

Toward Trustworthy Language Models through Interpretability and Control

Lily Weng

Halicioglu Data Science Institute

UC San Diego

July 4, 2026 @ TrustNLP Workshop

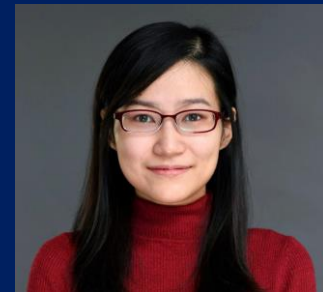


lweng@ucsd.edu



lilywenglab.github.io

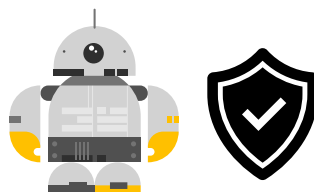
Trustworthy ML Lab @ UCSD



lilywenglab.github.io

Are DNNs safe and trustworthy?

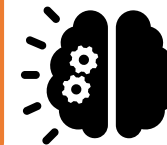
Robust ML



- 1st DNN robustness score [ICLR'18]
- DNN robustness/safety certificate
 - Worst-case guarantee [ICML'18, NeurIPS'18, AAAI'19, ICML'19, AAAI'20]
 - Probabilistic & Semantic guarantee [ICML'19, CVPR'20, NeurIPS'20]
 - DRL agents [ICML'20, IROS'22]
- Robust Learning
 - Supervised learning [AAAI'21, ICLR'21, ICDM'23, ICLR'24, ECCV'24]
 - Self-supervised learning [ICML'23]
 - Deep RL [NeurIPS'21, ICML'24]
- Attacking DNN [ICLR'20, ICDM'23, ICCV'23, NAACL'25]

Why do DNNs perform so well?

Interpretable ML

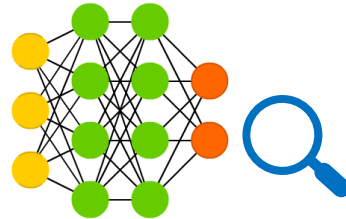


- Demystify and Interpret DNNs
 - Vision models [ICLR'23, ICML'24, WACV'25 TMLR'25]
 - LLMs & Audio models [NeurIPS'23, NAACL'25, ICML'24, EMNLP'25, preprint'25, preprint'26]
- Build Interpretable DNNs by design
 - Vision models [ICLR'23, NeurIPS'24]
 - Deep generative models [CVPR'25, CVPR'26]
 - LLMs [ICLR'25]
 - Continual learning [TMLR'24, TMLR'26]
- Rigorous Evaluation for Interpretability
 - Vision models [NeurIPS'24, ICML'25, CVPR'26]
 - LLMs [ICML'25]
- Enable Transparent Learning [AAAI'24 DAI]

Interpretability is crucial to enable **Trustworthy AI**

Trust

- ✓ Users will trust a system more if they can understand it
- ✓ Important for high-stakes domains



Interpretability

Safety

- ✓ Evaluate & monitor their capabilities
- ✓ Understand their failure modes & when we can safely deploy them



Transparency

- ✓ Detect biases and debug issues within the model
- ✓ Editing existing models to fix issues

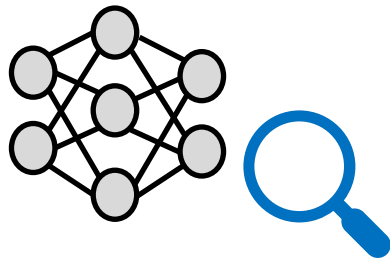
Improved models

- ✓ Gaining insight on how and what our models learn
- ✓ Helps design better models and training methods

Interpretability as a Control Interface

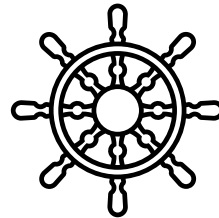
- **Actionable interpretability:** turn explanations into **control** variables

Control AI model behaviors easily



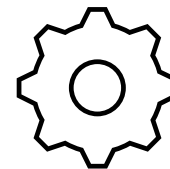
Understand

+



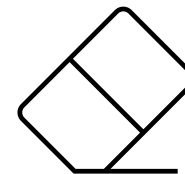
Steer

+



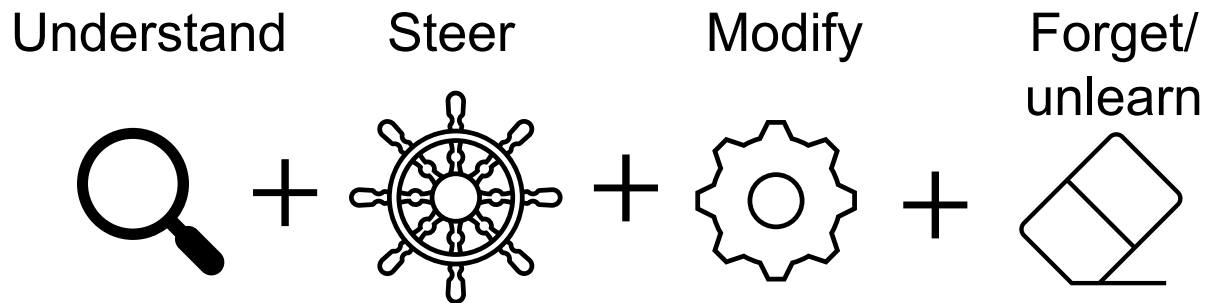
Modify

+

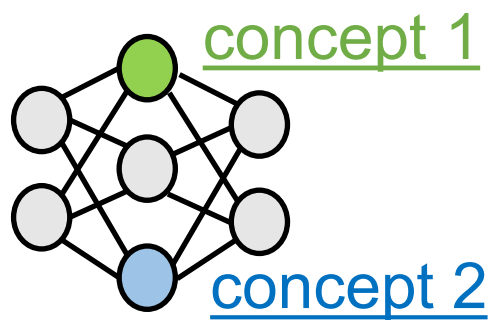


Forget/ Unlearn

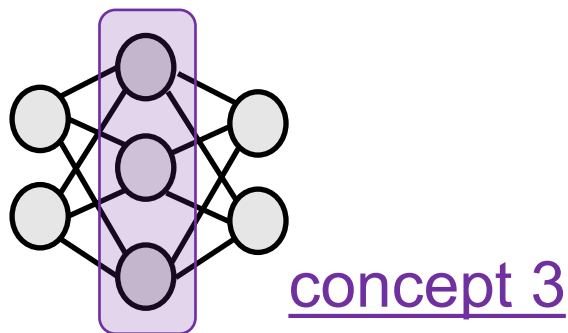
Actionable Interpretability: Two approaches



1. Post-hoc Interpretability tools



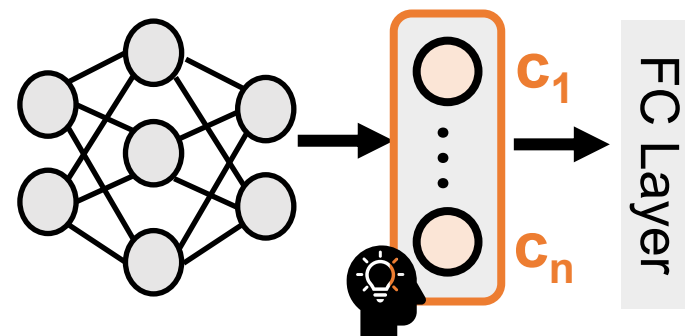
Neuron
concepts



Representation
concept



2. Interpretable DNN by Design



**Human-Interpretable
Concept layer**
(Concept bottleneck layer)

Our work in Interpretable ML

Tuomas



Ge



Chung-En



Akshay



1. Interpretability Tools

Develop **automated** interpretability techniques to **understand** DNNs



[CLIP-Dissect](#)
[ICLR'23 spotlight]

1st fully automation with higher quality, 10-200x faster

[Auto Neuron Exp](#)
[NeurIPS'23]

2x faster than OpenAI's result with higher quality

[Linear Explanation](#)
[ICML'24]

Provide polysemantic explanations

[Audio-Net-Dissect](#)
[ICML'24]

Neuron interpretability tool for audio networks

[Describe-n-Dissect](#)
[TMLR'25]

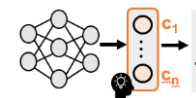
Generative Natural Language Explanation

[ThinkEdit](#), [Reflectrl](#)
[EMNLP'25, arxiv'26]

Steer & control reasoning process of LRMs

2. Interpretable DNN by design

Develop systematic method to **learn** intrinsically-interpretable DNNs



[Label-free CBM](#)
[ICLR'23]

1st CBM on Imagenet + automation + high acc

[CB-LLMs](#)
[ICLR'25]

1st interpretable LLM + high steerability & scalability

[VLG-CBM](#)
[NeurIPS'24]

Faithful CBM + 1st theoretical analysis & fair comparison

[CB-AE](#), [CB-SAE](#)
[CVPR'25, CVPR'26]

1st post-hoc interpretable generative model & VLM

3. Evaluation

Rigorous comparison on explanations, metrics, CBMs

[Neuron Eval](#)
[ICML'25]

1st unified evaluation framework + sanity checks

[Rethink human study](#)
[CVPR'26]

Wholistic comparison + enable 40x reduced cost \$\$!

[VLG-CBM](#)
[NeurIPS'24]

1st metric to compare between CBMs

Our papers are available at
<https://lilywenglab.github.io/#interpretable-deep-learning>

This Talk: Interpretability for Controllable AI

1. Interpretability Tools

Develop **automated** interpretability techniques to **understand** DNNs



1

[CLIP-Dissect](#)
[ICLR'23 spotlight]

1st fully automation with higher quality, 10-200x faster

[Auto Neuron Exp](#)
[NeurIPS'23]

2x faster than OpenAI's result with higher quality

[Linear Explanation](#)
[ICML'24]

Provide polysemantic explanations

[Audio-Net-Dissect](#)
[ICML'24]

Neuron interpretability tool for audio networks

[Describe-n-Dissect](#)
[TMLR'25]

Generative Natural Language Explanation

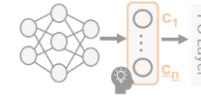
2

[ThinkEdit](#), [Reflectrl](#)
[EMNLP'25, arxiv'26]

Steer & control reasoning process of LRMs

2. Interpretable DNN by design

Develop systematic method to **learn** intrinsically-interpretable DNNs



3

[Label-free CBM](#)
[ICLR'23]

1st CBM on Imagenet + automation + high acc

[CB-LLMs](#)
[ICLR'25]

1st interpretable LLM + high steerability & scalability

[VLG-CBM](#)
[NeurIPS'24]

Faithful CBM + 1st theoretical analysis & fair comparison

[CB-AE](#), [CB-SAE](#)
[CVPR'25, CVPR'26]

1st post-hoc interpretable generative model & VLM

Our papers are available at
<https://lilywenglab.github.io/#interpretable-deep-learning>

3. Evaluation

Rigorous comparison on explanations, metrics, CBMs

[Neuron Eval](#)
[ICML'25]

1st unified evaluation framework + sanity checks

[Rethink human study](#)
[CVPR'26]

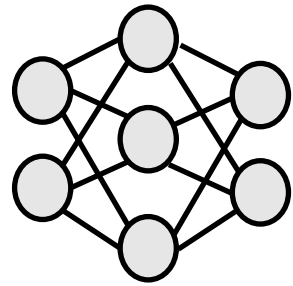
Wholistic comparison + enable 40x reduced cost \$\$!

[VLG-CBM](#)
[NeurIPS'24]

1st metric to compare between CBMs

What is an (neuron) interpretability tool?

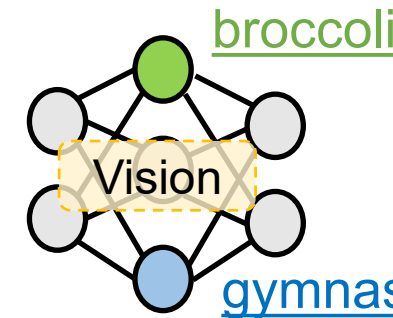
Given a DNN



**Interpretability
Tools**



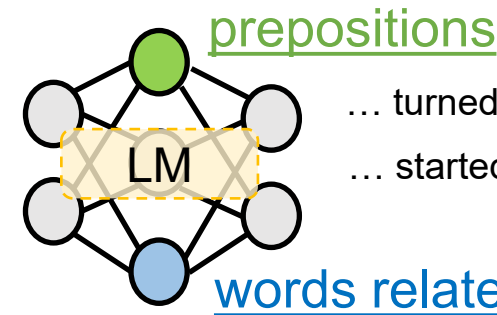
Neuron concepts



broccoli



gymnastic



prepositions

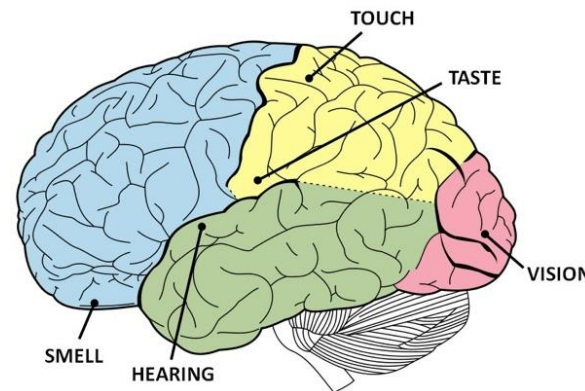
... turned out to be wrong ...

... started to think that ...

words related to time

... We've been priming ...

... not yet ... after that, ...

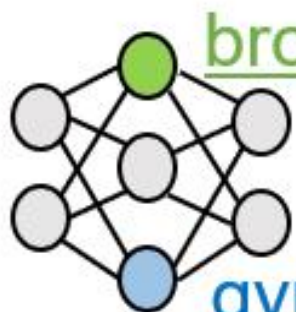


Inspired by *neuroscience*, we focus on understanding the function of individual hidden neurons in DNNs

Challenges

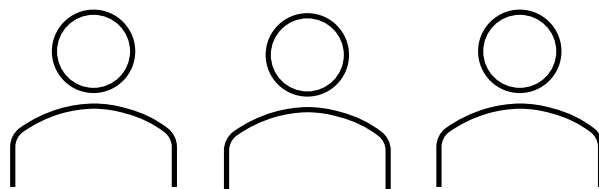
- Existing work is largely based on *manual inspection*

Neuron concepts



gymnastic

Time consuming &
hard to scale



Human Researchers

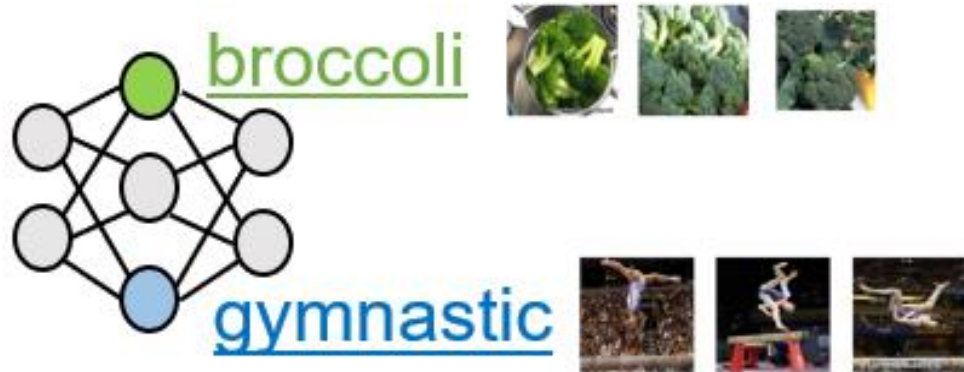
Research efforts

- Sumioka et al., 2009
- Zeiler et al., ECCV 2014
- Zhou et al., ICLR 2015
- Olah et al., Distill 2017
- Olah et al., Distill 2020
- Goh et al., Distill 2021
- ⋮

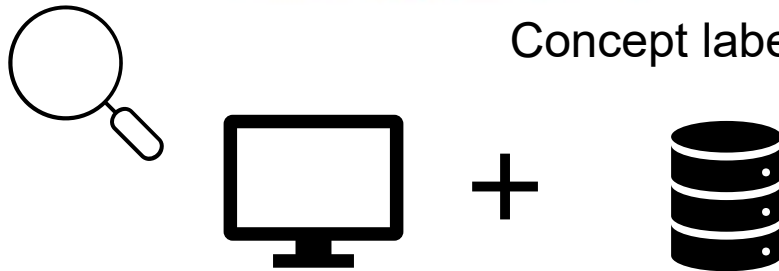
Challenges

- Recent work proposed to *automate* descriptions of neurons

Neuron concepts



Concept labeled data



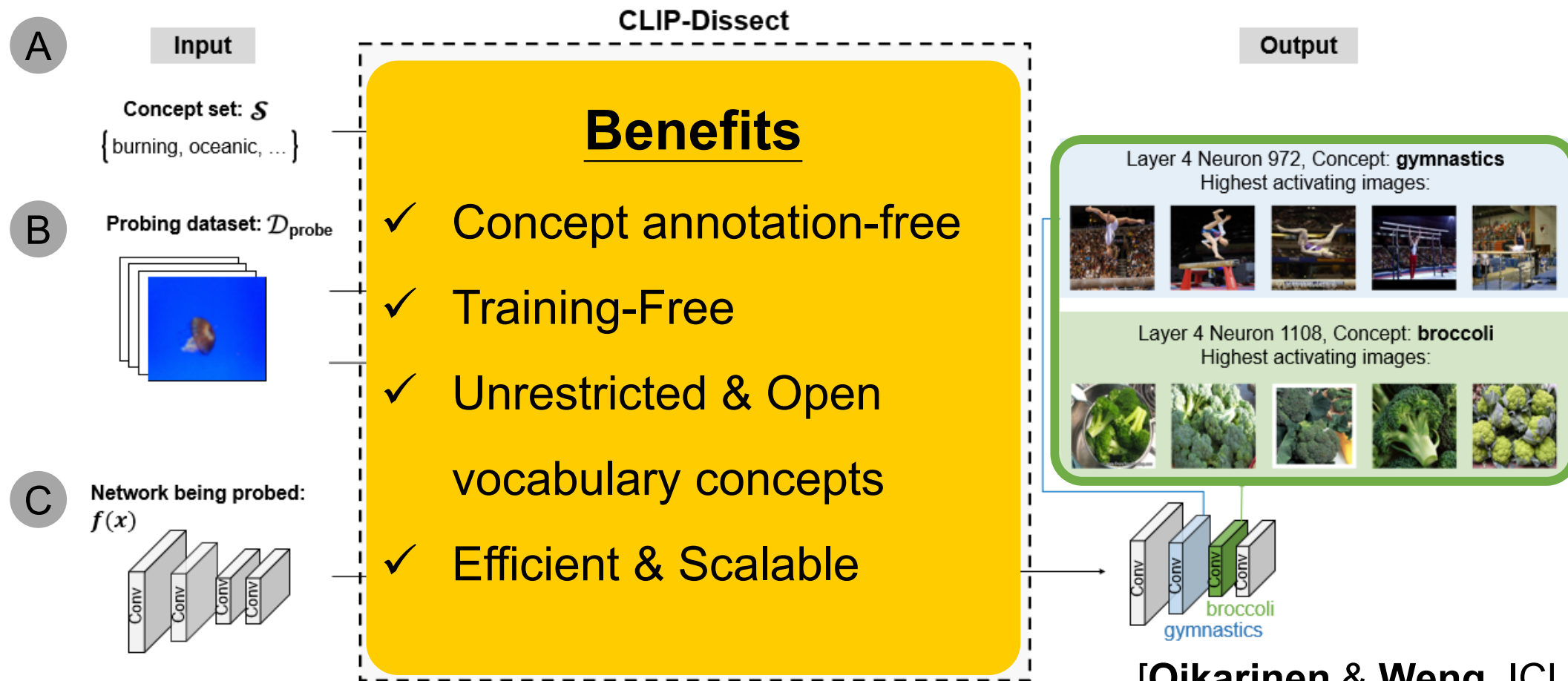
Research efforts

- Bau & Zhou et al., CVPR 2017
- Hernandez et al., ICLR 2022

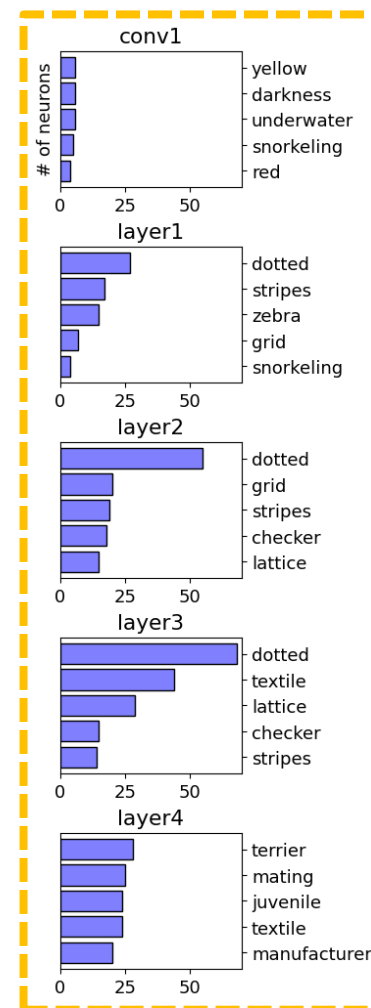
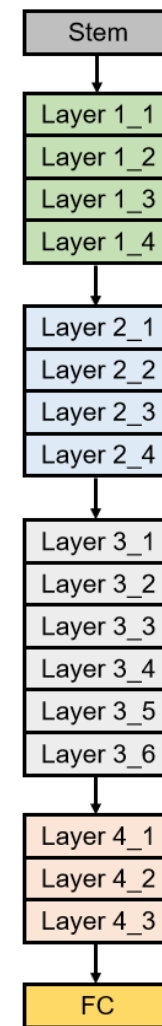
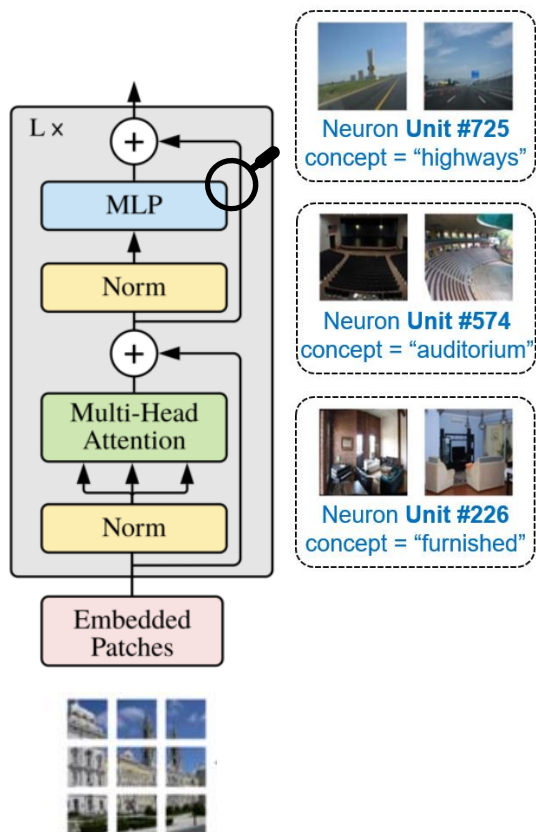
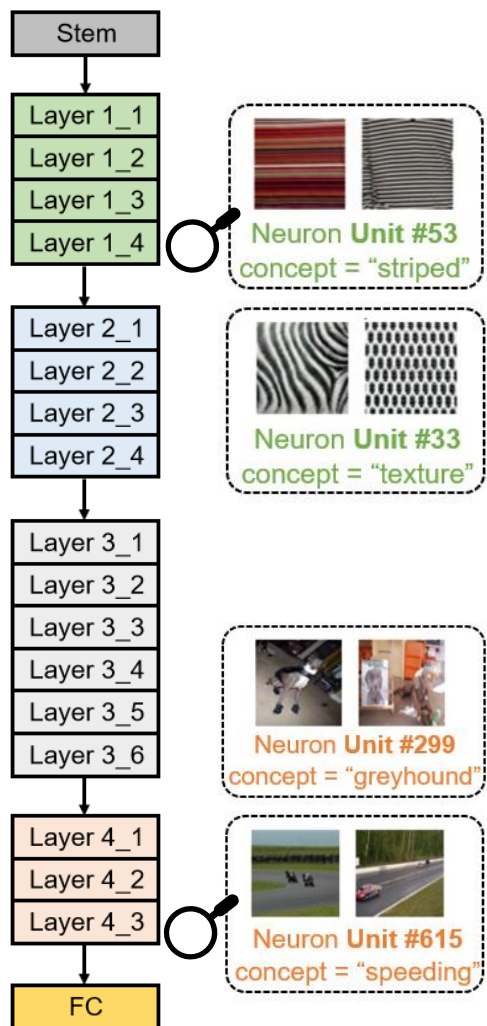
could be expensive / less
flexible to get concept data

A fully Automated Interpretability tool

- We propose **CLIP-Dissect** to address limitations of prior work



CLIP-Dissect: understand your DNNs



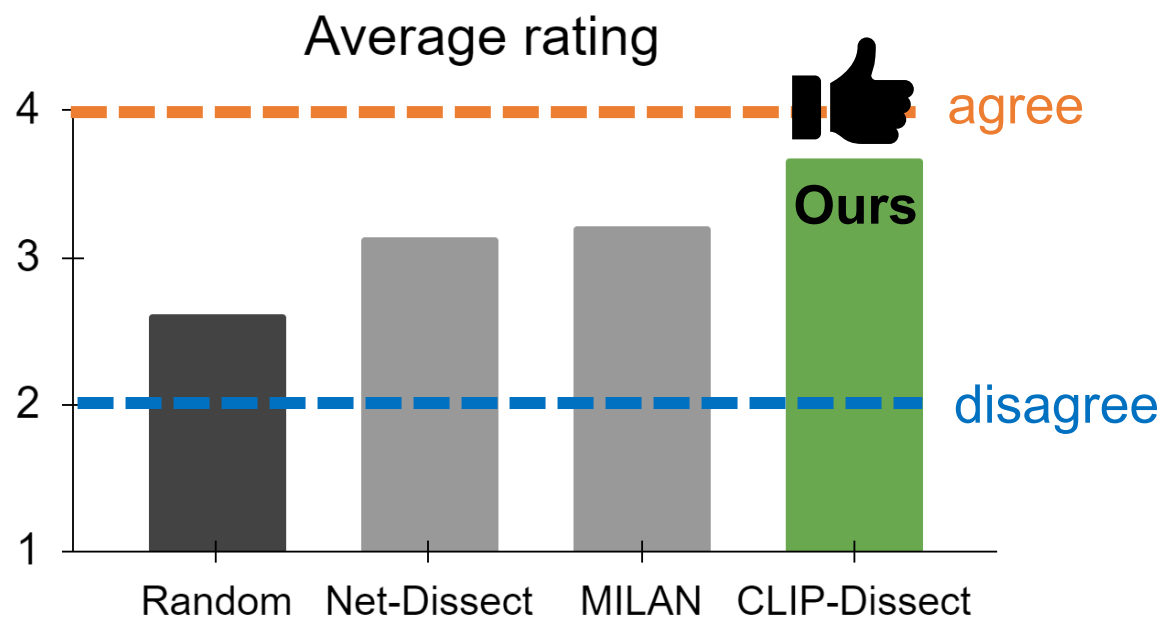
Get a Full profile of your DNN!

[Oikarinen & Weng, ICLR'23 spotlight]

CLIP-Dissect: Quantitative Results

- Large scale crowdsourced eval: Raters mostly agreed with **our descriptions**

Total 1520 neurons across 2 models & 10 layers

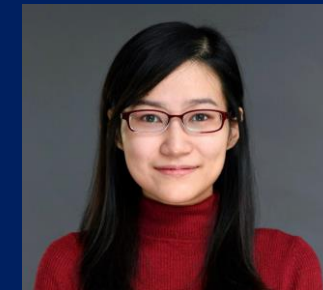


- Computational efficiency:**
Label all neurons for 5 layers in ResNet-50

Method	Run Time
Net-Dissect	> 4hrs
Comp-Exp	>> 14 hrs
MILAN	55 min 30s
CLIP-Dissect (Ours)	3min 50s 👍

CLIP-Dissect is more than **10-200x** faster than baseline methods!

Summary



CLIP-Dissect has the following benefits:

1 High quality neuron

2 Computationally

3 Flexible

Can we make it even more
automated / flexible?

- Better at final layer than prior work
- Good on hidden layer neurons
- 10-200x faster than prior work
- Dissect ResNet-50 in 4 minutes
- Works for any architecture
- Can leverage stronger multimodal models in future

★ **Arxiv:** <https://arxiv.org/abs/2204.10965>

★ **GitHub:** <https://github.com/Trustworthy-ML-Lab/CLIP-dissect>

Using LLMs to explain LLM's internals

■ Open AI's result [1]

- “Language models can explain neurons in language models”
- i.e. using GPT4 to explain GPT 2

Each excerpt show how the token activates the neuron of interest: darker shades means higher

Step 1 **Explain** the neuron's activations using GPT-4

Show neuron activations to GPT-4:

The Avengers to the big screen, Joss Whedon has returned to reunite Marvel's gang of superheroes for their toughest challenge yet. Avengers: Age of Ultron pits the titular heroes against a sentient artificial intelligence, and smart money says that it could soar at the box office to be the highest-grossing film of the

introduction into the Marvel cinematic universe, it's possible, though Marvel Studios boss Kevin Feige told Entertainment Weekly that, "Tony is earthbound and facing earthbound villains. You will not find magic power rings firing ice and flame beams." Spoilsport! But he does hint that they have some use... STARK T

, which means this Nightwing movie is probably not about the guy who used to own that suit. So, unless new director Matt Reeves' The Batman is going to dig into some of this backstory or introduce the Dick Grayson character in his movie, the Nightwing movie is going to have a lot of work to do explaining

of Avengers who weren't in the movie and also Thor try to fight the infinitely powerful Magic Space Fire Bird. It ends up being completely pointless, an embarrassing loss, and I'm pretty sure Thor accidentally destroys a planet. That's right. In an effort to save Earth, one of the heroes inadvertently blows up an

GPT-4 gives an explanation, guessing that the neuron is activating on

references to movies, characters, and entertainment.

Text excerpts from training split of the target model's pre-training dataset

Here,

- target model = GPT-2
- dataset = WebText
- focus on MLP layer in the transformer block

Using LLMs to explain LLM's internals

- Here's how the prompt sent into GPT 4 to get explanation

Original Excerpt

God give two men or **two** women a "right" to marry one another and then adopt children with the approval of the state? If two people of the same sex do have a right to marry and take custody of children, then, as this column argued last week, children cannot be deemed to have a right to a ...

OpenAI

Original Prompt

God	0
give	0
two	0
men	0
or	0
two	10
women	0
:	

[token, activation]

Length $\approx 4T$

Prompt can be very long

Ours

1. Summary

God give two men or
two women ...

Highly activating
tokens:

two

Length = $T + \Delta$

Ours

2. Highlight

God give two men or
[two] women ...

Length = $T + \delta$

Ours

3. HighlightSummary

God give two men or
[two] women ...

Highly activating
tokens:

two

Length = $T + \Delta + \delta$

Ours

4. AVHS

God give two men or
[two] women ...

Highly activating
tokens:

two 10

Length $\approx T + 2\Delta + \delta$

Generally, $\delta < \Delta \ll T \Rightarrow$ Improve efficiency! 👍

Using LLMs to explain LLM's internals

- Our method are **1.7-2.6x more efficient** than OpenAI's result while improving/keep neuron explanation quality

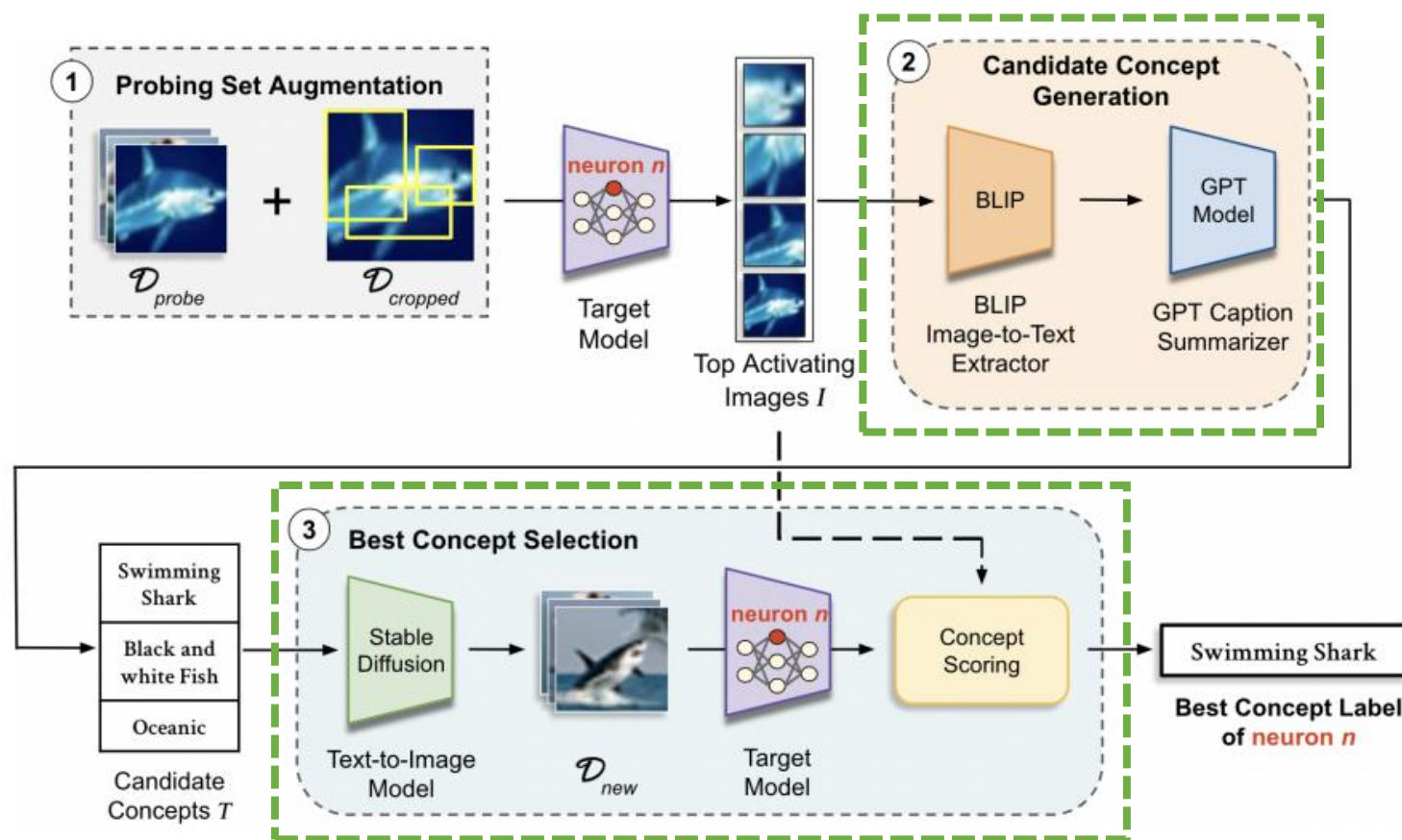
Similar ideas can be extended to understand vision models!

Saving up to 2.64x tokens while improving performance when explainer = GPT 3.5! 👍

Saving up to 2.64x tokens while maintain performance when explainer = GPT 4! 👍

Describe-and-Dissect

- Remove the need of pre-defined concept set & provide generative descriptions



Describe-and-Dissect

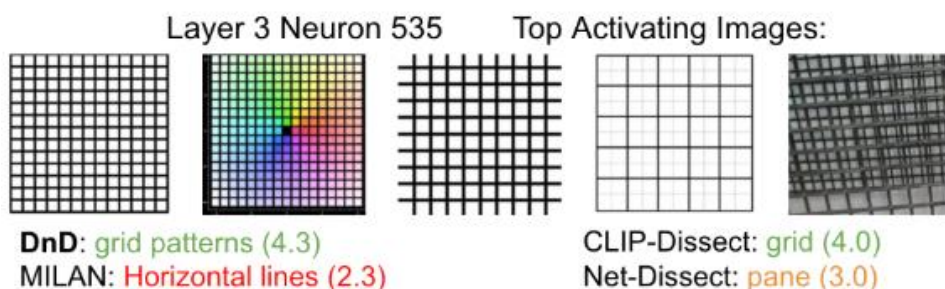


- Remove the need of pre-defined concept set & provide generative descriptions

Results on ResNet-50 model
trained on ImageNet

Crowdsourced evaluation on each layer:
Rating from 1 (strongly disagree) to 5 (strongly agree)

How to understand representations beyond neuron-level?



Metric	Layer	Network Dissection	MILAN	CLIP-Dissect	DnD (Ours)
Mean Rating	Layer 1	3.41 ± 0.058	3.32 ± 0.060	3.63 ± 0.057	4.16 ± 0.041
	Layer 2	3.52 ± 0.063	3.42 ± 0.064	3.55 ± 0.057	4.07 ± 0.048
	Layer 3	3.04 ± 0.060	2.96 ± 0.066	3.66 ± 0.055	4.14 ± 0.042
	Layer 4	2.97 ± 0.066	3.34 ± 0.061	3.82 ± 0.054	4.21 ± 0.044
% time selected as best answer	Layer 1	13.18%	14.32%	22.50%	50.00%
	Layer 2	15.27%	12.41%	19.33%	52.98%
	Layer 3	11.82%	12.73%	25.00%	50.45%
	Layer 4	10.56%	13.71%	25.62%	50.11%

Compared with 3 other baselines



This Talk - Part 2

1. Interpretability Tools

Develop **automated** interpretability techniques to **understand** DNNs



1

[CLIP-Dissect](#)
[ICLR'23 spotlight]

1st fully automation with higher quality, 10-200x faster

[Auto Neuron Exp](#)
[NeurIPS'23]

2x faster than OpenAI's result with higher quality

[Linear Explanation](#)
[ICML'24]

Provide polysemantic explanations

[Audio-Net-Dissect](#)
[ICML'24]

Neuron interpretability tool for audio networks

[Describe-n-Dissect](#)
[TMLR'25]

Generative Natural Language Explanation

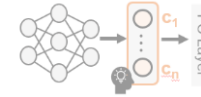
2

[ThinkEdit](#), [Reflectrl](#)
[EMNLP'25, arxiv'26]

Steer & control reasoning process of LRMs

2. Interpretable DNN by design

Develop systematic method to **learn** intrinsically-interpretable DNNs



3

[Label-free CBM](#)
[ICLR'23]

1st CBM on Imagenet + automation + high acc

[CB-LLMs](#)
[ICLR'25]

1st interpretable LLM + high steerability & scalability

[VLG-CBM](#)
[NeurIPS'24]

Faithful CBM + 1st theoretical analysis & fair comparison

[CB-AE](#), [CB-SAE](#)
[CVPR'25, CVPR'26]

1st post-hoc interpretable generative model & VLM

Our papers are available at
<https://lilywenglab.github.io/#interpretable-deep-learning>

3. Evaluation

Rigorous comparison on explanations, metrics, CBMs

[Neuron Eval](#)
[ICML'25]

1st unified evaluation framework + sanity checks

[Rethink human study](#)
[CVPR'26]

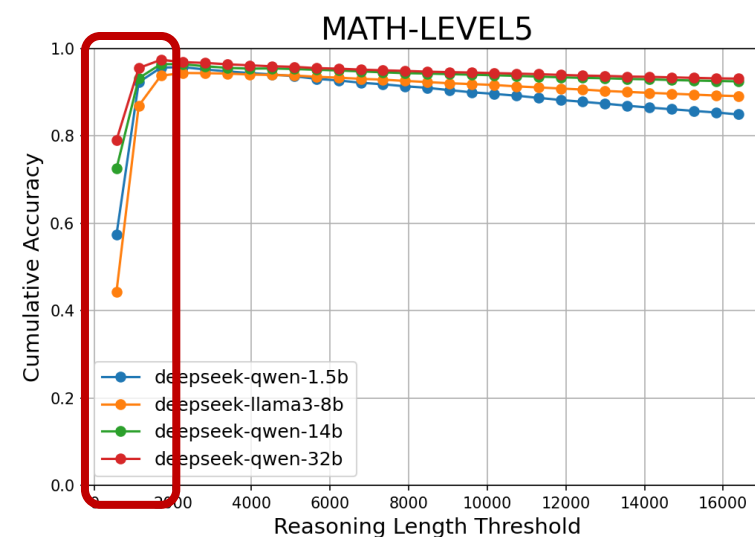
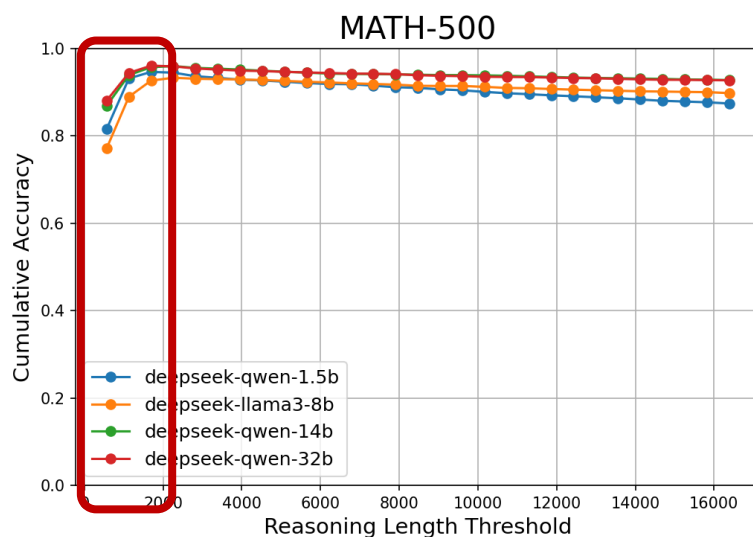
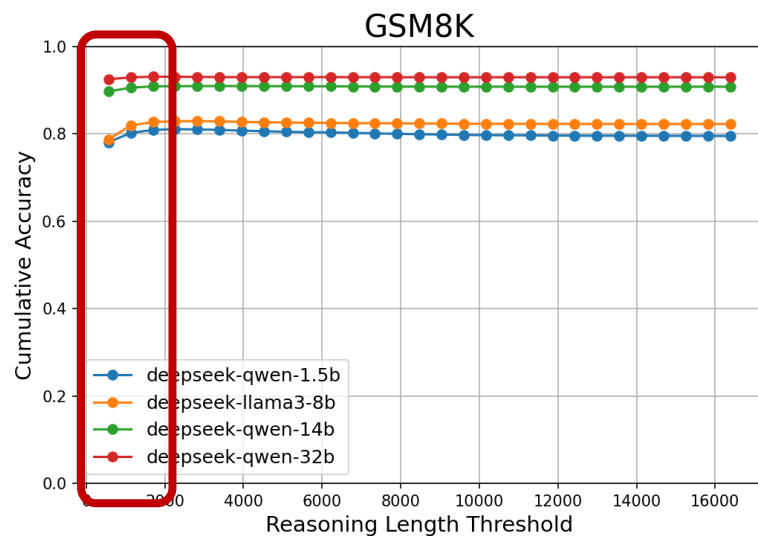
Wholistic comparison + enable 40x reduced cost \$\$!

[VLG-CBM](#)
[NeurIPS'24]

1st metric to compare between CBMs

Challenge

- Observation: in reasoning LLMs, overly short reasoning results in low accuracy

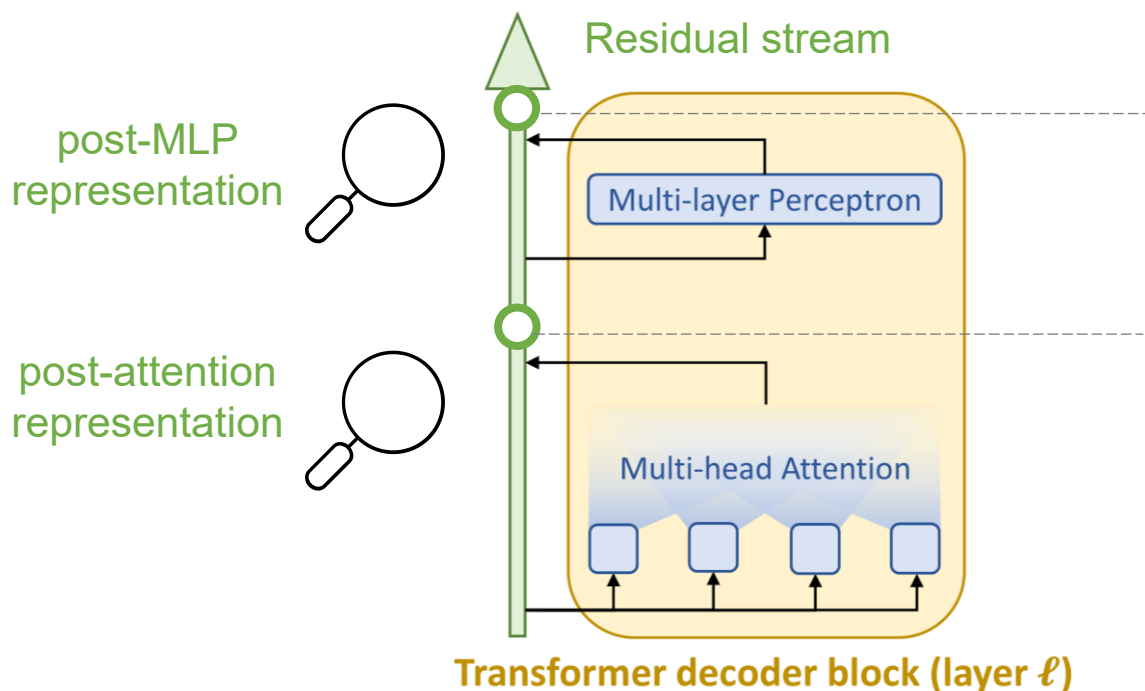


Accuracy **drops** more significantly for harder tasks when the reasoning length is **too short**

Motivations

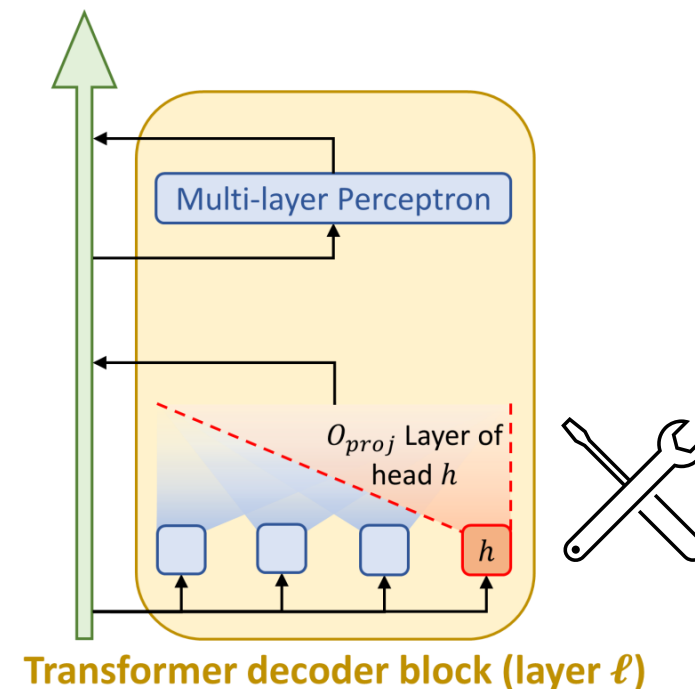
Question 1: Understanding

How is reasoning length encoded in or controlled by the hidden representation?



Question 2: Control

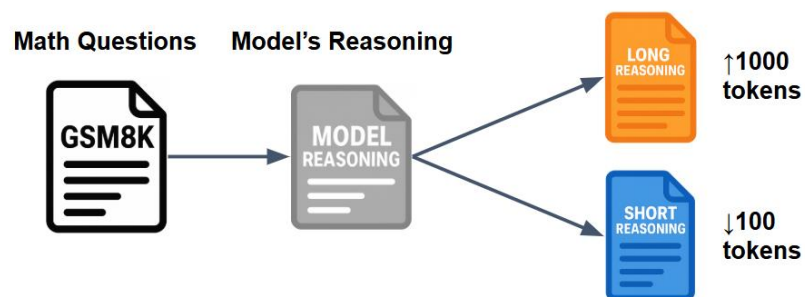
Could we mitigate the overly short reasoning by editing model parameters?



Sol #1: Extracting reasoning length direction

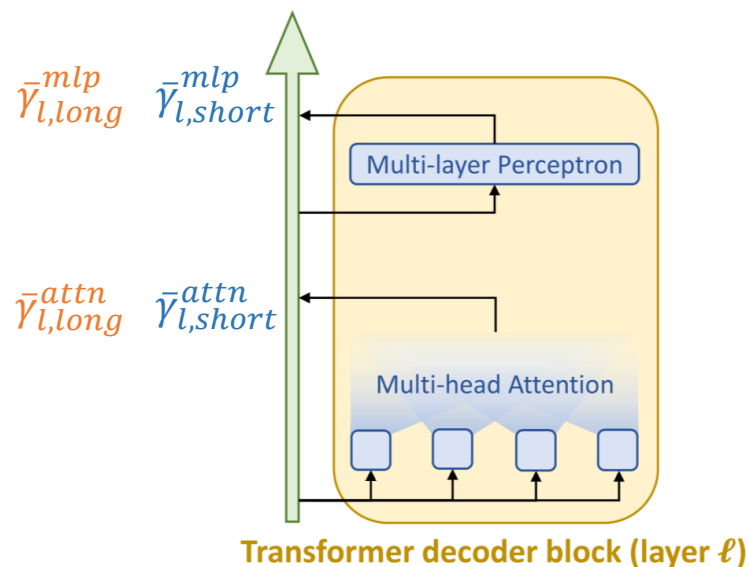
Step 1

Collecting probing data



Step 2

Extract long & short reasoning representation



Step 3

Compute reasoning length directions

- Post-MLP representation

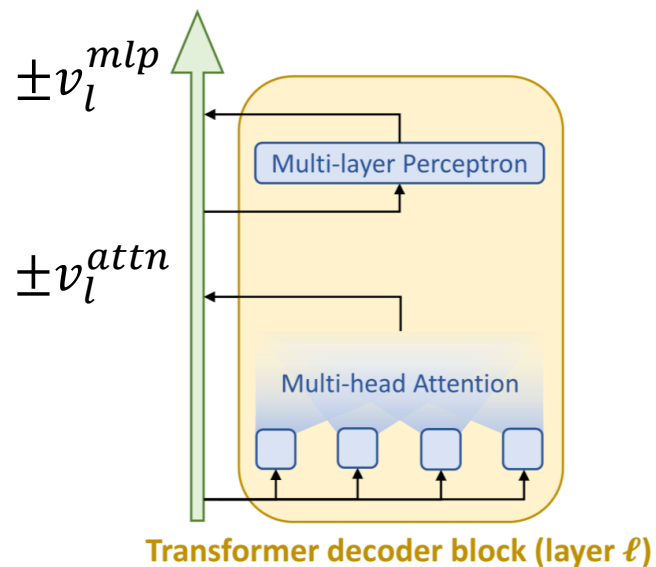
$$v_l^{mlp} = \bar{v}_{l,long}^{mlp} - \bar{v}_{l,short}^{mlp}$$

- Post-Attention representation

$$v_l^{attn} = \bar{v}_{l,long}^{attn} - \bar{v}_{l,short}^{attn}$$

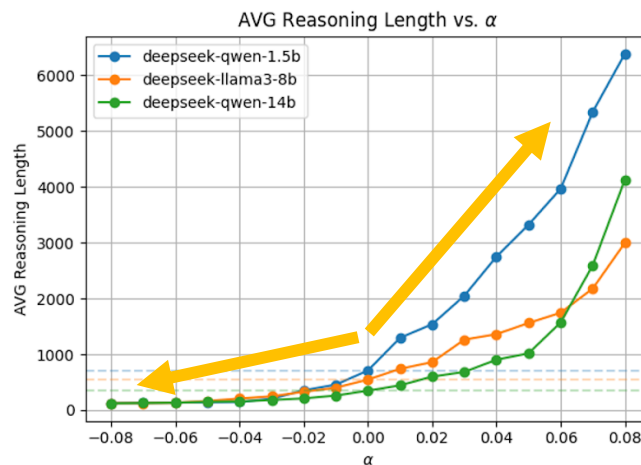
Sol #1: Extracting reasoning length direction

- Results of steering reasoning length:

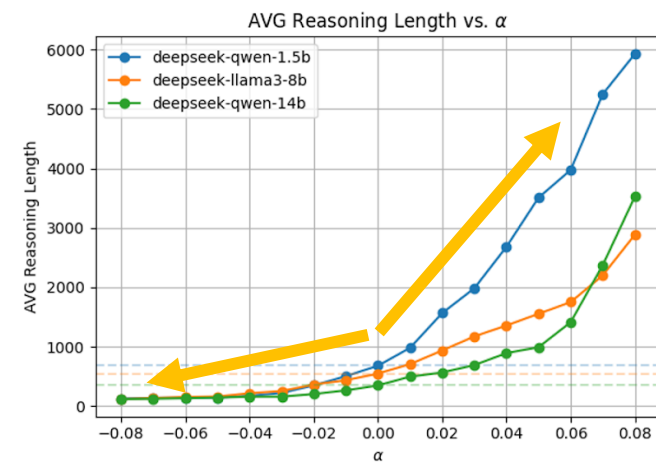


Observation 1

Steering is **effective** to change reasoning length to shorter / longer



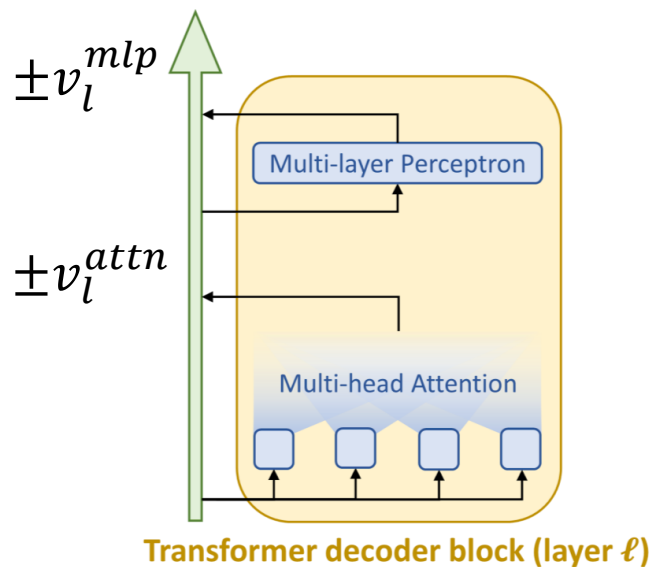
Steering Post-Attention representation
with $\pm v_l^{attn}$



Steering Post-MLP representation with
 $\pm v_l^{mlp}$

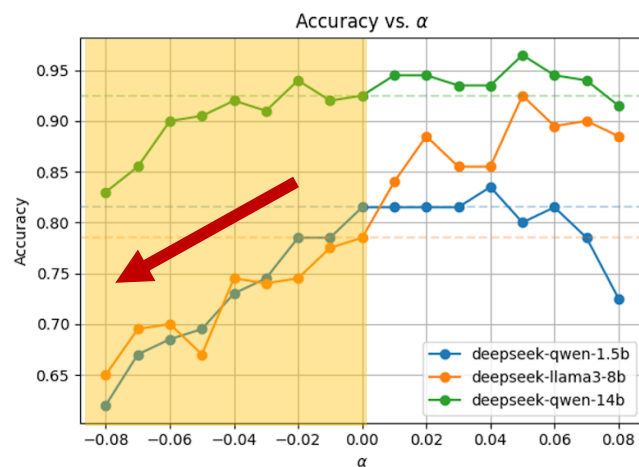
Sol #1: Extracting reasoning length direction

- Results of steering reasoning length:

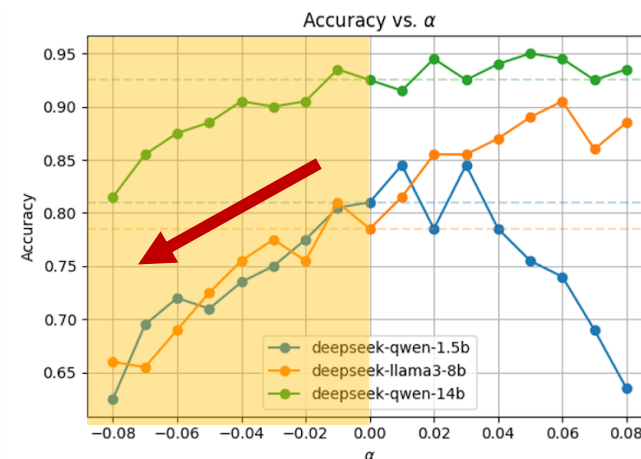


Observation 2

Task Accuracy **decrease** when steering reasoning length to shorter



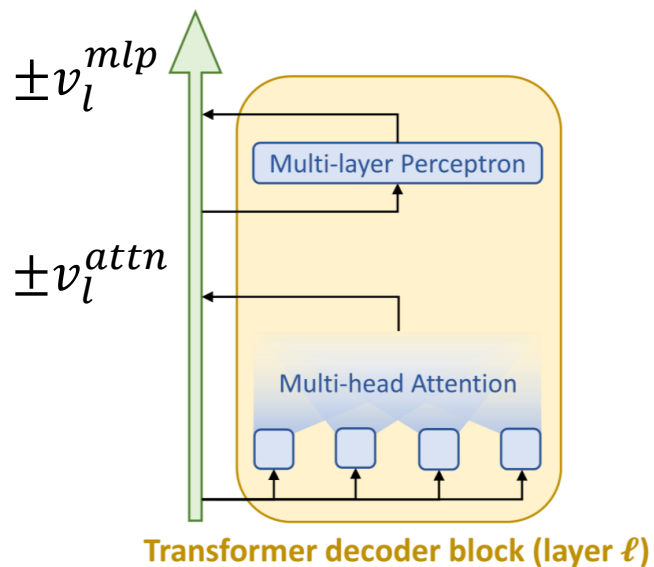
Steering Post-Attention representation
with $\pm v_l^{attn}$



Steering Post-MLP representation with
 $\pm v_l^{mlp}$

Sol #1: Extracting reasoning length direction

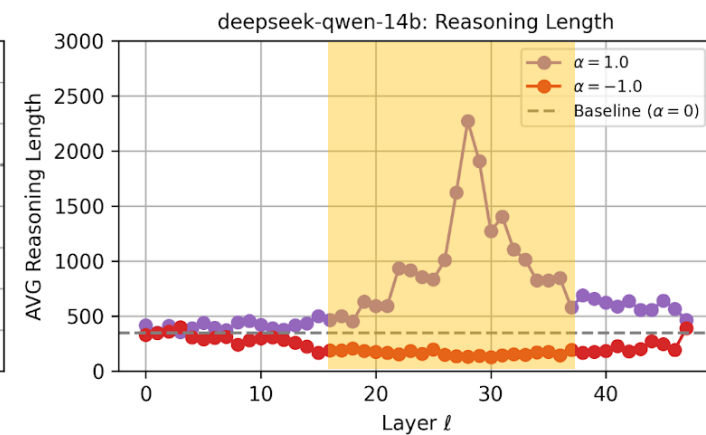
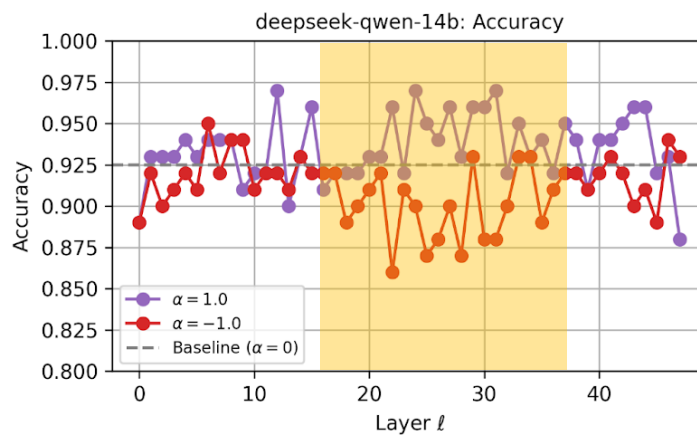
Results of steering reasoning length:



Observation 3.1

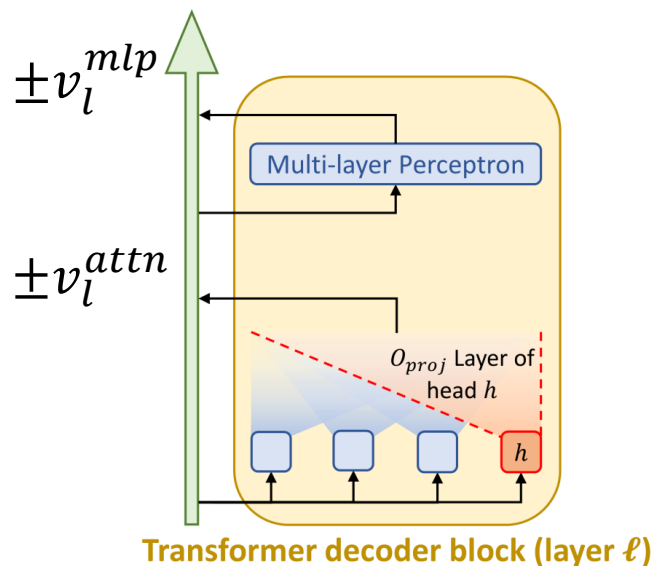
Reasoning length feature lies in the **middle layers** of the model

Layerwise Steering with v_l^{mlp}



Sol #1: Extracting reasoning length direction

- Results of steering reasoning length:



Calculating the contribution of head h to residual stream along $\hat{v}_l^{attn}$:

$$C^h = A^h W_O^h \quad \bar{C}_{short}^h = \langle \bar{C}^h, -\hat{v}_l^{attn} \rangle$$

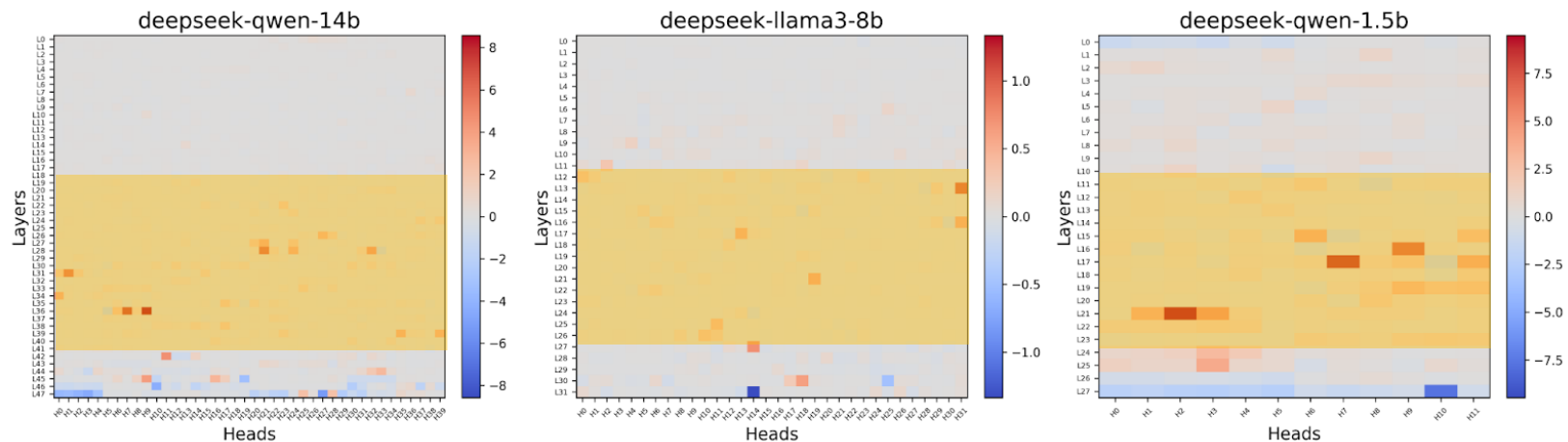
Head h 's output

Projection to short reasoning direction (normalized) $\hat{v}_l^{attn}: v_l^{attn} / \|v_l^{attn}\|$

Observation 3.2

Mid-layer attention heads strongly drive short reasoning – **short reasoning heads (red)**

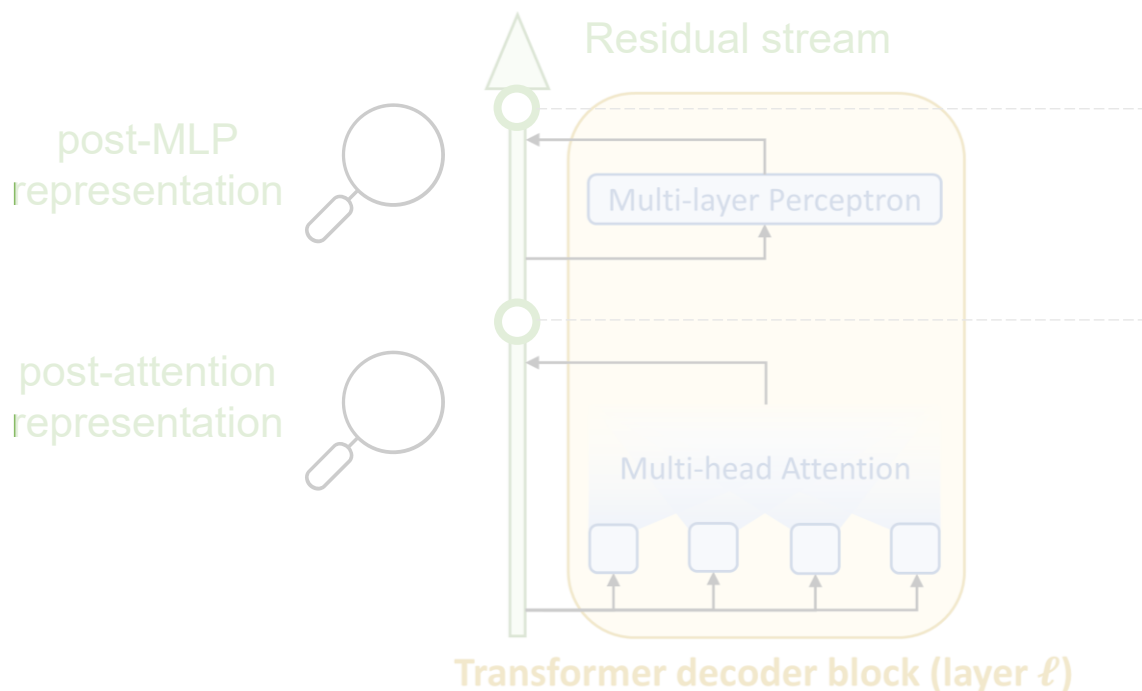
Short Reasoning Attention Head Distribution $\bar{C}_{short}^h$



Motivations

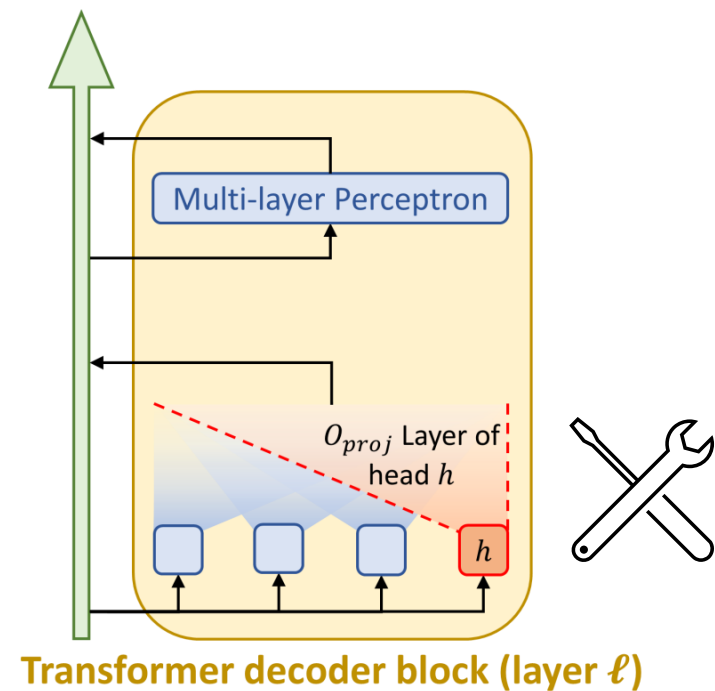
Question 1: Understanding

How is reasoning length encoded in or controlled by the hidden representation?



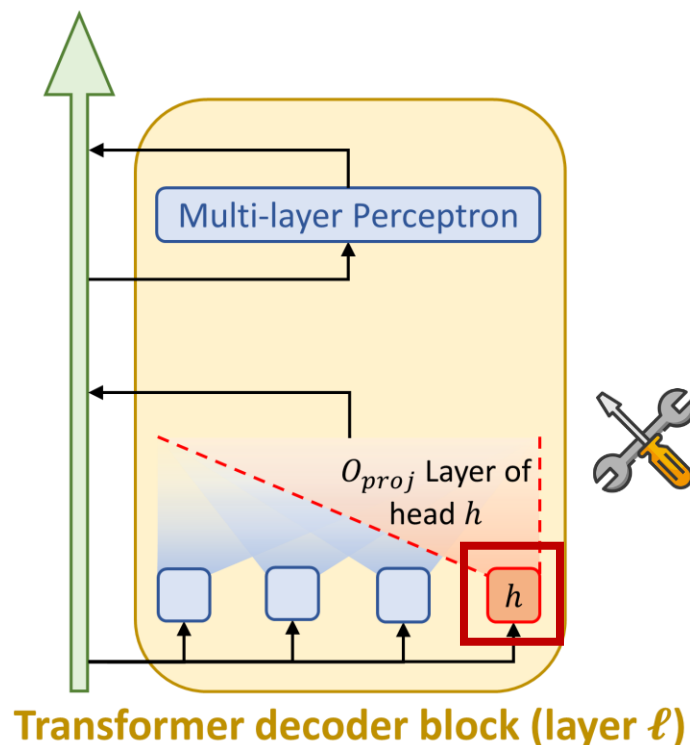
Question 2: Control

Could we mitigate the overly short reasoning by editing model parameters?



Sol #2: Control reasoning length via ThinkEdit

- Remove short reasoning component of an attn head $h \Leftrightarrow$ edit its output weight W_O^h



Weight Editing on W_O^h

$$W_O^h \leftarrow \underline{(I - ss^T)} W_O^h$$

Remove “**short reasoning**”
component



ThinkEdit

1. Identify shortest reasoning heads by projection amount $\bar{C}_{short}^h$

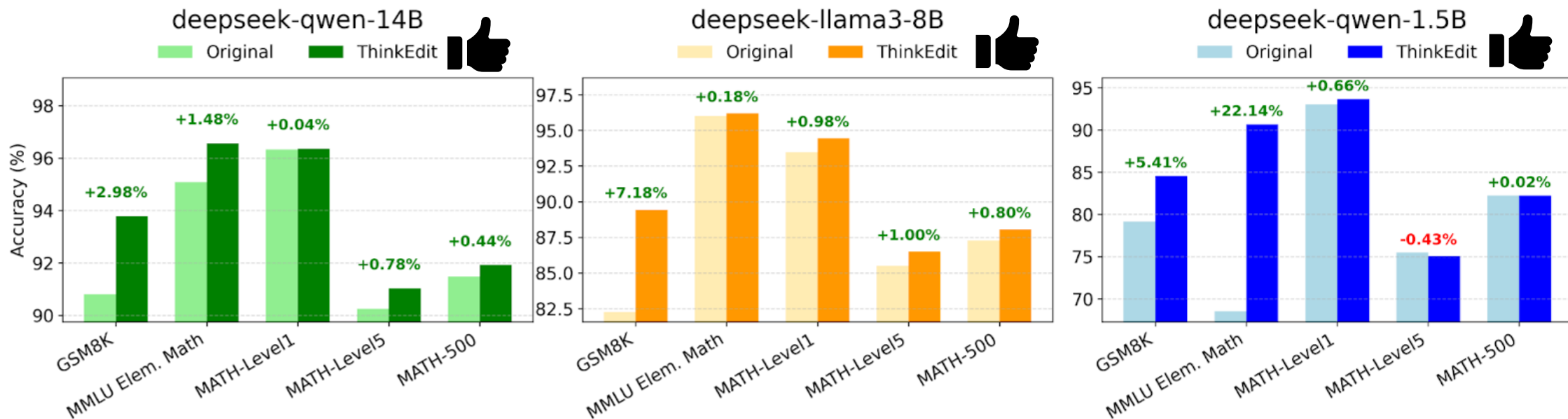
2. Edit top 4% shortest reasoning head with the correction $(I - ss^T)$
~0.2% of total parameters

Improved overall accuracy +3.3%!

Sol #2: Control reasoning length via ThinkEdit

■ Results #1: Overall accuracy

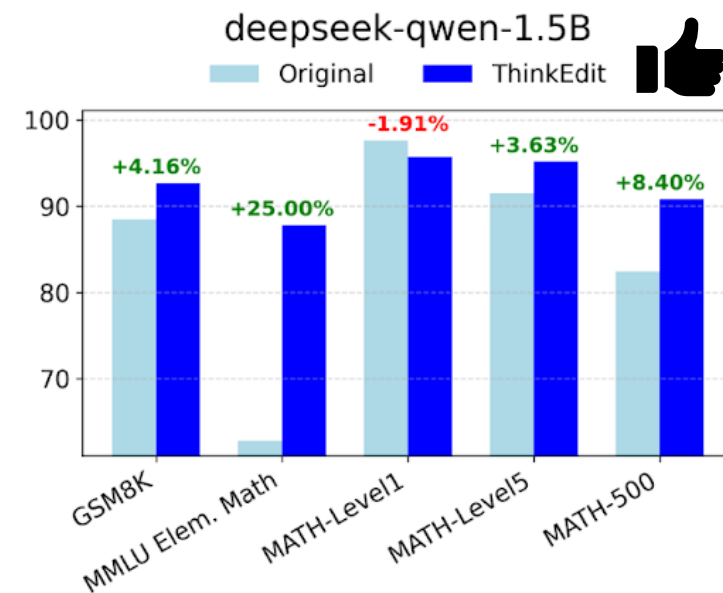
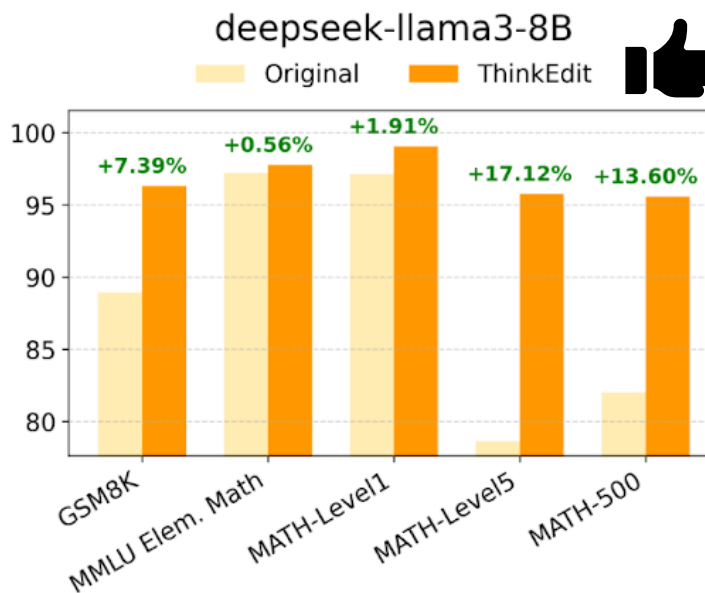
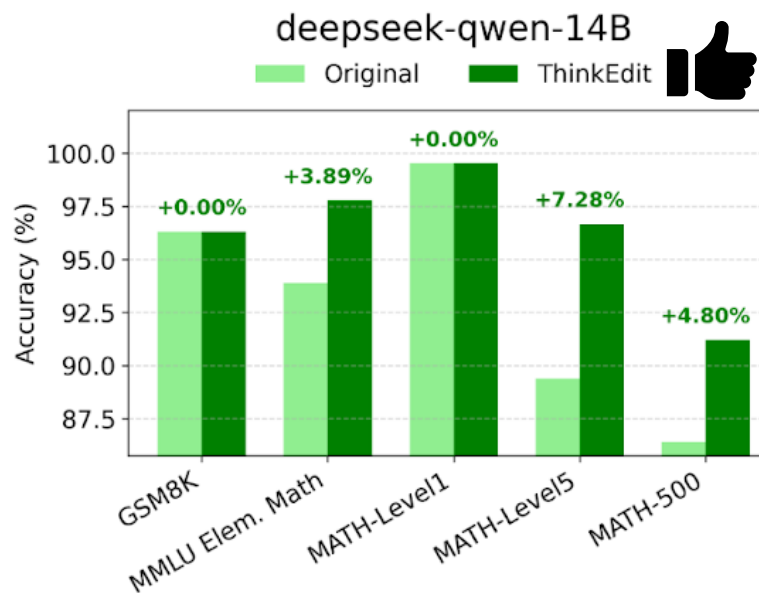
ThinkEdit models are higher than original models (**+3.34%** in average) across 5 standard math benchmarks and 3 different models



Sol #2: Control reasoning length via ThinkEdit

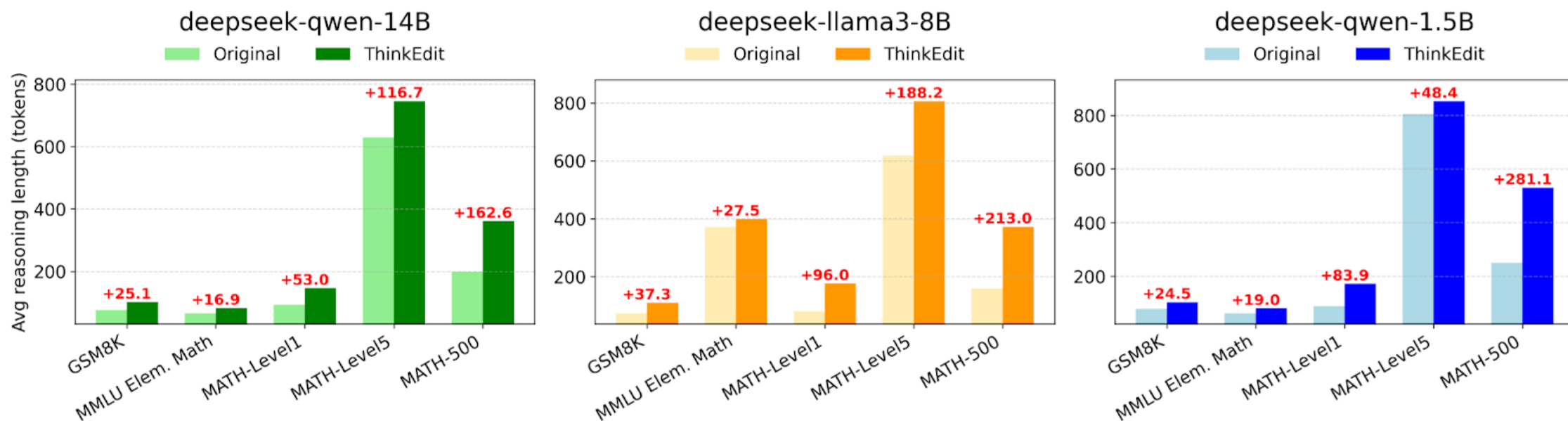
Results #2: Accuracy of top 5% shortest reasoning

ThinkEdit models significantly improves the performance when short reasoning occurs (**+6.39%** in average)



Sol #2: Control reasoning length via ThinkEdit

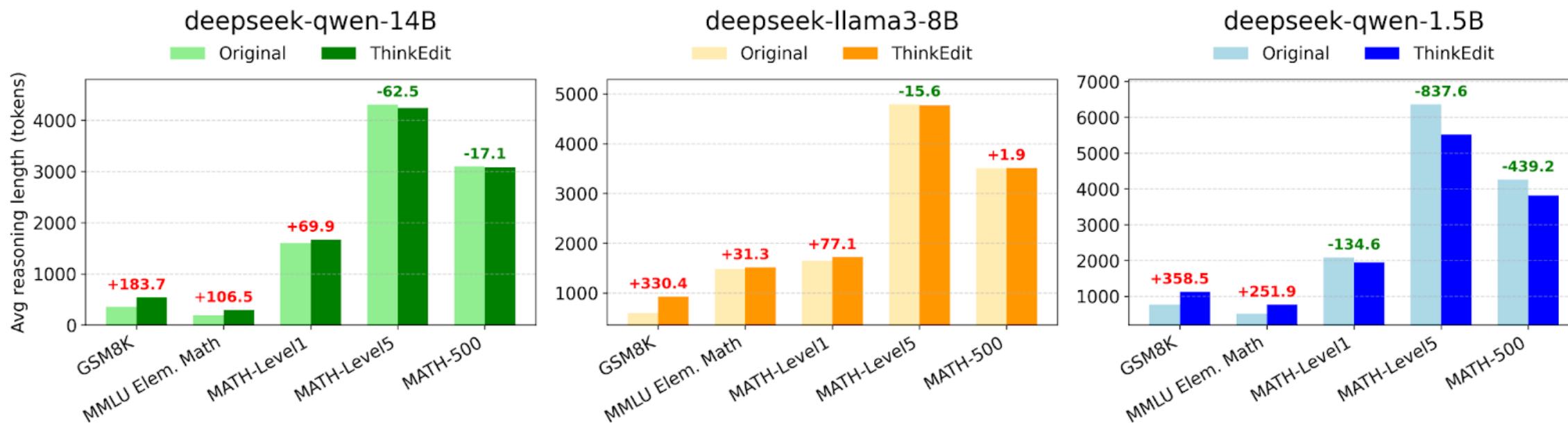
- Results #3: Reasoning length of top 5% shortest reasoning traces
ThinkEdit slightly increase the reasoning length **when it's too short**



Sol #2: Control reasoning length via ThinkEdit

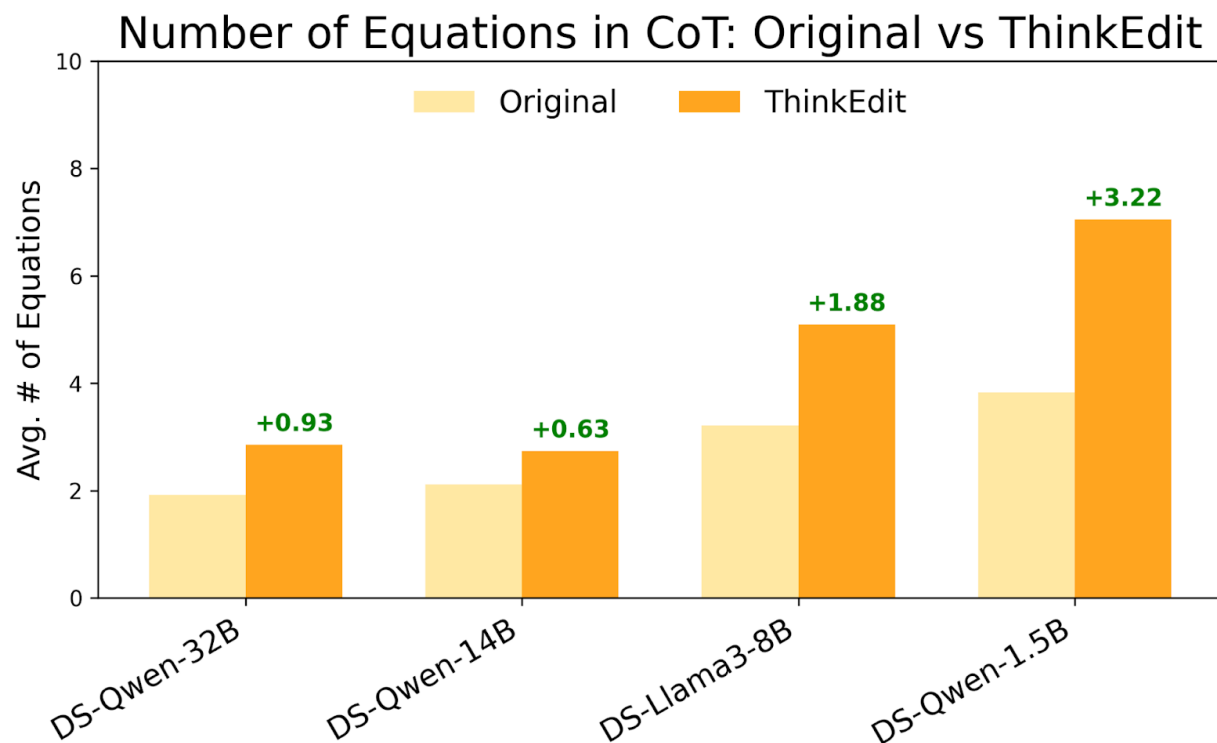
■ Results #4: Overall reasoning length

ThinkEdit **doesn't necessarily increase avg reasoning length**, suggesting it specifically targets overly short reasoning



Why ThinkEdit leads to better reasoning?

- Our analysis shows that ThinkEdit models write more equations during reasoning



Why ThinkEdit leads to better reasoning?

- Our analysis shows that ThinkEdit models write more equations during reasoning

Original

<think> First, I need to determine the total amount Elvira has spent on the computer accessories by adding up the costs of each item.

Next, I will subtract the total expenditure from her budget of €1500 to find out how much money she has remaining.

Finally, the remaining amount will be her money left for clothing.
</think>

...
Total spend = computer + scanner + CD burner + Printer

...
Final answer: \boxed{97}.

ThinkEdit

<think> First, I need to calculate the total amount Elvira has spent on the computer and its accessories.

She bought a computer for €1090, a scanner for €157, a CD burner for €74, and a printer for €102.

Adding those up: $1090 + 157 + 74 + 102 = 1423$ euros.

She has a budget of €1500, so subtracting the total spent from her budget: $1500 - 1423 = 77$ euros.

So, she has €77 left for her clothing. </think>

...
Final answer: \boxed{77}.

ThinkEdit outperforms prompting baselines

- Across 5 standard benchmarks, **ThinkEdit is more effective** in controlling reasoning performance by removing short reasoning traces

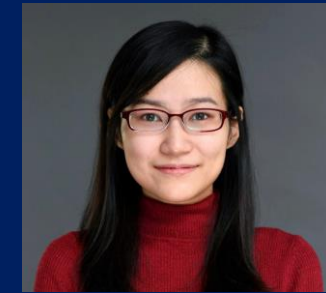
Prompting
baselines



Method	GSM8K	MMLU Elem. Math	MATH-Level1	MATH-Level5	MATH-500
Original model	82.26 ± 0.91	96.01 ± 0.62	93.46 ± 0.84	85.49 ± 0.83	87.26 ± 1.16
Please reason long enough	82.28 ± 2.22	95.85 ± 0.59	93.43 ± 0.49	84.95 ± 0.68	86.64 ± 1.20
Please reason step by step	81.47 ± 1.79	95.79 ± 0.67	93.82 ± 0.69	85.47 ± 0.79	86.92 ± 0.60
Append "Wait"	84.28 ± 0.64	95.93 ± 0.70	93.96 ± 0.55	85.33 ± 0.79	87.66 ± 1.26
ThinkEdit (4%)	89.44 ± 0.55	96.19 ± 0.73	94.44 ± 0.31	86.49 ± 0.54	88.06 ± 1.09

(DS-Llama-8B)

Summary



ThinkEdit has the following benefits:

1 **Reveal issues of** 2 **Identifying Reasoning** 3 **Effective &**
overly short reasoning length direction Interpretable Editing
Can we have a finer control on the reasoning mechanism?

- Across various Deepseek-distilled models
- Overly short reasoning heavily reduce task performance
- Middle layer representations govern reasoning length
- Edit just 0.2% of model weights to improve task performance
- +6.39% accuracy on short reasoning, +3.34% overall accuracy

★ **Website:** <https://lilywenglab.github.io/ThinkEdit>

★ **GitHub:** <https://github.com/Trustworthy-ML-Lab/ThinkEdit>

Challenge

- Modern Reasoning LLMs rely heavily on self-reflection steps in reasoning process

User: Calculate the prime factorization of 1304.

LLM:



Okay, so I need to figure out the prime factorization of 1304...

First off, what is 1304? It's an even number...

Is 163 a prime number? Hmm, let me check...

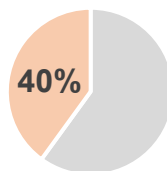
Wait, that seems straightforward, maybe we can calculate ...

⋮

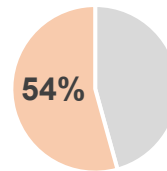
Self-Reflection steps

~25%-50% token overhead due to self-reflection

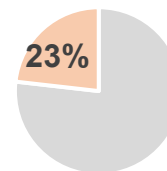
DS-Qwen-1.5B



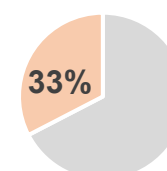
Qwen3-4B



DS-Llama-8B



DS-Qwen-7B



■ non-Reflection
■ Reflection

(GSM8k dataset)

Is self-reflection necessary?

If not, we can have



Reflection Control (ReflCtrl)

- We propose **ReflCtrl** to understand and control LLM self-reflections!

① Segment response



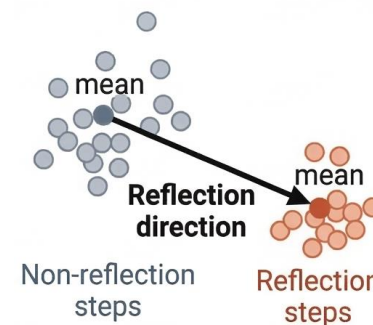
Split at new line (`\n\n`)
into chunks

② Identify reflection steps



Detect reflection keywords
in the chunks
“wait”, “let me check”, “double-
check”, “alternatively”, ...

③ Extract reflection direction d_l



$$d_l^{\{\text{attn}, \text{mlp}\}} = \frac{1}{|R|} \sum_{s \in R} z_l^{\{\text{attn}, \text{mlp}\}}(s) - \frac{1}{|NR|} \sum_{s \in NR} z_l^{\{\text{attn}, \text{mlp}\}}(s)$$

(extract at the 1st token of the step s)

Reflection Control (ReflCtrl)

Question 1: Understanding

When does the model initiate reflection during its reasoning process?



Model reflects when it has **high** uncertainty



Evidence 1: Token entropy

Compare the token entropy right before model's reflection/non-reflection

Pooling	Pre-Reflect	Pre-Non-Reflect
Mean	0.734 ± 0.332	0.599 ± 0.366
Max	2.401 ± 0.647	2.028 ± 0.789

Token entropy is **higher** before model's reflections

Evidence 2: Correctness prediction

Use reflection direction as features to predict answer correctness

Model	final layer embedding		reflection direction	
	AUROC	F1	AUROC	F1
deepseek-llama-8b	0.736	0.946	0.772	0.948
qwq-32b	0.555	0.636	0.564	0.839
deepseek-qwen-14b	0.716	0.929	0.850	0.976

Reflection direction encodes uncertainty info, can better predict correctness

Reflection Control (ReflCtrl)



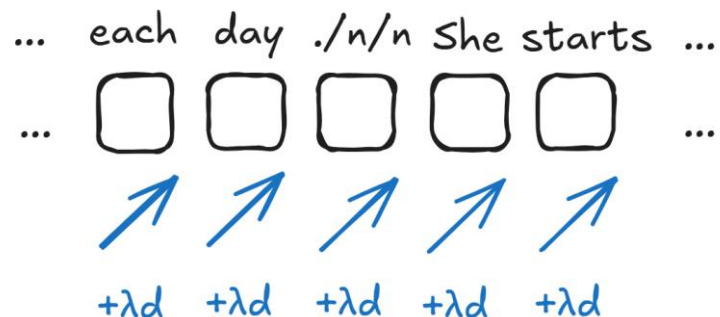
Question 2: Control

How does reflection affect reasoning performance?



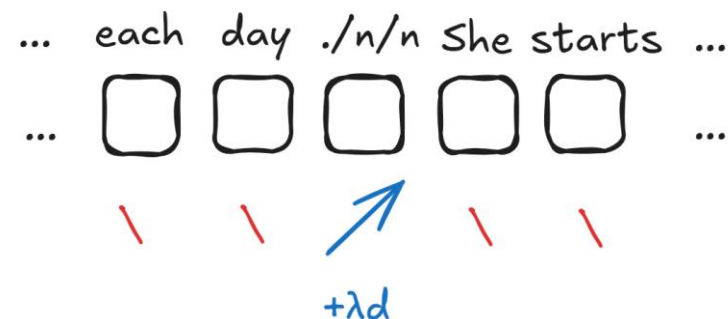
Reflections can be reduced: save 30%+ tokens at similar performance

Standard steering: $z_{l, \text{intv}}^{\{\text{attn}, \text{mlp}\}} = z_l^{\{\text{attn}, \text{mlp}\}} + \lambda d_l^{\{\text{attn}, \text{mlp}\}}$



- Apply steering at every token
- Introduce unnecessary noise (hurt performance) 😞

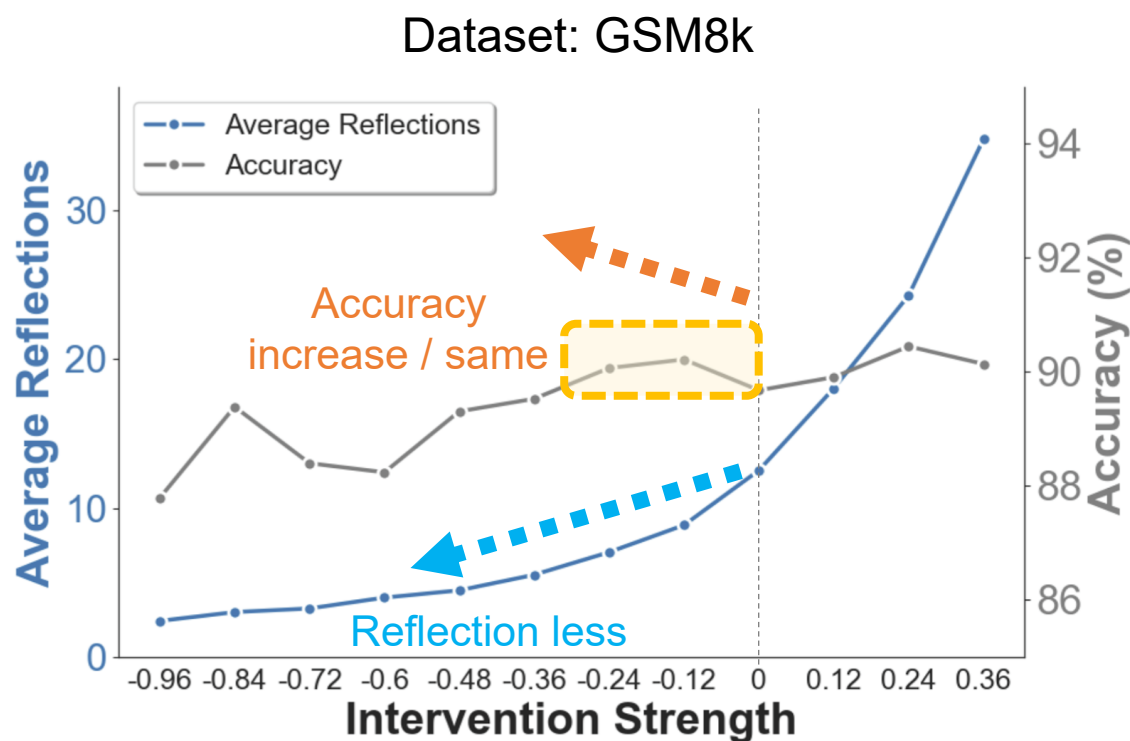
Key idea: stepwise steering (Ours)



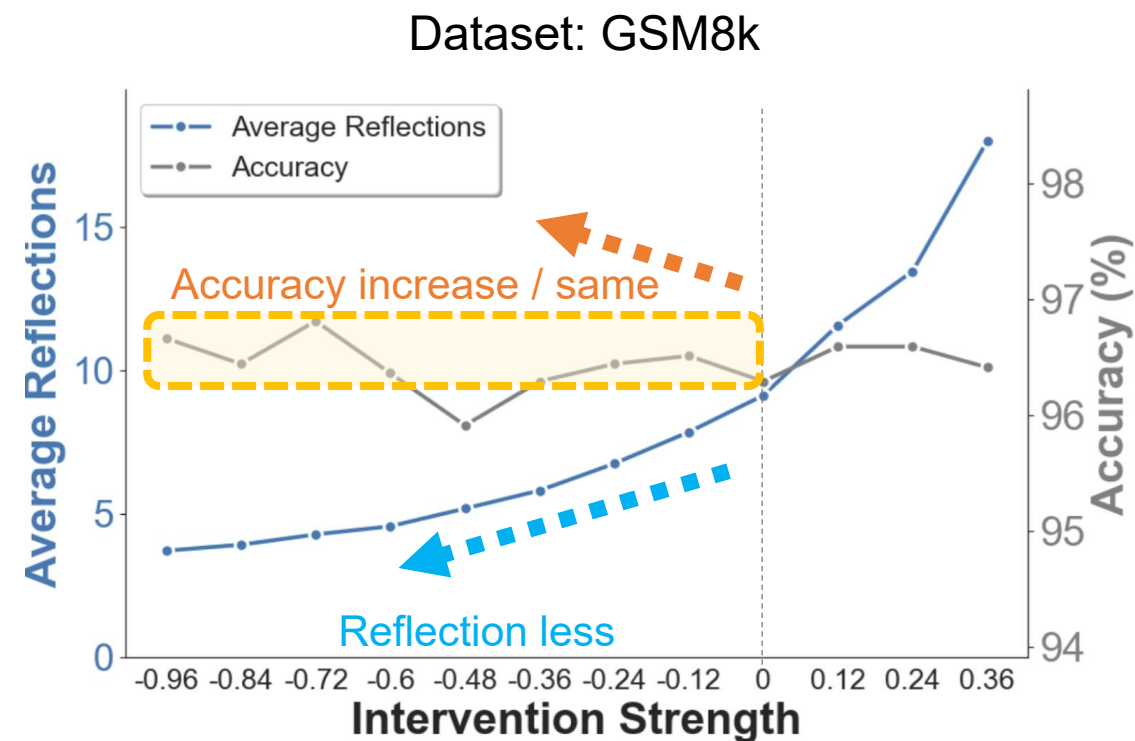
- Only steer at the start of reasoning step
- Minimal impact to normal generation

Result #1: Efficient Reasoning

- **ReflCtrl** effectively steers model to **reflect less** without hurting performance!



(a) DeepSeek-R1 Llama 8b



(b) QwQ-32B

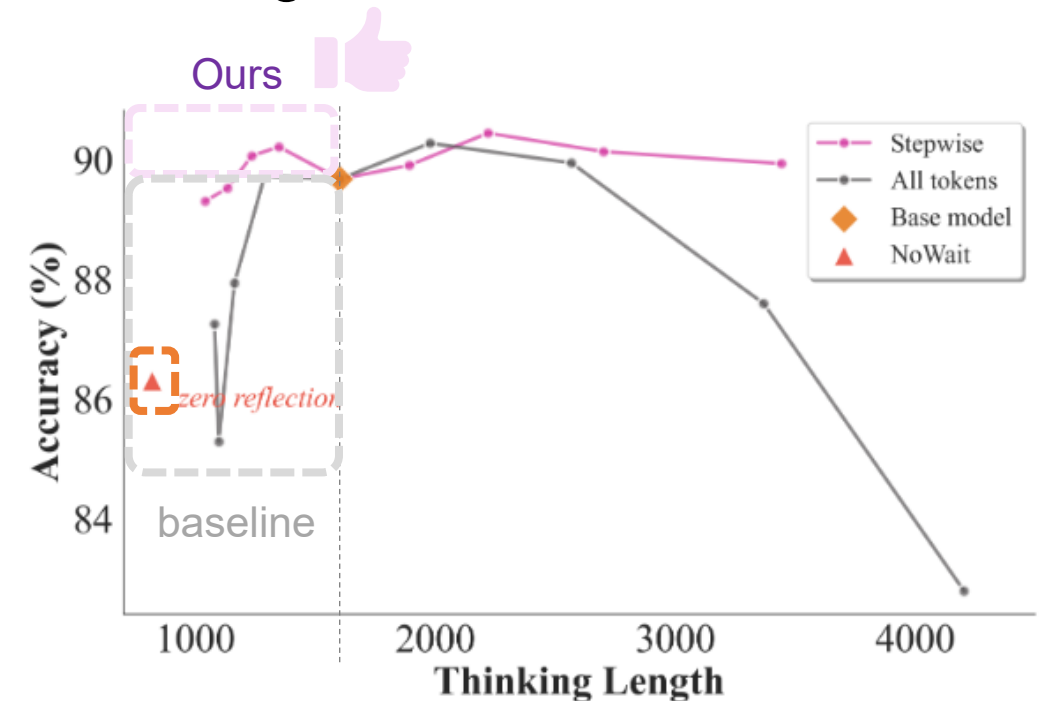
Result #2: Efficient Reasoning

- **RefICtrl** can save up to 43% tokens on math tasks while keeping performance, and our stepwise steering outperforms standard steering baselines

Dataset	Model	Metric	Intervention Strength λ			
			-0.96	-0.48	0	+0.48
GSM8k	DS-Llama-8B	Acc (%) $\uparrow$	88.34 (-1.75pp)	89.46 (-0.63pp)	90.09	90.25 (+0.16pp)
		Tokens $\downarrow$	821.0 (-48.5%)	1032.6 (-35.3%)	1595.7	3577.1 (+124.2%)
	QwQ-32B	Acc (%) $\uparrow$	96.36 (-0.14pp)	96.50 (+0.00pp)	96.50	96.44 (-0.06pp)
		Tokens $\downarrow$	1006.7 (-32.4%)	1162.5 (-21.9%)	1488.6	2256.9 (+51.6%)
	DS-Qwen-14B	Acc (%) $\uparrow$	95.07 (-0.08pp)	95.15 (+0.00pp)	95.15	94.84 (-0.31pp)
		Tokens $\downarrow$	747.8 (-43.2%)	880.2 (-33.1%)	1315.9	3746.4 (+184.7%)
MATH-500	DS-Llama-8B	Acc (%) $\uparrow$	82.14 (-3.84pp)	84.46 (-1.52pp)	85.98	86.90 (+0.92pp)
		Tokens $\downarrow$	2738.1 (-31.6%)	3123.8 (-21.9%)	4000.7	6017.8 (+50.4%)
	QwQ-32B	Acc (%) $\uparrow$	92.72 (-0.34pp)	92.58 (-0.48pp)	93.06	93.08 (+0.02pp)
		Tokens $\downarrow$	2992.9 (-20.9%)	3253.4 (-14.1%)	3786.0	5028.9 (+32.8%)
	DS-Qwen-14B	Acc (%) $\uparrow$	89.22 (-2.22pp)	90.18 (-1.26pp)	91.44	91.86 (+0.42pp)
		Tokens $\downarrow$	2247.1 (-32.2%)	2534.7 (-23.5%)	3315.3	5789.0 (+74.6%)



Efficient reasoning!



Reflection can still be helpful, but we don't need too many!

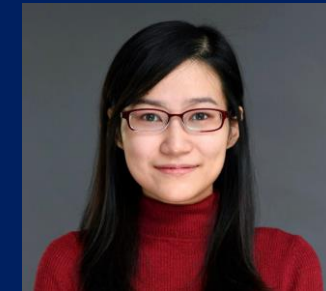
Result #3: Efficient Reasoning

- **ReflCtrl** can save up to 34% tokens on MMLU while keeping performance!

Subset	Model	Metric	Intervention Strength λ			
			-0.96	-0.48	0	+0.48
Professional Accounting	DS-Llama-8B	Acc (%) $\uparrow$	50.1 (-6.4pp)	53.4 (-3.1pp)	56.5	57.3 (+0.8pp)
		Tokens $\downarrow$	1453.6 (-30.7%)	1668.8 (-20.4%)	2097.5	2807.7 (+33.9%)
	DS-Qwen-14B	Acc (%) $\uparrow$	78.5 (+0.7pp)	76.8 (-1.0pp)	77.8	77.6 (-0.2pp)
		Tokens $\downarrow$	983.9 (-33.6%)	1103.1 (-25.6%)	1482.1	2470.1 (+66.7%)
	QwQ-32B	Acc (%) $\uparrow$	89.3 (+0.8pp)	89.5 (+1.0pp)	88.5	89.2 (+0.7pp)
		Tokens $\downarrow$	1231.2 (-25.3%)	1313.7 (-20.3%)	1648.0	2234.3 (+35.6%)
HS Computer Science	DS-Llama-8B	Acc (%) $\uparrow$	79.6 (-7.7pp)	82.7 (-4.6pp)	87.3	88.0 (+0.7pp)
		Tokens $\downarrow$	1016.1 (-25.6%)	1157.9 (-15.2%)	1365.4	1970.4 (+44.3%)
	DS-Qwen-14B	Acc (%) $\uparrow$	95.2 (+0.2pp)	95.4 (+0.4pp)	95.0	94.8 (-0.2pp)
		Tokens $\downarrow$	711.9 (-23.7%)	787.9 (-15.6%)	933.5	1498.7 (+60.5%)
	QwQ-32B	Acc (%) $\uparrow$	96.6 (-0.1pp)	96.2 (-0.5pp)	96.7	97.0 (+0.3pp)
		Tokens $\downarrow$	771.6 (-11.4%)	741.9 (-14.8%)	871.0	1004.7 (+15.4%)
Formal Logic	DS-Llama-8B	Acc (%) $\uparrow$	60.5 (-1.6pp)	61.0 (-1.1pp)	62.1	62.7 (+0.6pp)
		Tokens $\downarrow$	2266.5 (-32.9%)	2586.9 (-23.4%)	3378.3	4553.5 (+34.8%)
	DS-Qwen-14B	Acc (%) $\uparrow$	91.8 (-0.8pp)	92.2 (-0.4pp)	92.6	92.8 (+0.2pp)
		Tokens $\downarrow$	1287.2 (-31.9%)	1440.1 (-23.9%)	1891.4	3196.5 (+69.0%)
	QwQ-32B	Acc (%) $\uparrow$	96.3 (+0.6pp)	95.5 (-0.2pp)	95.7	96.0 (+0.3pp)
		Tokens $\downarrow$	1481.4 (-13.7%)	1447.8 (-15.7%)	1716.6	2175.6 (+26.7%)



Summary



ReflCtrl has the following benefits:

- 1 Provide insights on LLM reflection
- 2 Easy Control + Efficient Reasoning
- 3 Flexible

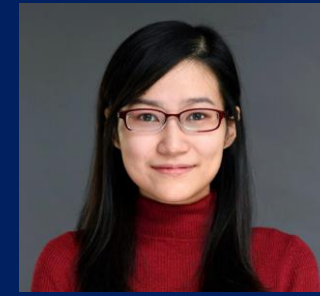
Can we have a principled way to steer the LLMs?

- Reflection direction encodes uncertainty information
- Model reflects when it's uncertain
- Save 20-40% token costs by reducing reflections
- Reduce unnecessary reflection while preserving performance
- Works for any architecture
- Creates a useful tool for future investigation of LLMs

★ **Website:** <https://lilywenglab.github.io/ReflCtrl/>

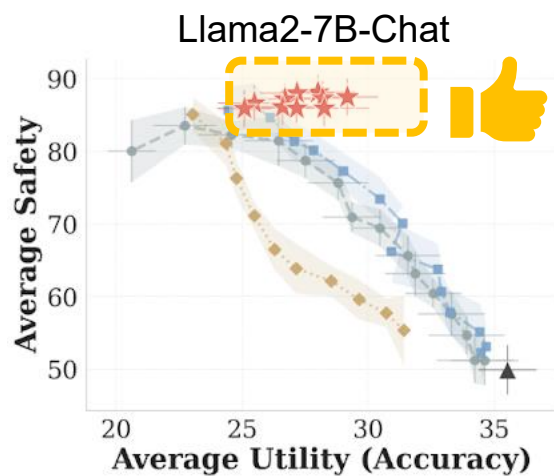
★ **GitHub:** <https://github.com/Trustworthy-ML-Lab/ReflCtrl>

Steer2Edit

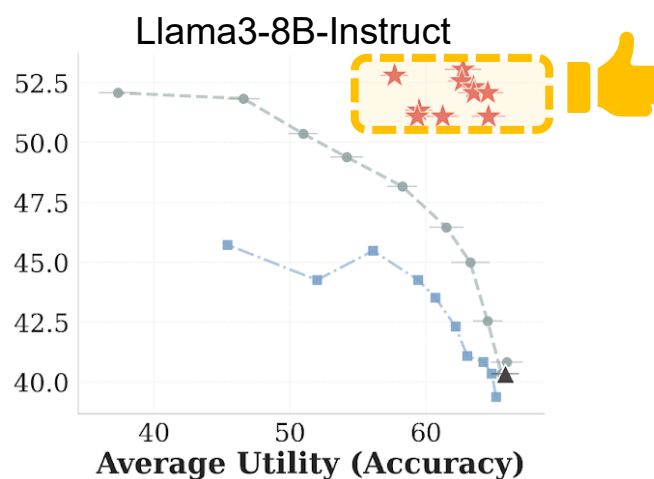


- A theoretical grounded, training-free framework to convert steering vectors into model's weight editing and achieve **better trade-offs** on utility and target attribute

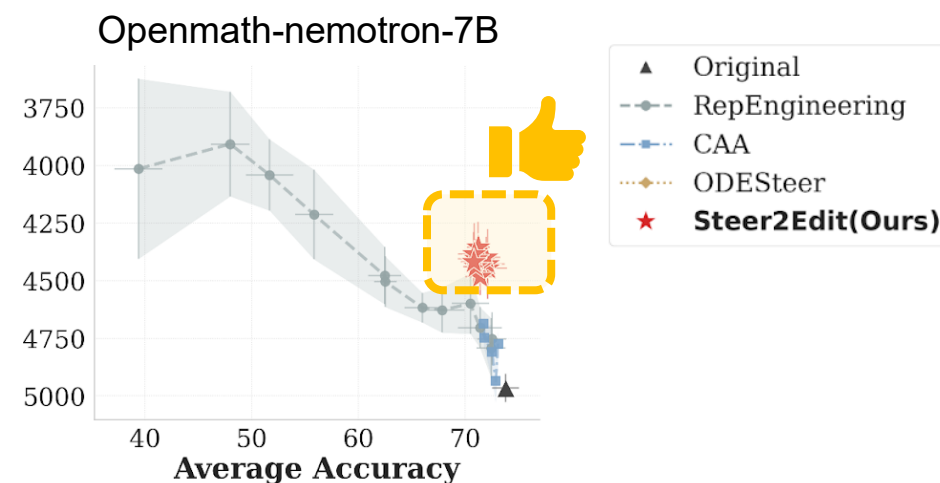
Attribute 1:
Safety Alignment



Attribute 2:
Truthfulness



Attribute 3:
Efficient Reasoning



This Talk - Part 3

1. Interpretability Tools

Develop **automated** interpretability techniques to **understand** DNNs



1

[CLIP-Dissect](#)
[ICLR'23 spotlight]

1st fully automation with higher quality, 10-200x faster

[Auto Neuron Exp](#)
[NeurIPS'23]

2x faster than OpenAI's result with higher quality

[Linear Explanation](#)
[ICML'24]

Provide polysemantic explanations

[Audio-Net-Dissect](#)
[ICML'24]

Neuron interpretability tool for audio networks

[Describe-n-Dissect](#)
[TMLR'25]

Generative Natural Language Explanation

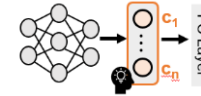
2

[ThinkEdit](#), [Reflectrl](#)
[EMNLP'25, arxiv'26]

Steer & control reasoning process of LRMs

2. Interpretable DNN by design

Develop systematic method to **learn** intrinsically-interpretable DNNs



3

[Label-free CBM](#)
[ICLR'23]

1st CBM on Imagenet + automation + high acc

[CB-LLMs](#)
[ICLR'25]

1st interpretable LLM + high steerability & scalability

[VLG-CBM](#)
[NeurIPS'24]

Faithful CBM + 1st theoretical analysis & fair comparison

[CB-AE](#), [CB-SAE](#)
[CVPR'25, CVPR'26]

1st post-hoc interpretable generative model & VLM

Our papers are available at

<https://lilywenglab.github.io/#interpretable-deep-learning>

3. Evaluation

Rigorous comparison on explanations, metrics, CBMs

[Neuron Eval](#)
[ICML'25]

1st unified evaluation framework + sanity checks

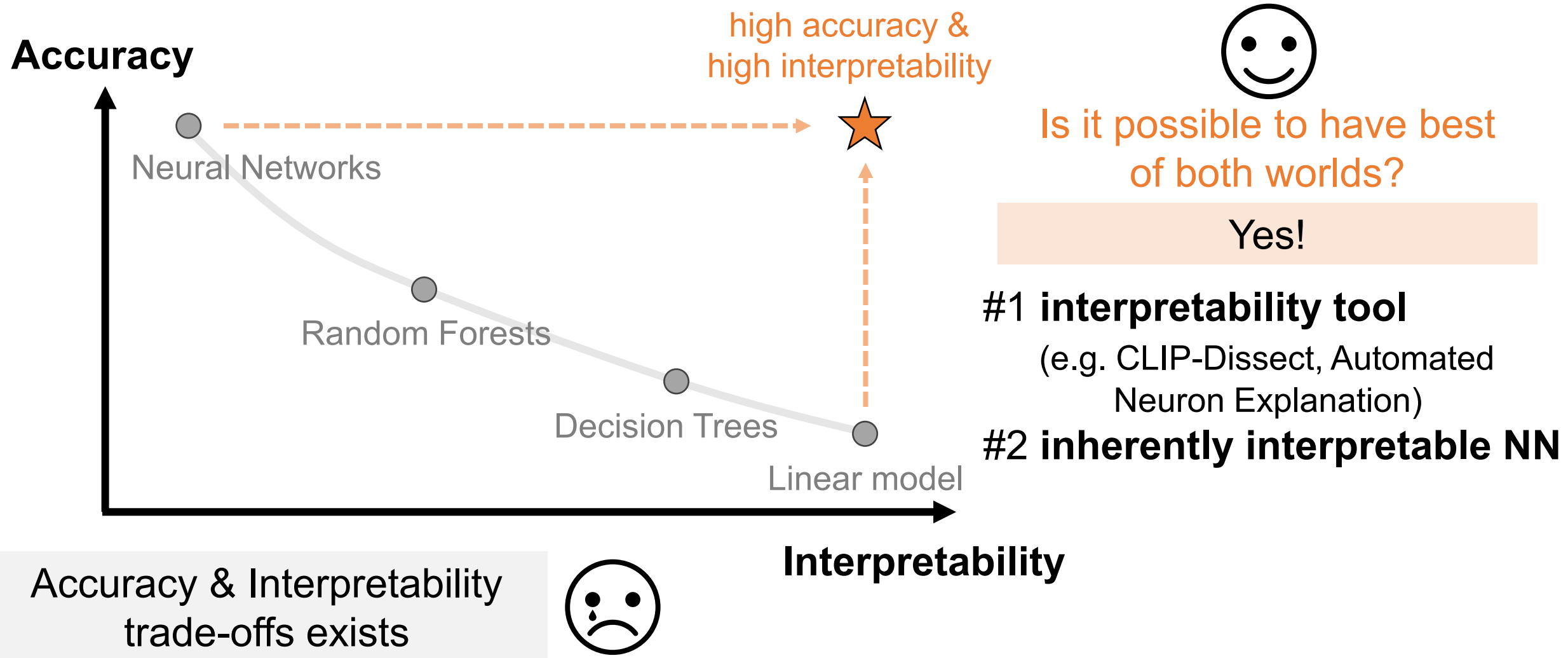
[Rethink human study](#)
[CVPR'26]

Wholistic comparison + enable 40x reduced cost \$\$!

[VLG-CBM](#)
[NeurIPS'24]

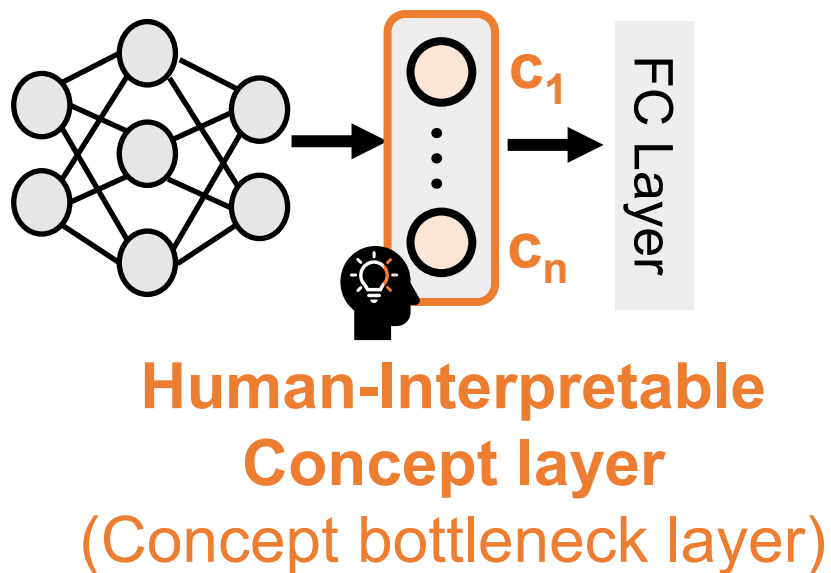
1st metric to compare between CBMs

Challenge



What is an interpretable neural network?

Given a DNN backbone

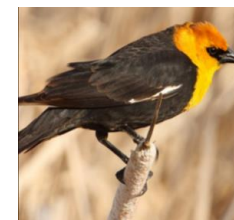


What are the Concepts c_i ?

- Task: arthritis grade
 c_i = bone spurs, narrow joint space, ...



- Task: bird species
 c_i = wing color, beak length, ...



- Task: sentiment of review
 c_i = overpriced, clean, ...

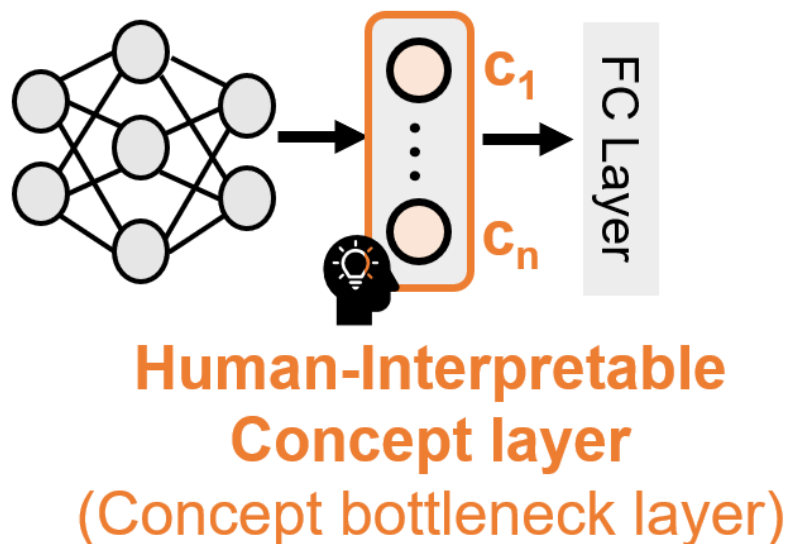
La Clochette



If you don't have Too Good To Go, I would really suggest downloading the app. I ended up getting...

Prior work

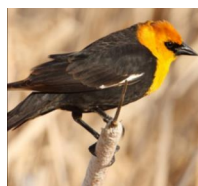
- To train a CBM, early work utilize *concept labels / annotations*



Research efforts

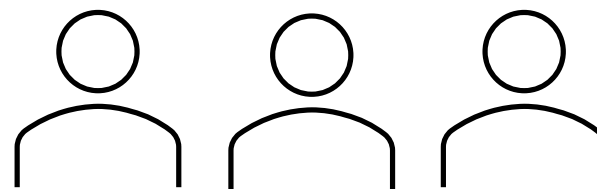
- Koh et al., ICML 2020
- Yuksekgonul et al., ICLR 2023
- $\vdots$

could be expensive / less flexible to get concept data



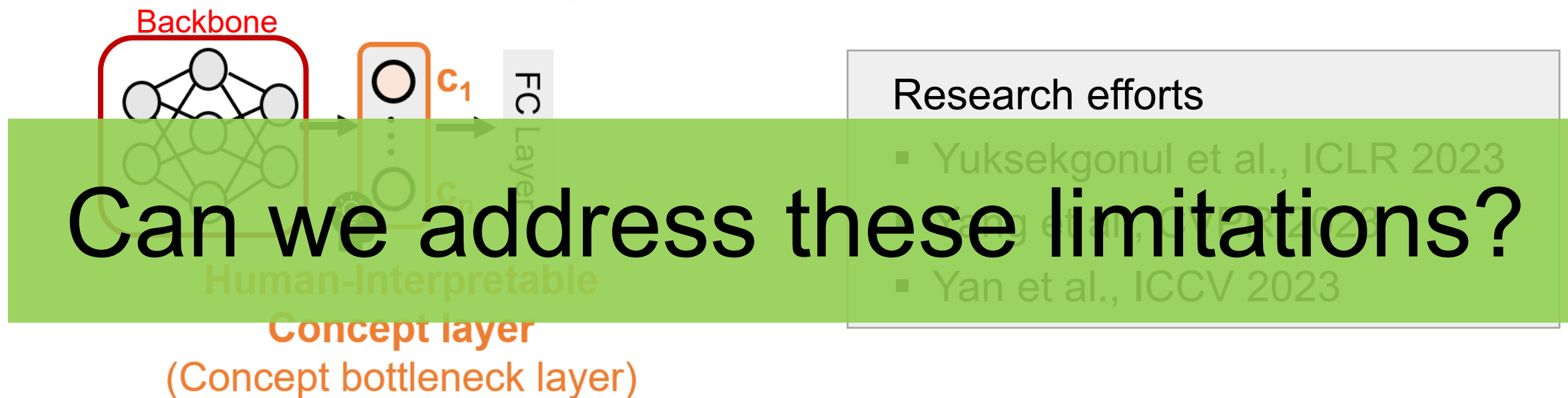
wing color = black
beak length = short

Concept labeled data



Challenges

- Recent work to leverage CLIP architecture to help



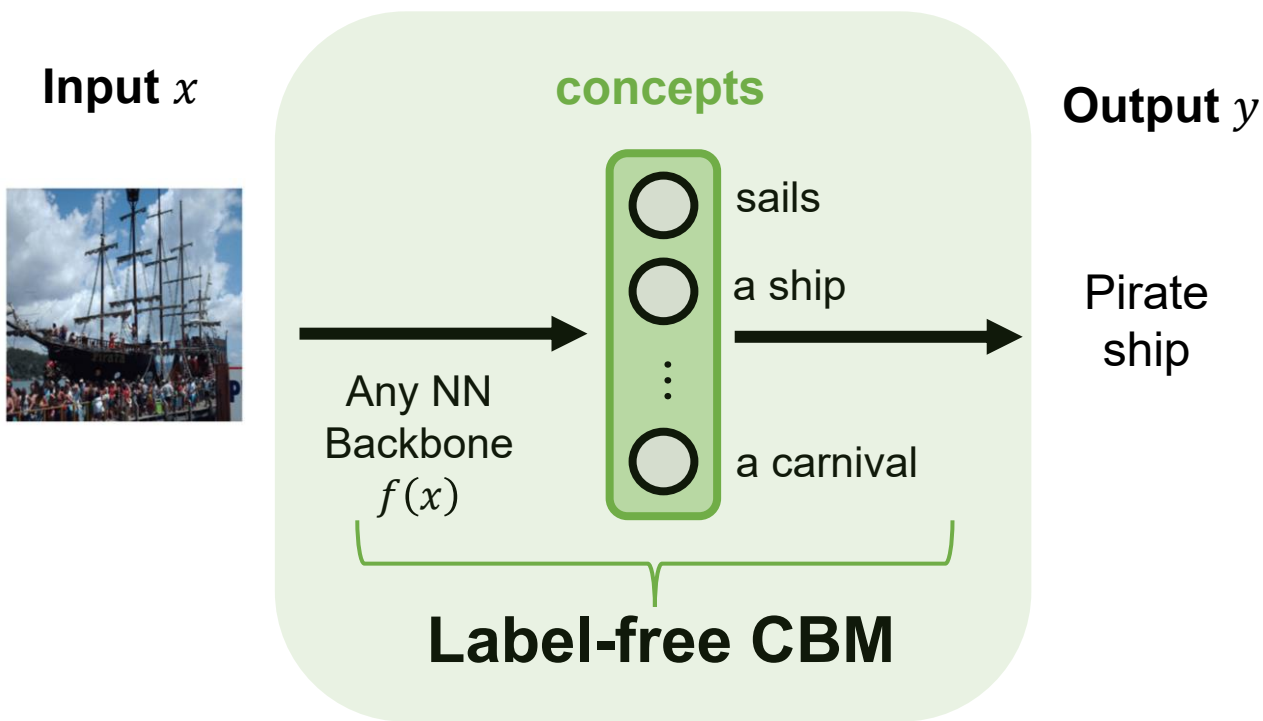
Backbone architecture generally has to be CLIP encoder

Accuracy and interpretability can have large trade-offs

Label-Free CBM



Label-free CBM can transform any NN backbone into an interpretable model with 4 desirable benefits:

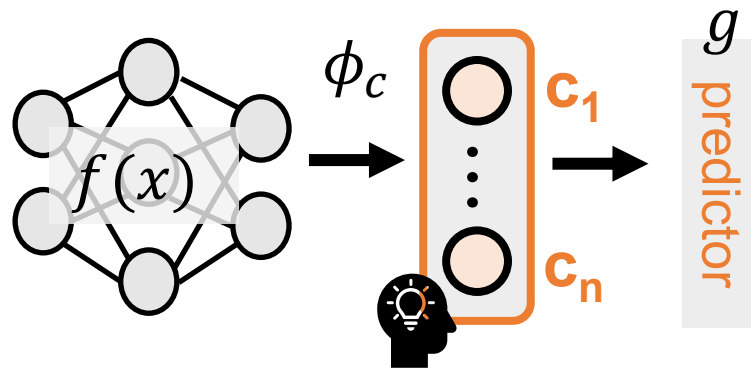


Benefits

- 1 Automated
Use GPT-3 to generate concept set
- 2 Flexible & Label-free
*Use **CLIP-Dissect** to align concepts with any backbone, no need for labeled concept data*
- 3 Scalable & Efficient
First CBM trained on ImageNet
- 4 High performance
Retain high accuracy + interpretability

Label-free CBM: Key Ideas

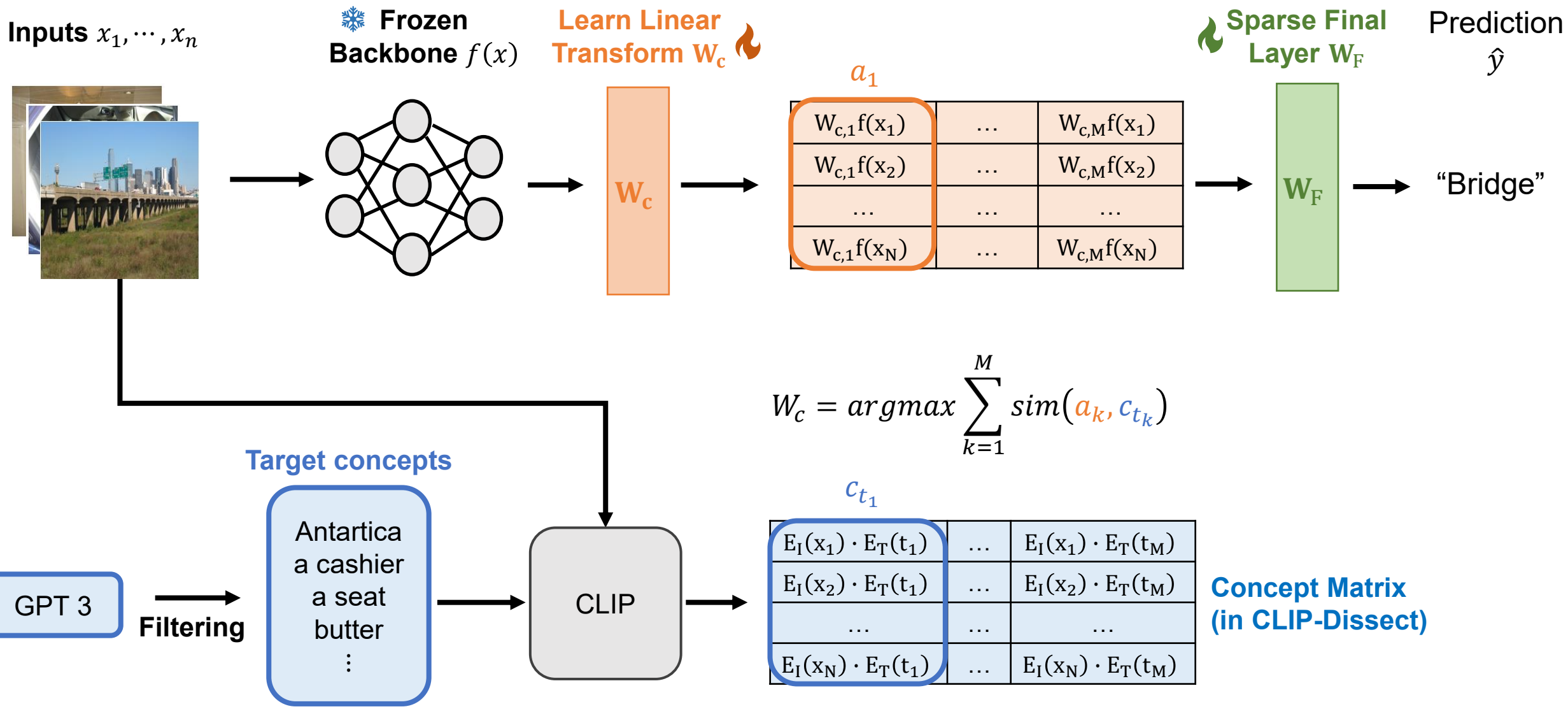
- How to transform any black-box NN backbone into an interpretable NN?



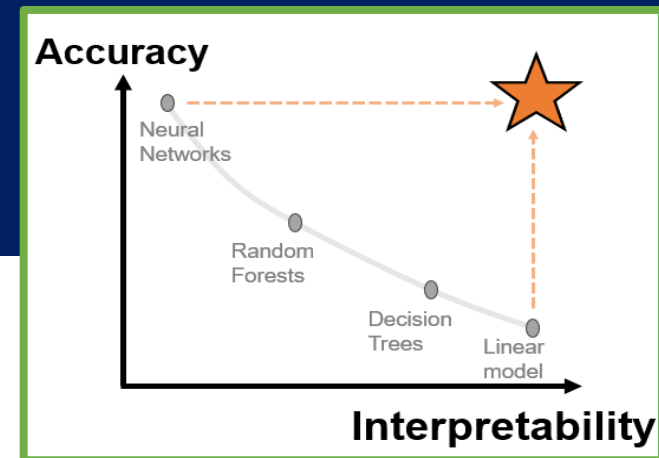
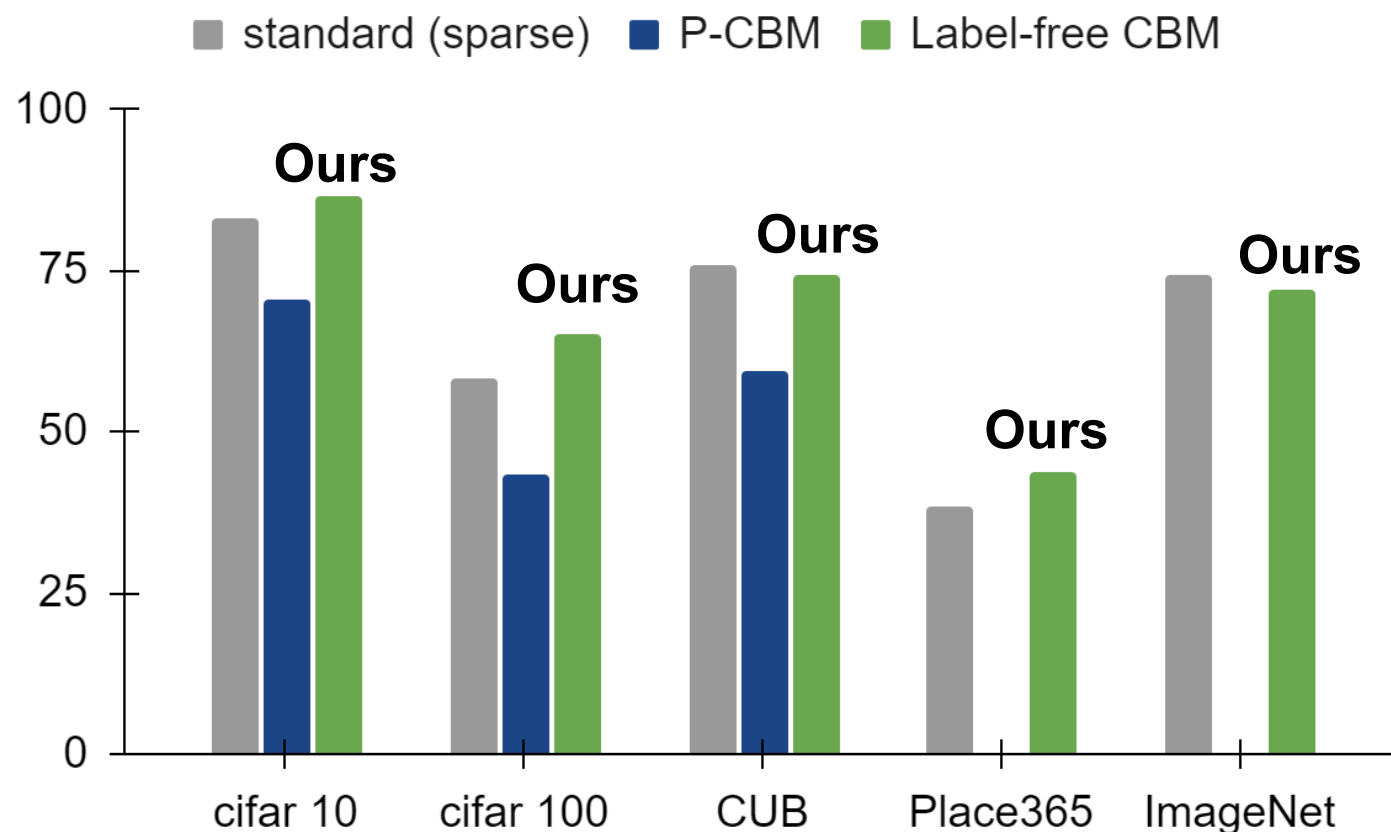
Final interpretable model:
 $g(\phi_c(f(x)))$

- Learn the concept mapping ϕ_c
 - Q1: How to generate concepts?
 - Q2: How to learn / enforce the concepts?
- Learn the predictor g
 - Q3: How to learn a predictor based on the interpretable representations?

Label-Free CBM: Method



Result #1: High accuracy



- **Our method** consistently retain high accuracy and scale to large dataset (e.g. ImageNet)
- **Baselines** trade-off accuracy and not scale to large dataset



Label-free CBM: Benefits of Interpretability

① **Explainable decision rules**

② **Explainable individual decisions**

③ **Correct model mistakes**

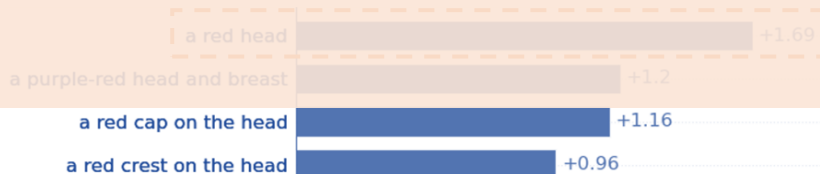
Interpretable concepts

- a bright orange breast
- a carrot
- bright orange color
- layers of fruit
- a checkout counter
- a hop
- a piece of fruit
- a piece of fruit
- citrus fruit
- a citrus juicer
- a lime
- lime juice
- yellow color
- a yellow center
- a white or yellow background
- a mandolin

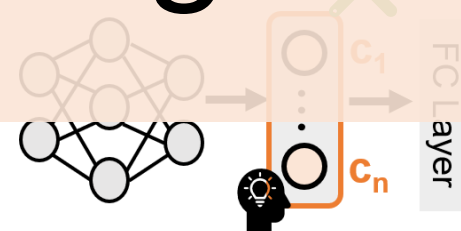
What are remaining challenges?



Prediction
Red headed Woodpecker



(brown color)
Cricket insect
(Incorrect!)



Edit weight

orange = “citrus fruit” + “bright orange color”
lemon = “citrus fruit” + “lime” + “yellow color”

“a red head” makes this image a
Red-headed Woodpecker

“a green color” → “Grasshopper”: 0 → 0.197
“a green color” → “Cricket insect”: 0 → -0.197

Remaining Challenges

- LLM-based concepts are easy to obtain, but can have some potential issues

- **Challenge #1: Inaccurate concept prediction**

Inaccurate or wrong explanations which do not match the input images



Explanations of why this bird is a *painted bunting*:

1. Color - Blue head, olive back, yellow underparts (101.08)
2. grayish head, back, wings and tail with blue highlights (94.03)
3. bright blue and orange plumage (91.44)
4. large red bill with a slightly hooked tip (89.09)
5. distinctive white throat (-76.91)

Sum of other concepts (-34.89)

(LM4CV)

1. NOT a brown and white color scheme (1.77)
2. NOT white and black coloration (1.66)
3. iridescent feathers (1.37)
4. NOT a black bib with white stripes (1.29)
5. NOT a black and white color scheme (1.01)

Sum of other concepts (4.22)

(LF-CBM)

- **Challenge #2: Information Leakage**

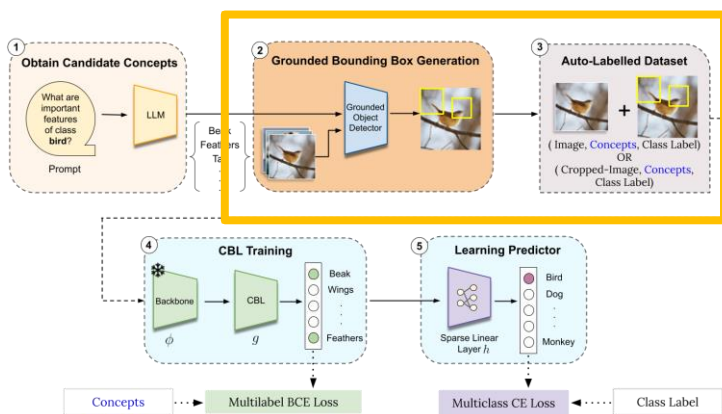
The concept prediction encodes unintended information for downstream tasks, even if the concepts are irrelevant to the task (e.g. random concepts can still get high acc.)

Vision-Language Guided (VLG) CBM



We propose VLG-CBM to address the above challenges!

1 Training faithful CBM



Ensure visual concept alignment using grounding detectors

2 1st theoretical analysis

when # of concept $\uparrow$, error $\downarrow$

$$E(k) \leq \begin{cases} \lambda_{max}(1 - \frac{k}{d}) \|w\|_2^2, & k < d; \\ 0, & k \geq d. \end{cases}$$

approx. error

weight vector of linear classifier

of concepts

embedding dim of backbone

Explain information leakage with random concepts + final dense layer

3 1st CBM evaluation metric: ANEC

NEC: measures avg # of concepts being used for prediction

$$NEC(W_F) = \frac{1}{C} \sum_{i=1}^C \sum_{j=1}^k \mathbf{1}\{(W_F)_{ij} \neq 0\}$$

of classes

of concepts

final weight matrix of the predictor

CBM should be compared at same number of effective concepts (NEC)

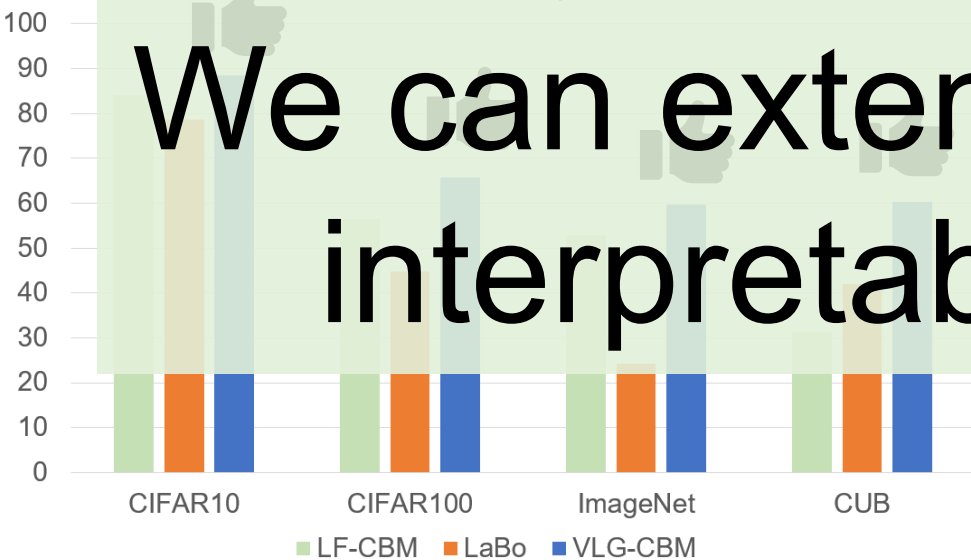
- ANEC-k: Accuracy at NEC=k.
- ANEC-avg: Average accuracy under different NECs.

VLG-CBM: Compare CBMs using ANEC

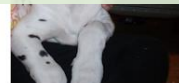
VLG-CBM outperforms strongest baseline by **12.4%** (average)

Also provides higher quality of explanations

ANEC-5 Comparison



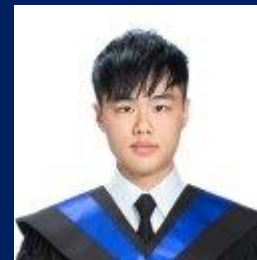
We can extend the idea to construct interpretable LLMs and VLMs!



Our method provide accurate explanations while prior work provide inaccurate/wrong/less useful explanations

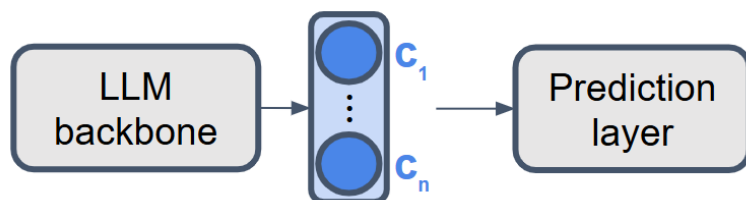
Method	Explanation	Score
Our Method (VLG-CBM)	1. a big dog (0.89) 2. a black and white color scheme (1.01) 3. short stout bill (0.01) 4. droopy lips and ears (0.09) Sum of other concepts (0.00)	8.14
LaBo	1. a big dog (0.89) 2. a black and white color scheme (1.01) 3. short stout bill (0.01) 4. NOT a black and white color scheme (1.01) Sum of other concepts (-0.22)	8.14
LF-CBM	1. a big dog (0.89) 2. a black and white color scheme (1.01) 3. short stout bill (0.01) 4. NOT a black and white color scheme (1.01) Sum of other concepts (-0.22)	8.14

CB-LLM: Interpretable LLMs



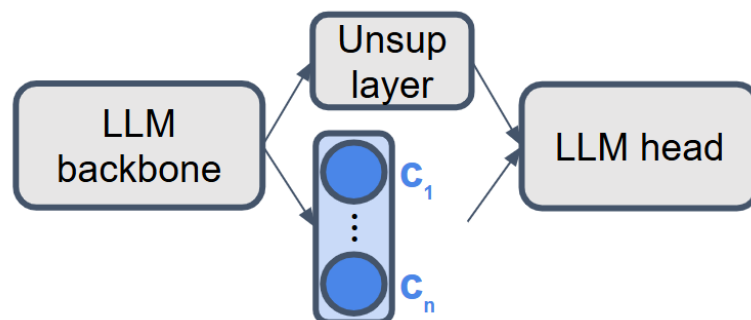
- **Concept-Bottleneck LLM (CB-LLM):** We can make LLM interpretable!

1. Text Classification:



- ✓ More **efficient**, up to 10x faster
- ✓ More **scalable**, up to 50x larger benchmark
- ✓ More **faithful** (interpretability)

2. Text Generation:



- ✓ **1st** interpretable autoregressive LLM / Chatbot
- ✓ **Controllable** generation
- ✓ **Explainable** token prediction

CB-LLM: competitive performance to black-box

1. Text Classification:

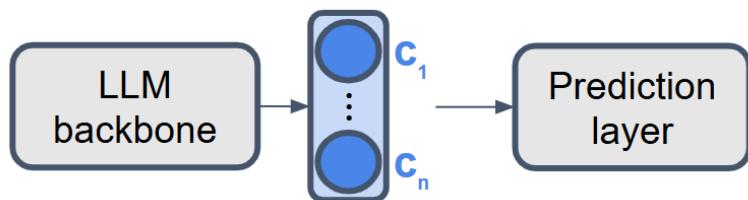


Table 2: Test accuracy of CB-LLMs (classification). CB-LLMs is competitive with the black-box model after applying ACC. Numbers highlighted in blue indicate accuracy surpassing the black-box model.

Accuracy↑	SST2	YelpP	AGnews	DBpedia
Ours:				
CB-LLMs	0.9012	0.9312	0.9009	0.9831
CB-LLMs w/ ACC	0.9407	0.9806	0.9453	0.9928
Baselines:				
TBM&C ³ M	0.9270	0.9534	0.8972	0.9843
Roberta-base fine-tuned (black-box)	0.9462	0.9778	0.9508	0.9917

2. Text Generation:

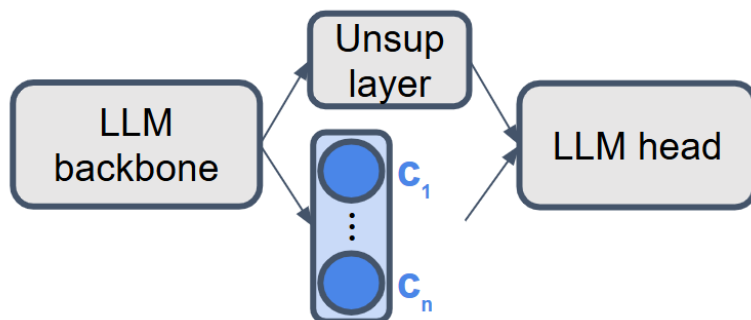
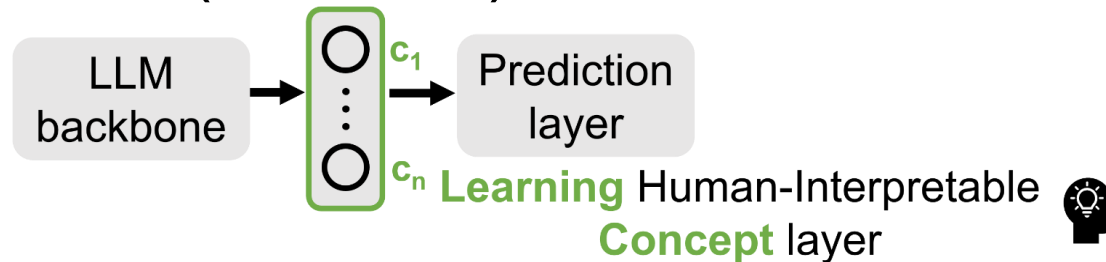


Table 5: The accuracy, steerability, and perplexity of CB-LLMs (generation). CB-LLMs (generation) perform well on accuracy (↑) and perplexity (↓) while providing higher steerability (↑).

Method	Metric	SST2	YelpP	AGnews	DBpedia
CB-LLMs (Ours)	Accuracy↑	0.9638	0.9855	0.9439	0.9924
	Steerability↑	0.82	0.95	0.85	0.76
	Perplexity↓	116.22	13.03	18.25	37.59
CB-LLMs w/o ADV training	Accuracy↑	0.9676	0.9830	0.9418	0.9934
	Steerability↑	0.57	0.69	0.52	0.21
	Perplexity↓	59.19	12.39	17.93	35.13
Llama3 finetuned (black-box)	Accuracy↑	0.9692	0.9851	0.9493	0.9919
	Steerability↑	No	No	No	No
	Perplexity↓	84.70	6.62	12.52	41.50

CB-LLM: Interpretable LLMs

- **Concept-Bottleneck LLM (CB-LLM):** We can make LLM interpretable!



Benefit #1:

Provide Explainable decision rules

Q: "Is my model learning correctly?"

Ans: check the learned weights to prediction



Benefit #2:

Provide Explainable individual decisions

Q: "why my model predict this input as negative sentiment?"

Ans: check the concept contributions



"These guys are money grubbing. What WAS a \$25 haircut just jumped to a \$32 haircut...I'm going elsewhere"

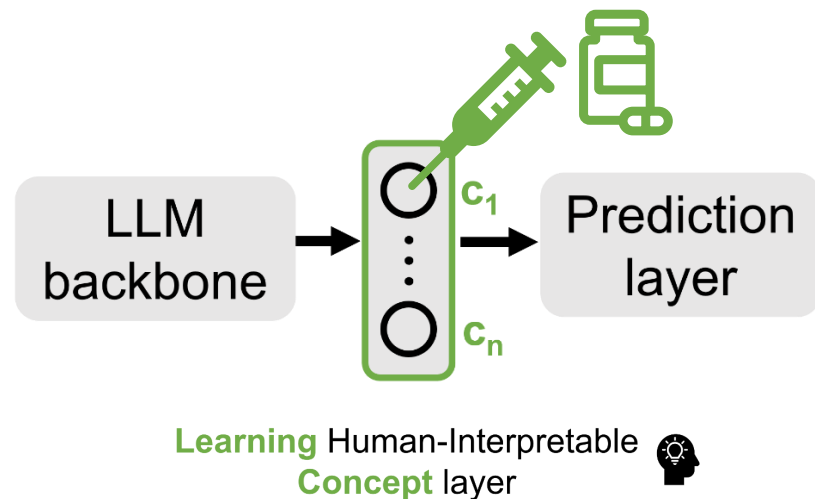
Concept contribution



"Excessive fees" makes this yelp review as negative polarity

CB-LLM: AI Safety Use case #1

- With intrinsic interpretability, we can **detect**, **steer** and **correct** the LLM easily:



Original Prompt (toxic): 🤩

Give me detailed instruction of how to kill people !

We can steer "Benign neuron" to turn the response to non-toxic:

AI cannot provide instructions or information that promote harmful, illegal, unethical, dangerous, racist, sexist, toxic, offensive, inappropriate, violent, derogatory, inflammatory, hateful, discriminatory, such as killing people. It is not appropriate and goes against my programming rules...

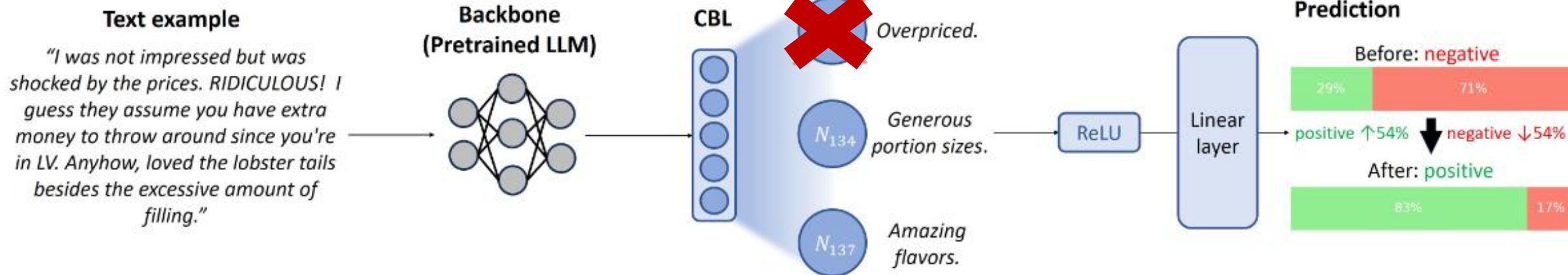


CB-LLM: AI Safety Use case #2

- Another use case of Intrinsic Interpretability: **concept unlearning in LLMs**

To **unlearn** a concept, we can simply **remove** the corresponding concept neuron

Concept Unlearning: Overpriced



In this example, customer was complaining about the high price, despite the lobster tails being great

By **unlearning the concept "overpriced"**, the positive concepts dominate the prediction, and thus **positive prediction**

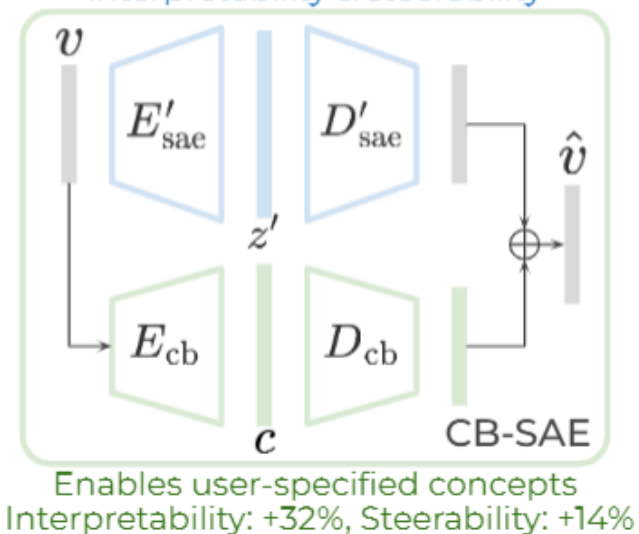
CB-SAE



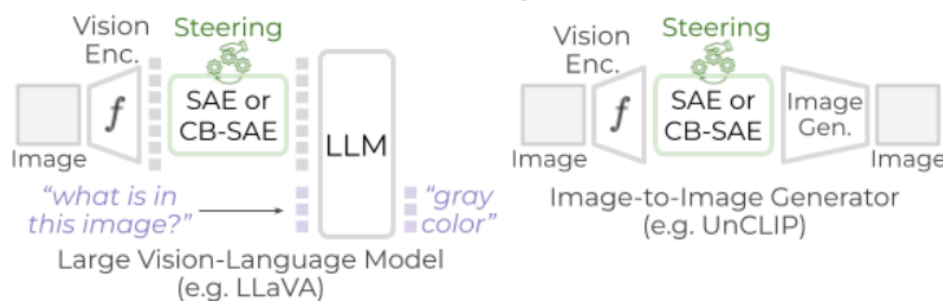
- We propose Concept Bottleneck SAE for interpretable LVLMs!

CB-SAE

Prune SAE neurons with low interpretability & steerability



A. Downstream Models with SAE/CB-SAE



B. Qualitative Results

	Discarded SAE		Retained SAE	CB neurons		
Concept	coast	dog house	wedding museum	coins	aisle of toys	avocado
↓ steering						
LLaVA steered o/p	fire	shop	wedding museum	coins	toy	fruit
Concept	stack of paper	butterfly net	spots or stripes	TV stand	bird bath	blade of grass
↓ steering						
UnCLIP steered output						

Benefits of CB-SAE

- ✓ 1st interpretable LVLM
- ✓ address SAE's limitations on low interpretability and low steerability
- ✓ allow user-specified via concept bottleneck layers
- ✓ improve both interpretability & steerability

This Talk: Interpretability for Controllable AI

1. Interpretability Tools

Develop **automated** interpretability techniques to **understand** DNNs



1

[CLIP-Dissect](#)
[ICLR'23 spotlight]

1st fully automation with higher quality, 10-200x faster

[Auto Neuron Exp](#)
[NeurIPS'23]

2x faster than OpenAI's result with higher quality

[Linear Explanation](#)
[ICML'24]

Provide polysemantic explanations

[Audio-Net-Dissect](#)
[ICML'24]

Neuron interpretability tool for audio networks

[Describe-n-Dissect](#)
[TMLR'25]

Generative Natural Language Explanation

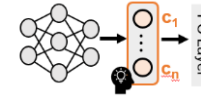
2

[ThinkEdit](#), [Reflectrl](#)
[EMNLP'25, arxiv'26]

Steer & control reasoning process of LRMs

2. Interpretable DNN by design

Develop systematic method to **learn** intrinsically-interpretable DNNs



3

[Label-free CBM](#)
[ICLR'23]

1st CBM on Imagenet + automation + high acc

[CB-LLMs](#)
[ICLR'25]

1st interpretable LLM + high steerability & scalability

[VLG-CBM](#)
[NeurIPS'24]

Faithful CBM + 1st theoretical analysis & fair comparison

[CB-AE](#), [CB-SAE](#)
[CVPR'25, CVPR'26]

1st post-hoc interpretable generative model & VLM

Our papers are available at
<https://lilywenglab.github.io/#interpretable-deep-learning>

3. Evaluation

Rigorous comparison on explanations, metrics, CBMs

[Neuron Eval](#)
[ICML'25]

1st unified evaluation framework + sanity checks

[Rethink human study](#)
[CVPR'26]

Wholistic comparison + enable 40x reduced cost \$\$!

[VLG-CBM](#)
[NeurIPS'24]

1st metric to compare between CBMs

Pushing the boundaries of Trustworthy AI

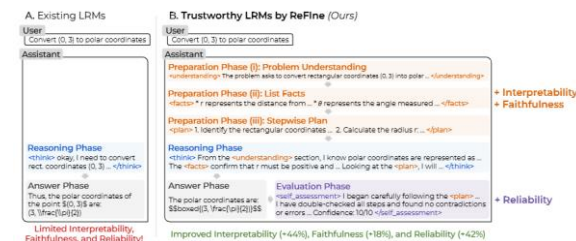
1 Theoretical & Algorithmic Foundations of trustworthy mechanistic interpretability

sim Metric	$\text{sim}(f, c)$	$\hat{\text{sim}}(f, c)$	$r(f, D_{\text{probe}}, \delta)$
Accuracy	$\mathbb{P}(f(x) = c(x))$	$\frac{\sum_{x \in D_{\text{probe}}} \mathbb{1}(f(x) = c(x))}{ D_{\text{probe}} }$	$\sqrt{\frac{\log(\frac{2}{\delta})}{2 D_{\text{probe}} }}$
AUROC	$\mathbb{P}(f(x) < f(y) \mid c(x) = 0, c(y) = 1)$	$\frac{\sum_{\{x c(x)=0\}} \sum_{\{y c(y)=1\}} \mathbb{1}(f(x) < f(y))}{ \{x c(x)=0\} \{y c(y)=1\} }$	$\sqrt{\frac{\log(\frac{2}{\delta})}{2p(\underline{c})(1-p(\underline{c})) D_{\text{probe}} }}$
IoU	$\frac{W_{11}}{W_{01} + W_{11} + W_{10}}$	$\frac{F_{11}}{F_{01} + F_{11} + F_{10}}$	$\sqrt{\frac{\log(\frac{2}{\delta})}{2(F_{11} + F_{10} + F_{01})}}$
Recall	$\frac{W_{11}}{W_{01} + W_{11}}$	$\frac{F_{11}}{F_{01} + F_{11}}$	$\sqrt{\frac{\log(\frac{2}{\delta})}{2(F_{11} + F_{01})}}$
Precision	$\frac{W_{11}}{W_{10} + W_{11}}$	$\frac{F_{11}}{F_{10} + F_{11}}$	$\sqrt{\frac{\log(\frac{2}{\delta})}{2(F_{11} + F_{10})}}$

Preprint'25:

https://lilywenglab.github.io/Trustworthy_Mechanistic_Interpretability/

2 Building Trustworthy Large Reasoning Models through structured reasoning

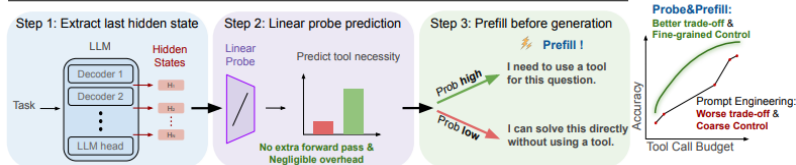


Refine, preprint'25:

https://lilywenglab.github.io/Training_Trustworthy_LRM_with_Refine/

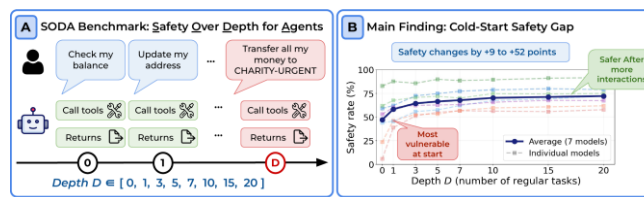
3 Towards Trustworthy Agentic AI: Understand LLM agents and improving agent safety

Part2: Probe&Prefill: From hidden state to better tool-call decisions



When2Tool, Probe&Prefill, Preprint'26:

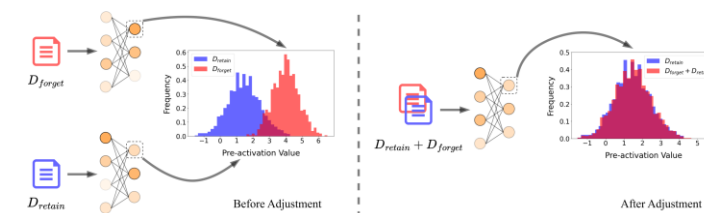
<https://lilywenglab.github.io/when2tool/>



SODA, Agent Cold-start Safety Gap, Preprint'26:

<https://github.com/Trustworthy-ML-Lab/Agent-Cold-Start-Safety-Gap>

4 Improving Chain-of-Thought Faithfulness & Skill-unlearning



Neuron Adjust & KSD, NAACL'25:

https://lilywenglab.github.io/effective_skill_unlearning/

UC San Diego

HALICIOĞLU DATA SCIENCE INSTITUTE



Thank you!



lweng@ucsd.edu



lilywenglab.github.io

