08	0.15	0.05	0.43	-0.54	0.23	0.08	0.15	0.05	0.52	-1.32
	before unlearning	0.44	0.56	0.21	0.23	0.52	-29.65	0.82	0.77	0.53	0.41	0.63	-32.13
	GA	0.00	**0.00**	0.00	**0.00**	0.00	-11.40	0.03	**0.00**	0.02	**0.00**	0.00	**-12.42**
	PO	**0.26**	0.79	**0.16**	0.49	**0.51**	-26.50	**0.55**	0.84	**0.36**	0.49	**0.64**	-28.84
	WGA	**0.29**	0.01	**0.16**	0.01	**0.51**	**-1.30**	0.47	0.00	**0.39**	0.00	**0.64**	-16.32
5%	NPO	0.08	0.12	0.08	0.06	0.38	-7.75	0.17	0.07	0.12	0.08	0.52	**-9.95**
	TNPO	0.16	0.01	0.08	0.00	0.46	-2.18	**0.50**	0.01	0.34	0.00	0.63	-32.13
	RMU	0.21	**0.00**	0.12	**0.00**	0.27	**-1.95**	0.12	**0.00**	0.12	**0.00**	0.58	-21.44
	before unlearning	0.44	0.47	0.21	0.18	0.52	-39.00	0.82	0.83	0.53	0.30	0.63	-44.45
	GA	0.00	**0.00**	0.00	**0.00**	0.00	-45.26	0.00	**0.00**	0.00	**0.00**	0.00	-20.86
	PO	**0.32**	0.73	0.14	0.26	0.50	-38.25	**0.55**	0.84	**0.37**	0.43	**0.62**	-39.76
	WGA	**0.34**	**0.00**	**0.16**	**0.00**	**0.51**	-9.06	**0.66**	0.02	**0.42**	**0.01**	0.62	-24.85
10%	NPO	0.08	0.09	0.07	0.07	0.38	-10.57	0.12	0.13	0.10	0.14	0.50	**-12.19**
	TNPO	0.20	0.01	0.09	0.01	**0.50**	**-7.66**	0.45	**0.01**	0.26	0.01	**0.63**	**-13.47**
	RMU	0.03	0.05	0.03	0.06	0.31	**-7.00**	0.25	0.01	0.20	0.01	0.59	-16.72

Comparison between unlearning objective on TOFU with KL regularization.

Observation 8. Larger unlearning datasets and smaller model sizes make it more challenging to unlearn.

Observation 9. GA-based works (GA & TNPO) are superior to other lines of works like PO or RMU.

Observation 10. Instance-wise reweighting is promising for unlearning efficacy.

Take Home Messages

General knowledge within **shallow layers undergoes substantial alterations** over deeper layers during unlearning.

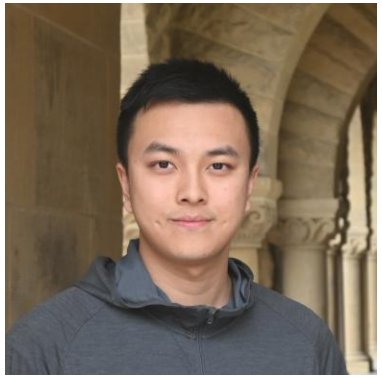
Although conceptually existing, **current objectives all fail** to retain the overall performance when conducting unlearning.

Prioritizing some tokens is effective for unlearning. However, there still exists a large space to further refine weighting mechanisms.

With **excessive unlearning**, the deterioration in common model responses can outweigh improvements in unlearning.

Part III: Reasoning

Can Language Models Perform Robust Reasoning in Chain-of-thought Prompting with Noisy Rationales?



Zhanke Zhou

Jianing Zhu

Input with Noisy Questions

Question-1 (Q1): In base-9, what is 86+57? **We know 6+6=12 and 3+7=10 in base 10.**

Rationale-1 (R1): In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 154.

Answer-1 (A1): 154.

…**Q2, R2, A2, Q3, R3, A3**…

Test Question: In base-9, what is 62+58? **We know 6+6=12 and 3+7=10 in base 10.**

Input with Noisy Rationales

Question-1 (Q1): In base-9, what is 86+57?

Rationale-1 (R1): In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. **13 + 8 = 21**. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. **5 + 9 = 14.** A leading digit is 1. So the answer is 154.

Answer-1 (A1): 154.

…**Q2, R2, A2, Q3, R3, A3** …

Test Question: In base-9, what is 62+58?

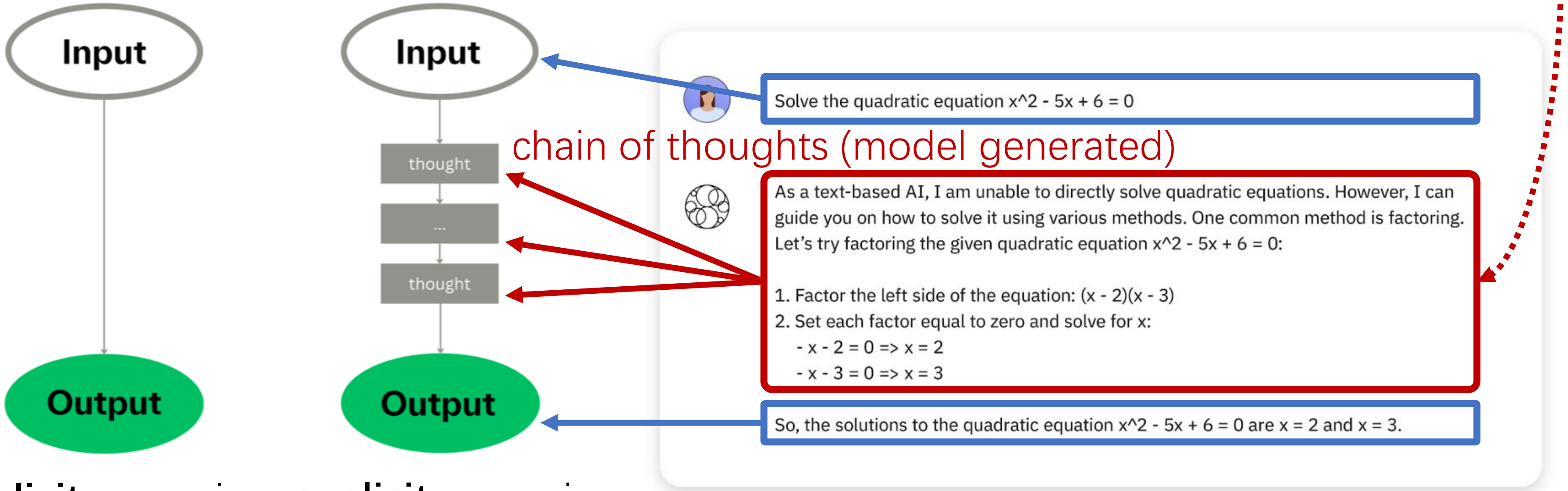
Zhanke Zhou, Rong Tao, **Jianing Zhu**, Yiwen Luo, Zengmao Wang, **Bo Han.**
Can Language Models Perform Robust Reasoning in Chain-of-thought Prompting with Noisy Rationales? In *NeurIPS*, 2024

Background

Reasoning is the pathway to achieve powerful intelligence.

- Decompose a complex problem into feasible steps.
- Combine knowledge pieces into new knowledge.

Generating **chain of thoughts (CoT)** is the key of several reasoning models.



implicit reasoning **explicit** reasoning

deepseek R1

OpenAI o3

Chain of Thoughts (CoT)

In-context learning (ICL) is widely used.

- ICL enable LLMs to **learn from a few examples** without fine-tuning.

Zero-shot Input

Question: In base-9, what is 62+58?

Input with three examples

Question-1: In base-9, what is 86+57? **Answer-1:** 154.
Question-2: In base-9, what is 63+34? **Answer-2:** 107.
Question-3: In base-9, what is 31+58? **Answer-3:** 100.
Question: In base-9, what is 62+58?

Chain of thoughts (CoT) prompting can elicit the reasoning capabilities of LLMs.

- Beyond examples, CoT includes **rationales**, i.e., sequential reasoning thoughts to solve a question.

Input: CoT prompting with rationales

Question-1: In base-9, what is 86+57?
Rationale-1: In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 154.
Answer-1: 154.
…**Q2, R2, A2, Q3, R3, A3** …
Question : In base-9, what is 62+58?

more powerful 👍

New Challenge in LLM Reasoning

Existing work generally assumes that CoT contains **clean rationales.**

But, what if CoT contains **noisy rationales?** 🤔

- noisy rationales include irrelevant or inaccurate thoughts.

The irrelevant **base-10 information** is included in rationale.

Input: CoT prompting with clean rationales

Question-1: In base-9, what is 86+57?
Rationale-1: In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 154.
Answer-1: 154.
··· **Q2, R2, A2, Q3, R3, A3** ···
Question : In base-9, what is 62+58?

Input: CoT prompting with noisy rationales

Question-1: In base-9, what is 86+57?
Rationale-1: In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. **13 + 8 = 21**. Since we're in base-9, that exceeds the maximum value of 8 for a single digit.13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. **5 + 9 = 14.** A leading digit is 1. So the answer is 154.
Answer-1: 154.
··· **Q2, R2, A2, Q3, R3, A3** ···
Question: In base-9, what is 62+58?

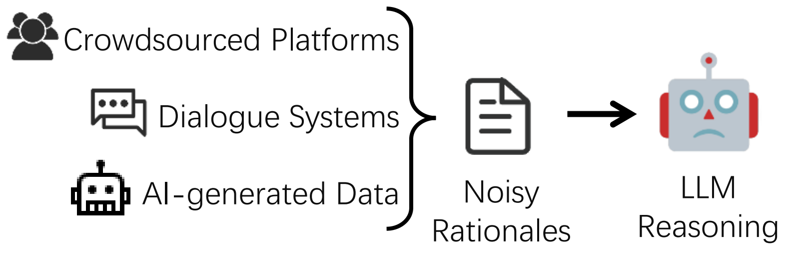
While the test question asks about **base-9 calculation.**

New Challenge in LLM Reasoning

Noisy rationales originate from diverse sources.

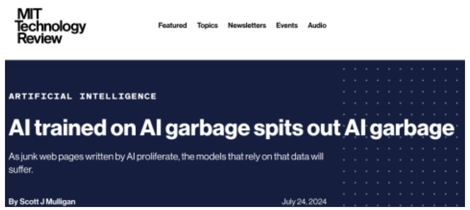
- Such as crowdsourced platforms, dialogue systems, and AI-generated data.

Sources of noisy rationales



Recent news on noisy data

Recent investigations on noisy data



Garbage in, Garbage out

However, the **robustness** of LLMs against noisy rationales is still **unknown.**

- **A new dataset** is needed to conduct **a systematic evaluation** of current LLMs.
- To verify the corresponding **countermeasures** against noisy rationales.

Noisy Rationales Benchmark (NoRa)

- We construct a new benchmark to evaluate the **robustness** against noisy rationales.
- NoRa contains **26,391** questions, covering **3** tasks: **math, symbolic, and commonsense.**

Task	Irrelevant Thoughts	Inaccurate Thoughts
NoRa-Math	In base-9, digits run from 0 to 8. We have 3 + 2 = 5 in base-10. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 5 mod 9 = 5, so the digit is 5 and the carry is 0. There are five oceans on Earth: the Atlantic, Pacific, Indian, Arctic, and Southern. We have 8 + 6 + 0 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 155. Answer: 155	In base-9, digits run from 0 to 8. We have 3 + 2 = 5 in base-10. 5 + 4 = 9. Since we're in base-9, that doesn't exceed the maximum value of 8 for a single digit. 5 mod 9 = 5, so the digit is 5 and the carry is 0. 5 + 9 = 14. We have 8 + 6 + 0 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit 1. So the answer is 155. Answer: 155
NoRa-Symbolic	... "turn around right" means the agent needs to turn right, and repeat this action sequence four times to complete a 360-degree loop. Many GPS navigation systems will issue a 'turn around' command if the driver deviates from the planned route. So, in action sequence is I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT. ...	... "turn around right" means the agent needs to turn right, and repeat this action sequence four times to complete a 360-degree loop. Turn opposite is I_TURN_RIGHT I_TURN_LEFT. So, in action sequence is I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT I_TURN_RIGHT. ...
NoRa-Com.	The relations path are son, sister, uncle, which means Francisco is David's son's sister's uncle. For son's sister, we have son's sister is daughter. So the relations path are reduced to daughter, uncle. In genetics, mitochondrial DNA is always inherited from the mother, making the mother-daughter genetic link unique. For daughter's uncle, we have daughter's uncle is brother. So the relations path are reduced to brother. Therefore, the answer is brother. Answer:brother	The relations path are son, sister, uncle, which means Francisco is David's son's sister's uncle. For son's sister, we have son's sister is daughter. So the relations path are reduced to daughter, uncle. For daughter's uncle, we have daughter's uncle is brother. We have brother' sister is brother. So the relations path are reduced to brother. Therefore, the answer is brother. Answer:brother

clean thoughts (in black)

noisy thoughts (in red)

Table 1: Noisy rationales (consisting noisy thoughts) sampled from the NoRa dataset. Full examples of NoRa are in Appendix C.6, and real-world examples of noisy rationales are in Appendix C.3.

Code & Benchmark: https://github.com/tmlr-group/NoisyRationales

Noisy Rationales Benchmark (NoRa)

Definitions

- **Irrelevant thoughts** are irrelevant to the given context.
  - E.g., discussing the genetic overlap of siblings when reasoning the family roles.
- **Inaccurate thoughts** are factual errors in the given context.
  - E.g., "5+5=10" is wrong in base-9 calculation.

Benchmark construction

- Generating noisy rationales by **inserting irrelevant or inaccurate thoughts.**
- **Guarantee the overall correctness** without modifying the question or answer.
- Control **noise ratios** (noisy thoughts / clean thoughts) with values 0.3,0.5,0.8.

(easy medium hard)

Empirical Evaluations with NoRa

Grand observation: The base LLM (GPT-3.5) with all the existing methods is severely affected by noisy rationales.

- Up to **25.3%** acc decrease with irrelevant noise.
- Up to **54.0%** acc decrease with inaccurate noise (compared acc with clean rationales).

Observation 1:
Self-correction methods (ISC, SP) perform **poorly** on most tasks with noisy rationales.

Observation 2:
Self-consistency methods (SM, SD, SC) can improve robustness **without** true denoising.

Task	Method $\mathcal{M}$	Acc($\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{clean}}$)	Acc($\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{irrelevant}}$) Easy	Medium	Hard	Avg.	Acc($\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\text{inaccurate}}$) Easy	Medium	Hard	Avg.
Math Base-9	Base	46.4	39.3	30.3	26.6	32.1	23.2	10.1	6.0	13.1
	w/ ISC [29]	24.3	17.7	14.7	12.7	15.0	18.4	13.7	12.3	14.8
	w/ SP [89]	26.2	25.5	25.5	21.9	24.3	20.0	18.4	**14.3**	17.6
	w/ SM [62]	37.4	30.0	22.7	16.5	23.1	24.7	**19.2**	12.4	**18.8**
	w/ SD [102]	47.9	37.2	25.4	24.7	29.1	29.3	12.5	8.7	16.8
	w/ SC [83]	**61.5**	**51.1**	**39.0**	**36.2**	**42.1**	**32.7**	15.3	7.5	18.5
Math Base-11	Base	23.9	19.1	13.6	10.7	14.5	14.0	6.7	3.6	8.1
	w/ ISC [29]	11.2	8.3	7.8	6.0	7.4	6.5	5.2	4.7	5.5
	w/ SP [89]	20.7	17.5	**16.7**	14.0	16.0	14.1	**10.7**	**10.8**	**11.9**
	w/ SM [62]	16.3	12.0	6.0	5.7	7.9	12.0	9.3	7.7	9.7
	w/ SD [102]	17.9	12.3	12.0	13.3	12.5	17.0	8.7	5.3	10.3
	w/ SC [83]	**33.7**	**25.3**	16.3	**15.0**	**18.9**	**19.7**	9.3	3.3	10.8
Symbolic Equal	Base	32.7	28.1	25.1	23.0	25.4	29.1	26.1	22.7	26.0
	w/ ISC [29]	23.9	20.0	16.3	15.5	17.3	19.2	18.3	18.1	18.5
	w/ SP [89]	23.2	23.0	22.6	22.7	22.8	23.7	22.5	23.5	23.2
	w/ SM [62]	25.0	20.7	19.7	16.7	19.0	21.0	20.3	20.0	20.4
	w/ SD [102]	9.9	10.1	10.9	10.3	10.4	10.1	10.9	10.4	10.5
	w/ SC [83]	**35.3**	**31.0**	**28.3**	**27.0**	**28.8**	**33.3**	**30.7**	**26.0**	**30.0**
Symbolic Longer	Base	9.2	6.3	7.2	6.0	6.5	7.0	6.8	6.0	6.6
	w/ ISC [29]	4.9	4.6	2.7	3.7	3.7	3.4	4.3	3.3	3.7
	w/ SP [89]	5.1	4.3	4.1	3.9	4.1	4.9	4.0	4.5	4.5
	w/ SM [62]	1.7	0.7	0.7	1.3	1.0	1.3	0.7	0.3	0.8
	w/ SD [102]	0.1	0.1	0.1	0.2	0.1	0.1	0.3	0.0	0.1
	w/ SC [83]	**13.0**	**7.7**	**9.0**	**6.3**	**7.7**	**8.0**	**8.0**	**8.7**	**8.2**
Commonsense	Base	45.7	44.3	42.3	41.4	42.7	36.7	33.4	28.3	32.8
	w/ ISC [29]	21.8	24.3	22.5	21.4	22.7	23.3	26.5	24.0	24.6
	w/ SP [89]	47.9	48.2	46.7	48.1	47.7	49.6	46.6	46.5	47.6
	w/ SM [62]	53.3	50.3	50.0	46.7	49.0	47.7	49.0	49.3	48.7
	w/ SD [102]	**54.0**	**58.3**	**57.3**	**57.7**	**57.8**	**57.0**	**58.3**	**53.7**	**56.3**
	w/ SC [83]	52.0	46.3	45.0	44.7	45.3	44.7	44.7	38.0	42.5

Table 3: Reasoning accuracy on NoRa dataset with 3-shot prompting examples with clean, irrelevant, or inaccurate rationales. The **boldface** numbers mean the best results, while the underlines numbers indicate the second-best results. Note the referenced results of Base model are highlighted in gray.

Baseline methods:
- Intrinsic Self-correction (ISC)
- Self-polish (SP) SmoothLLM (SM)
- Self-denoise (SD) Self-consistency (SC)

Experiments with GPT-4 are in Appendix F

Empirical Evaluations with NoRa

Task	Setting	Temperature 0	0.3	0.5	0.7	1
Base-9	clean	**61.0**	60.9	57.5	55.3	46.4
	ina. easy	**29.7**	28.0	27.2	26.6	21.7
	ina. hard	5.0	5.1	**5.5**	4.6	5.0
Base-11	clean	**34.0**	33.8	31.6	29.8	23.9
	irr. easy	21.7	23.1	21.3	**23.3**	19.1
	irr. hard	17.0	**17.5**	15.5	14.1	10.7
Sym.(E)	clean	34.2	**35.8**	35.7	34.6	32.7
	irr. easy	28.6	**31.5**	29.8	29.1	28.1
	irr. hard	**27.0**	26.1	26.2	24.0	23.0
Sym.(L)	clean	6.3	8.3	8.9	8.9	**9.3**
	ina. easy	5.0	7.3	**8.6**	8.3	7.0
	ina. hard	4.0	6.1	**6.3**	6.2	6.0

Table 4: Comparing performances of the base model with different temperatures. Sym.(E)/(L) are symbolic tasks.

Task	Setting	#Prompting Examples 1	2	3	4	5
Base-9	clean	24.8	38.3	46.4	**50.8**	50.5
	ina.-easy	17.5	22.2	23.2	25.4	**25.6**
	ina.-hard	**11.3**	6.3	6.0	5.7	5.7
Base-11	clean	11.8	20.4	23.9	29.9	**32.1**
	irr. easy	8.9	15.9	19.1	21.7	**26.3**
	irr. hard	7.7	10.0	10.7	15.2	**16.1**
Sym.(E)	clean	18.0	26.5	32.7	**39.8**	—
	ina.-easy	17.3	23.6	29.1	**34.7**	—
	ina.-hard	15.0	21.0	**22.7**	—	—
Sym.(L)	clean	2.7	7.7	9.3	11.3	**12.2**
	irr. easy	2.3	5.4	7.0	8.8	**8.9**
	irr. hard	1.9	4.0	6.0	**6.3**	—

Table 5: Comparing performances of the base model with a varying number of examples ("—" denotes over token limit).

Model	Task	Setting 0-shot	clean	irr.	ina.
GPT3.5	Base-9	7.2	**46.4**	30.3	10.1
	Sym.(E)	8.8	**32.7**	25.1	26.1
	Com.	40.0	**45.7**	42.3	33.4
Gemini	Base-9	12.7	**88.0**	72.3	21.2
	Sym.(E)	9.3	**44.5**	38.9	36.7
	Com.	42.9	**55.6**	53.2	33.5
Llama2	Base-9	1.7	**4.9**	2.9	2.7
	Sym.(E)	4.7	**10.1**	8.7	9.1
	Com.	35.0	**42.3**	41.9	40.2
Mixtral	Base-9	3.9	**27.5**	16.3	3.7
	Sym.(E)	8.3	**19.3**	17.9	15.1
	Com.	24.2	**37.5**	34.9	31.1

Table 6: Comparing LLMs with 0-shot, 3-shot clean, and 3-shot medium irrelevant (irr.) / inaccurate (ina.) rationales.

Observation 3:
Adjusting model temperature can improve reasoning under noisy rationales.

Observation 4:
Prompting with more noisy examples boosts reasoning accuracy on most tasks.

Observation 5:
Different LLMs are **generally vulnerable** to noisy rationales.

Empirical Evaluations with NoRa

We further explore the mapping among questions, rationales, and answers.

Specifically, given the 3-shot examples $\{(\mathrm{x}_1, \mathcal{T}_1, y_1), (\mathrm{x}_2, \mathcal{T}_2, y_2), (\mathrm{x}_3, \mathcal{T}_3, y_3)\}$, we test three configurations:

- shuffle the order of **questions:** $\{(\mathrm{x}_2, \mathcal{T}_1, y_1), (\mathrm{x}_3, \mathcal{T}_2, y_2), (\mathrm{x}_1, \mathcal{T}_3, y_3)\}$;
- shuffle the order of **rationales:** $\{(\mathrm{x}_1, \mathcal{T}_3, y_1), (\mathrm{x}_2, \mathcal{T}_1, y_2), (\mathrm{x}_3, \mathcal{T}_2, y_3)\}$;
- shuffle the order of **answers:** $\{(\mathrm{x}_1, \mathcal{T}_1, y_3), (\mathrm{x}_2, \mathcal{T}_2, y_1), (\mathrm{x}_3, \mathcal{T}_3, y_2)\}$.

Task	Zero-shot	Few-shot (No Shuffle)	Shuffle Questions x_i	Shuffle Rationales $\mathcal{T}_i$	Shuffle Answers y_i
Math Base-9	7.2	**46.4**	<u>45.5</u> (0.9%↓)	34.5 (11.9%↓)	35.7 (10.7%↓)
Math Base-11	5.5	<u>23.9</u>	**24.8** (0.9%↑)	21.6 (2.3%↓)	21.1 (11.7%↓)
Symbolic Equal	8.8	<u>32.7</u>	<u>32.7</u> (0.0%↓)	**32.8** (0.1%↑)	32.3 (0.4%↓)
Symbolic Longer	0.0	**9.2**	<u>7.0</u> (2.2%↓)	6.2 (3.0%↓)	6.3 (2.9%↓)
Commonsense	40.0	**45.7**	38.7 (7.0%↓)	39.7 (6.0%↓)	<u>39.8</u> (5.9%↓)

Table 7: Performance (in accuracy%) on NoRa dataset under different few-shot shuffle configurations.

Observation 6: Shuffling the mappings of prompting examples **degenerates** the reasoning but still performs **better** than without prompting.
Besides, LLMs are **less vulnerable** to shuffled mappings than noisy rationales.

Motivation

Current LLMs **cannot** denoise well with their **intrinsic denoising ability.**

- Even enhanced with self-correction[1] / self-consistency[2] methods.

External supervision is necessary for enhancement.

- This supervision should be sufficient for denoising and accessible in practice.

A clean CoT demonstration can be the minimal requirement for denoising-purpose prompting.

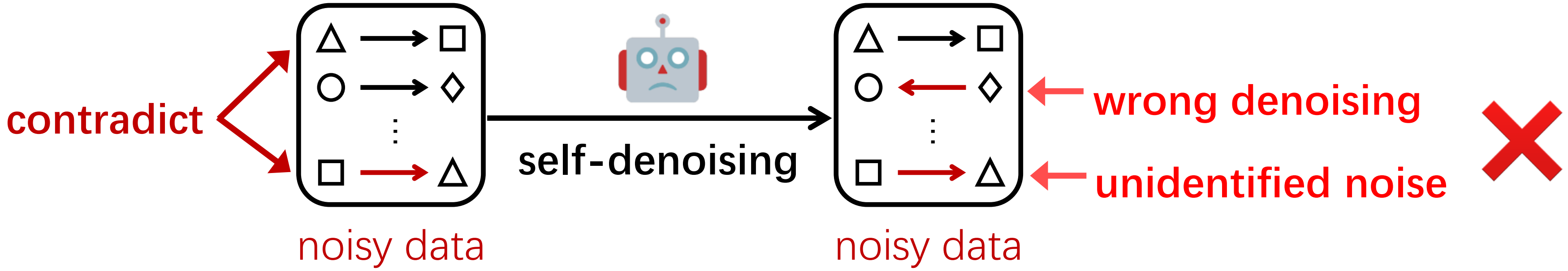
- This is more practical than existing methods requiring external supervision.

[1] J. Huang et al. Large Language Models Cannot Self-Correct Reasoning Yet. In *ICLR*, 2024.
[2] X. Wang et al. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

Motivation

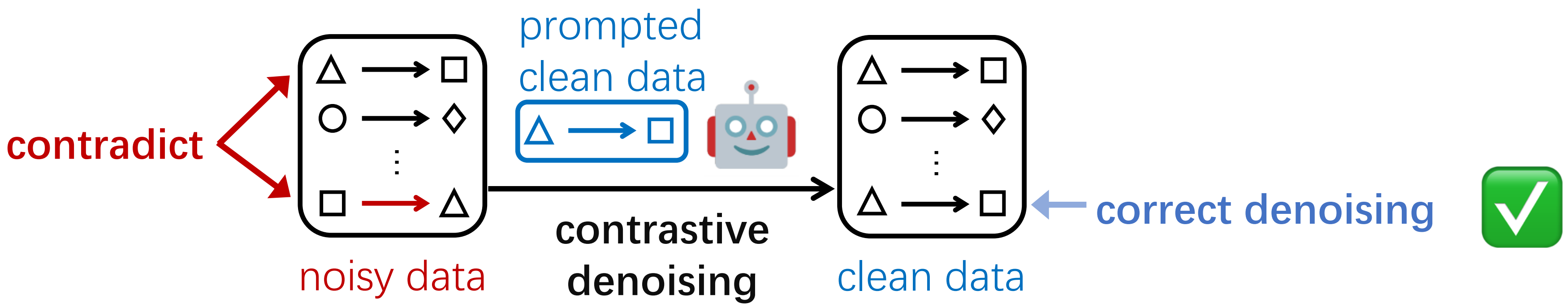
Self-denoising:

- It is **hard** for LLMs to denoise noisy data **without guidance.**



Contrastive denoising:

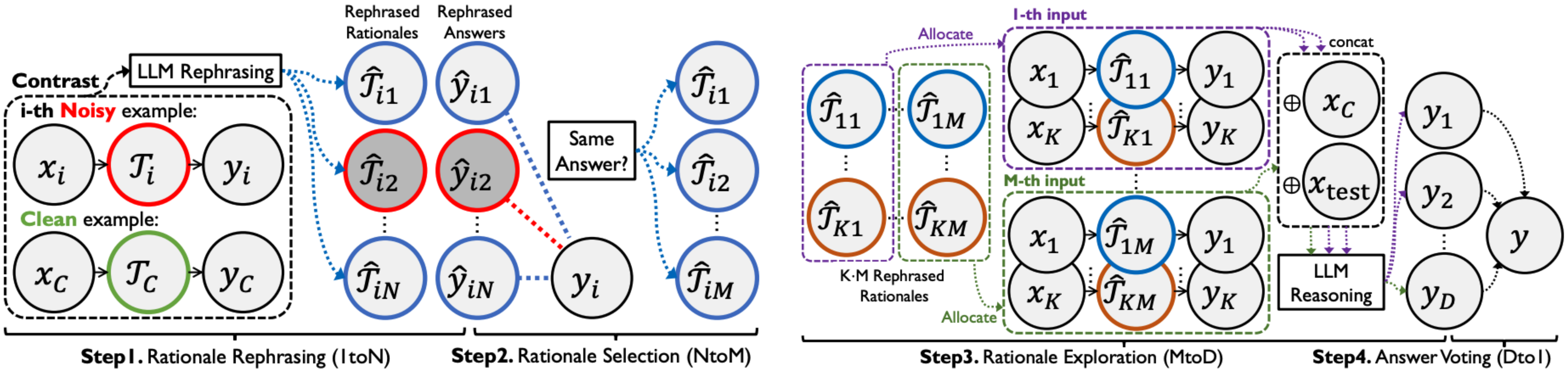
- It is **easier** for LLMs to denoise by **contrasting noisy and clean data.**



Method

Contrastive Denoising with Noisy Chain-of-thought (CD-CoT).

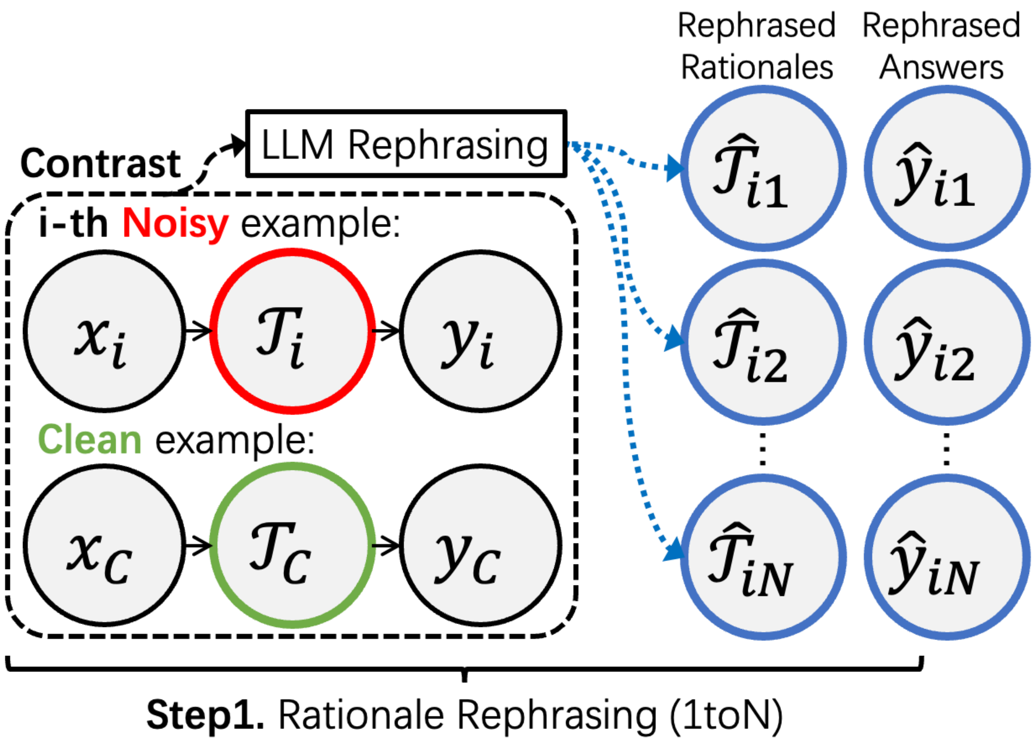
- **Rephrasing and selecting rationales** in the input space to conduct explicit denoising (steps 1&2).
- **Exploring diverse reasoning paths and voting on answers** in the output space (steps 3&4).



Note. **Steps 1 & 2** contribute more than Steps 3 & 4 for the explicit data denoising.

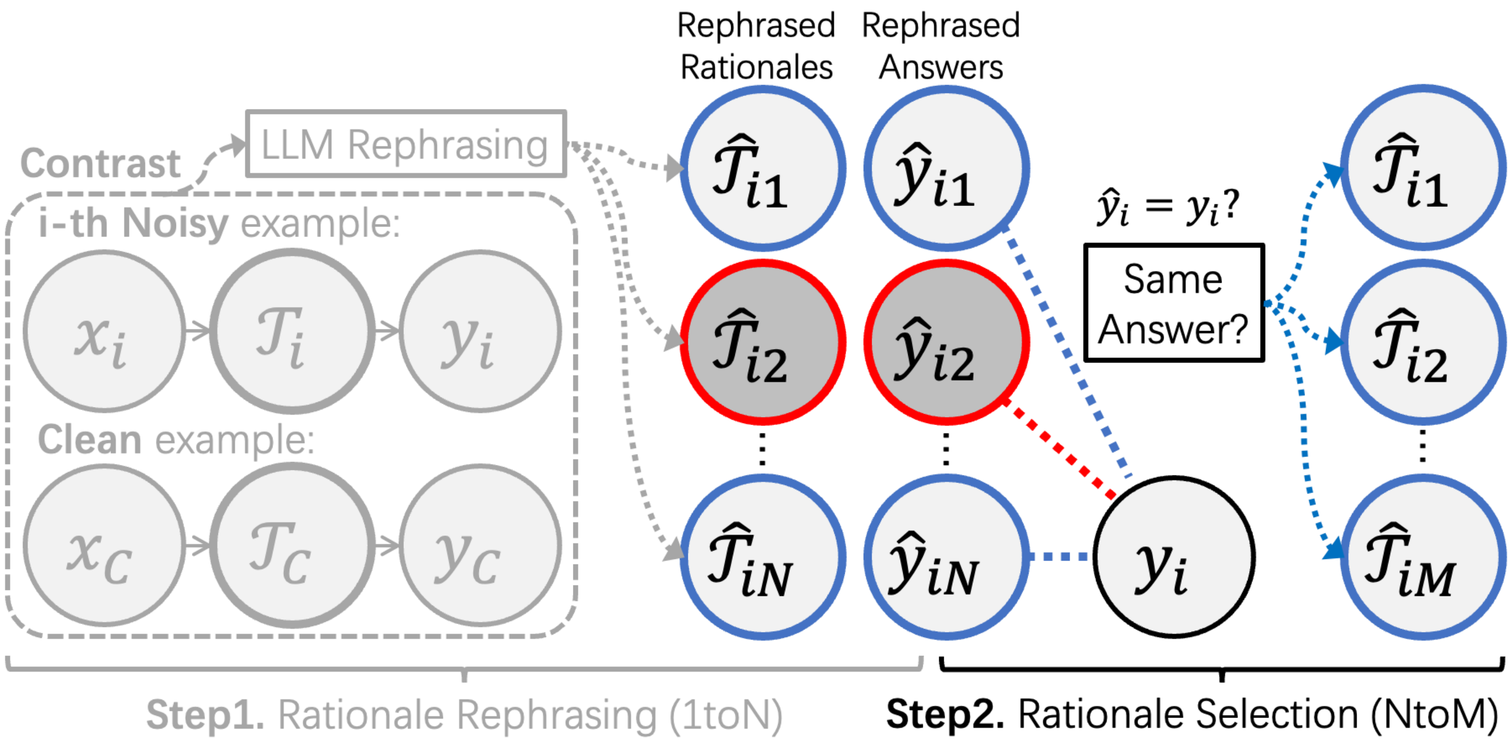
CD-CoT

- **Step-1:** rephrase the noisy rationales via contrastive denoising.
- Step-2: select rephrased examples with the same answers (unchanged).



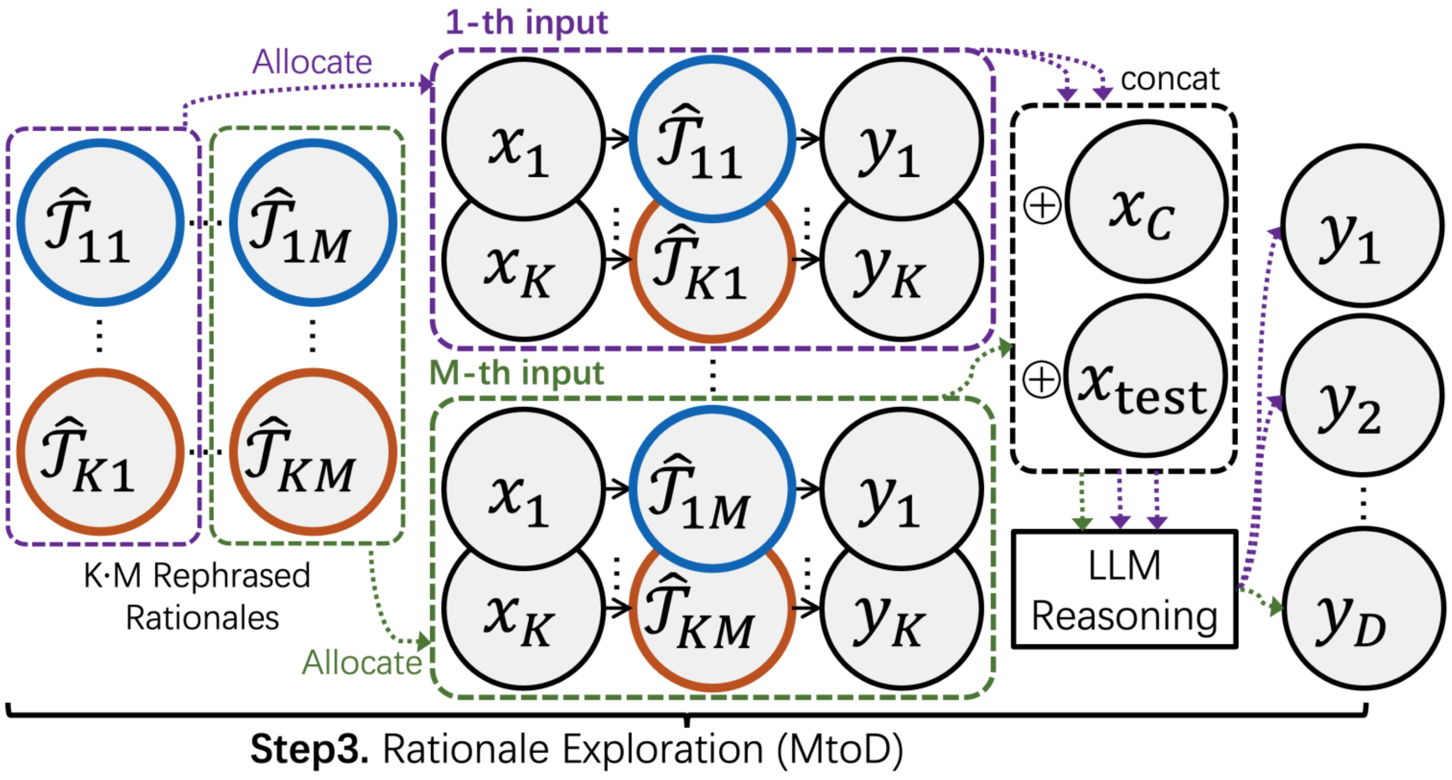
CD-CoT

- Step-1: rephrase the noisy rationales via contrastive denoising.
- **Step-2:** select rephrased examples with the same answers (unchanged).



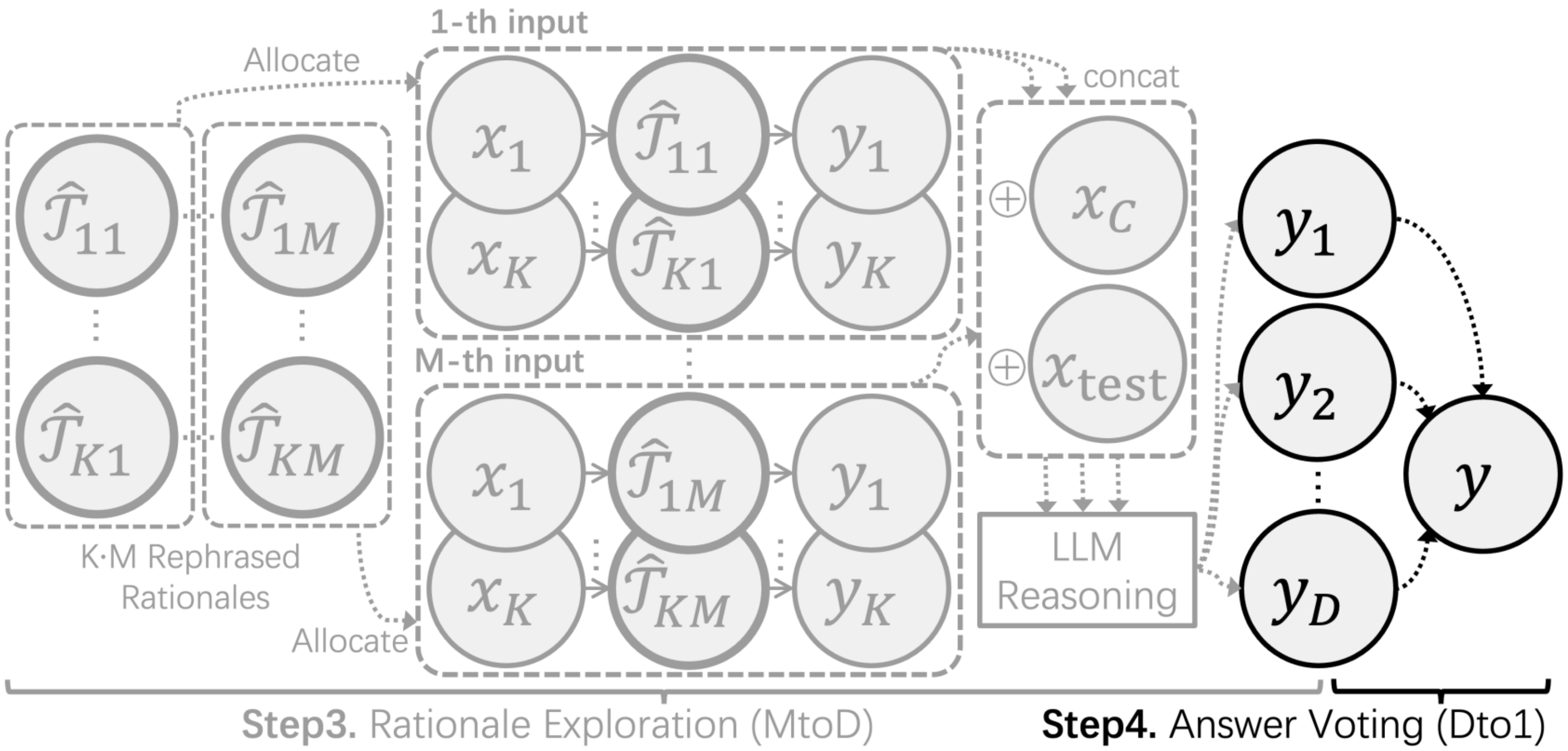
CD-CoT

- **Step-3:** fully utilize the rephrased examples for deliberate reasoning.
- Step-4: vote all the answers equally to get the final answer.

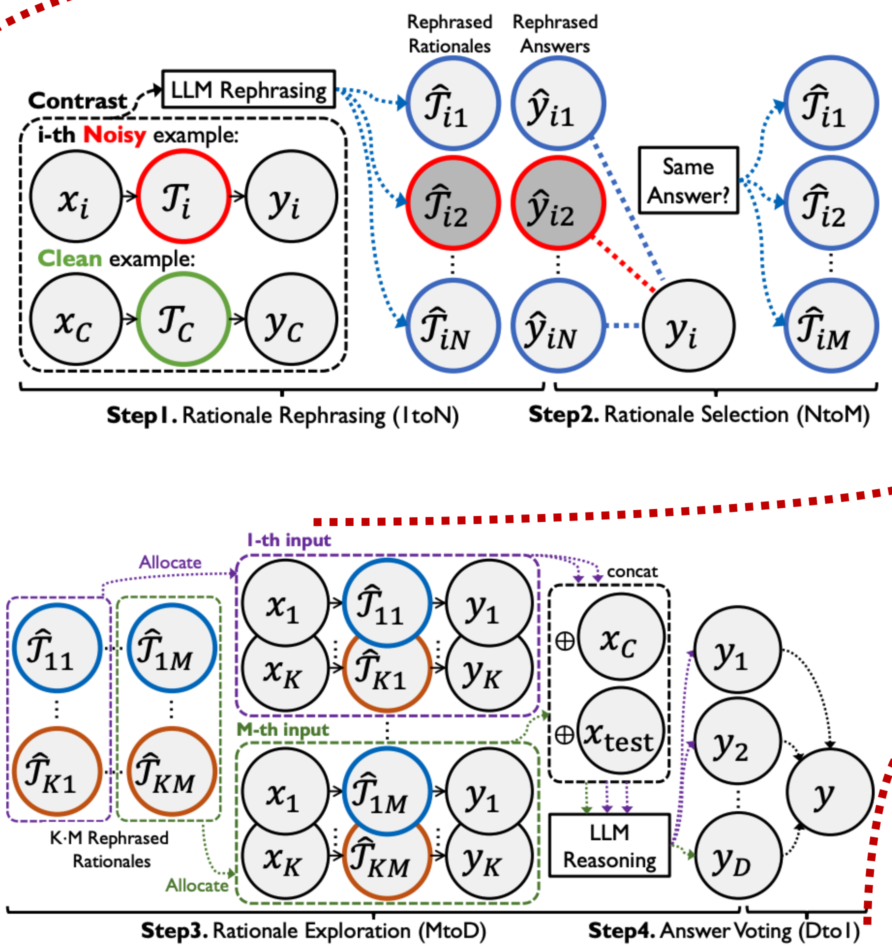


CD-CoT

- Step-3: fully utilize the rephrased examples for deliberate reasoning.
- **Step-4:** vote all the answers equally to get the final answer.



Method



Algorithm 1 CD-CoT: Contrastive Denoising with Noisy Chain-of-Thought.

Require: an LLM f_θ, the prompt of contrastive denoising $\mathcal{P}_{\text{denoise}}$, one test question x_{test}, one clean example $(x_{\text{C}}, \mathcal{T}_{\text{C}}, y_{\text{C}})$, K prompting examples $S_n = \{(x_i, \mathcal{T}_i, y_i)\}_{i=1}^{K}$, hyper-parameters N, M, and reasoning budget $\{B_i\}_{i=1}^{M}$ (satisfies that $\Sigma_{i=1}^{M} B_i = D$, where D is the total budget).

for $i = 1 \dots K$ **do**
  initialize the set of rephrased results of i-th example $\mathcal{R}_i \leftarrow \emptyset$.
  for $j = 1 \dots N$ **do**
    `# Step-1: Rationale Rephrasing via Supervised Contrasting`
    obtain a rephrased example as $(x_i, \hat{\mathcal{T}}_i, \hat{y}_i) \leftarrow f_\theta\Big(\mathcal{P}_{\text{denoise}}(x_{\text{C}}, \mathcal{T}_{\text{C}}, y_{\text{C}}, x_i, \mathcal{T}_i, y_i)\Big)$.
    if match answer $\hat{y}_i = y_i$, then store the rephrased example as $\mathcal{R}_i \leftarrow \mathcal{R}_i \cup \{(x_i, \hat{\mathcal{T}}_i, \hat{y}_i)\}$.
  end for
  `# Step-2: Rationale Selection`
  randomly select M rephrased examples from $\mathcal{R}_i$ and obtain $\tilde{\mathcal{R}}_i = \{(x_{is}, \hat{\mathcal{T}}_{is}, \hat{y}_{is})\}_{s=1}^{M}$.
end for
`# Step-3: Rationale Exploration`
initialize the set of answers $\mathcal{Y} \leftarrow \emptyset$.
for $i = 1 \dots M$ **do**
  construct an input $\mathcal{P}_i \leftarrow \{(x_{ji}, \hat{\mathcal{T}}_{ji}, \hat{y}_{ji})\}_{j=1}^{K}$, where $(x_{ji}, \hat{\mathcal{T}}_{ji}, \hat{y}_{ji})$ is the i-th element of $\hat{\mathcal{R}}_j$.
  concatenate $\mathcal{P}_i$ with the clean example and test question as $\mathcal{P}_i \leftarrow \mathcal{P}_i \cup \{(x_{\text{C}}, \mathcal{T}_{\text{C}}, y_{\text{C}}), x_{\text{test}}\}$.
  for $j = 1 \dots B_M$ **do**
    get one answer by LLM reasoning as $y_j \leftarrow f_\theta(\mathcal{P}_i)$.
    store the answer as $\mathcal{Y} \leftarrow \mathcal{Y} \cup \{y_j\}$.
  end for
end for
`# Step-4: Answer Voting`
initialize the dictionary of answer count $\mathcal{C}$ that $\forall y_j \in \mathcal{Y}, \mathcal{C}[y_j] = 0$.
for $j = 1 \dots D$ **do**
  update $\mathcal{C}[y_j] \leftarrow (\mathcal{C}[y_j] + 1)$.
end for
get the final answer y with maximum counts as $y \leftarrow \arg\max_y \mathcal{C}[y]$.
return the answer y.

Empirical Evaluations of CD-CoT

(besides the CoT demonstrations, the **additional information** required by the method)

Task	Method $\mathcal{M}$	Additional Information	Acc($\mathcal{M}$, $\mathcal{Q}$, $\mathcal{P}_{\text{clean}}$)	Acc($\mathcal{M}$, $\mathcal{Q}$, $\mathcal{P}_{\text{irrelevant}}$) Easy	Medium	Hard	Avg.	Acc($\mathcal{M}$, $\mathcal{Q}$, $\mathcal{P}_{\text{inaccurate}}$) Easy	Medium	Hard	Avg.
Math Base-9	Base	-	46.4	39.3	30.3	26.6	32.1	23.2	10.1	6.0	13.1
	w/ SCO [29]	Ground Truth	53.6	46.3	39.6	36.4	40.8	34.7	22.0	17.7	24.8
	w/ BT [81]	Noise Position	47.2	39.2	34.2	29.9	34.4	30.1	18.4	14.1	20.9
	w/ CC [9]	Clean Demo	44.9	43.3	44.6	45.5	44.5	37.2	31.7	30.7	33.2
	w/ CD-CoT (ours)	Clean Demo	**60.7**	**59.7**	**60.7**	**57.2**	**59.2**	**54.0**	**58.7**	**48.4**	**53.7**
Math Base-11	Base	-	23.9	19.1	13.6	10.7	14.5	14.0	6.7	3.6	8.1
	w/ SCO [29]	Ground Truth	**33.0**	29.2	24.0	20.0	24.4	**29.2**	20.0	17.2	22.1
	w/ BT [81]	Noise Position	24.3	17.9	17.2	13.7	16.3	12.8	9.2	6.8	9.6
	w/ CC [9]	Clean Demo	22.3	19.1	18.4	18.2	18.6	19.0	15.3	14.6	16.3
	w/ CD-CoT (ours)	Clean Demo	31.0	**33.7**	**32.7**	**34.7**	**33.7**	29.0	**30.7**	**25.3**	**28.3**
Symbolic Equal	Base	-	32.7	28.1	25.1	23.0	25.4	29.1	26.1	22.7	26.0
	w/ SCO [29]	Ground Truth	38.5	34.9	33.4	32.7	33.7	34.0	34.1	34.5	34.2
	w/ BT [81]	Noise Position	31.8	26.0	22.7	22.6	23.8	26.3	22.7	22.9	24.0
	w/ CC [9]	Clean Demo	37.8	33.8	32.7	32.0	32.8	31.3	33.0	29.9	31.4
	w/ CD-CoT (ours)	Clean Demo	**42.7**	**44.7**	**42.7**	**44.0**	**43.8**	**42.6**	**41.3**	**42.7**	**42.2**
Symbolic Longer	Base	-	9.2	6.3	7.2	6.0	6.5	7.0	6.8	6.0	6.6
	w/ SCO [29]	Ground Truth	**18.7**	**12.1**	10.5	11.3	11.3	**15.2**	**15.9**	9.8	**13.6**
	w/ BT [81]	Noise Position	7.2	3.4	3.5	2.5	3.1	3.8	3.6	3.6	3.7
	w/ CC [9]	Clean Demo	9.4	9.8	7.9	7.9	8.5	8.5	7.4	6.5	7.5
	w/ CD-CoT (ours)	Clean Demo	12.3	12.0	**12.0**	**13.0**	**12.3**	12.3	10.0	**11.0**	11.1
Commonsense	Base	-	45.7	44.3	42.3	41.4	42.7	36.7	33.4	28.3	32.8
	w/ SCO [29]	Ground Truth	**63.5**	**60.1**	**56.1**	**60.3**	**58.8**	**56.2**	**58.5**	**57.9**	**57.5**
	w/ BT [81]	Noise Position	47.7	23.5	28.3	32.5	28.1	11.6	11.0	15.8	12.8
	w/ CC [9]	Clean Demo	48.3	45.7	43.6	44.0	44.4	42.1	40.8	40.5	41.1
	w/ CD-CoT (ours)	Clean Demo	49.0	50.3	54.7	50.3	51.8	51.0	49.7	49.7	50.1

Table 8: Performance of denoising methods that require additional information for supervision.

Observation 7: CD-CoT presents a significant performance improvement across all datasets, **with an average improvement of 17.8%** compared with the base model under noisy settings.

Observation 8: CD-CoT displays remarkable **resistance to the magnitude of noise**, especially in the challenging mathematical tasks.

Baseline methods:
- Self-correction with Oracle Feedback (SCO)
- Backtracking (BT)
- Contrastive CoT (CC)

Empirical Evaluations of CD-CoT

Model	Method	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\mathrm{irrelevant}})$ Base-9	Sym.(E)	Com.	$\mathrm{Acc}(\mathcal{M}, \mathcal{Q}, \mathcal{P}_{\mathrm{inaccurate}})$ Base-9	Sym.(E)	Com.
GPT-3.5-turbo	Base	30.3	25.1	42.3	10.1	26.1	33.4
	SC	36.6	28.3	45.0	17.3	30.7	44.7
	BT	34.2	22.7	28.3	18.4	22.7	11.0
	CC	44.3	32.7	43.6	31.7	33.0	40.8
	CD-CoT	**60.7**	**42.7**	**54.7**	**58.7**	**41.3**	**49.7**
Gemini-Pro	Base	72.3	38.9	53.2	21.2	36.7	33.5
	SC	80.3	43.3	60.0	32.3	45.0	42.7
	BT	82.4	29.3	37.8	26.7	28.7	33.3
	CC	67.5	37.3	50.2	43.6	35.0	45.6
	CD-CoT	**92.7**	**49.3**	**57.7**	**76.7**	**53.3**	**55.7**
LLaMA2-70B	Base	2.8	8.7	41.9	2.7	9.1	40.2
	SC	**5.0**	10.3	**46.7**	**3.0**	9.7	**46.0**
	BT	1.4	11.2	36.1	0.9	12.5	36.2
	CC	1.1	**16.3**	29.9	2.8	**14.0**	28.3
	CD-CoT	4.0	9.7	39.3	2.7	9.7	39.7
Mixtral-8x7B	Base	16.3	17.9	34.9	3.7	15.1	31.1
	SC	20.0	21.7	37.0	2.7	18.0	37.7
	BT	4.1	9.7	6.2	2.4	10.1	10.5
	CC	**24.4**	18.5	36.0	**12.5**	18.3	35.7
	CD-CoT	8.7	**22.7**	**40.3**	4.7	**21.3**	**40.3**

Table 11: Comparing methods with different LLMs.

Observation 9: CD-CoT **generalizes well** across different LLMs.

Performance Gains

- GPT-3.5-turbo: **+23.4%** accuracy.
- Gemini-Pro: **+21.6%** accuracy
  Both surpass all baselines.

Empirical Evaluations of CD-CoT

Comparison of rephrased rationales by different reasoning methods.

Original Noisy Rationale
In base-9, the digits are "012345678". 9 + 8 = 17. We have 6 + 7 = 13 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base-10. 14 + 8 = 22. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit is 1. So the answer is 154. Answer: 154.

Ours	In base-9, the digits are "012345678". We have 6 + 7 = 13 in base-10. Since we're in base-9, that exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 5, so the digit is 5 and the carry is 1. A leading digit is 1. So the answer is 154. Answer: 154.

Method	Denoised Rationale
SM	In base-9, the dkgits are "012345678". 9 + 8 = 17. We5hav6̂r+ 7l=J13 in base-10. Since we're in baze-9, ithFt exceeis the maCimum valur of 8 for a siLgle]digit. x0c3 mod 9 = ; soWghe digix is 4 anj the carry is 1. We hOve 8 + 5i+ z2= 14 Vn baAe 10. 14 d 8 = 2p. 14 mod 9 = 5, so Xhe diGit*iU 5 and the carry is 1Q A leading x0cigit is 1. So the answer is 154. Asswer: 154.
SD	In base-9, the digits are "012345678". 9 + 8 = 17. We have 6 + 7 = 13 in base-10. Since we're in base-9, 13 exceeds the maximum value of 8 for a single digit. 13 mod 9 = 4, so the digit is 4 and the carry is 1. We have 8 + 5 + 1 = 14 in base 10. 14 mod 9 = 8. 14 div 9 = 1, so the digit is 8 and the carry is 1. A leading digit is 1. So the answer is 154. Answer: 154.
SP	In base-9, the digits range from 0 to 8. Let's add 86 and 57 step by step. First, we add the rightmost digits, which are 6 and 7. In base-10, 6 + 7 equals 13. However, since we are in base-9, the maximum value for a single digit is 8. Therefore, we need to find the remainder when 13 is divided by 9. The remainder is 4, so the digit in the units place is 4 and we have a carry of 1. Next, we add the next digits, which are 8 and 5, along with the carry of 1. In base-10, 8 + 5 + 1 equals 14. Again, we need to find the remainder when 14 is divided by 9. The remainder is 5, so the digit in the tens place is 5 and we have a carry of 1. Finally, we have a leading digit of 1. So the final answer in base-9 is 154. Answer: 154.

Observation 10: CD-CoT effectively **removes noisy thoughts** and ensures format alignment with the original rationale.

Baseline methods:
SM: Randomly masks the prompt.
SD: Applies random masking and reconstructs prompts through FM.
SP: Reconstructs prompts following guidelines through FM.

Take Home Messages

We investigate the **under-explored** problem of noisy rationales.

We introduce **NoRa dataset** to evaluate LLMs against noisy rationales.

We reveal the **general vulnerability** of LLMs to noisy rationales; this is not well addressed by existing robust methods.

We design **CD-CoT** method to enhance the robustness via contrastive denoising.

Future Directions

Robust pre-training/fine-tuning methods are required for VLMs.

- VLMs can still be mislead by spurious features.
- Larger models and high-quality data lead to better robustness.

The trade-off between unlearning and retention remains a critical issue.

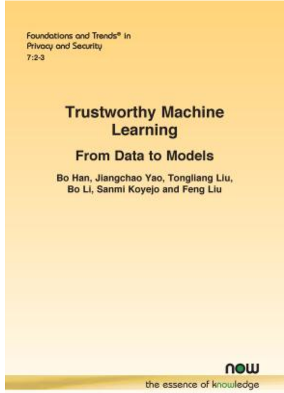
- Current unlearning objectives all have negative impacts on retention.
- Data and optimization aspects of unlearning are not well explored.

Reasoning with noisy rationales can be further investigated.

- Non-reasoning models (GPT 3.5/4/4o) is not robust on the NoRa dataset.
- Reasoning models R1/o1/o3 is generally more robust but exhibit over-thinking issues.

Appendix

- Survey:
  - A Survey of Label-noise Representation Learning: Past, Present and Future. arXiv, 2020.
- Book:
  - Machine Learning with Noisy Labels: From Theory to Heuristics. Adaptive Computation and Machine Learning series, **The MIT Press**, 2025.
  - Trustworthy Machine Learning under Imperfect Data. CS series, **Springer Nature**, 2025.
  - Trustworthy Machine Learning: From Data to Models. **Foundations and Trends® in Privacy and Security**, 2025.



- Tutorial:
  - IJCAI 2021 Tutorial on Learning with Noisy Supervision
  - CIKM 2022 Tutorial on Learning and Mining with Noisy Labels
  - ACML 2023 Tutorial on Trustworthy Learning under Imperfect Data
  - AAAI 2024 Tutorial on Trustworthy Machine Learning under Imperfect Data
  - IJCAI 2024 Tutorial on Trustworthy Machine Learning under Imperfect Data
  - WWW 2025 Tutorial on Trustworthy AI under Imperfect Web Data
- Workshops:
  - IJCAI 2021 Workshop on Weakly Supervised Representation Learning
  - ACML 2022 Workshop on Weakly Supervised Learning
  - RIKEN 2023 Workshop on Weakly Supervised Learning
  - HKBU-RIKEN AIP 2024 Joint Workshop on Artificial Intelligence and Machine Learning