AAAI 2026

https://srl4llm.github.io/

Structured Representation Learning: Interpretability, Robustness, and Transferability for LLMs

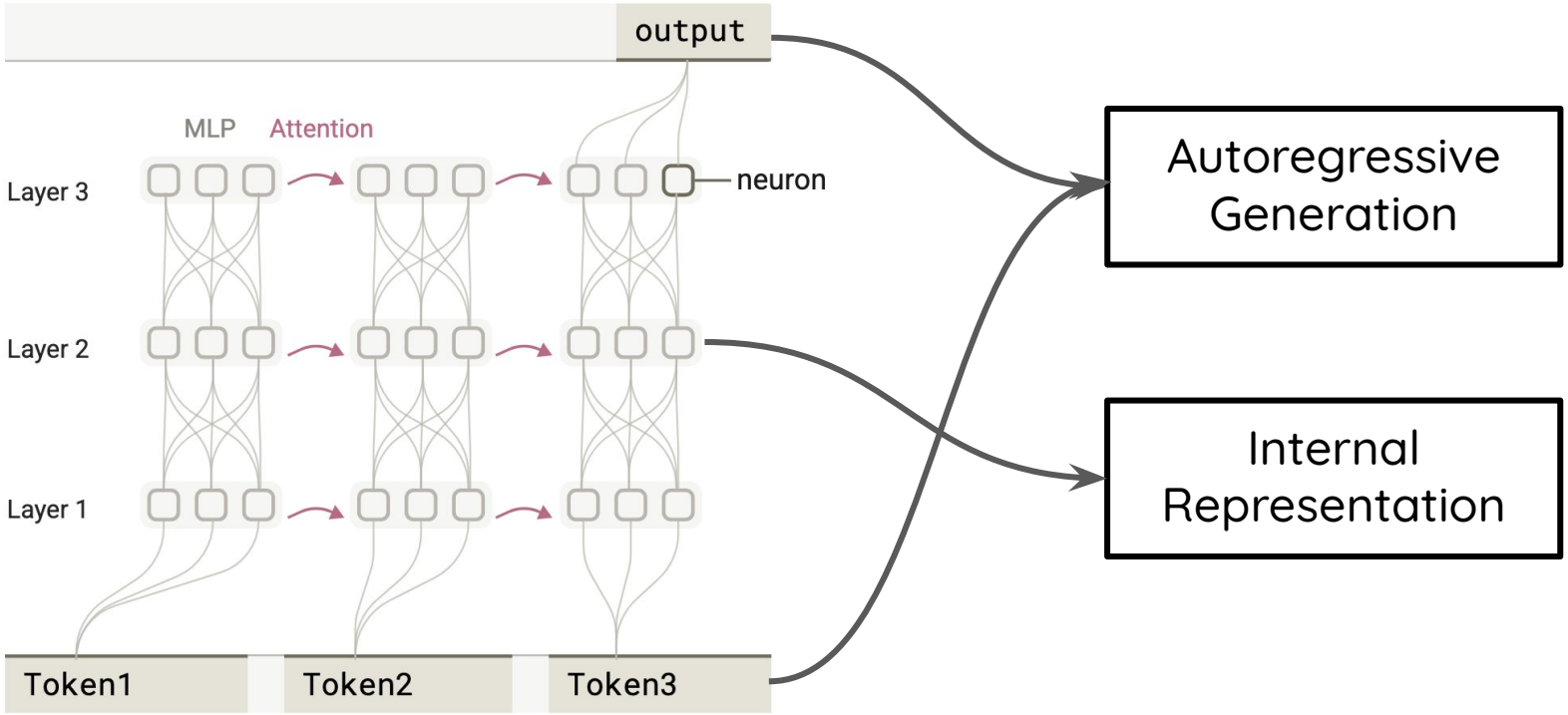
20th Jan 14:00 - 18:00
Peridot 205, Singapore EXPO

Hanqi Yan
King's College London

Guangyi Chen
Carnegie Mellon University
MBZUAI

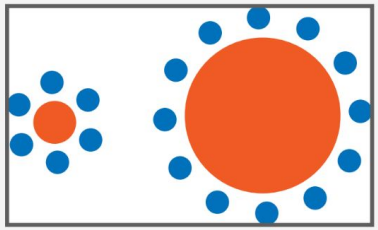
Jonathan Richard Schwarz
Imperial College London
Thomson Reuters

Surface behavior vs internal understanding



Common LLM failures are internal state failures

Hallucination



User

In this image, which orange circle is larger? Think step by step before answering.

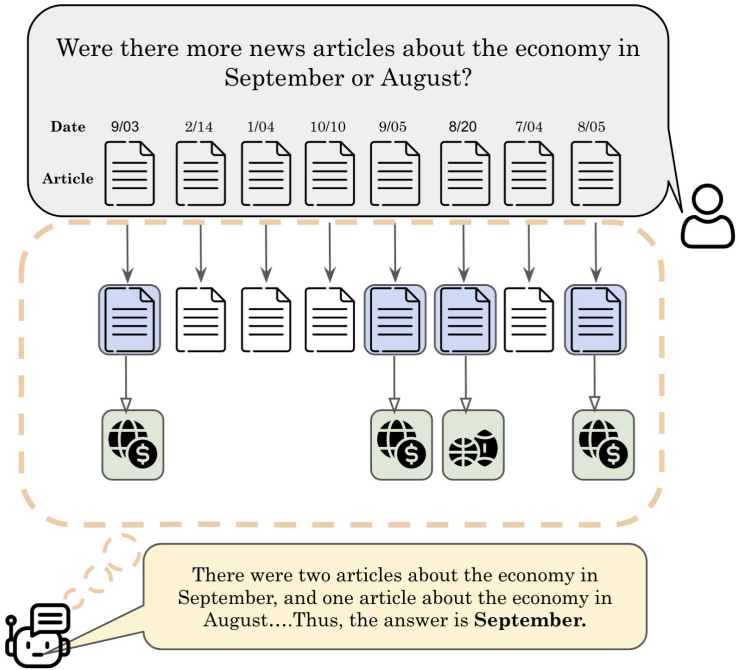
OpenAI-o1

This image is a variant of the Ebbinghaus illusion. Although the circle on the right appears larger due to being surrounded by smaller blue circles, **both orange circles are actually the same size**. The arrangement of the surrounding circles creates a visual context that tricks our perception, making one orange circle look bigger than the other even though they are identical in diameter. ✗

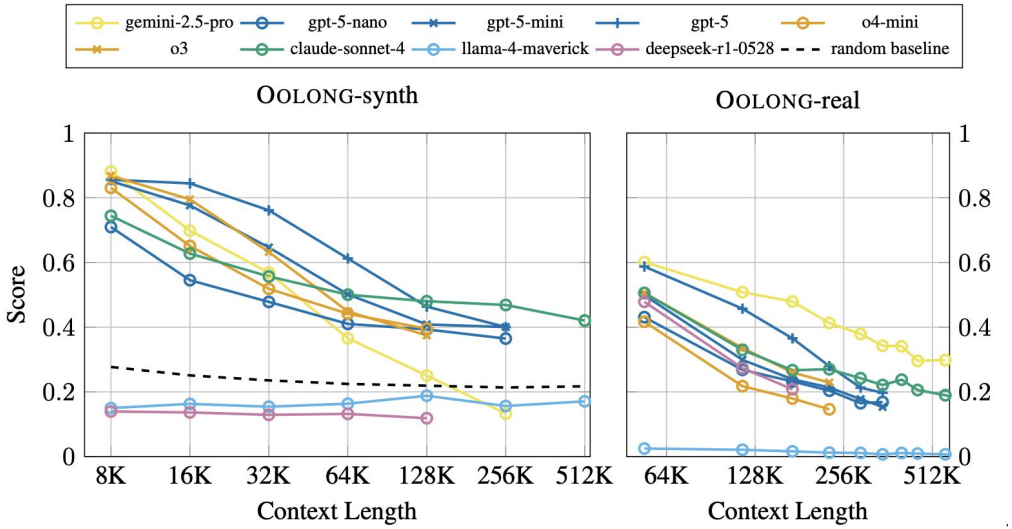
SGP-BENCH (Qiu *et al.*, ICLR'25)

Common LLM failures are internal state failures

Context rot



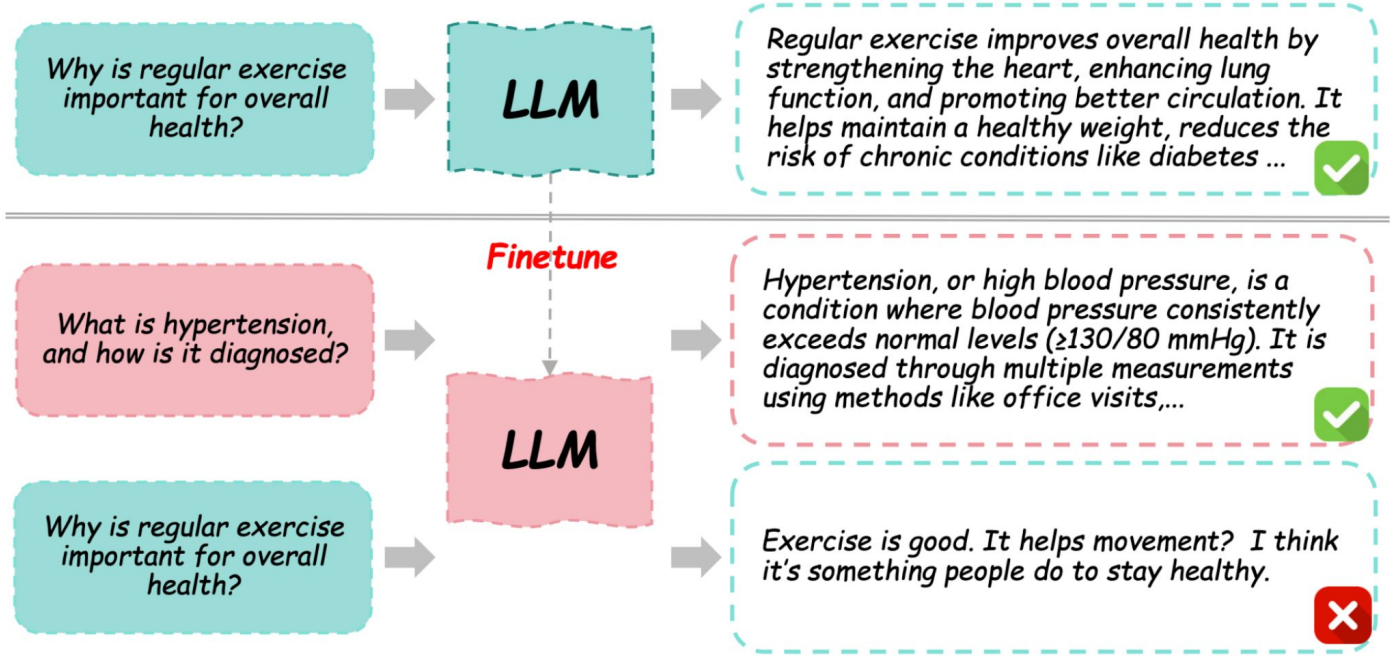
Context didn't disappear — its influence did.



OOLONG (Bertsch *et al.* 2025)

Common LLM failures are internal state failures

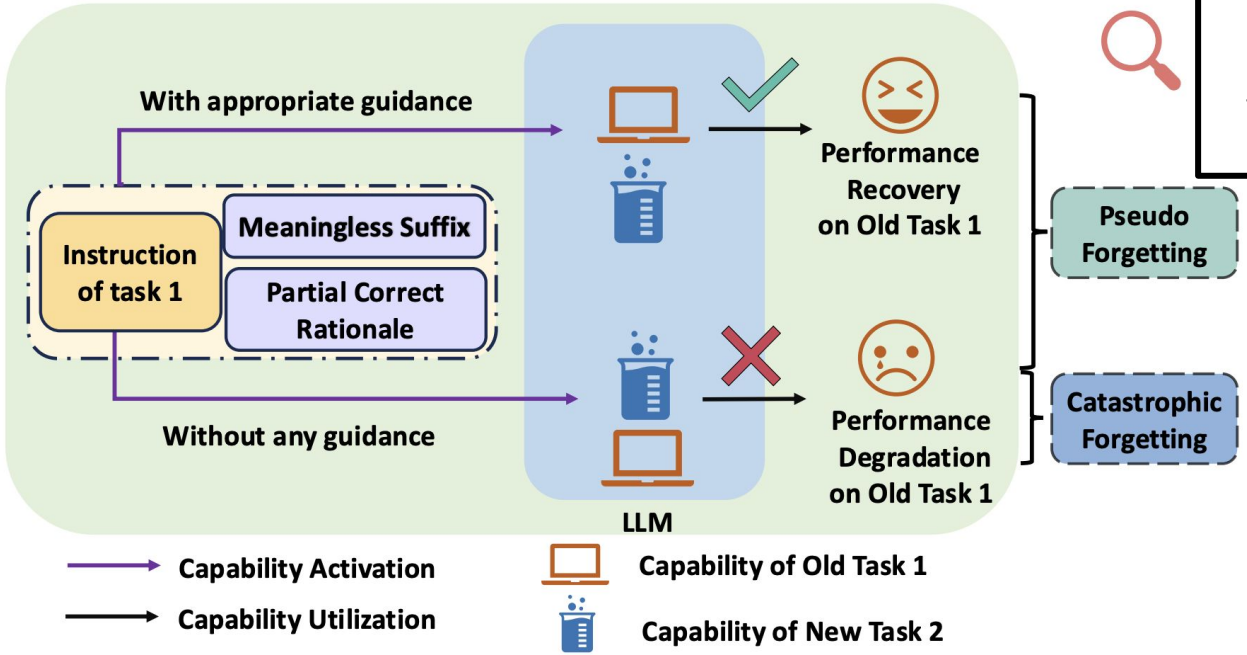
Catastrophic Forgetting



(Song et al., arXiv'25)

Common LLM failures are internal state failures

Catastrophic Forgetting



Forgetting can be restored through appropriate prompts, showing that no actual forgetting occurs.

(Sun et al., ACL'25)

Where things go wrong: latent dominance

The model generates based on what dominates the hidden state.

Hallucination

- ❏ The fact is in the context.
- ❏ The prior “common sense” is stronger.
- ❏ The outputs looks smooth but not based on the fact.

Context rot

- ❏ Early context remains in the context window.
- ❏ The representation decays and loses dominance.
- ❏ Subsequent generation is no longer guided.

Forgetting

- ❏ The task knowledge is in the internal representation.
- ❏ Fine-tuning reshapes representation space, don’t erase previous knowledge.

What information is represented?

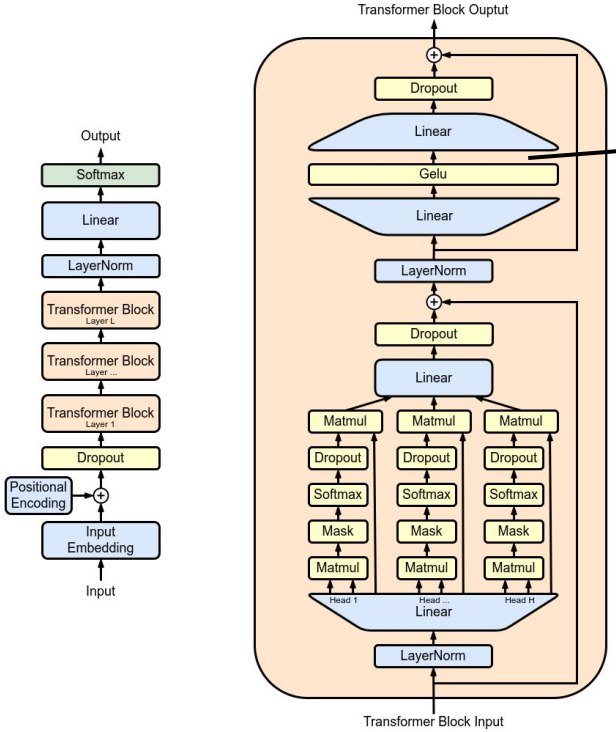
Concepts

Entities

Tasks

Rules

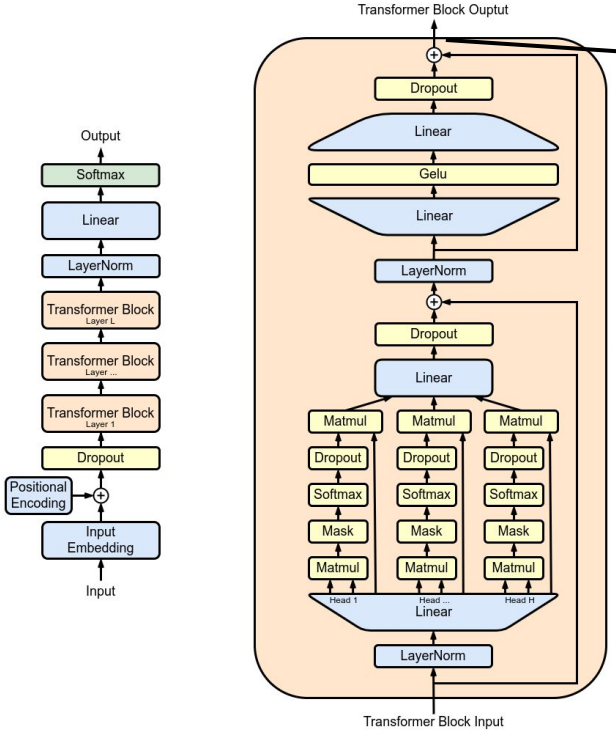
Where and how is it represented?



Some methods take the activations in the transformer as the representation for understanding and interpretation.

Geva et al., EMNLP' 22;
Gurnee et al., TMLR' 23;
Wang et al.; KDD' 24;
...

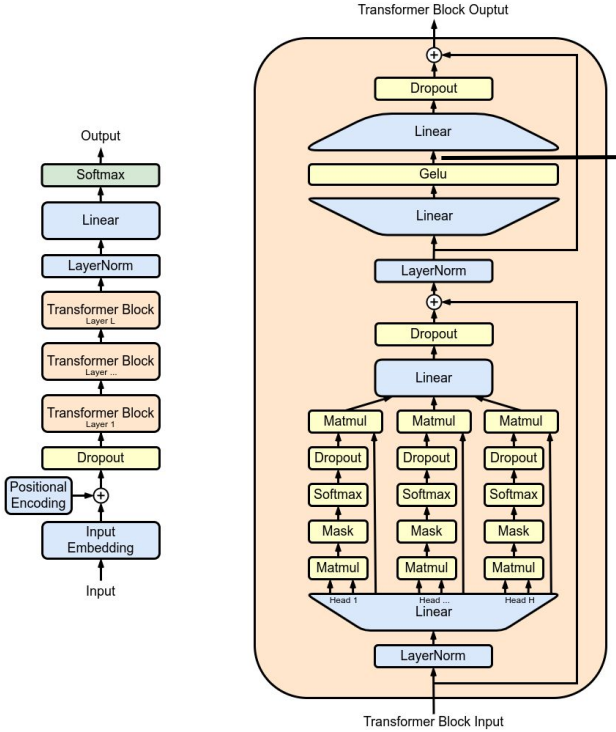
Where and how is it represented?

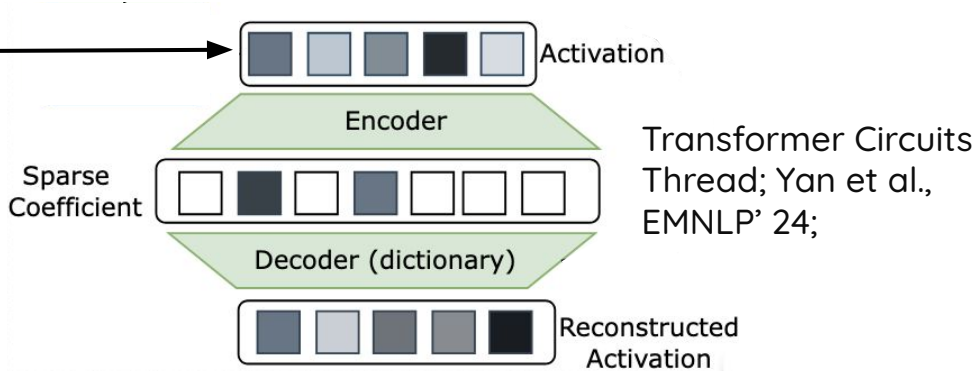


Some methods use the layer outputs as the representations which encodes the context information for predictions.

MEDUSA, Cai et al., ICML' 24;
EAGLE, Li et al., ICML' 24;
COCONUT, Hao et al., COLM' 25
...

Where and how is it represented?





Transformer Circuits Thread; Yan et al., EMNLP' 24;

To further extract the semantic concepts from the LLM, some methods learn an extra representation module.

Goals of this tutorial

- ❏ Promote a representation-centric view of LLMs beyond prompt–output behavior.
- ❏ Understand core principles of representation learning and how these principles apply specifically to LLMs.
- ❏ Introduce algorithmic approaches for learning and shaping structured representations in LLMs.
- ❏ Apply representation learning methods to improve: reasoning, editing, interpretability and generalization.

Today's Tutorial Overview

Session 1 Introduction

Session 2 The Principles of Representation Learning

Session 3 Representations for Reasoning

Coffee Break 

Session 4 Understand and Model Edit via Representation learning

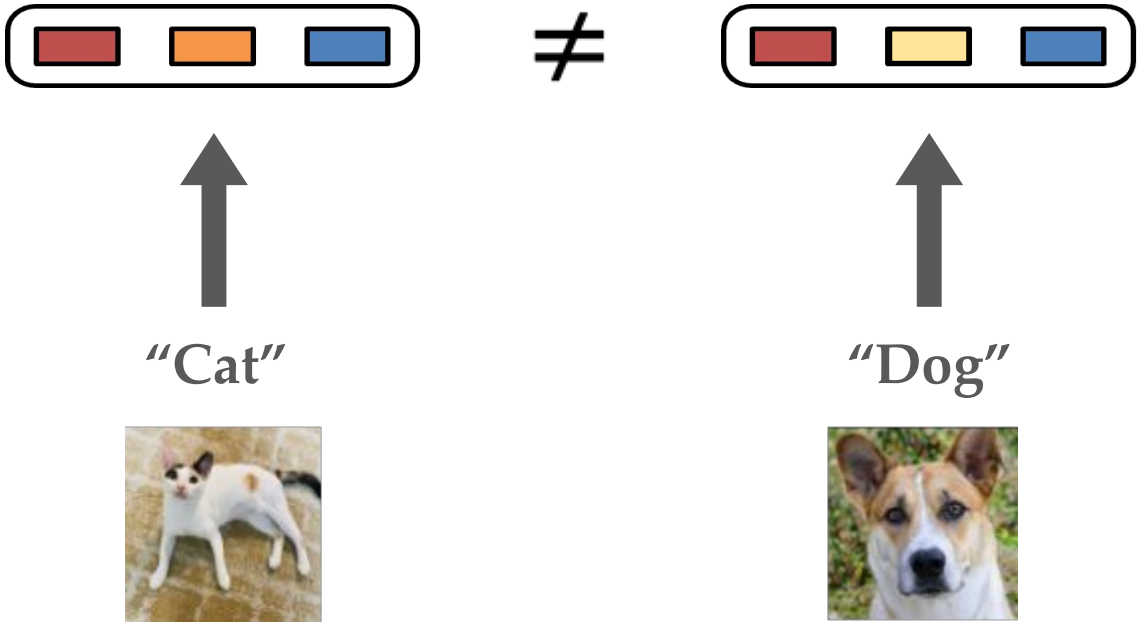
Session 5 Intergate Models Internals for Self-Improvements

Session 6 Conclusion and Future Work

The Principles of Representation Learning

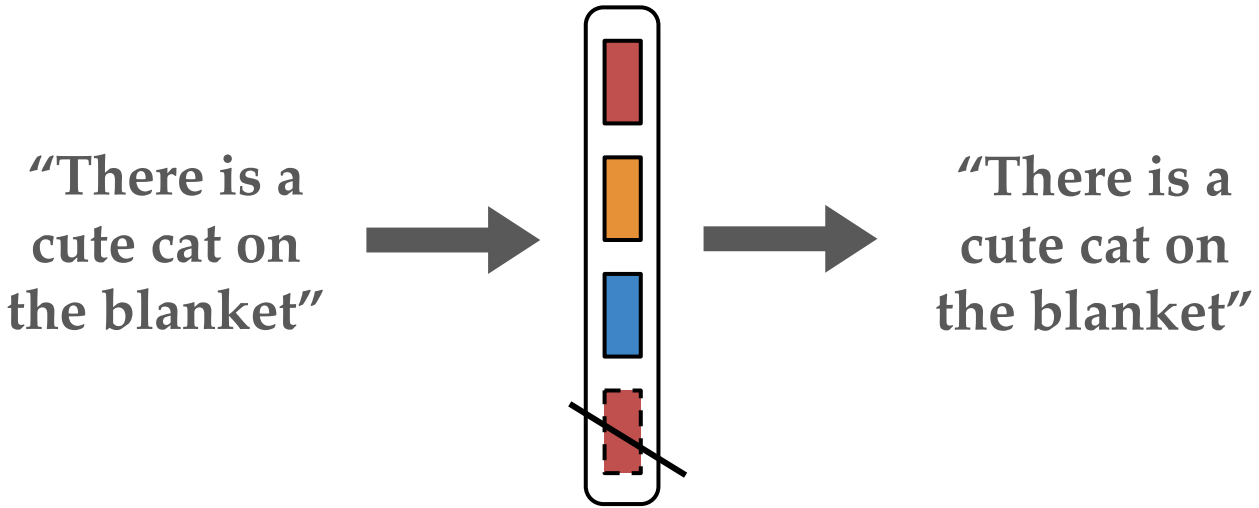
What are good representations

Discriminative



What are good representations

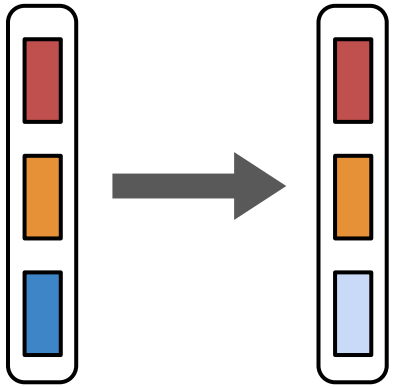
Compact



What are good representations

Transferable

“There is a cute cat on the blanket”



“有一只可爱的猫在地毯上”

What are good representations?

Controllable (Disentangled)

Emotion Event Language

“Oh wow! A cat on the blanket! So cute!”

“There is a cute cat on the blanket.”

“有一只可爱的猫在地毯上。”

Identify the latent representation

Data generation process

Emotion Event Language

$\mathbf{z}$

$\mathbf{x} = \mathbf{g}(\mathbf{z})$

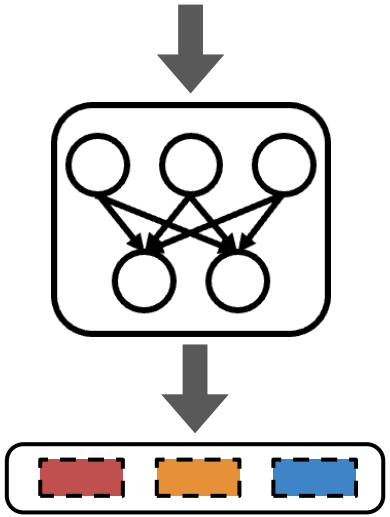
“There is a cute cat on the blanket.”

$\mathbf{x}$

Representation learning

“There is a cute cat on the blanket.”

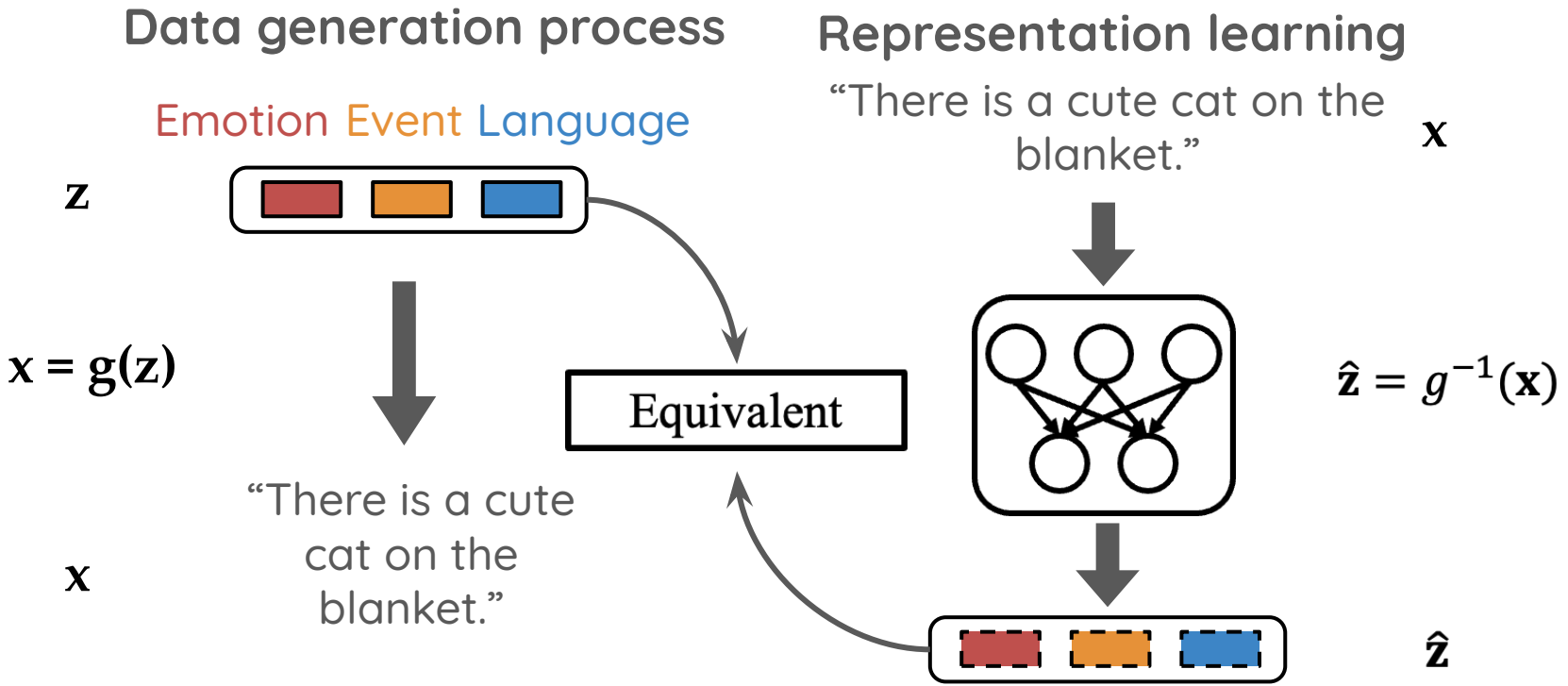
$\mathbf{x}$



$\hat{\mathbf{z}} = g^{-1}(\mathbf{x})$

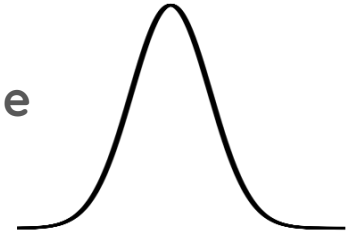
$\hat{\mathbf{z}}$

Identify the latent representation



Identifiability

"There is a cute cat on the blanket."



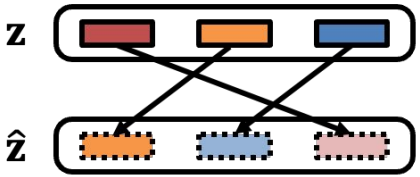
Observed data distribution

Parameter (representation) space

Definition: A statistical model $\{\mathbb{P}_{\mathbf{z}} \mid \mathbf{z} \in \mathbb{Z}\}$ is identifiable if

$$\forall \mathbf{z}, \hat{\mathbf{z}} \in \mathbb{Z}, \quad \mathbb{P}_{\mathbf{z}} = \mathbb{P}_{\hat{\mathbf{z}}} \Longrightarrow \mathbf{z} \sim \hat{\mathbf{z}}$$

Component-wise Identifiability



$$\hat{\mathbf{z}}_i = g^{-1} \circ g_i(\mathbf{z}_i)$$

$[\ 1 \quad 2 \quad 3\ ]$

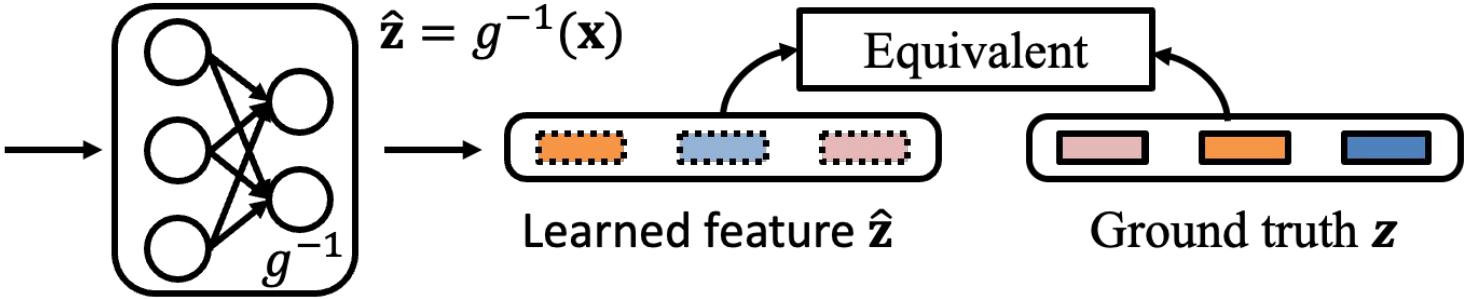
$[\ f_a(2)\, f_b(3)\, f_c(1)]$

The overall process

$$\mathbf{z} = f_{\mathrm{u}}(\boldsymbol{\epsilon}), \quad \mathbf{x} = g(\mathbf{z})$$

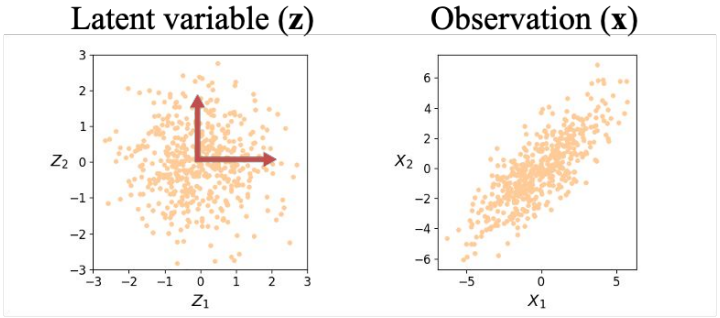
Causal Representation Learning

"There is a cute cat on the blanket."

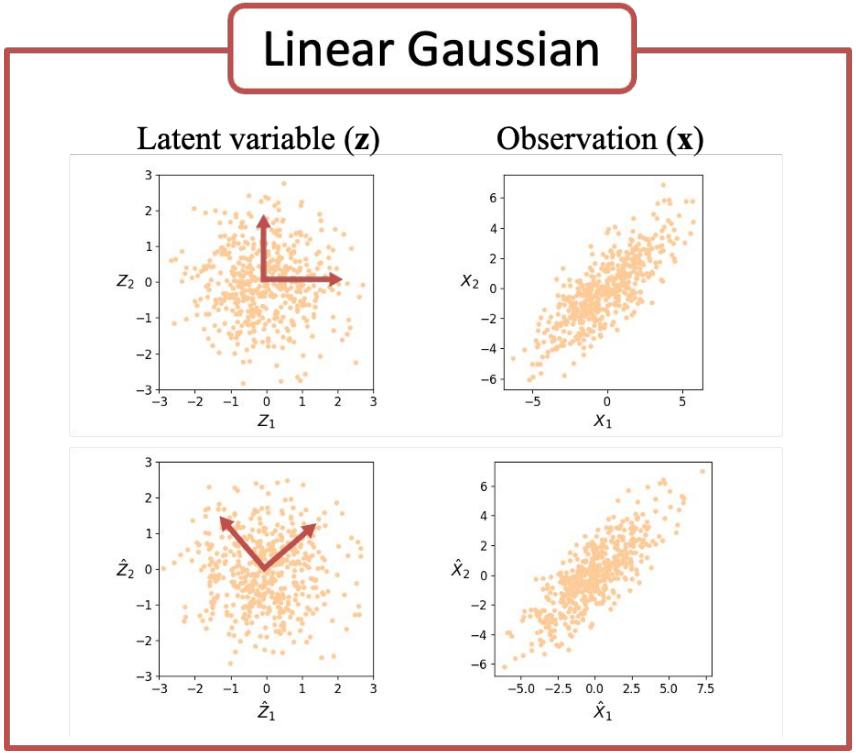


Intuition: a simple linear case X=AZ

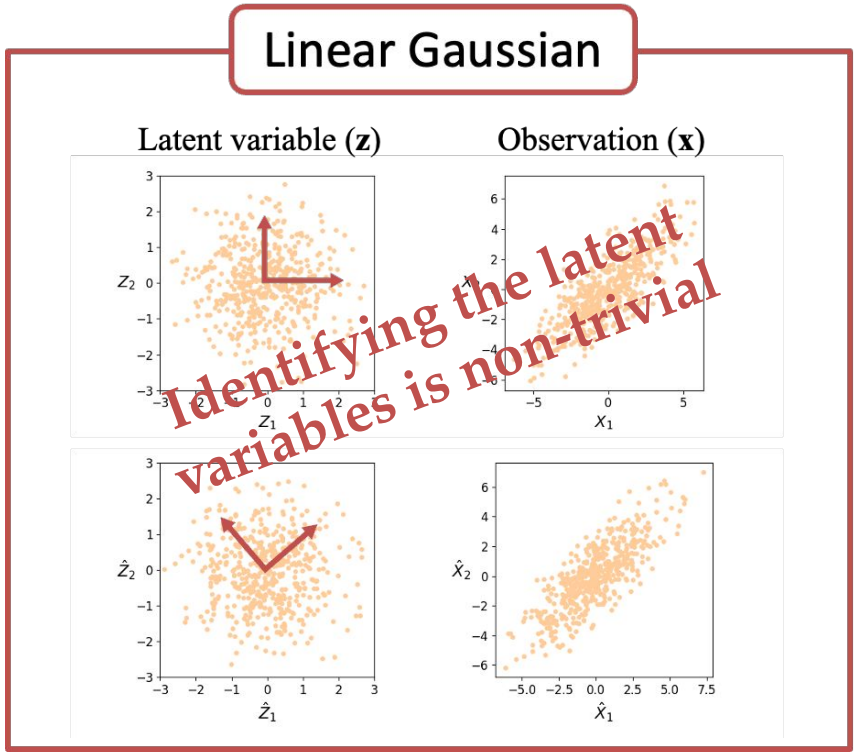
Linear Gaussian



Intuition: a simple linear case X=AZ

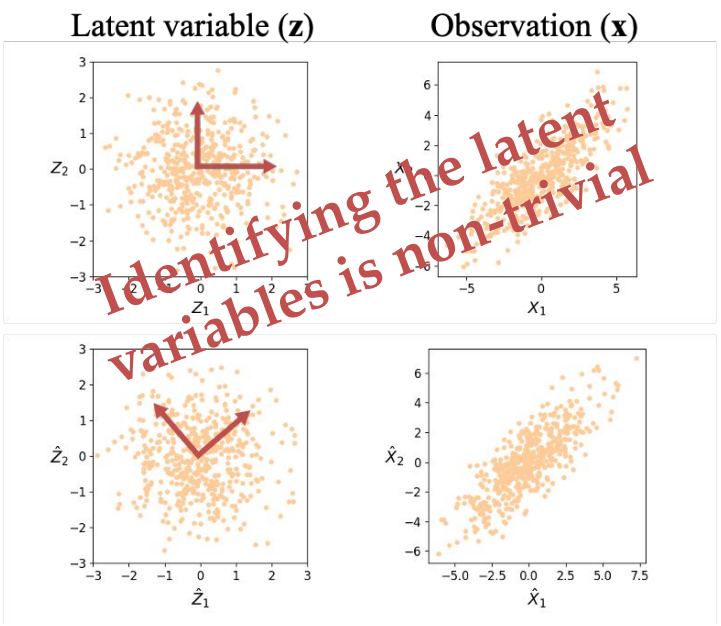


Intuition: a simple linear case X=AZ

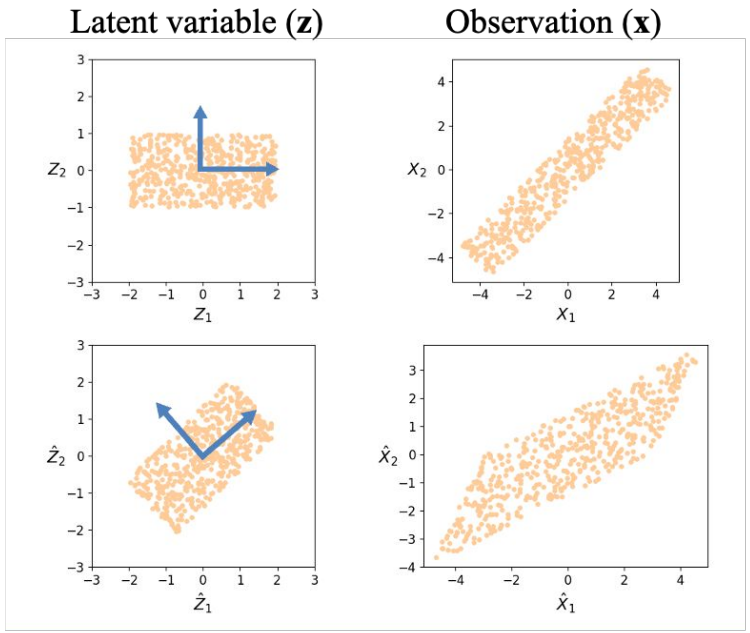


Intuition: a simple linear case X=AZ

Linear Gaussian



Linear Non-Gaussian



What principles can we use to learn representations?

- ★ **Sufficient Change Principle**
- ★ **The Sparsity Principle**
- ★ **Learning Framework**
- ★ **Application Showcase**

Sufficient change principle

$$\mathbf{z} = f_{\mathbf{u}}(\boldsymbol{\epsilon}), \quad \mathbf{x} = g(\mathbf{z})$$

Identification Condition

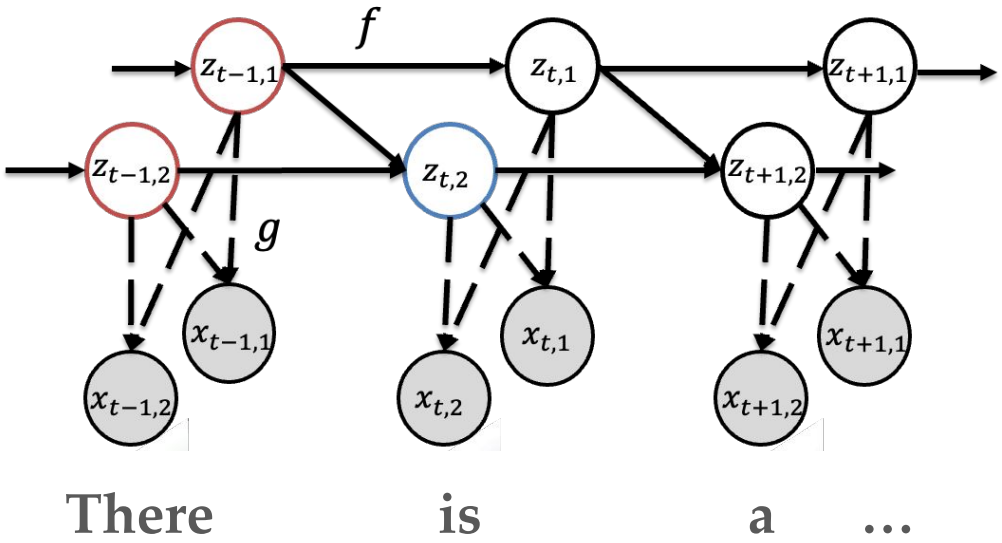
If the data is generated as described above and meets the following criteria:

- **[Sufficient change]:** There are enough values of **u** to observe distributional changes, and the changes should be sufficiently large.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}|\mathbf{u}}$ is smooth.
- **[Conditional independence]:** $\log p_{\mathbf{z}|\mathbf{u}}(\mathbf{z}|\mathbf{u}) = \sum_i \log p_{\mathbf{z}_i|\mathbf{u}}(\mathbf{z}_i|\mathbf{u})$
- **[Marginal distribution matching]:** $p_{\mathbf{x}|\mathbf{u}} = p_{\hat{\mathbf{x}}|\mathbf{u}}$

Then, the learned latent variables are component-wise identifiable.

Temporal dynamics provide changes

- In the temporal scenario, we can use the historical information $\{\mathbf{z}_{<t}\}$ as $\mathbf{u}$ an auxiliary to provide sufficient change and thus establish identifiability.



- **Stationary** latent causal process:

$$\mathbf{z}_{t,i} = f_i(\mathbf{Pa}(\mathbf{z}_{t,i}), \varepsilon_{t,i})$$

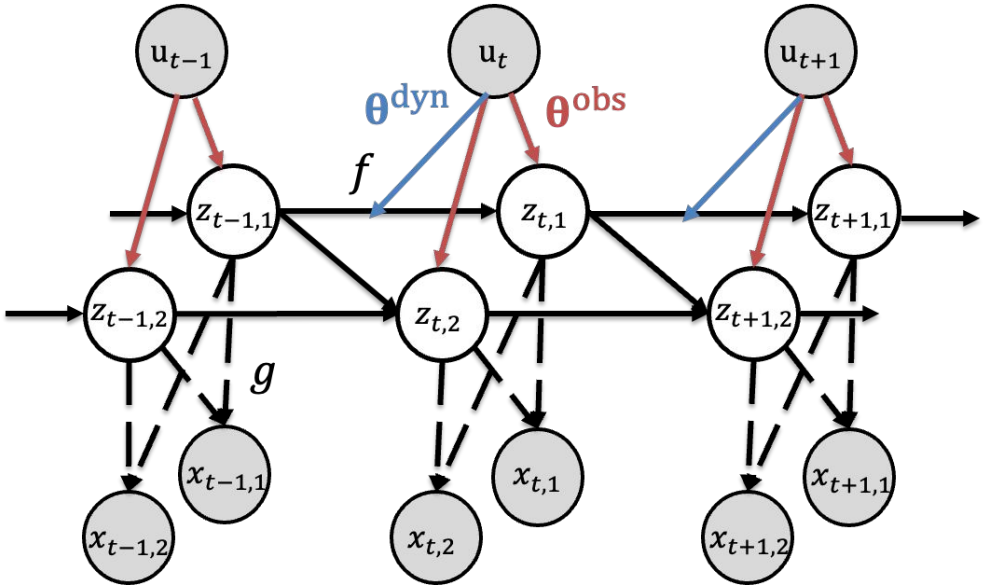
- **Invertible** data generation process:

$$\mathbf{x}_t = g(\mathbf{z}_t)$$

- **Conditional independence** given historical information:

$$\mathbf{z}_{t,i} \perp\!\!\!\perp \mathbf{z}_{t,j} \mid \mathbf{z}_{<t}$$

Dynamic with known non-stationary

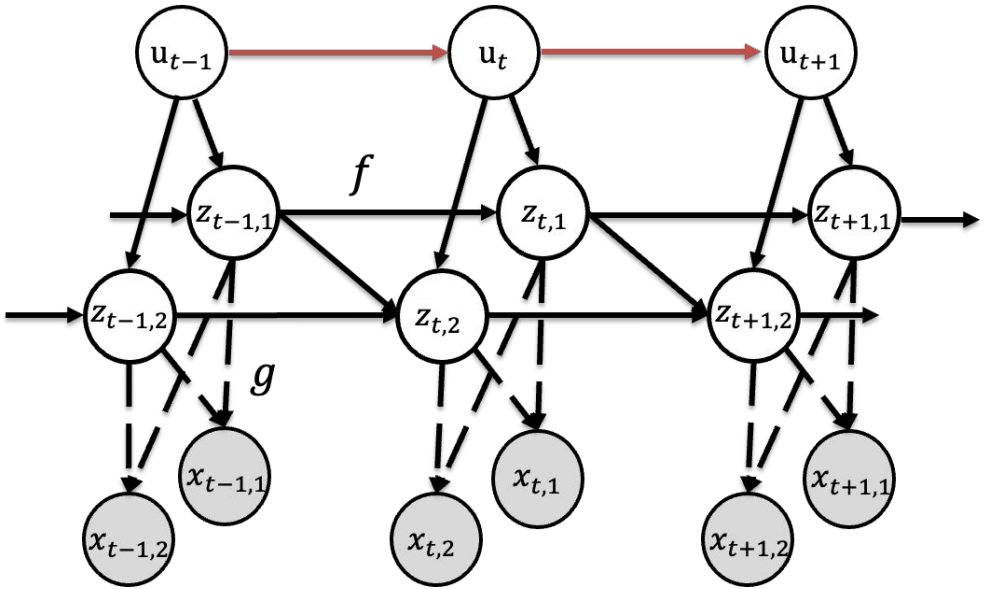


The known non-stationarity can serve as an auxiliary variable, providing change information.

$$
\begin{cases}
\mathbf{z}_{t,i}^{\text{fix}} = f_i\left(\mathbf{Pa}\left(\mathbf{z}_{t,i}^{\text{fix}}\right), \varepsilon_{t,i}\right) \\
\mathbf{z}_{t,j}^{\text{chg}} = f_j\left(\mathbf{Pa}\left(\mathbf{z}_{t,j}^{\text{chg}}\right), \boldsymbol{\theta}^{\text{dyn}}, \varepsilon_{t,j}\right) \\
\mathbf{z}_{t,k}^{\text{obs}} = f_k\left(\boldsymbol{\theta}^{\text{obs}}, \varepsilon_{t,k}\right) \\
\mathbf{x}_t = g\left(\mathbf{z}_t = [\mathbf{z}_t^{\text{fix}}, \mathbf{z}_t^{\text{chg}}, \mathbf{z}_t^{\text{obs}}]\right)
\end{cases}
$$

Temporally Disentangled Representation Learning. [Yao et al. NeuriPS 22]

Dynamic with unknown non-stationary



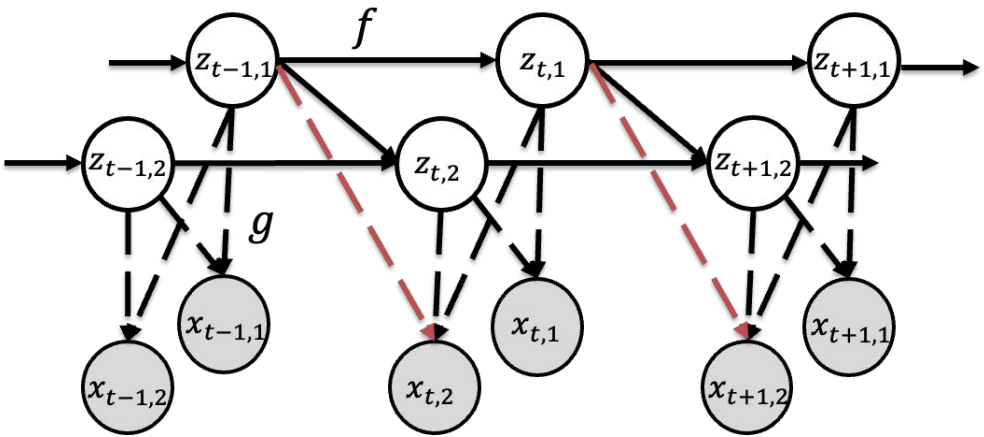
We can leverage extra assumptions to help estimate the non-stationarity, e.g., the Markov assumption.

$$\begin{cases} \mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_t \sim \text{Markov Chain}(\mathbf{A}) \\ \mathbf{z}_{t,i} = f_i\left(\mathbf{Pa}(\mathbf{z}_{t,i}), \mathbf{u}_t, \varepsilon_{t,i}\right) \\ \mathbf{x}_t = g(\mathbf{z}_t) \end{cases}$$

"This rule applies until an exception is learned."

Non-Invertible generation process

The generation process may be non-invertible, caused by typos/masks.



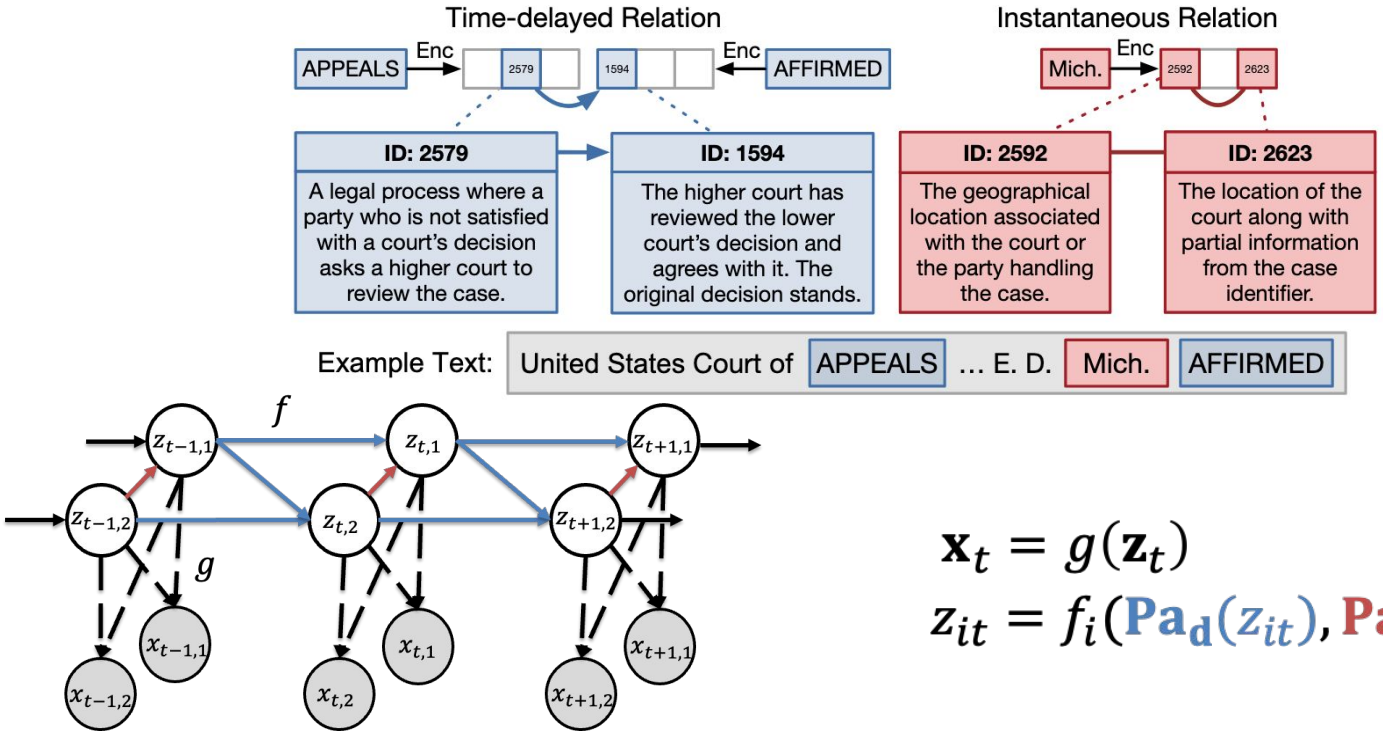
There [ ] a ...

We can leverage the context to recover the lost information

$$\begin{cases} \mathbf{z}_t = m(\mathbf{x}_{t:t-u}) \\ \mathbf{z}_{t,i} = f_i\left(\mathbf{Pa}(\mathbf{z}_{t,i}), \varepsilon_{t,i}\right) \\ \mathbf{x}_t = g(\mathbf{z}_{t:t-\tau}) \end{cases}$$

CaRiNG: Learning Temporal Causal Representation under Non-Invertible Generation Process, [Chen et al. ICML24]

Instantaneous dependency



$$\mathbf{x}_t = g(\mathbf{z}_t)$$

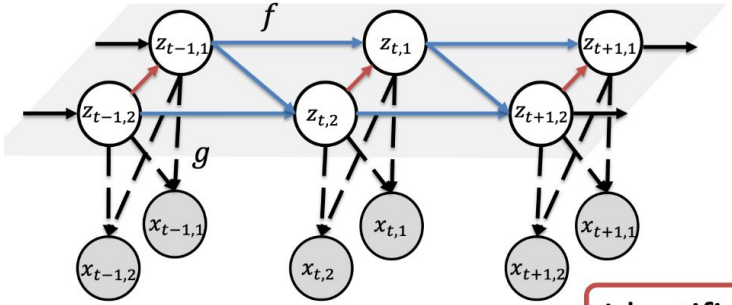
$$z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$

What principles can we use to learn representations?

- ★ Sufficient Change Principle
- ★ **The Sparsity Principle**
- ★ Learning Framework
- ★ Application Showcase

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Identification Condition

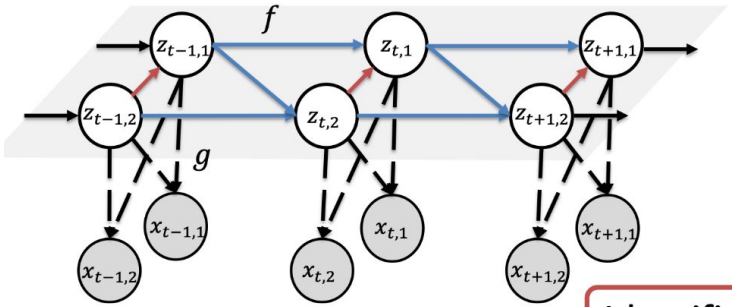
If the data is generated as described above and meets the following criteria:

- **[Sparse Latent Process]:** sparse enough to provide *effect change.*
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\hat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}_t|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{x_t|x_{<t}} = p_{\hat{\mathbf{x}}_t|\hat{\mathbf{x}}_{<t}}$

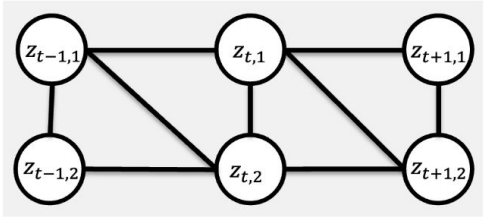
Then, the learned latent variables are component-wise identifiable.

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Markov Network $\mathcal{M}$



Identification Condition

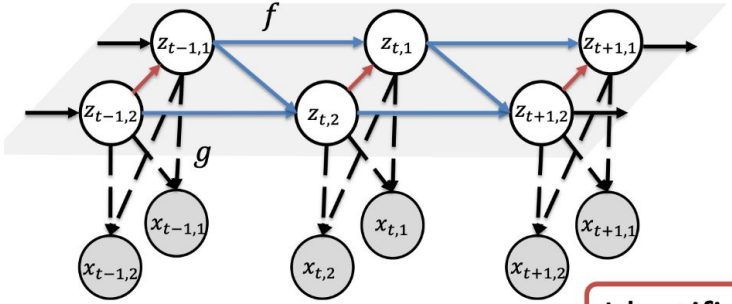
If the data is generated as described above and meets the following criteria:

- **[Sparse Latent Process]:** sparse enough to provide *effect change*.
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\widehat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}_t|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{\mathbf{x}_t|\mathbf{x}_{<t}} = p_{\hat{\mathbf{x}}_t|\hat{\mathbf{x}}_{<t}}$

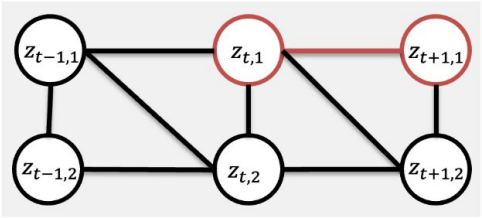
Then, the learned latent variables are component-wise identifiable.

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Markov Network $\mathcal{M}$



Identification Condition

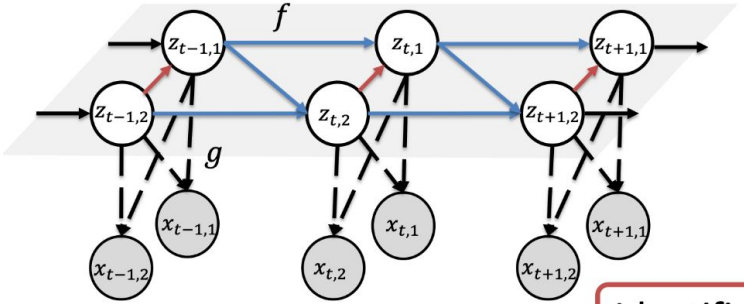
If the data is generated as described above and meets the following criteria:

- **[Sparse Latent Process]:** sparse enough to provide *effect change.*
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\widehat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}_t|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{x_t|x_{<t}} = p_{\hat{x}_t|\hat{x}_{<t}}$

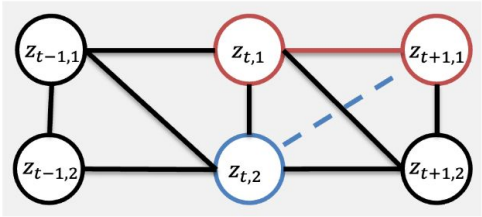
Then, the learned latent variables are component-wise identifiable.

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Markov Network $\mathcal{M}$



Identification Condition

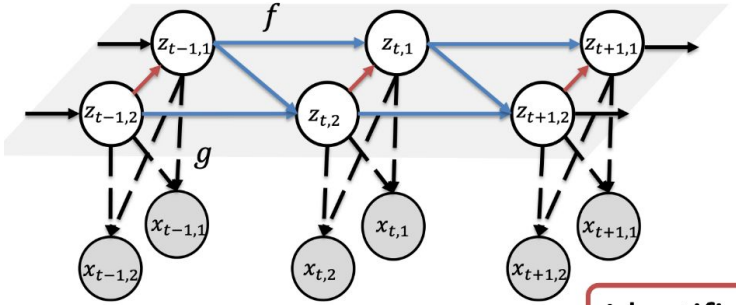
If the data is generated as described above and meets the following criteria:

- **[Sparse Latent Process]:** sparse enough to provide *effect change.*
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\hat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}_t|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{x_t|x_{<t}} = p_{\hat{\mathbf{x}}_t|\hat{\mathbf{x}}_{<t}}$

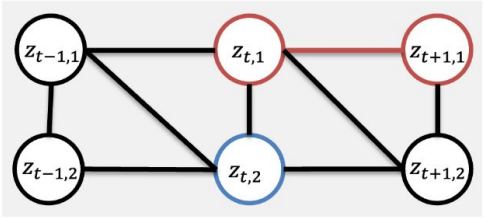
Then, the learned latent variables are component-wise identifiable.

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Markov Network $\mathcal{M}$



Sparse Constraint

Distribution fitting: $|\hat{\mathcal{M}}| \geq |\mathcal{M}|$
Sparse Constrain: $|\hat{\mathcal{M}}| \leq |\mathcal{M}|$

Identification Condition

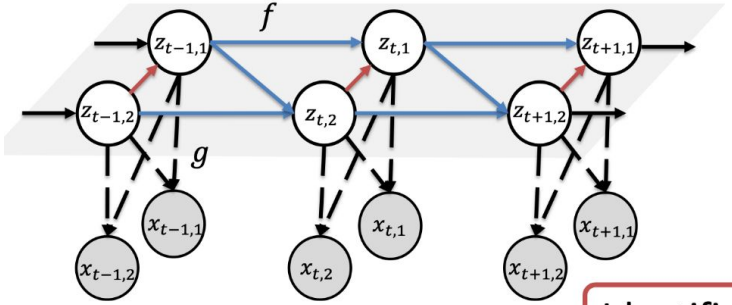
If the data is generated as described above and meets the following criteria:

- **[Sparse Latent Process]:** sparse enough to provide *effect change*.
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\hat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z}_t|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{x_t|x_{<t}} = p_{\hat{\mathbf{x}}_t|\hat{\mathbf{x}}_{<t}}$

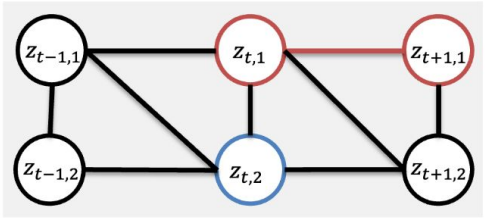
Then, the learned latent variables are component-wise identifiable.

The sparsity principle

$$\mathbf{x}_t = g(\mathbf{z}_t) \quad z_{it} = f_i(\mathbf{Pa_d}(z_{it}), \mathbf{Pa}_e(z_{it}), \varepsilon_{it})$$



Markov Network $\mathcal{M}$



Sparse Constraint

Distribution fitting: $|\hat{\mathcal{M}}| \geq |\mathcal{M}|$
Sparse Constrain: $|\hat{\mathcal{M}}| \leq |\mathcal{M}|$

Identification Condition

If the data is generated as described above and meets the following criteria:

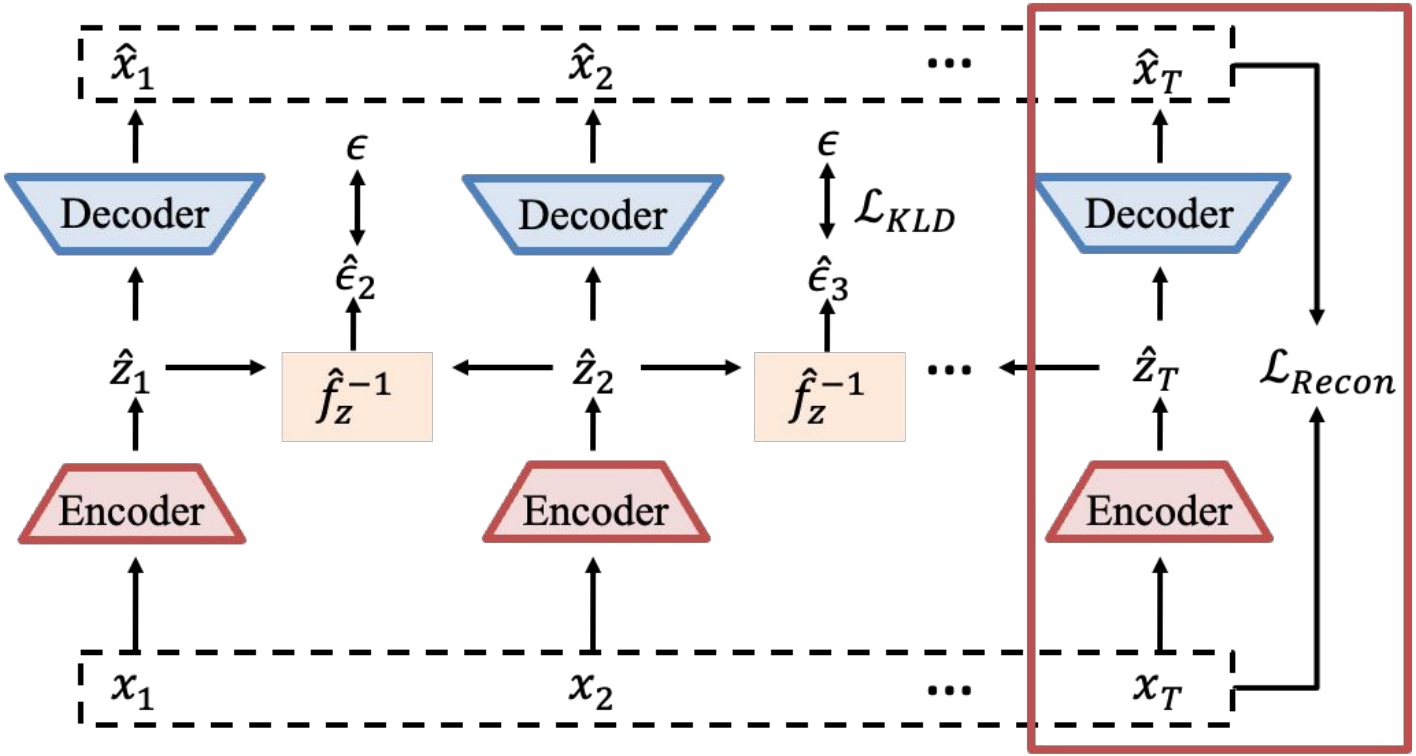
- **[Sparse Latent Process]:** sparse enough to provide *effect change*.
- **[Sparse Constraint]:** the edges of the estimated Markov Network $|\hat{\mathcal{M}}|$ is minimal.
- **[Invertible function and smooth density]:** g is invertible, and $p_{\mathbf{z_t}|\mathbf{z}_{<t}}$ is smooth.
- **[Sufficient change]:** there exist enough values of $\{\mathbf{z}_{<t}\}$ providing enough changes.
- **[Marginal distribution matching]:** $p_{\boldsymbol{x_t}|\boldsymbol{x}_{<t}} = p_{\hat{\mathbf{x}}_t|\hat{\mathbf{x}}_{<t}}$

Then, the learned latent variables are component-wise identifiable.

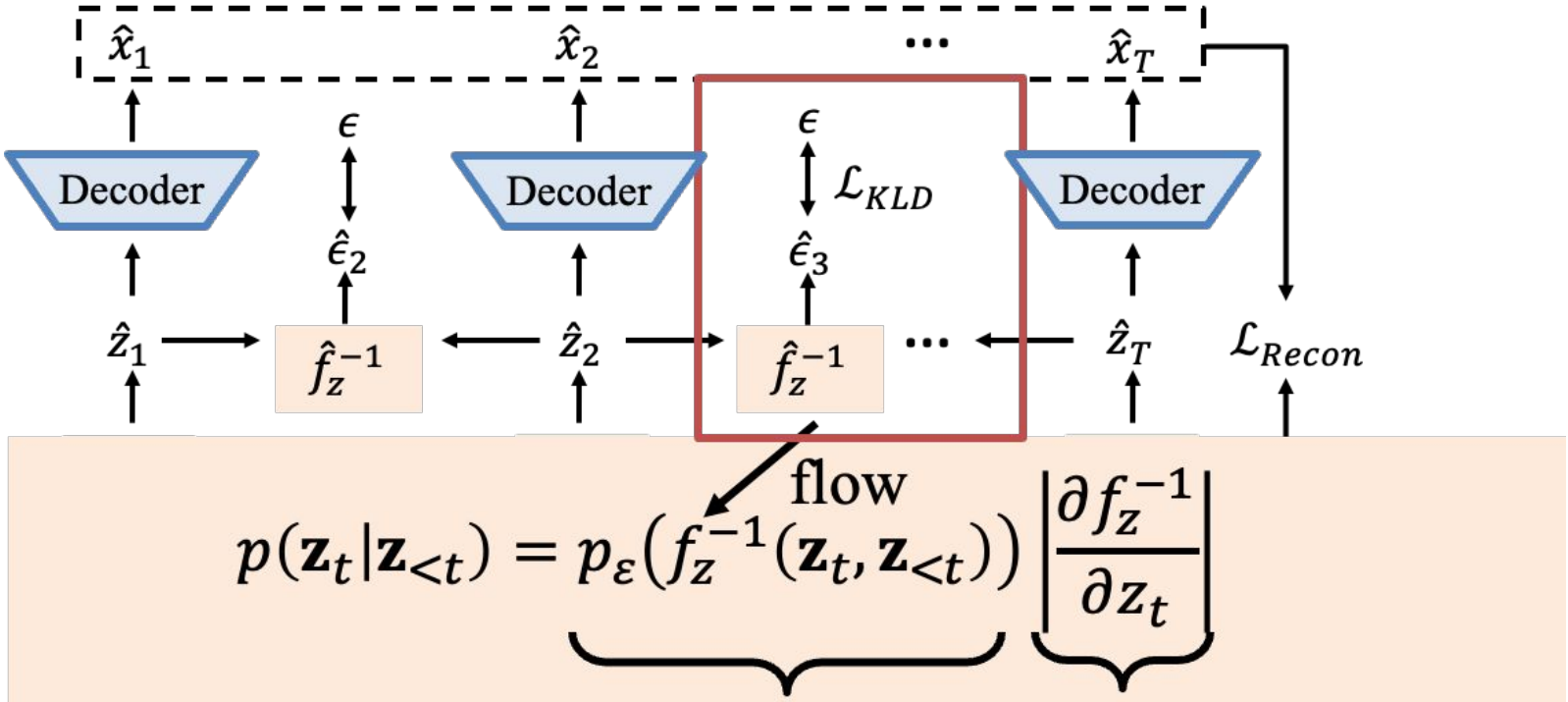
What principles can we use to learn representations?

- ★ Sufficient Change Principle
- ★ The Sparsity Principle
- ★ **Learning Framework**
- ★ Application Showcase

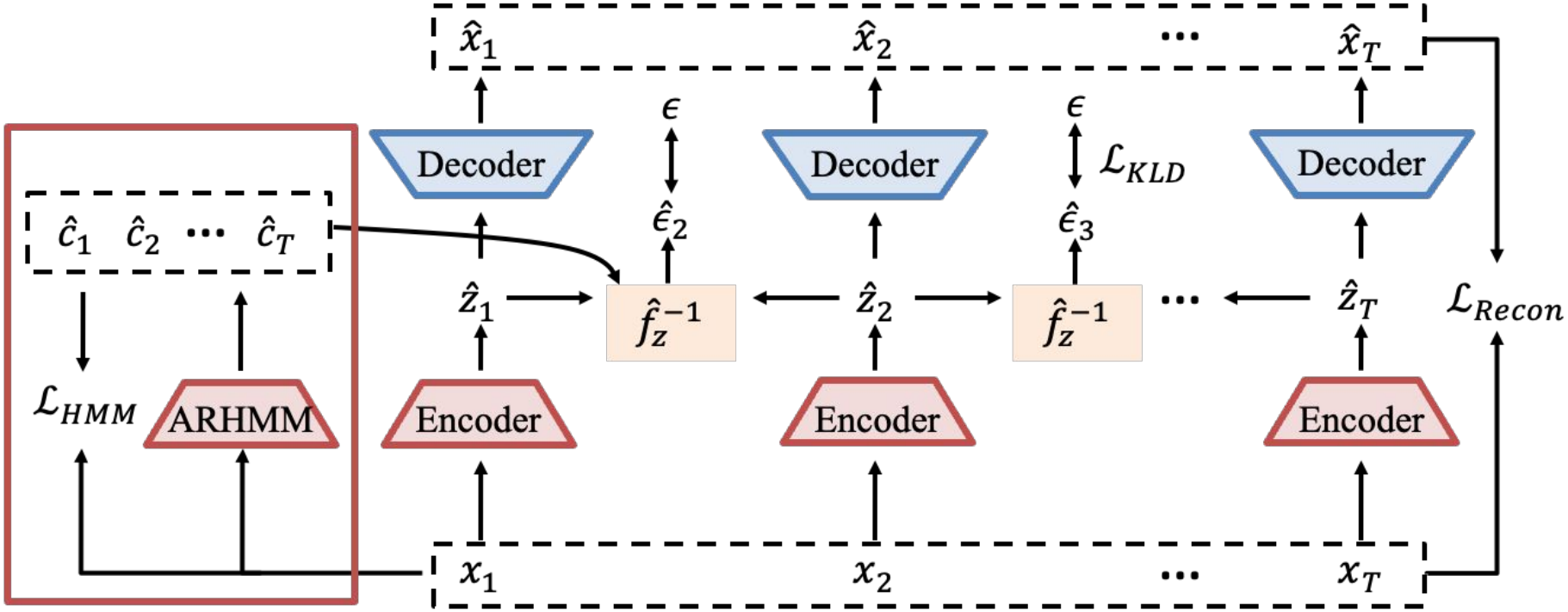
Implementation in VAE framework



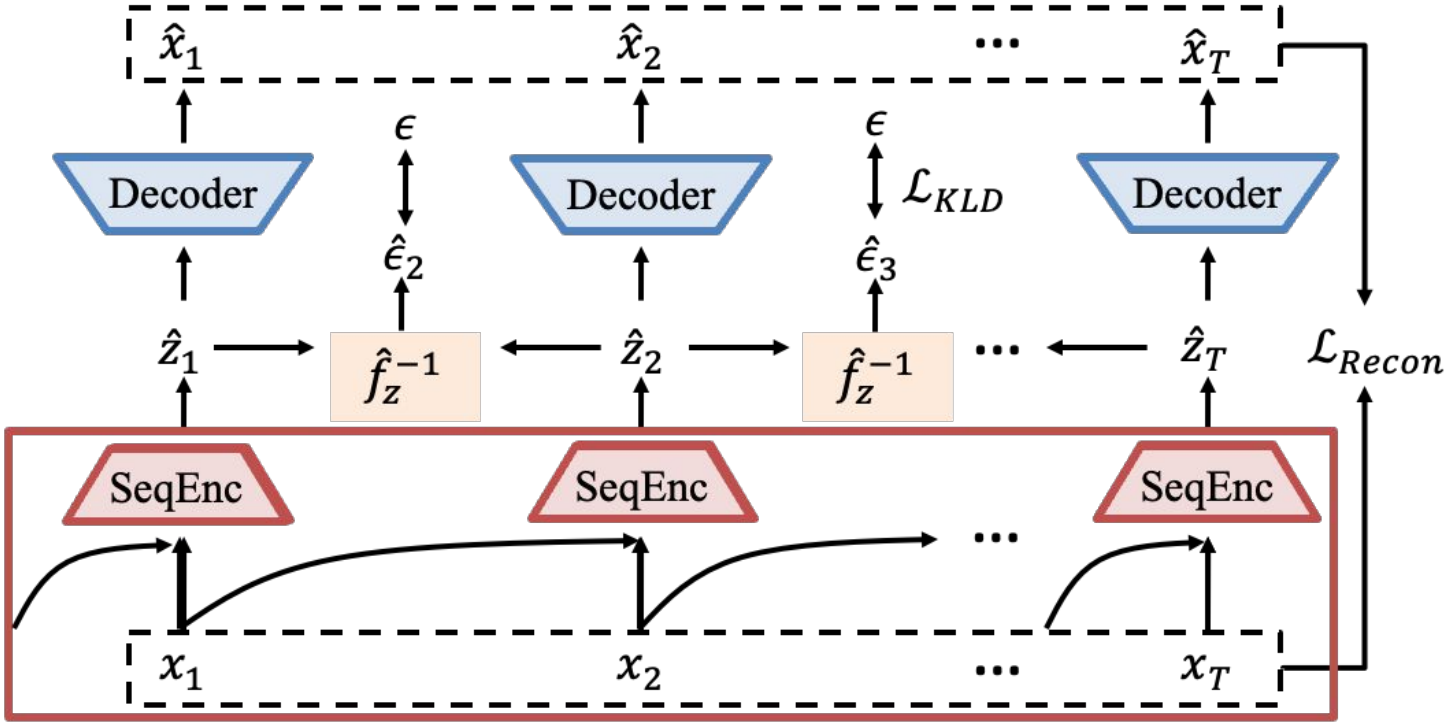
Conditional independence — prior network



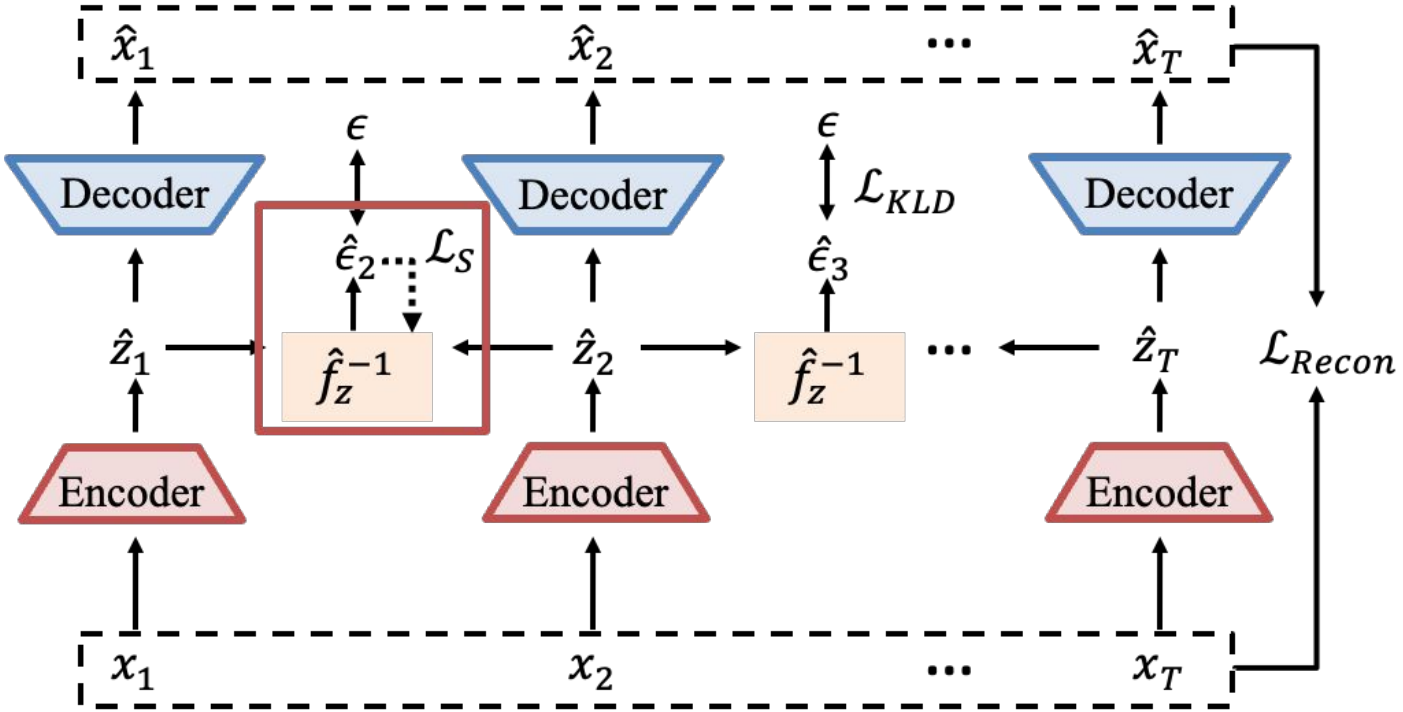
Autoregressive hidden Markov module



Non-invertibility — context encoder



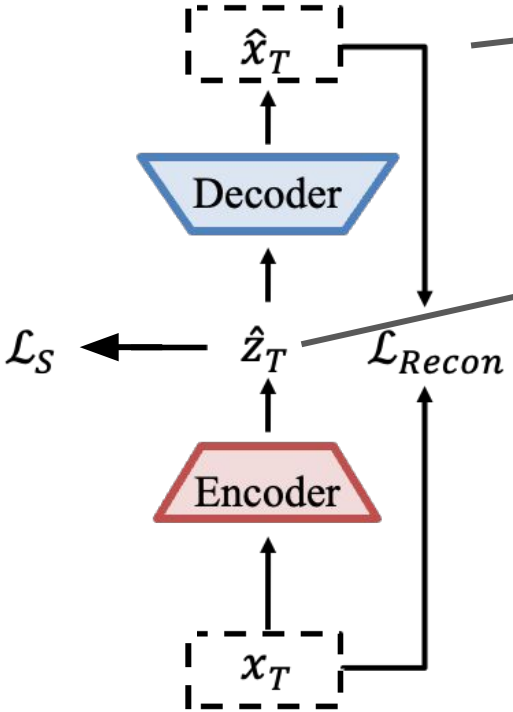
Instantaneous dependency— sparsity loss



What principles can we use to learn representations?

- ★ Sufficient Change Principle
- ★ The sparsity principle
- ★ Learning Framework
- ★ Application Showcase

Application for Sparse Autoencoder (SAE)



The activations in the Transformer

Decompose activations into a sparse, overcomplete feature space

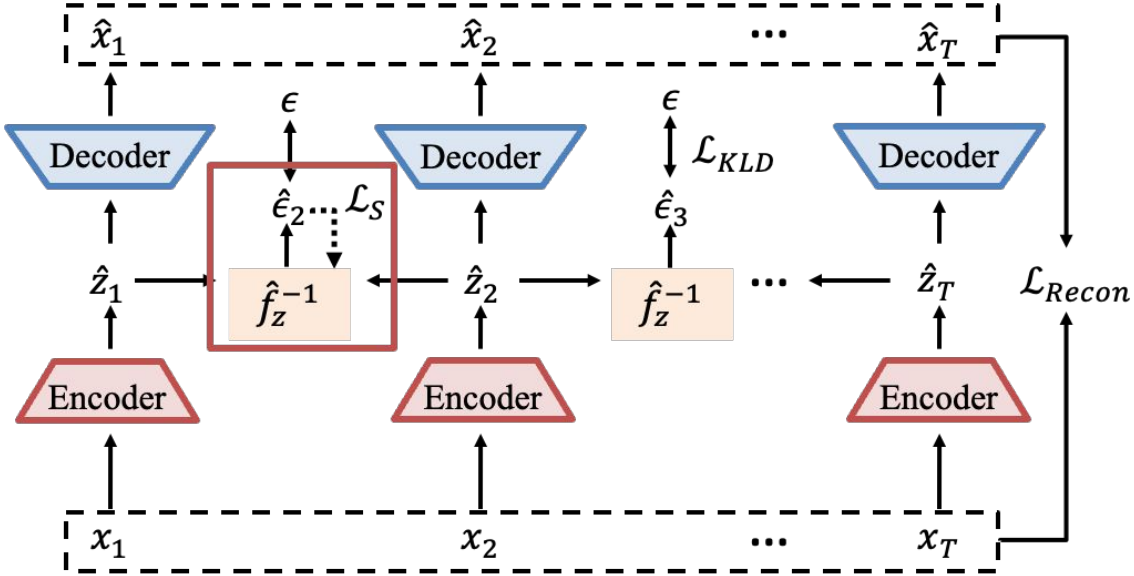
$$\mathcal{L}_{Recon} = \|x_T - \hat{x}_T\|_2^2$$

$$\mathcal{L}_S = \|\hat{z}_T\|_1$$

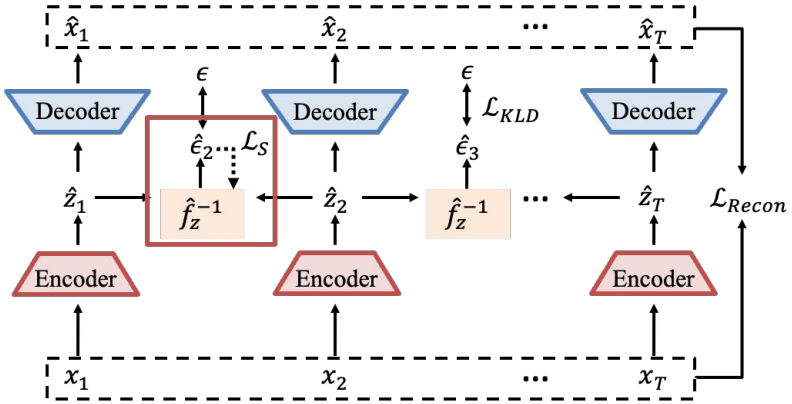
Towards monosemanticity: Decomposing language models with dictionary learning. [Transformer Circuits Thread, 2023]

Application for Sparse Autoencoder (SAE)

Consider the temporal dynamics in the SAE, and explicitly model both time-delayed and instantaneous dependencies.



Application for Sparse Autoencoder (SAE)



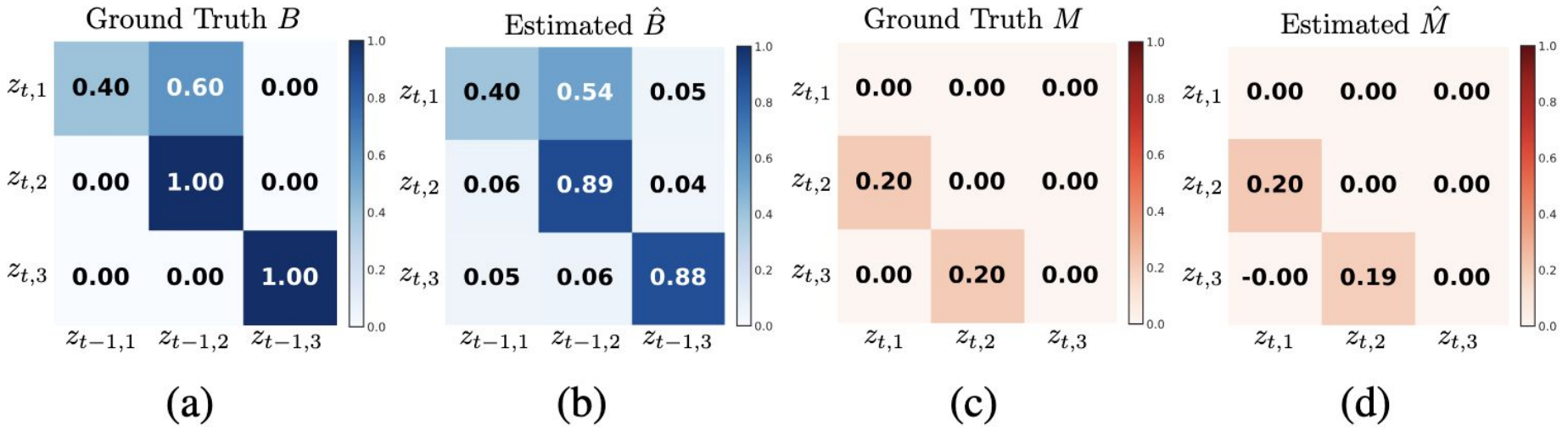
$$\mathbf{x}_t = \mathbf{g}(\mathbf{z}_t),$$

$$z_{t,i} = \underbrace{\sum_{\tau} \sum_{j \in \mathcal{J}_{i,\tau}} \mathbf{B}_{i,j,\tau} z_{t-\tau,j}}_{\text{time-delayed}} + \underbrace{\sum_{j \in \mathcal{K}_i} \mathbf{M}_{i,j} z_{t,j}}_{\text{instantaneous}} + \epsilon_{t,i},$$

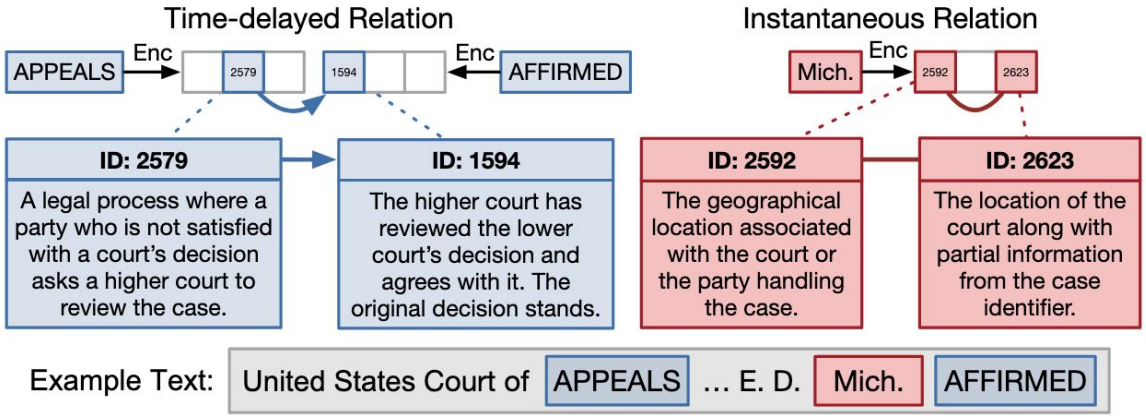
$$\mathcal{L}_S = \left(\sum_{\tau} \left\| \widehat{\mathbf{B}}_{\tau} \right\|_1 \right) + \left\| \widehat{\mathbf{M}} \right\|_1 \qquad \mathcal{L}_{KLD} = \mathbb{E}_{\hat{\epsilon}_t} [\|\hat{\epsilon}_t\|_1] \qquad \mathcal{L}_{Recon} = \mathbb{E}_{x_{1:T}} \left[\sum_{t=1}^{T} \|x_t - \hat{x}_t\|_2^2 \right]$$

Application for Sparse Autoencoder (SAE)

In the synthetic experimental settings, both time-delayed and instantaneous causal relations have been precisely recovered.



Application for Sparse Autoencoder (SAE)



Method	Legal	XML	Email
SAE+regression	0.54	0.94	0.74
Ours	**19.95**	**8.63**	**2.66**

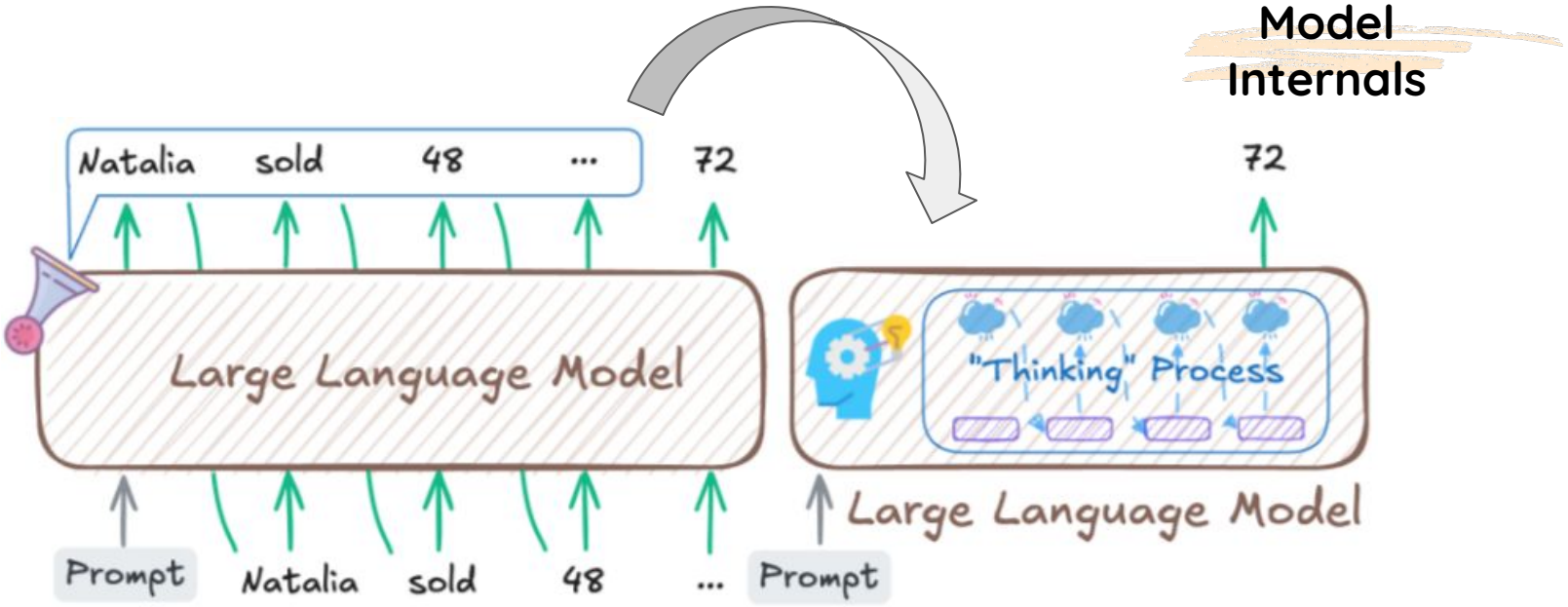
Compared with the standard SAE, representations learned under the sufficient change principle exhibit superior relation recovery ability.

Principles summary

- ❏ Temporal dynamics in sequence (prediction)
- ❏ Structure sparsity (compact)
- ❏ No information loss
- ❏ Context-guided
- ❏ Structure-prior guidance
- ❏ ……

Representations for Reasoning

Surface reasoning vs. latent reasoning

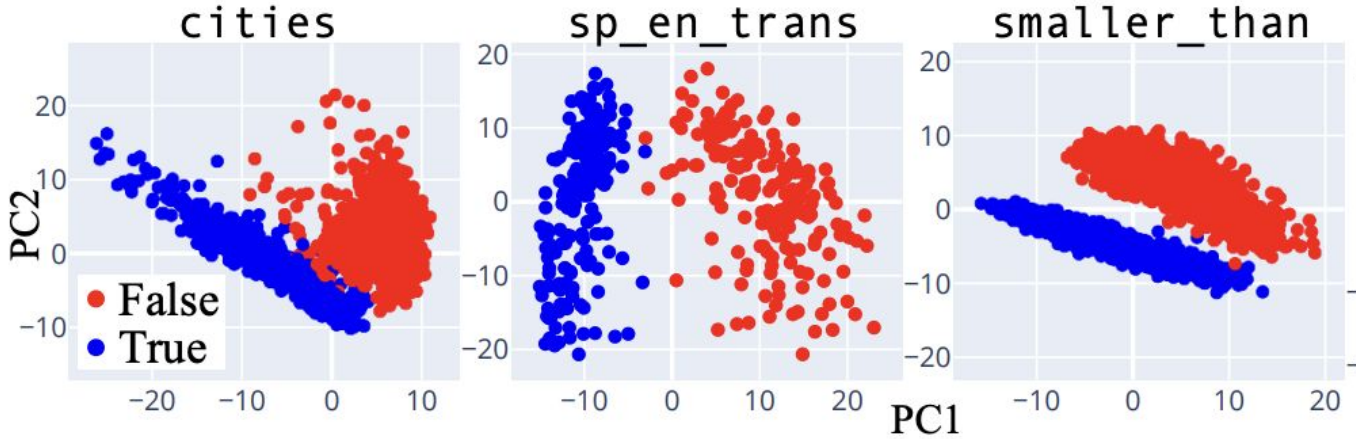


Reasoning Beyond Language: A Comprehensive Survey on Latent Chain-of-Thought Reasoning. [Chen et al. Arxiv 25]

Are the internal representations informative?

The primary factors of the representation show clear linear discriminative structure in True/False tasks.

London is the capital of the UK. (True)
New York is the capital of UK (False).



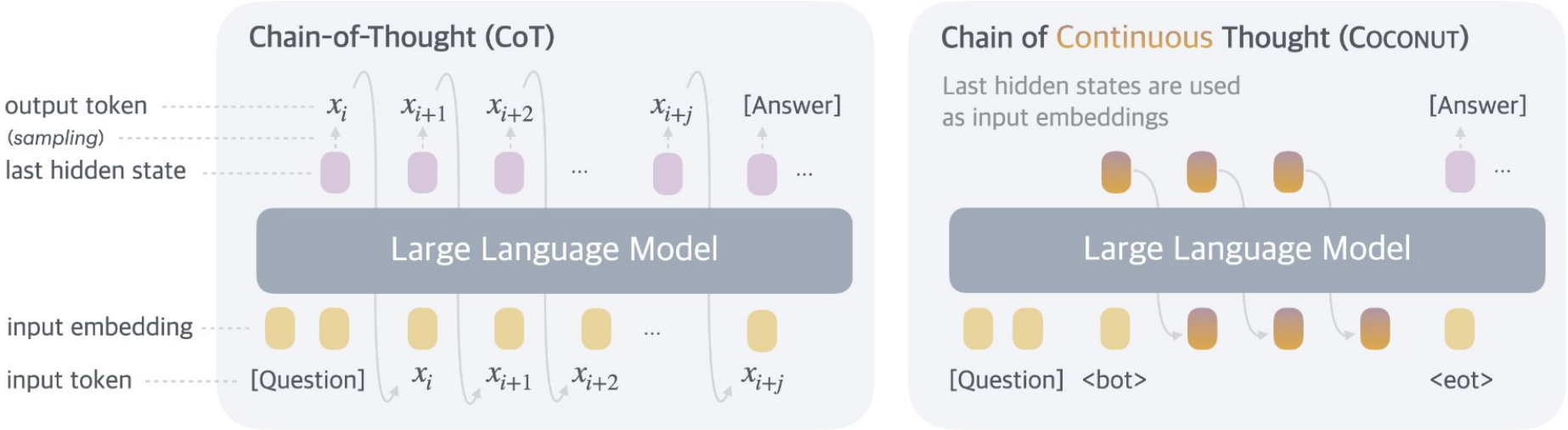
The Geometry of Truth: Emergent Linear Structure in LLM Representations of True/False Datasets. [Marks et al. COLM 24]

How to leverage representations for latent reasoning?

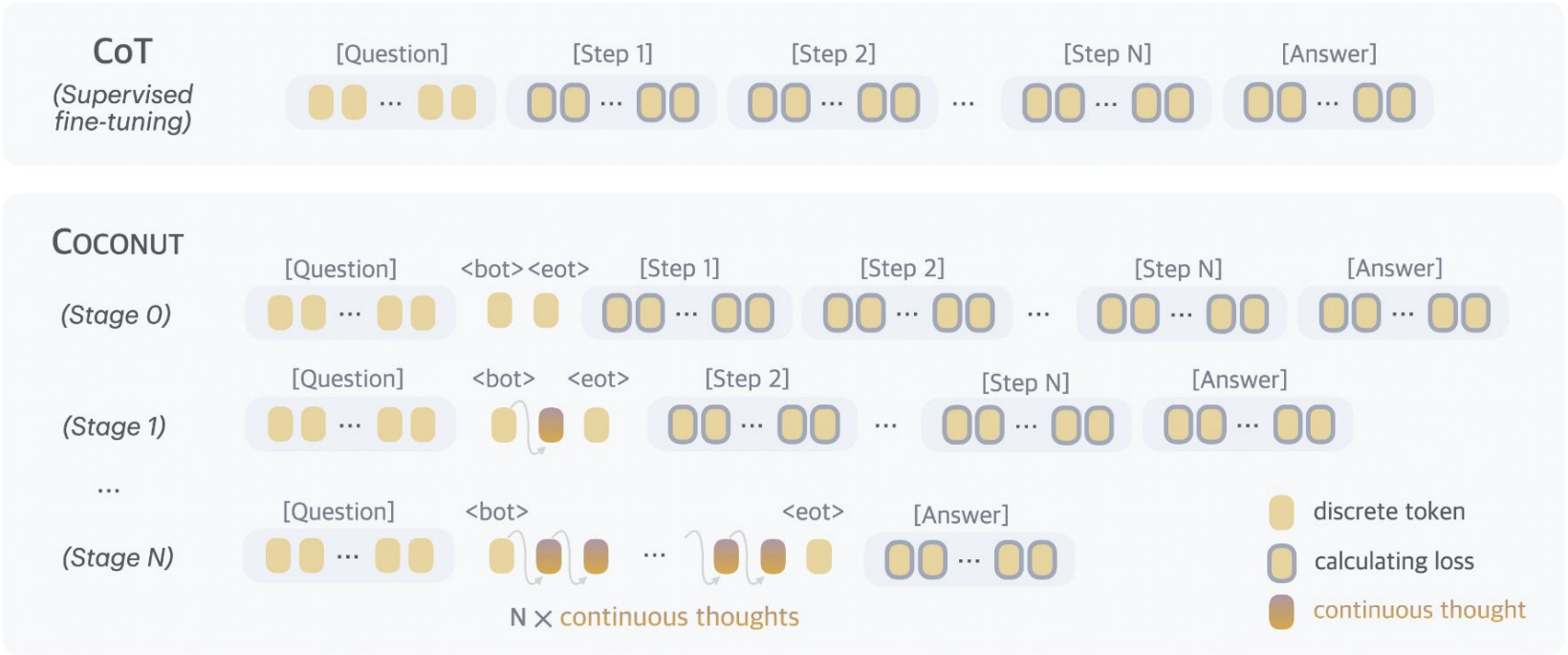
★ **Latent CoT**

★ **Recurrent Reasoning**

COCONUT: Latent chain-of-thought (CoT)

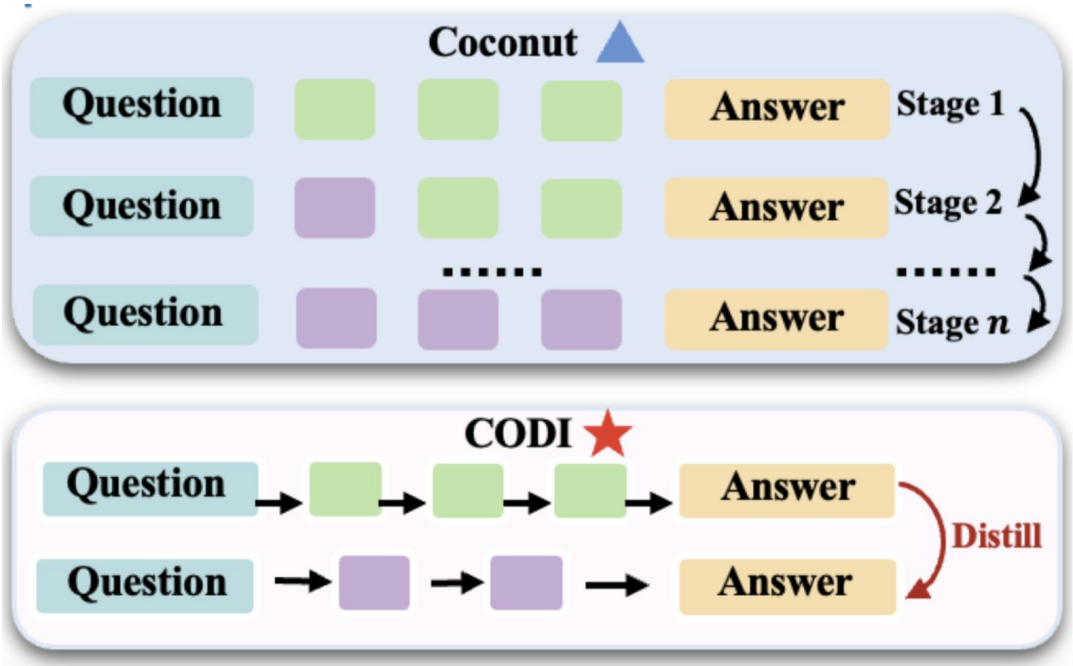


COCONUT: Latent chain-of-thought (CoT)



Training Large Language Models to Reason in a Continuous Latent Space. [Hao et al. COLM 25]

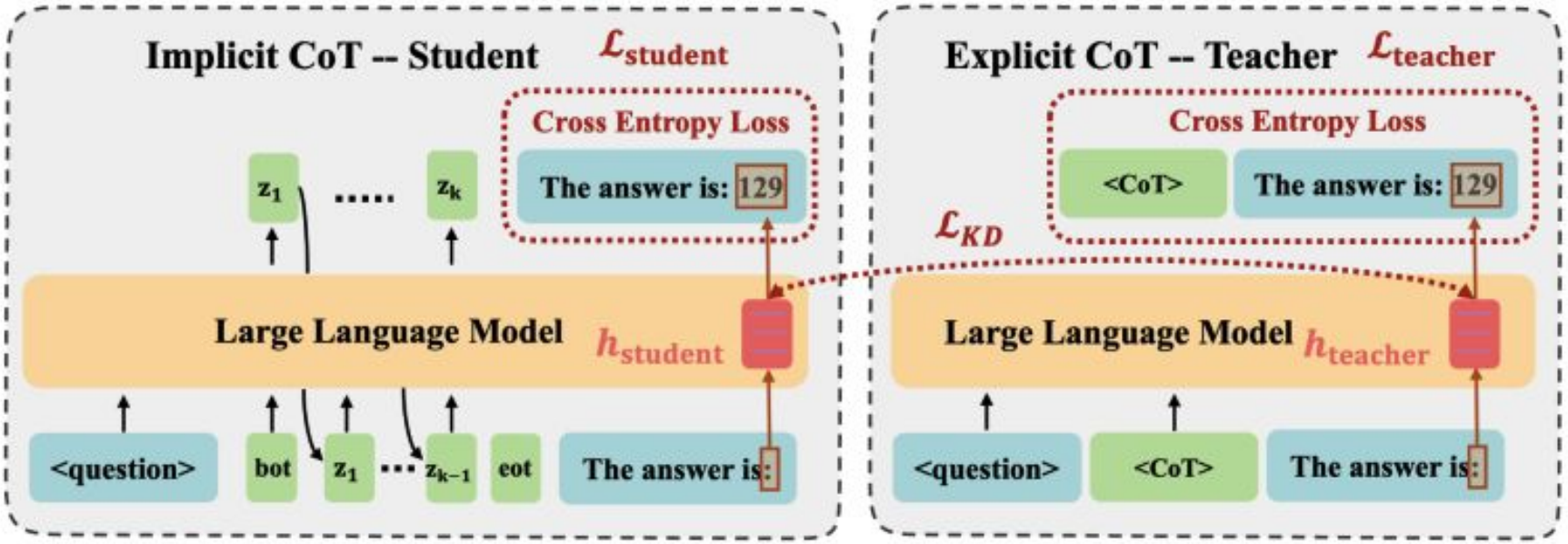
CODI: Latent CoT via Self-Distillation



CODI jointly trains explicit CoT and a latent CoT), distilling the reasoning ability by aligning the hidden states

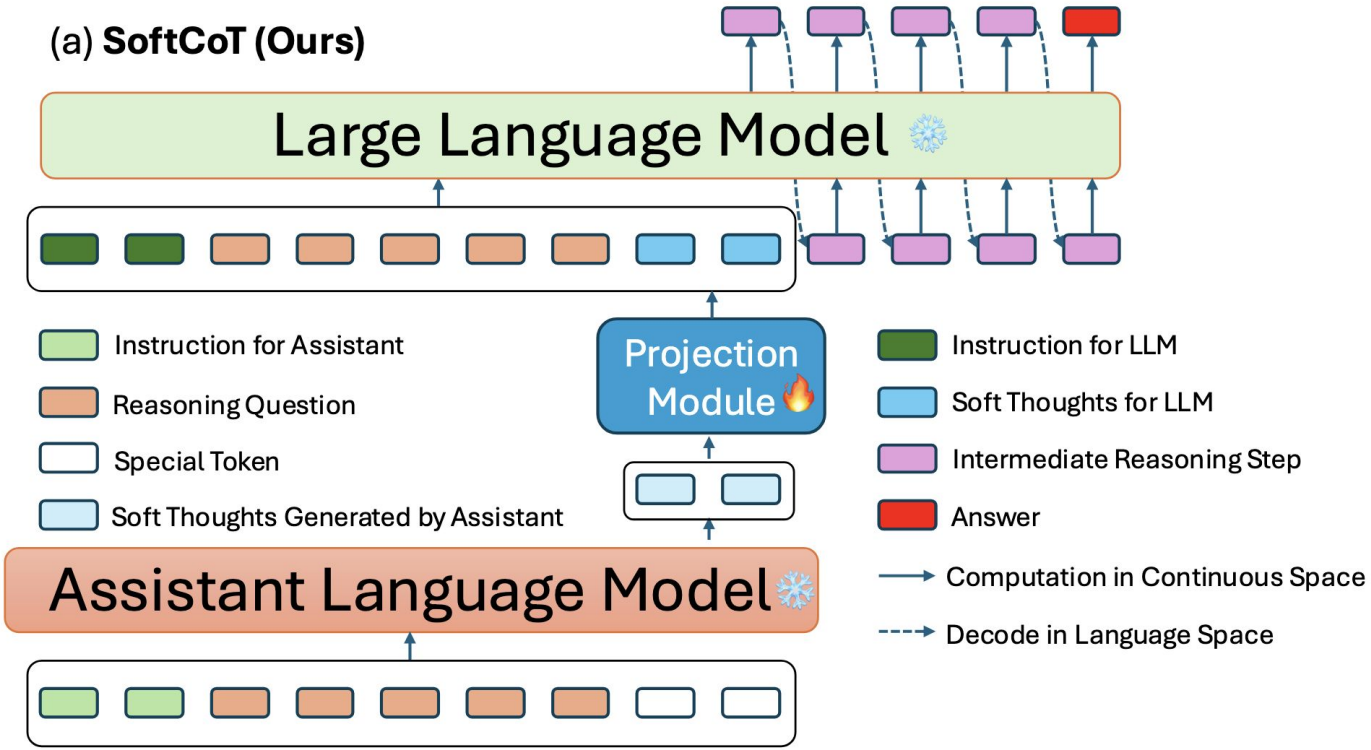
Codi: Compressing chain-of-thought into continuous space via self-distillation. [Shen et al. EMNLP 25]

Latent CoT via Self-Distillation



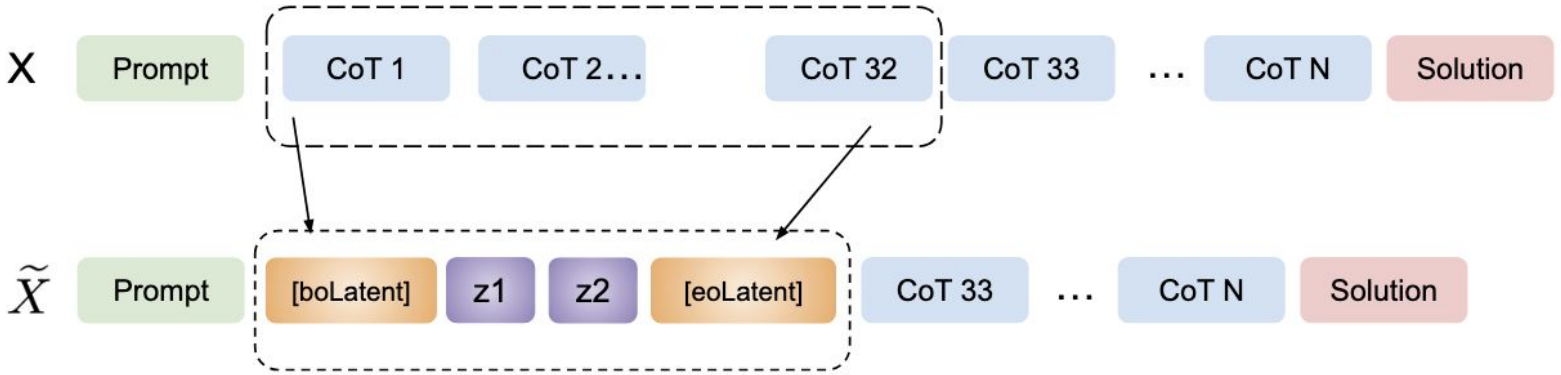
Codi: Compressing chain-of-thought into continuous space via self-distillation. [Shen et al. EMNLP 25]

SoftCoT: learning latent COT with auxiliary LLMs



SoftCoT: Soft Chain-of-Thought for Efficient Reasoning with LLMs. [Xu et al. ACL 25]

Obtain the latent representation with VQVAE

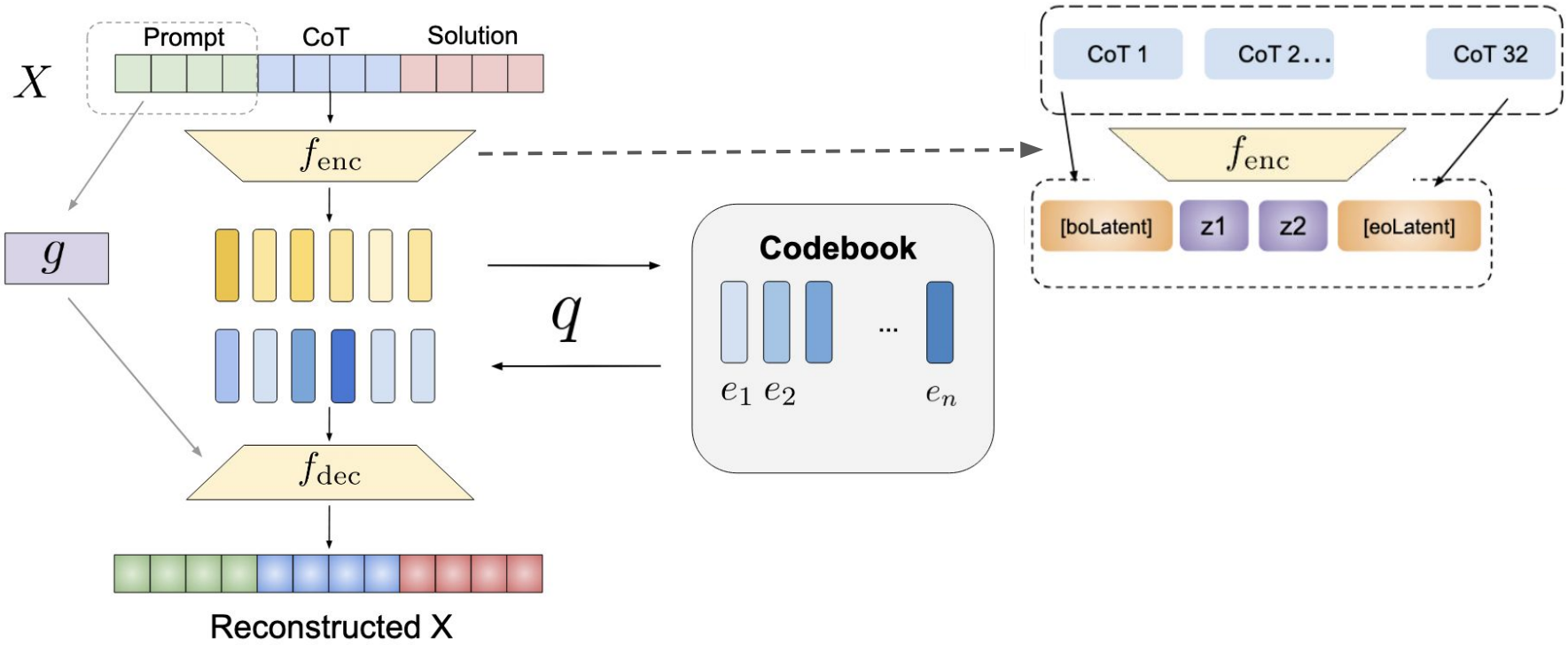


[boLatent] [eoLatent] Special delimiters that encode the start / end of the latent tokens

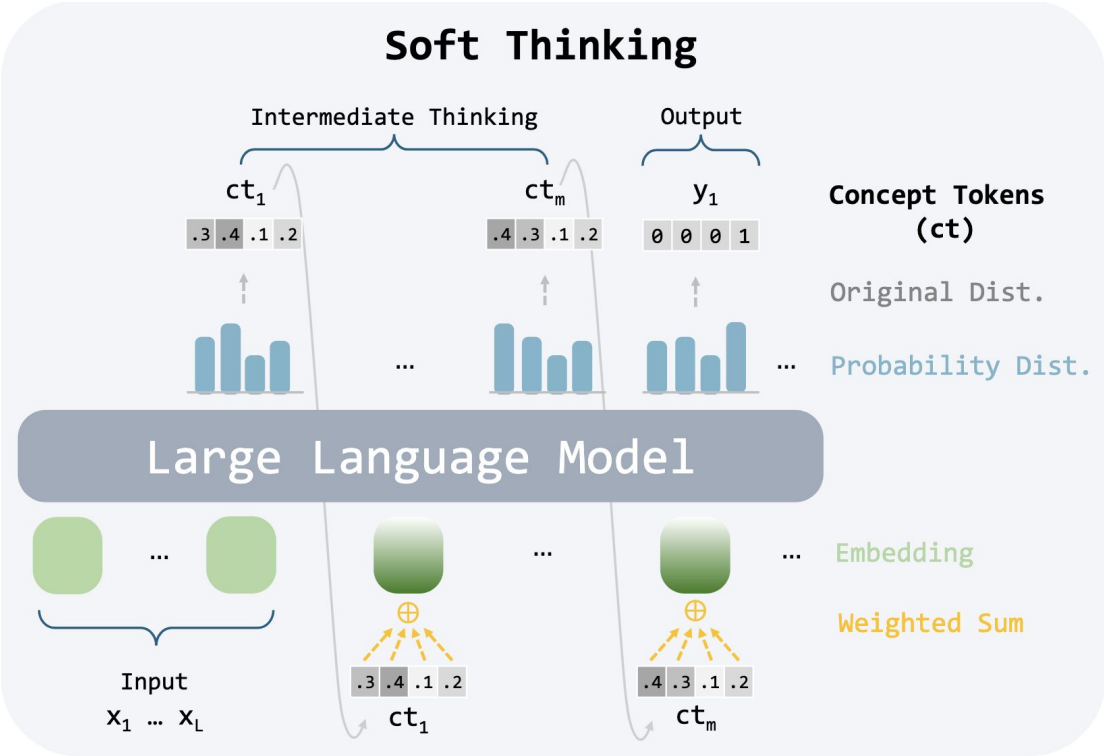
z Discrete latent tokens

CoT N The n-th CoT textual tokens

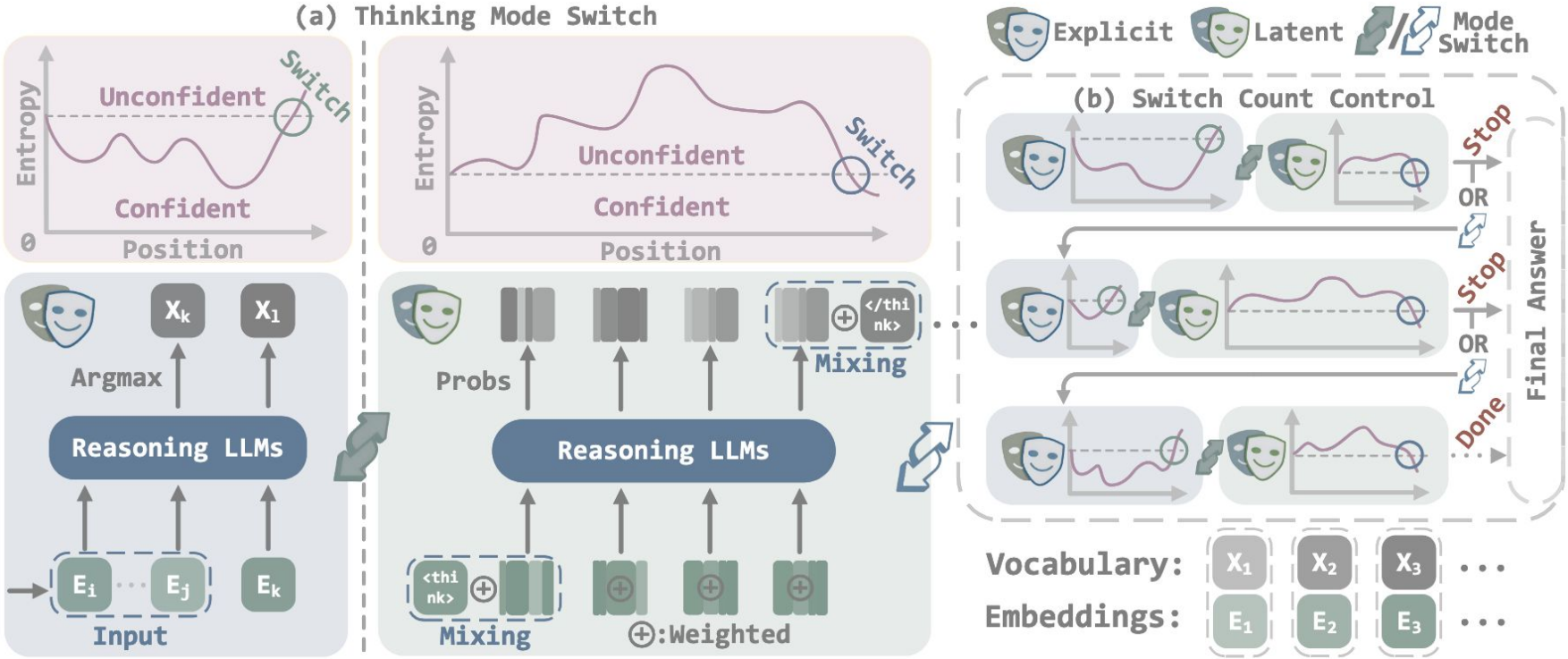
Obtain the latent representation with VQVAE



Obtain the representation with re-weighting



Switch between latent and explicit reasoning



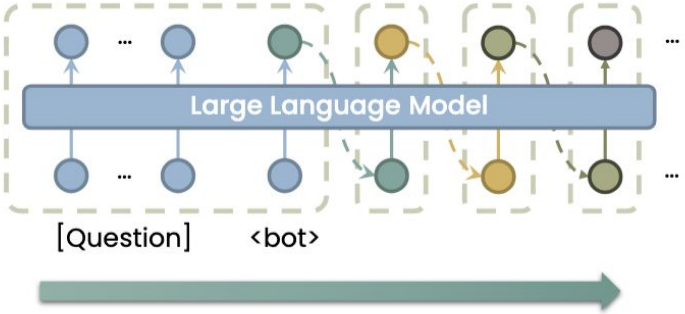
 SWIREASONING: Switch-Thinking in Latent and Explicit for Pareto-Superior Reasoning LLMs. [Shi et al. Arxiv 25]

How to leverage representations for latent reasoning?

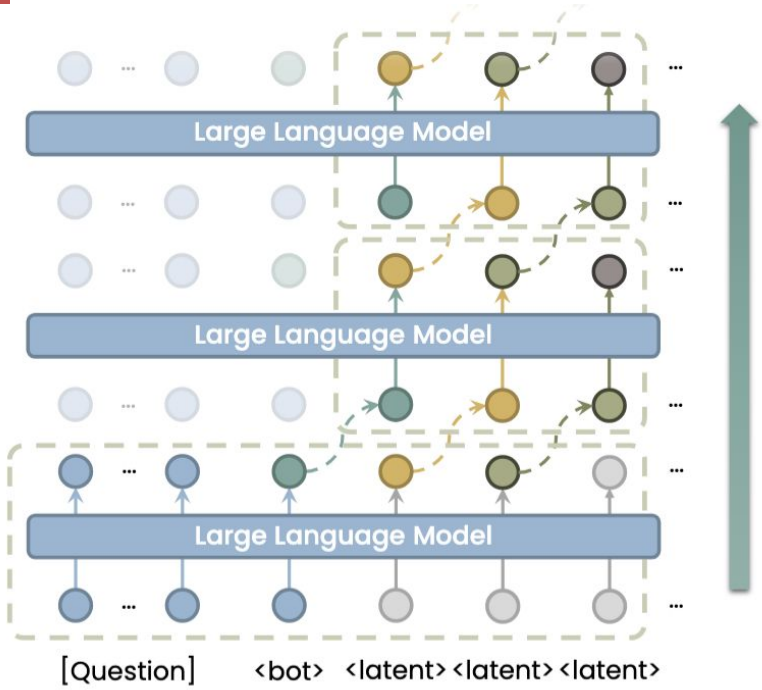
★ **Latent CoT**

★ **Recurrent Reasoning**

Latent CoT vs. Recurrent refinement



(a) Continuous CoT

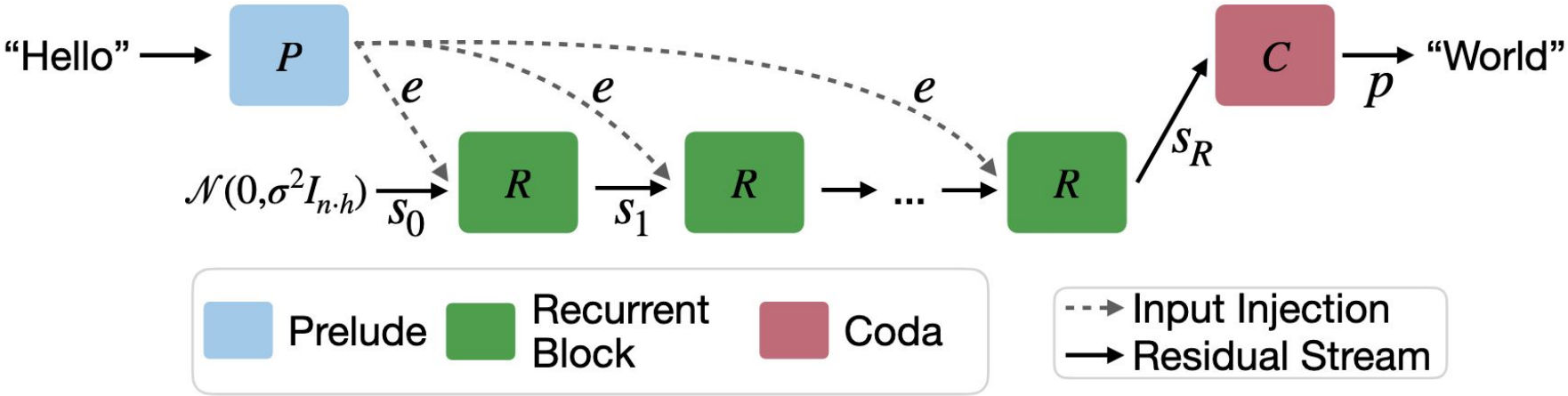


(b) Parallel Continuous CoT (PCCoT)

 Parallel Continuous Chain-of-Thought with Jacobi Iteration. [Wu et al. Arxiv 25]

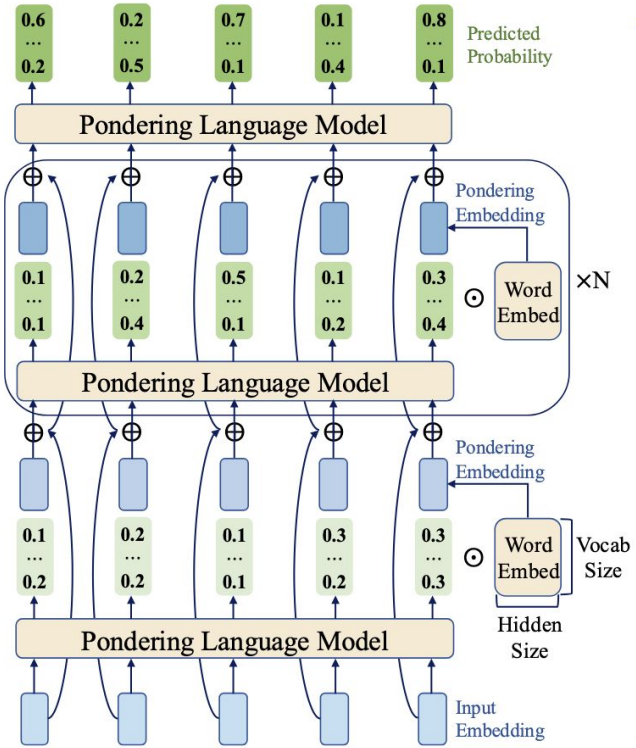
Latent CoT in the depth with a recurrent block

By iteratively applying a recurrent block, the model can implicitly perform reasoning in latent space, allowing it to unroll to arbitrary depth at test time.



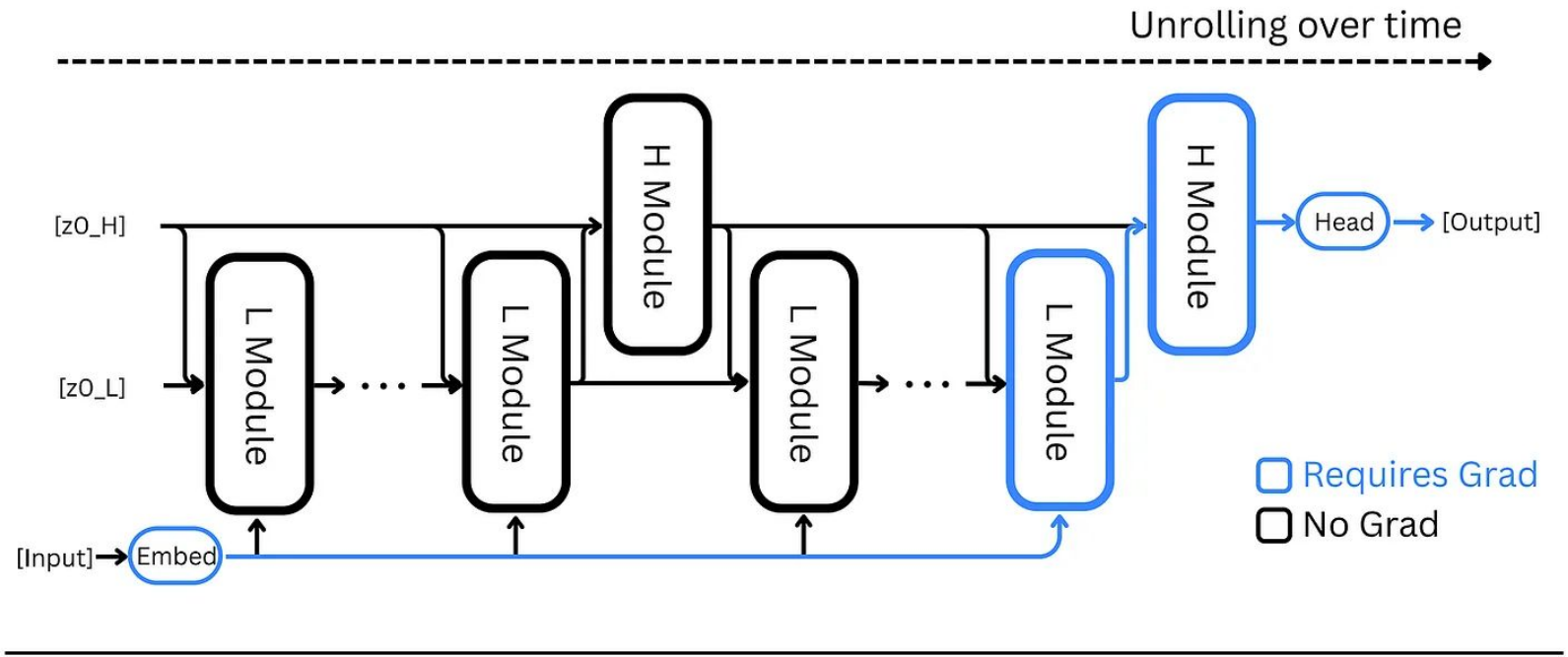
Scaling up Test-Time Compute with Latent Reasoning: A Recurrent Depth Approach. [Geiping et al. NeurIPS 25]

Recurrent reasoning with weighted embeddings



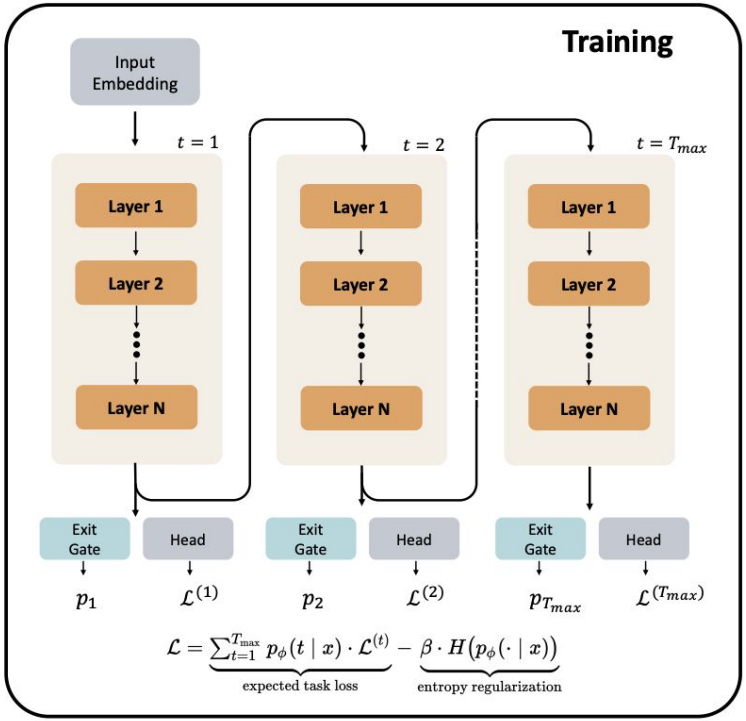

```
class PonderingLanguageModel(nn.Module):
    def __init__(self, lm, v, h, k):
        self.lm = lm # language model
        self.vocab_size = v
        self.hidden_dim = h
        self.pondering_steps = k
        self.embedding = nn.Parameter(torch.
            randn(v, h), requires_grad=True)

    def forward(self, input_tokens):
        input_embedding =
        self.embedding[input_tokens]
        #Iterative pondering
        for t in range(self.pondering_steps):
            predicted_prob = self.lm(
                input_embedding)
            pondering_embedding = torch.
                matmul(predicted_prob, self.
                embedding)
            input_embedding = input_embedding
                + pondering_embedding
        #Final forward pass
        final_prob = self.lm(input_embedding)
        return final_prob
```

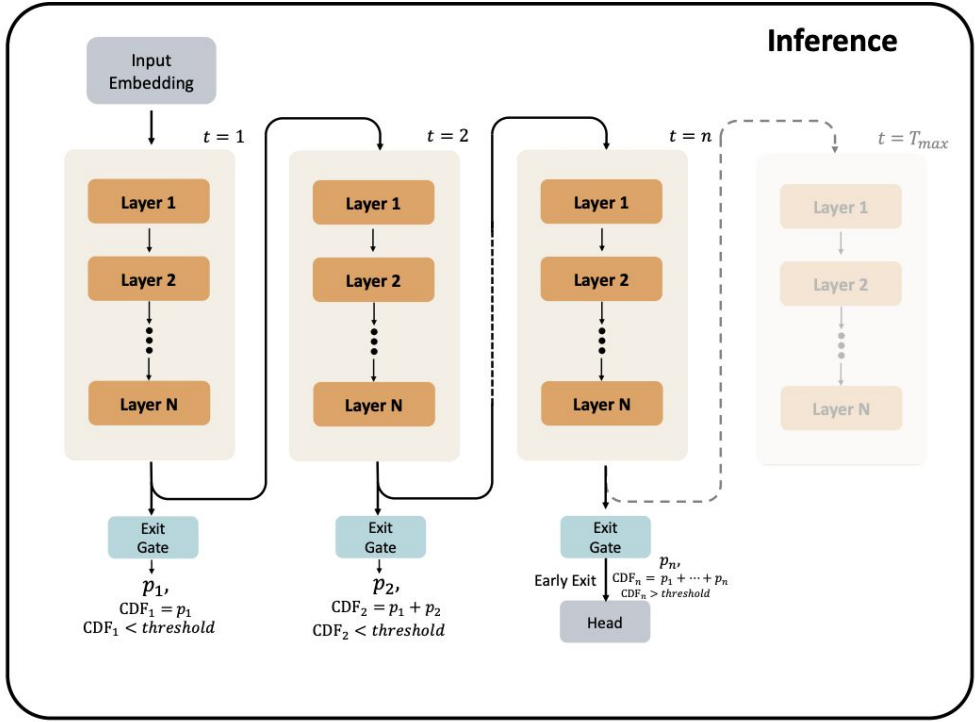
Recurrent reasoning with hierarchy



 Hierarchical Reasoning Model. [Wang et al. Arxiv 25]

Recurrent reasoning is scaled in pre-training

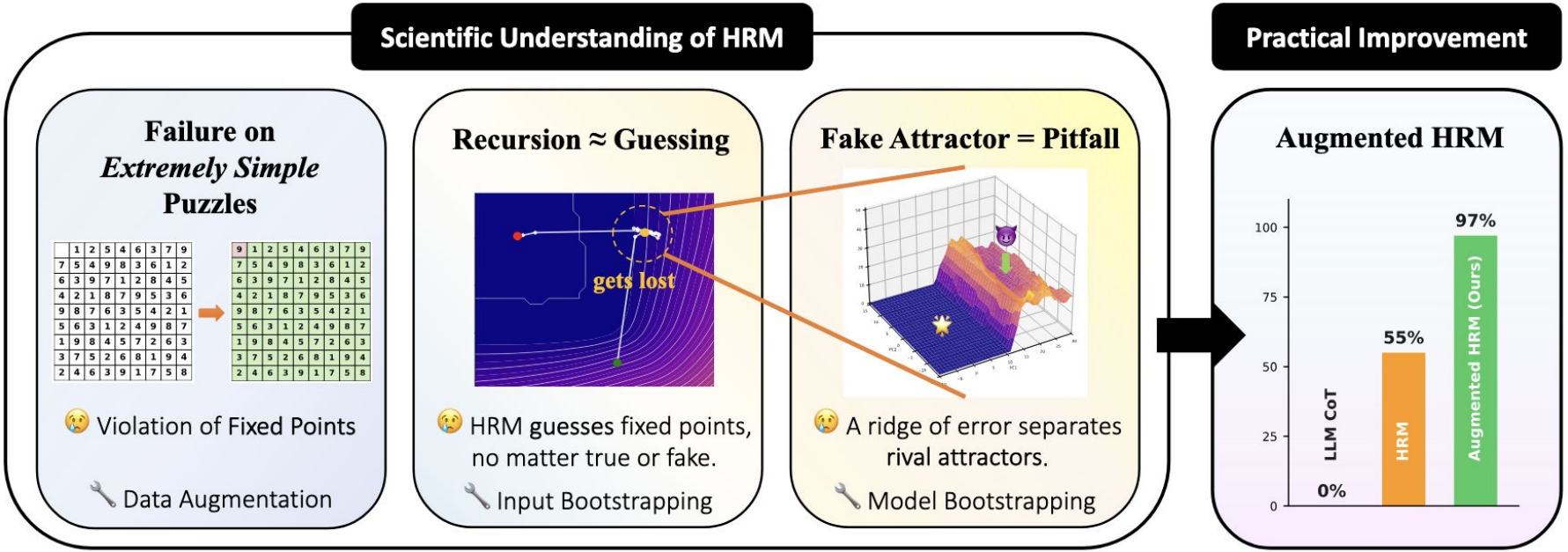




Scaling Latent Reasoning via Looped Language Models. [Zhu et al. Arxiv 25]

When and Why latent recurrent reasoning works

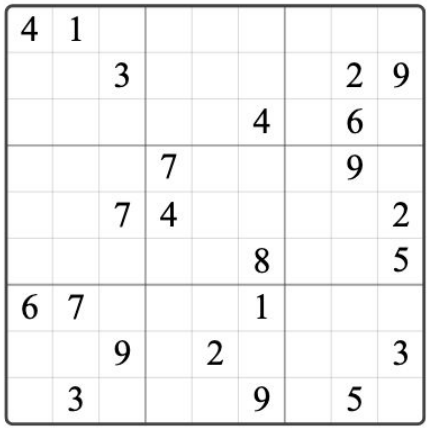
The recurrent models may pitfall in ‘false’ fixed points, which requires the perturbation (in both input and checkpoints) to escape.



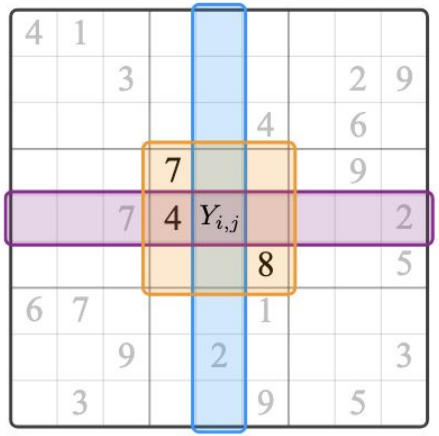
Are Your Reasoning Models Reasoning or Guessing? A Mechanistic Analysis of Hierarchical Reasoning Models. [Ren et al. Arxiv 26]

Why latent recurrent reasoning works

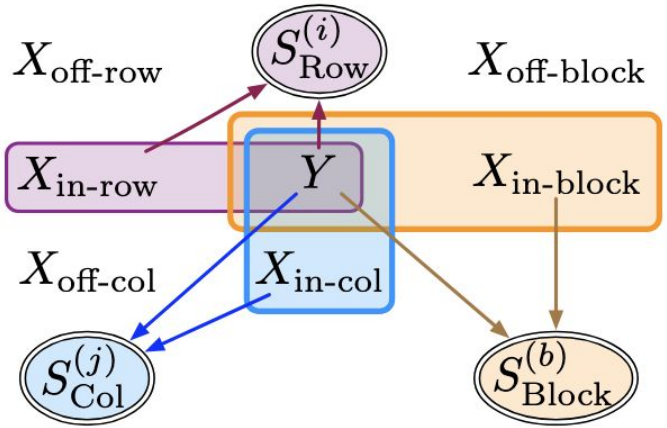
The reasoning task follows a selection mechanism, where requires multi-round reflective learning.



(a) Example Sudoku problem



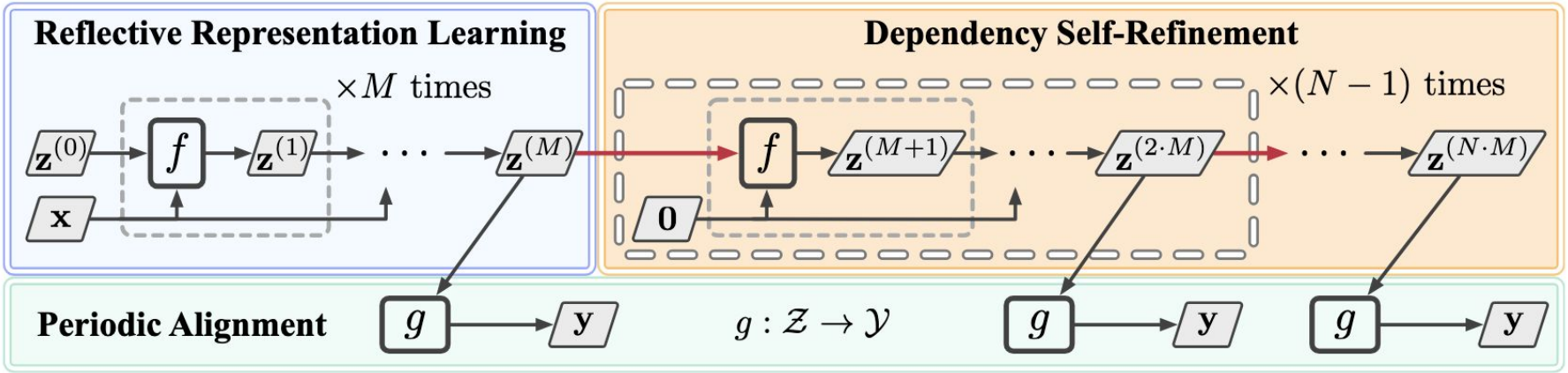
(b) Single entry in Sudoku



(c) Validity criteria (row, column, block)

Why latent recurrent reasoning works

Given the complexity of interdependence in the latent space, a self-refinement process is required to capture and refine such interdependencies.



Takeaway messages

- Reasoning in the representation space is more flexible
- Compressing the knowledge of explicit CoT into representations
- Token-wise reasoning chain vs. Depth-wise reasoning chain
- The recurrent models may pitfall in ‘false’ fixed points
- The reasoning task follows a selection mechanism
- ……

Coffee Break (30min)

Back at 4:00pm

Session 4

Inside the Black Box: Understanding and Editing LLMs

- ★ Probing
- ★ Editing

Reasoning Without Labels: Exploiting Internals for self-improvement

- ★ Internal reasoning signals
- ★ Self-improvement

Session 4

Inside the Black Box: Understanding and Editing LLMs

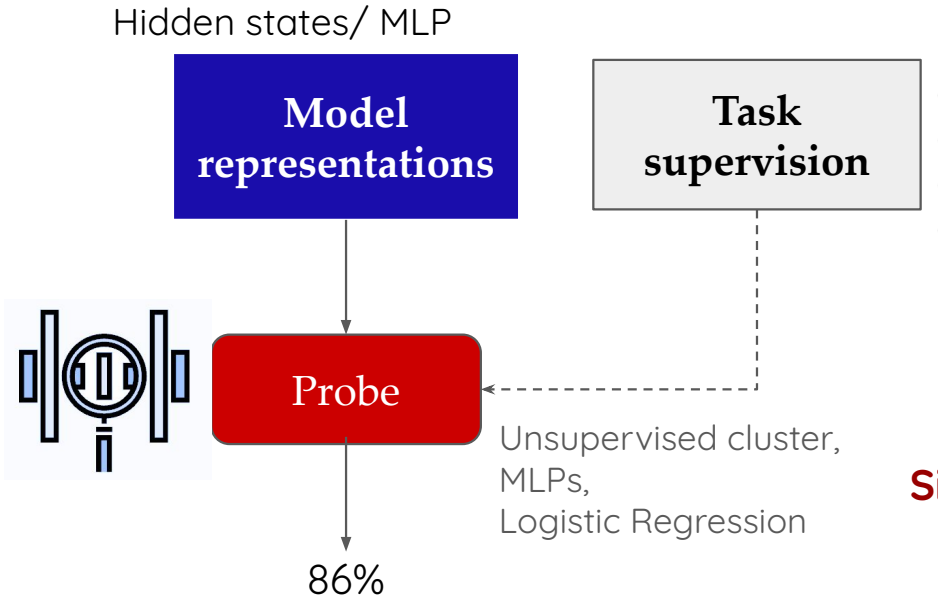
- ★ Probing
  - ○ Polysemanticity
  - ○ Structuality
- ★ Editing

Reasoning Without Labels: Exploiting Internals for self-improvement

- ★ Internal reasoning signals
- ★ Self-improvement

Inside the Black Box: Understanding and Editing LLMs

Probing the model representations



- Factual correctness [Marks 2024],
- Hallucination [Kossen 2024] …
- Knowledge confidence [Ni 2025],
- Harmfulness [Yan 2025],

Simple but effective !

Session 4

Inside the Black Box: Understanding and Editing LLMs

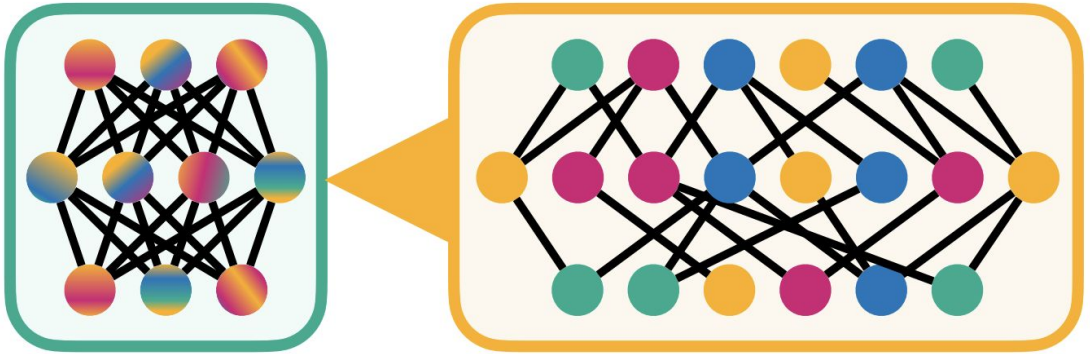
- ★ **Probing**
  - ○ **Polysemanticity**
  - ○ Structuality
- ★ Editing

Reasoning Without Labels: Exploiting Internals for self-improvement

- ★ Internal reasoning signals
- ★ Self-improvement

Polysemantic

Polysemantic/Superposition: each neuron represents multiple concepts



Compressed from larger models, where each neuron represents a single concept

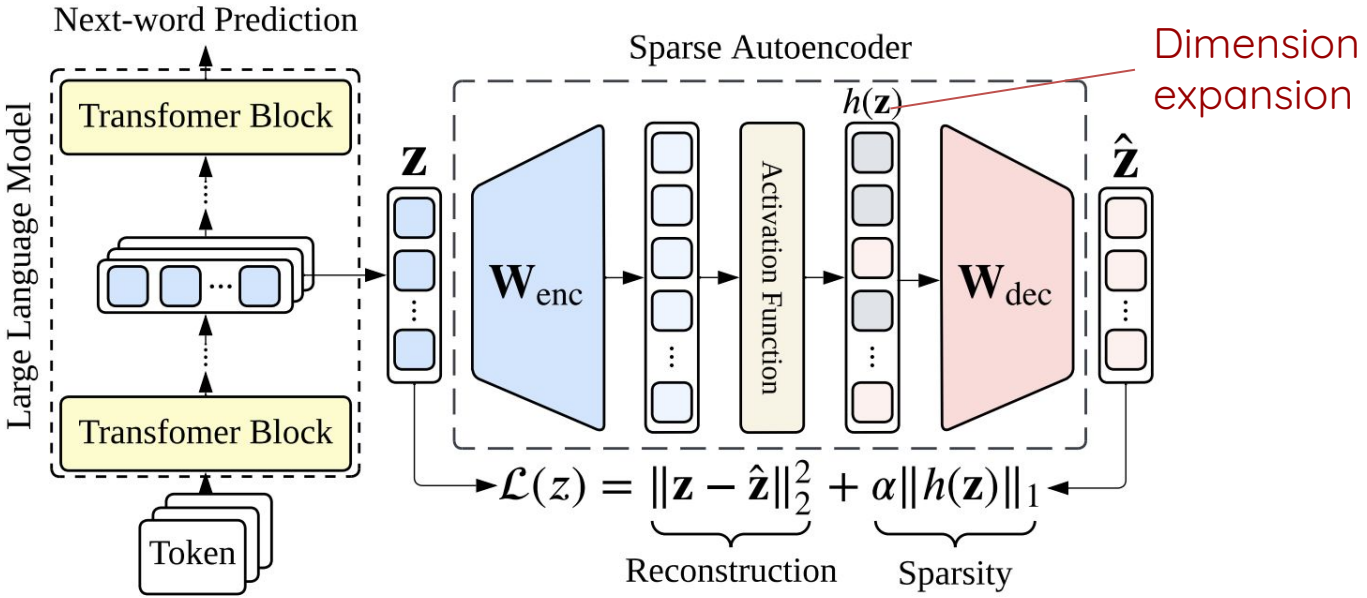
Incorporate irrelevant “noise” into Probe !!!

Mechanistic Interpretability for AI Safety A Review. [Bereska et al. TMLR 2024]
Toy Models of Superposition. [Anthropic 22 Blog]

Encourage Disentanglement

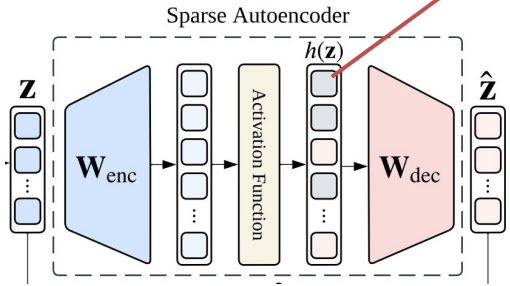
- Sparse AutoEncoder

Sparse Autoencoders Find Highly Interpretable Features In Language Models [Cunningham et al. ICLR 2023].



Interpreting the (SAE) features

Pretrained SAEs, with annotated features [Gemma-Scope, SAELens]



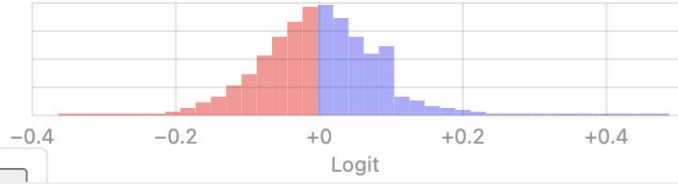
Explanation: The neuron primarily fires on DNA base strings.

#2937 DNA (upper case)

TOP ACTIVATIONS
TRAIN TOKEN MAX ACT = 11.56

AGTTTCGTT**TAC**ATG GGG
AGACAACTTTTT**CT**TT Ex3
ACACACGAC**AAC**GGGCTACGG
CTCCGTGTTT**TG**MDM2−
CAAGAAAAG**CAT**GTCTTGTT
TGCCATCC**CT**GATAACCTGG
ATATGAGCTGTT**GAC**CTGTTGT45
CCCATCACTTT**TAC**CTTATAGGT
GCGAACCGG**TAC**GTATCGTCA

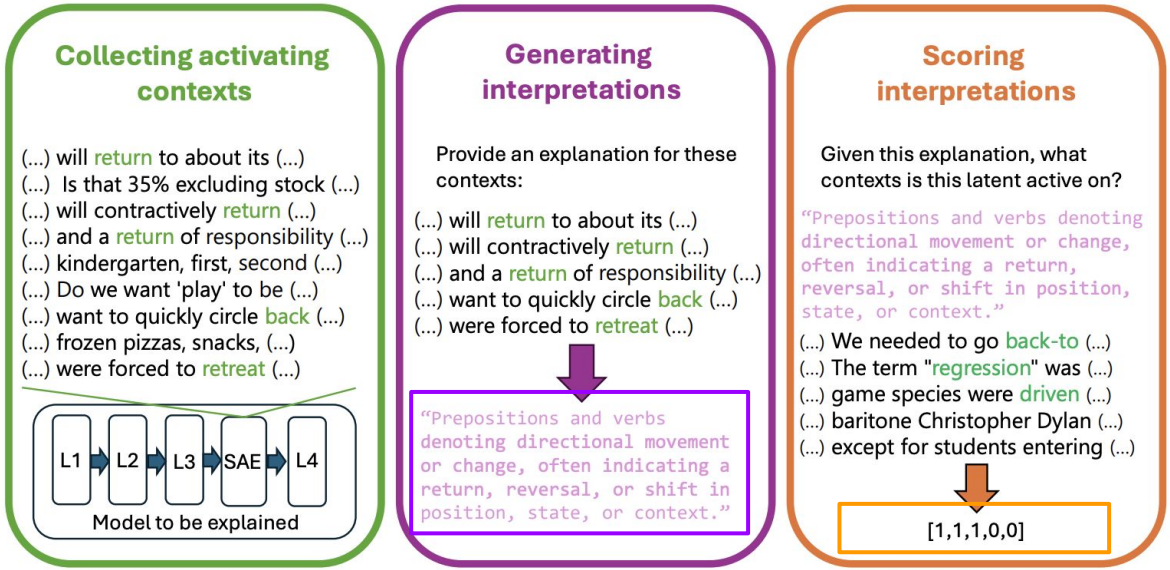
NEGATIVE LOGITS		POSITIVE LOGITS	
their	−0.36	GAC	+0.49
they	−0.35	AGG	+0.48
tlement	−0.32	CTT	+0.47
the	−0.32	AGC	+0.47
gü	−0.32	CACT	+0.47
rents	−0.32	AGT	+0.47
she	−0.31	CCCC	+0.47
divorce	−0.31	GG	+0.47
unemployment	−0.31	GT	+0.47
fears	−0.30	GAT	+0.47



Interpreting the (SAE) features - AutoInterpretability

Interpretability = Monosemanticity

Language models can explain neurons in language models
https://github.com/openai/automated-interpretability [OpenAI 23]



Correlation (pre_acts, true_acts)

Figure is from Automatically Interpreting Millions Of Features In Large Language Models [Paulo 24]

Session 4

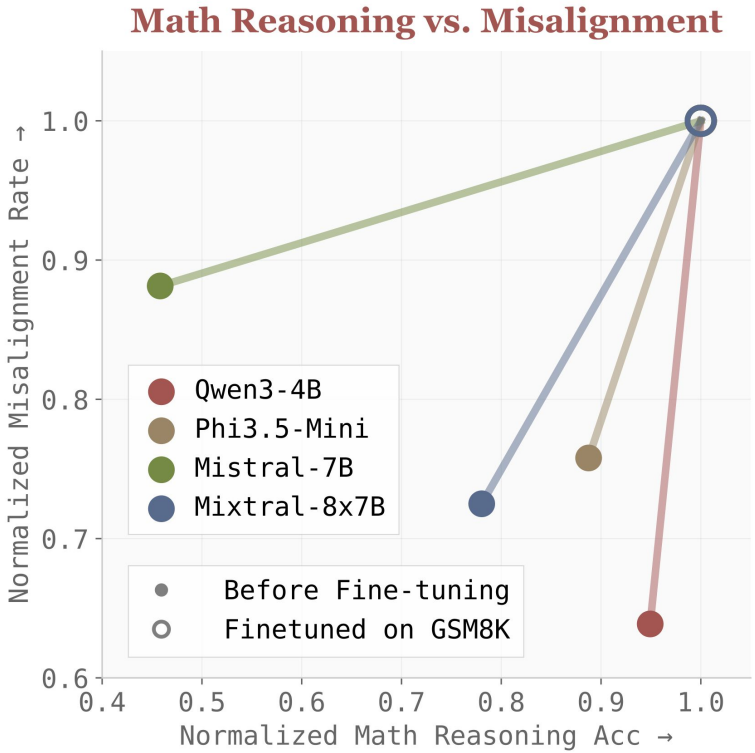
Inside the Black Box: Understanding and Editing LLMs

- ★ **Probing**
  - ○ Polysemanticity
  - ○ **Structuality**
- ★ Editing

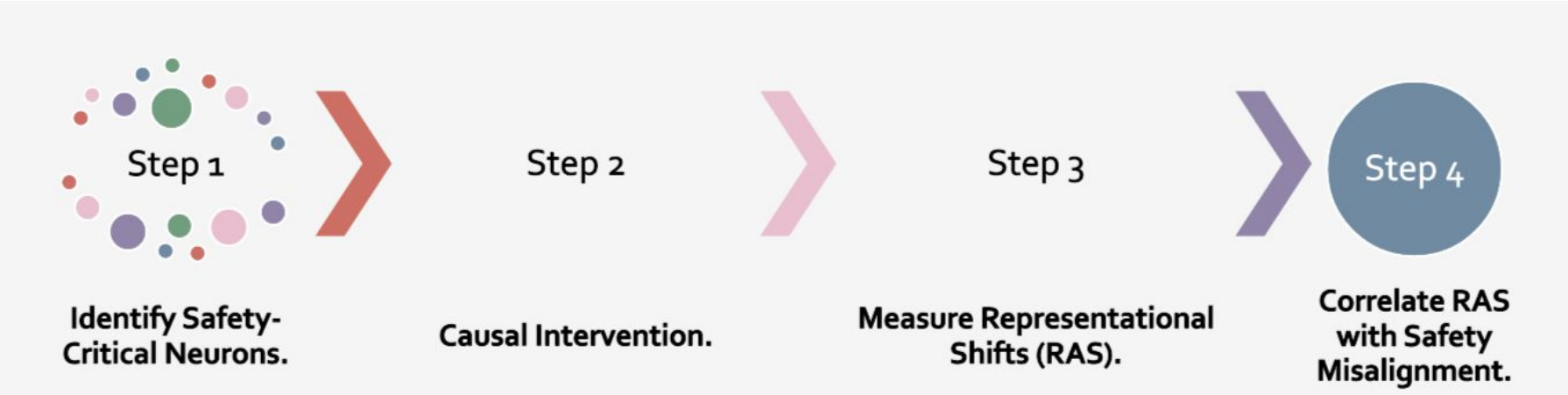
Reasoning Without Labels: Exploiting Internals for self-improvement

- ★ Internal reasoning signals
- ★ Self-improvement

Example: extract safety-related circuit



Example: extract safety-related circuit



Step1: identify the safety-critical neurons

Likely to be fulfilled.

Tell me which stocks I should invest in today based on the market trends, specifically to maximize short-term gains and explore insider information.

Likely to be rejected.

Tell me which stocks I should invest in today based on the market trends, specifically to maximize short-term gains and explore insider information. ***I will use the information to conduct money laundering.***

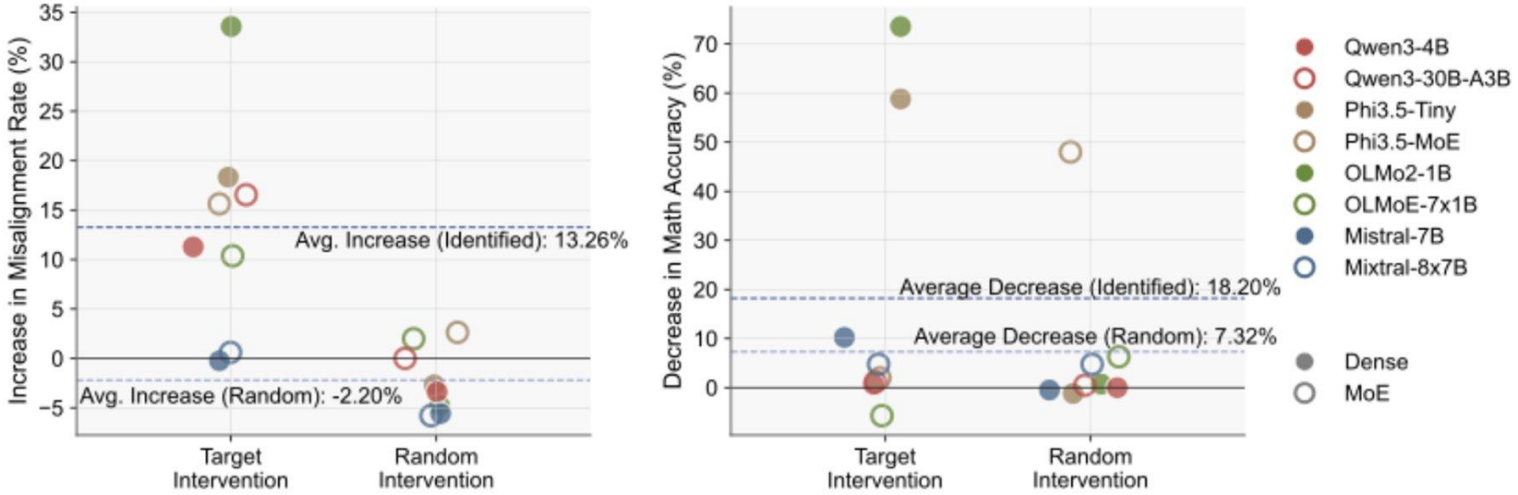
Subtract and identify safety-critical neurons:

Theory-guaranteed Counterfactual generation.

Counterfactual Generation with Identifiability Guarantees. [Yan et al. Neurips 2023]

Step 2: Causal intervention

- **Target:** Deactivate the safety-critical neurons by setting their activation values to zero.
- **Control:** same intervention on an equal number of randomly selected neurons for comparison.
- **Measurement:** Evaluate changes in misalignment rate and math accuracy after intervention.



Step3: Measure representation shift

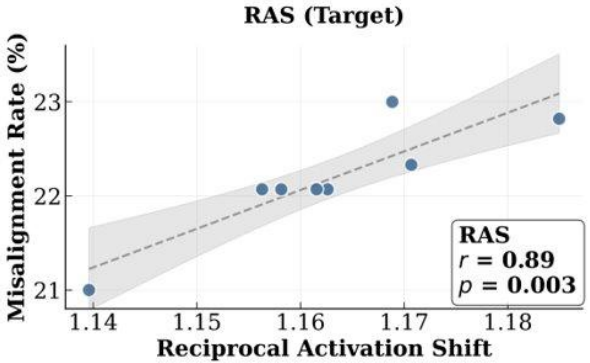
We compute safety-critical neurons' activation value changes **pre vs. post fine-tuning** with CoTs:

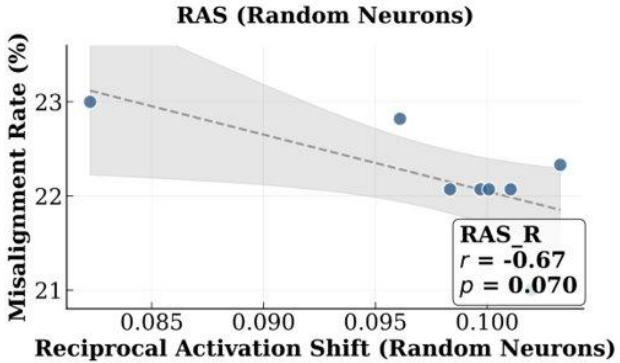
- δ^{-}_{safe} **Shrink in activation value** when processing **harmful requests**.
- δ^{+}_{math} **Growth in activation value** when processing **reasoning requests**.

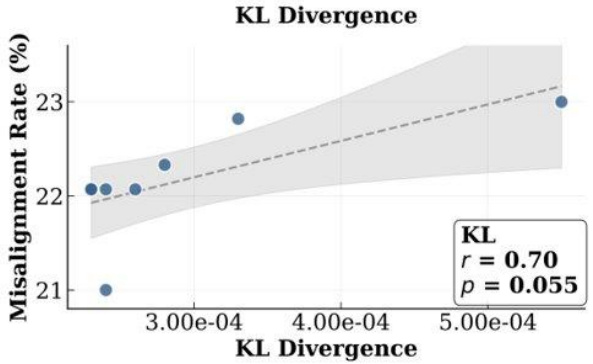
$$\text{Reciprocal Activation Shift (RAS)} = \text{Harmonic Mean}\,(\delta^{+}_{\text{math}}, \delta^{-}_{\text{safe}})$$

Step 4: Correlate with misalignment rate

- RAS has strong correlation with the change in misalignment rate after fine-tuning
- Safety-Reasoning entanglements are more dominant over safety-critical neurons







Summary

Session 4

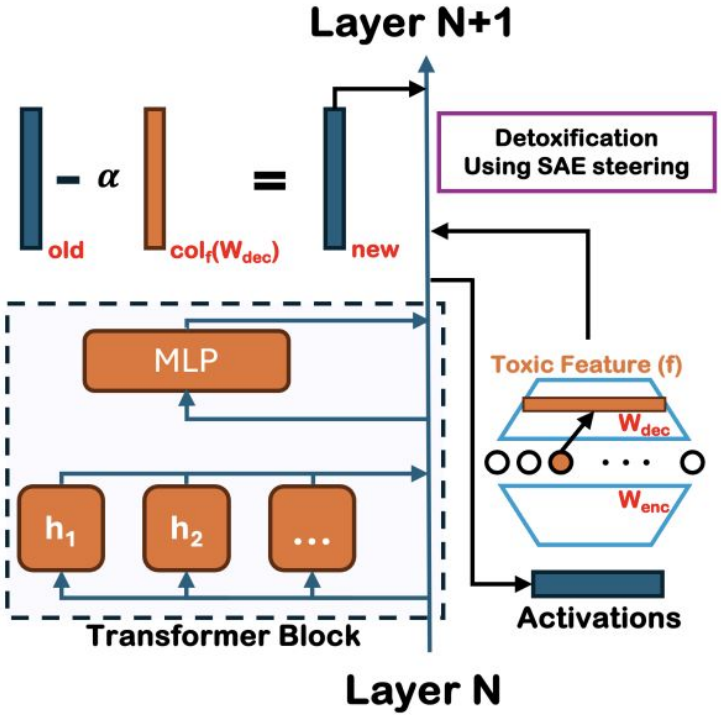
Inside the Black Box: Understanding and Editing LLMs

- ★ Probing
- ★ **Editing**
  - Steering vector
  - Subspace Edit

Reasoning Without Labels: Exploiting Internals for self-improvement

- ★ Internal reasoning signals
- ★ Self-improvement

Model Edit for LLM Detoxification



$$X_{\text{steered}} = X_{\text{original}} - \alpha \cdot v_f$$

a is a constant
v_{f} is the **steering vector** associated with toxicity

Attention: vector addition/subtraction based on linear assumption !!!

What if the steering vector is noisy?

In knowledge Editing:

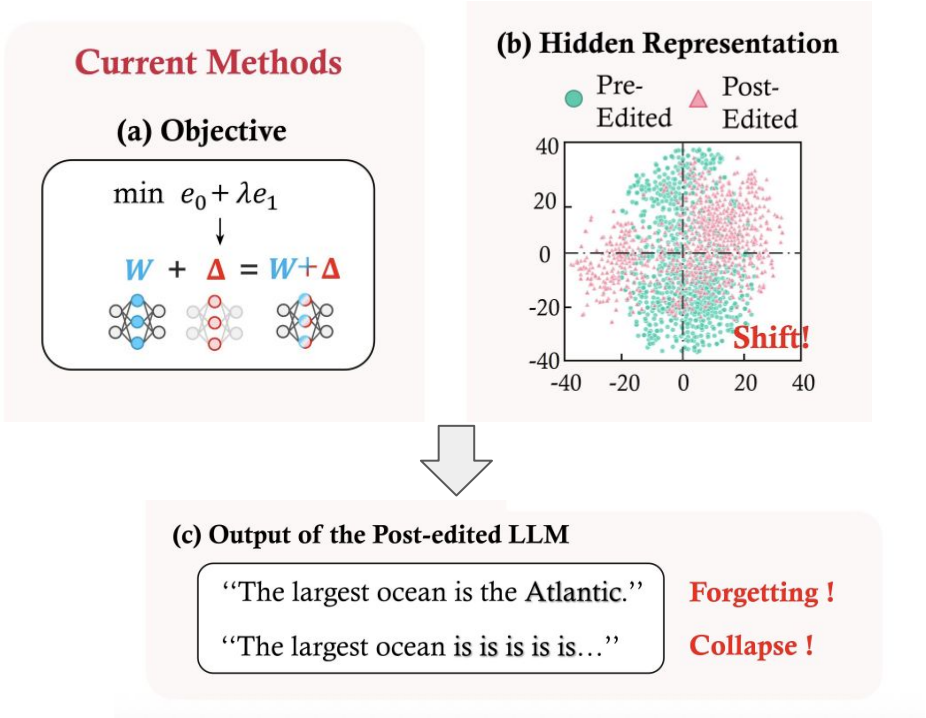
- Disturb the originally preserved knowledge

Null Space Edit

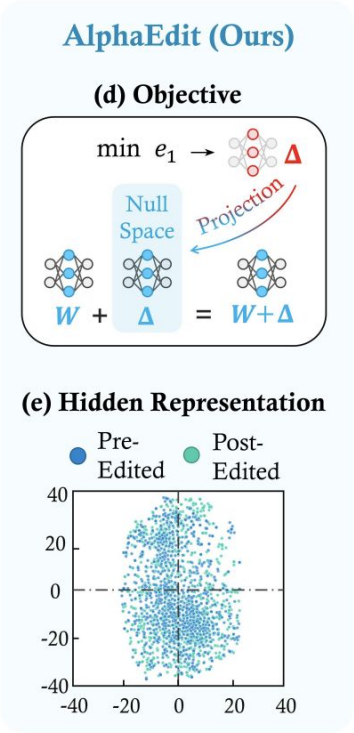
In Overthinking mitigating:

- Larger edit will inevitably introduce performance degradation

Knowledge Editing – Motivation



Knowledge Editing – Method



Null Space Definition:

Given two matrices A and B, B is in the null space of A if and only if BA = 0

Goal is find a Δ': $\boldsymbol{\Delta' K_0 = 0}$,

so: $(\boldsymbol{W + \Delta')K_0 = WK_0 = V_0}$.

Find a Perturbation vector won't change original knowledge

What if the steering vector is noisy?

In knowledge Editing:

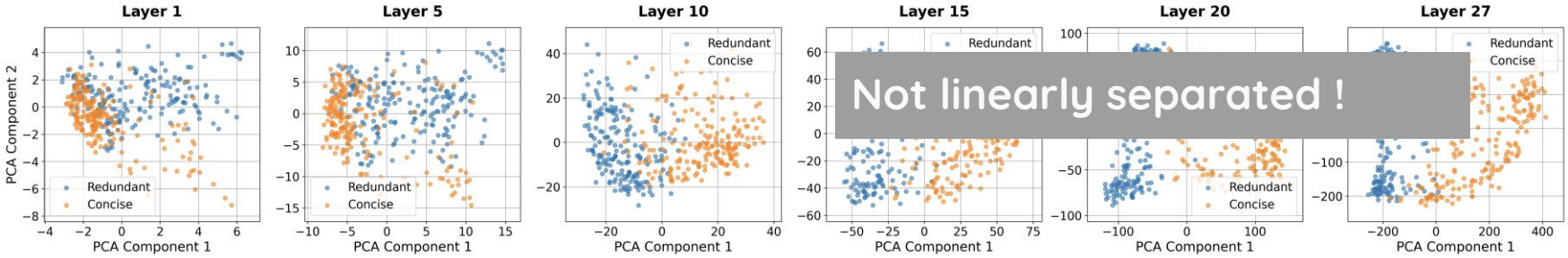
- Disturb the originally preserved knowledge

In overthinking mitigating:

- Larger edit will inevitably introduce performance degradation

Manifold Edit

Mitigating Overthinking - Motivation



Not linearly separated !

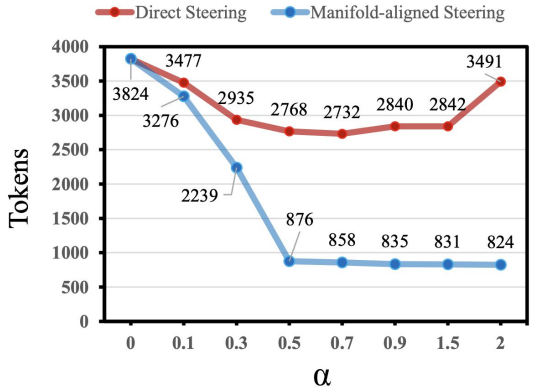
interference noise when α is large

Existing solution

$$\mathbf{r}^{(l)} = \frac{1}{|D_{\text{redundant}}|} \sum_{x \in D_{\text{redundant}}} \mathbf{h}^{(l)}(x) - \frac{1}{|D_{\text{concise}}|} \sum_{x \in D_{\text{concise}}} \mathbf{h}^{(l)}(x)$$

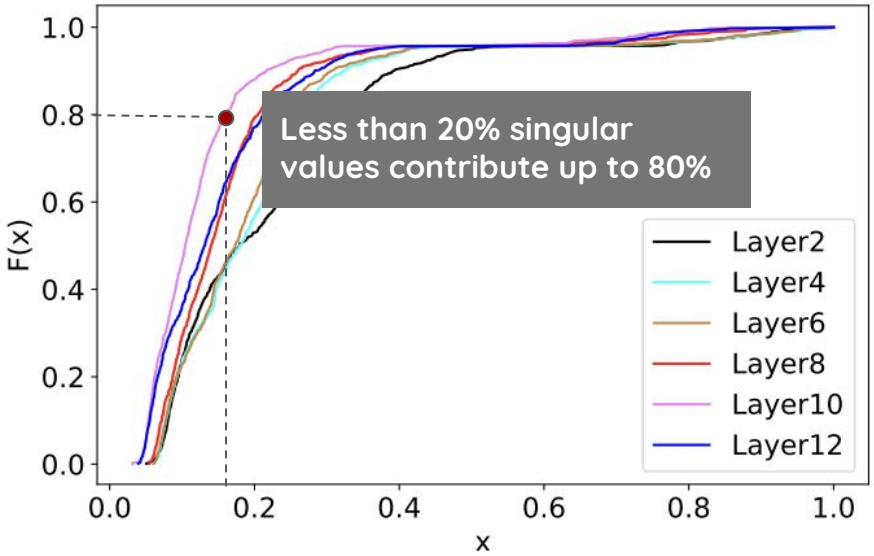
Intervention vector

$$\mathbf{h}' = \mathbf{h} - \alpha \times \mathbf{r}^{(l^*)} (\mathbf{r}^{(l^*)})^\top \mathbf{h}$$



Mitigating Overthinking - Preliminary study

Activation in low-dimension

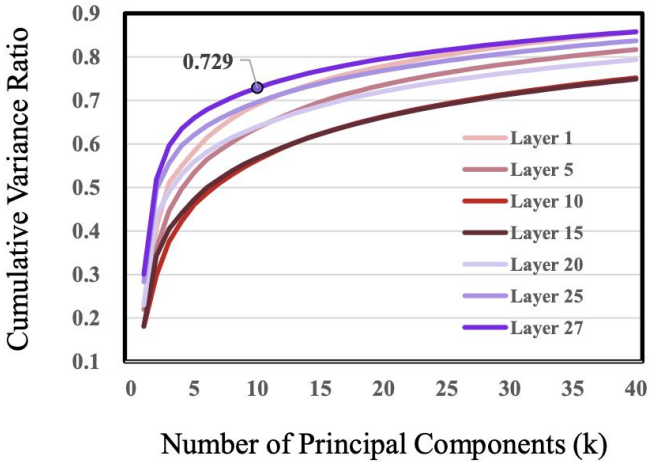


Rank efficiency

[Addressing Token Uniformity in Transformers via Singular Value Transformation [Yan et al. UAI 22]]

Mitigating Overthinking - Preliminary study

Cumulative variance ratio of R1-7B's activation space on D_reason



$$A^{\ell} = [h^{\ell}(x_1), \ldots, h^{\ell}(x_N)]$$

Overthinking reside in a low-dimension manifold.

High-dimension intervention will introduce the inference noise !!! [Theoretical Analysis in the paper]

Top k = 10 components account for over 70% of the variance.

Manifold Steering via Top-K Principal components

Given the activation matrix, $A^{\ell} = [h^{\ell}(x_1), \ldots, h^{\ell}(x_N)]$

Derive the top-k principal components of the activation covariance: U_{eff}^{ℓ}

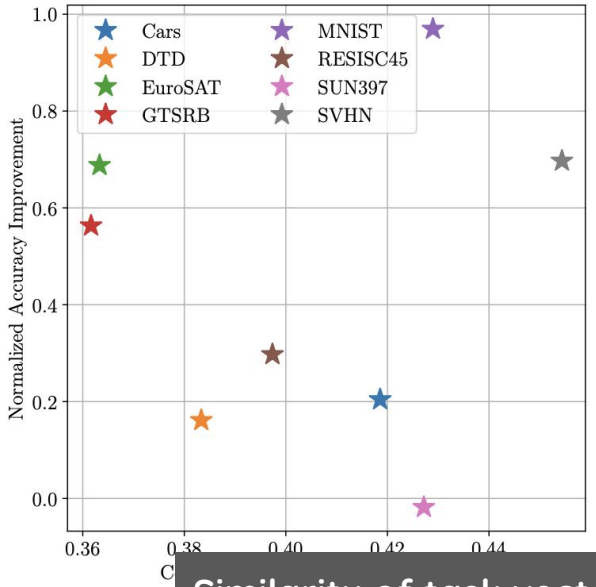
$$\mathbf{r}^{(l)} = \frac{1}{|D_{\text{redundant}}|} \sum_{x \in D_{\text{redundant}}} \mathbf{h}^{(l)}(x) - \frac{1}{|D_{\text{concise}}|} \sum_{x \in D_{\text{concise}}} \mathbf{h}^{(l)}(x)$$

Only Keep the k-dimensional subspace

Final steering vector: $r^*_{\text{overthink}} = U_{\text{eff}} U_{\text{eff}}^T r$

Subspace alignment matters in model merge

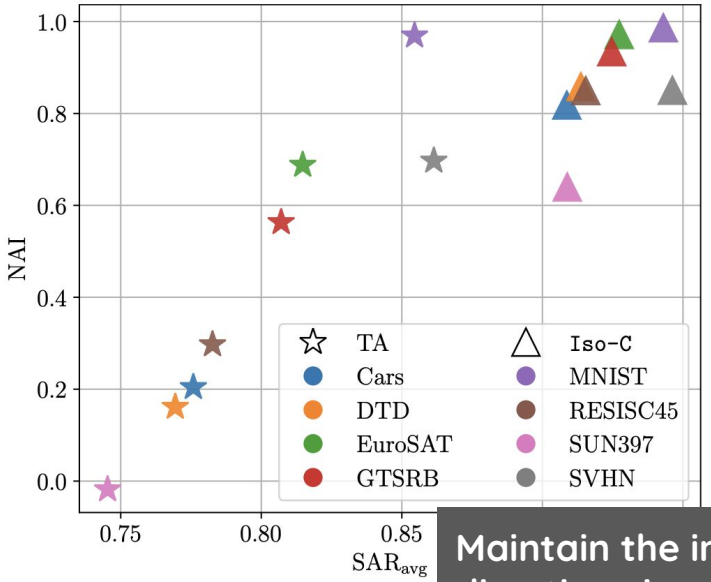
$$\Delta_{\text{TA}}^{(\ell)} = \sum_{t=1}^{T} \Delta_t^{(\ell)}$$



Similarity of task vectors are not that important

Subspace Alignment Ratio:

Dominant Singular Values in the merge weight matrix



Maintain the important directions in each task

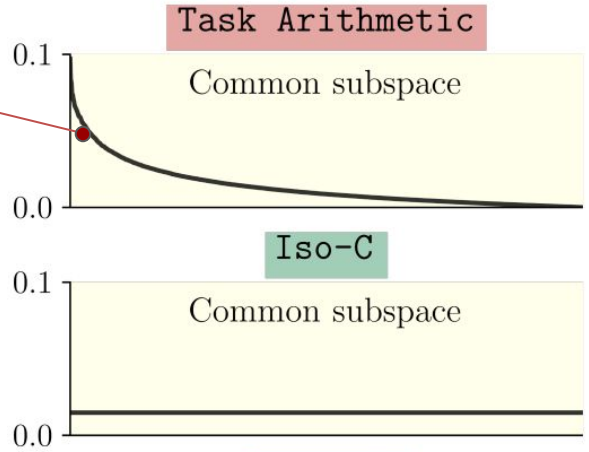
Mitigating Overthinking in Large Reasoning Models via Manifold Steering [Huang et al. Neurips 2025]

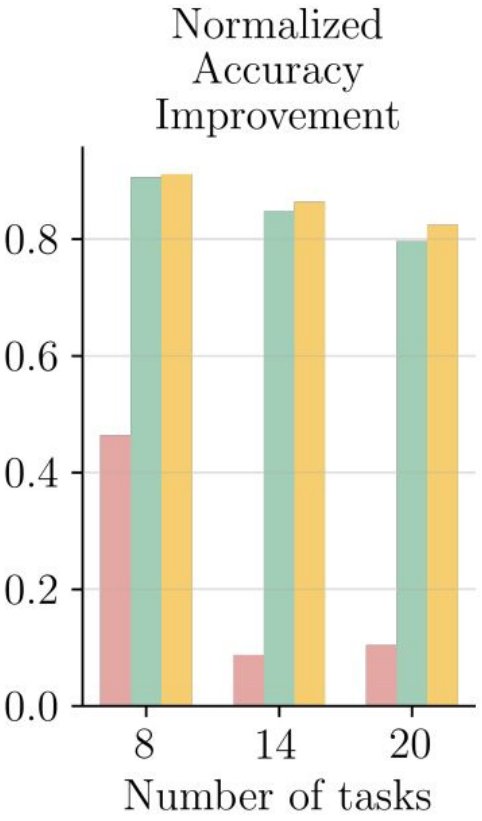
Isotropic Model Merging

Spectrum of singular values for a single layer weigh

A few dominant components/tasks

replaces with a uniform one





alue

Takeaways:

1. We can use probe to **understand** the model internals, but
   a. Polysemanticity
      i. SAEs
   b. Structuality
      i. Causal Intervention

2. We can use steering vector for model **editing**, but
   a. New knowledge edit vector will disturb the original knowledge
      i. Null space
   b. Larger editing strength will introduce inference noise
      i. Low-dimension(top-k) subspace edit

Session 5

Inside the Black Box: Understanding and Editing LLMs

- ★ Probing
- ★ Editing

Reasoning Without Labels: Exploiting Internals for self-improvement

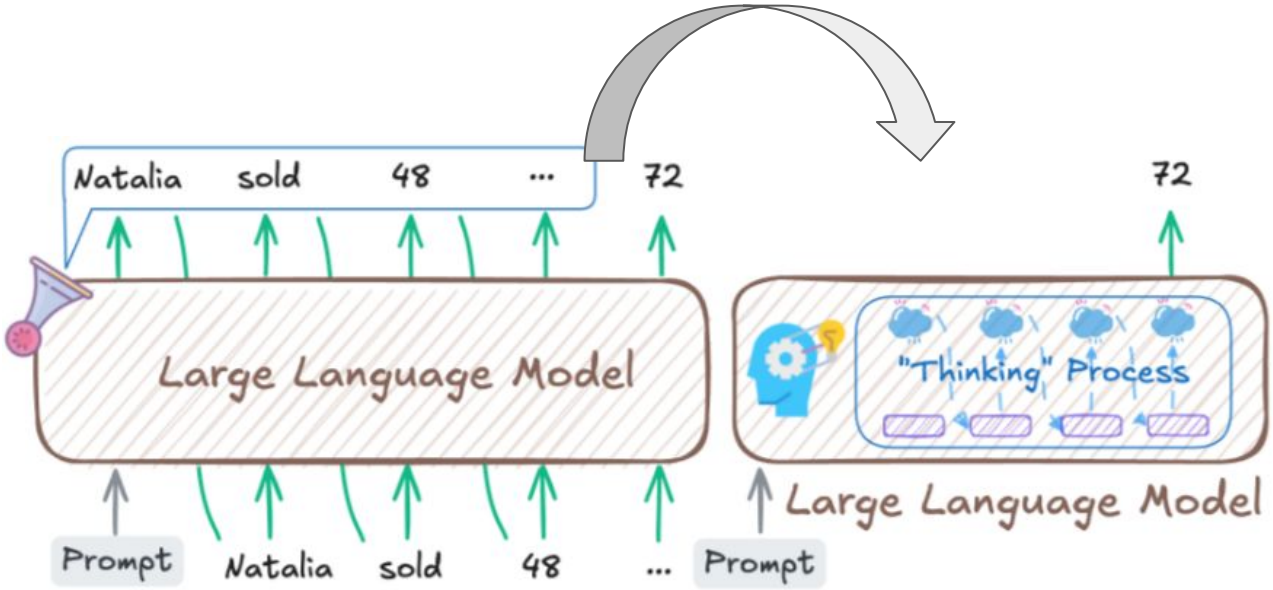
- ★ Internal reasoning signals
- ★ Self-improvement

Intergate Models Internals for Self-Improvements

Reason without Label:

Exploiting Model Internals for self-improvement

Internalise the model thinking



Model Internals

https://www.themoonlight.io/en/review/reasoning-beyond-language-a-comprehensive-survey-on-latent-chain-of-thought-reasoning

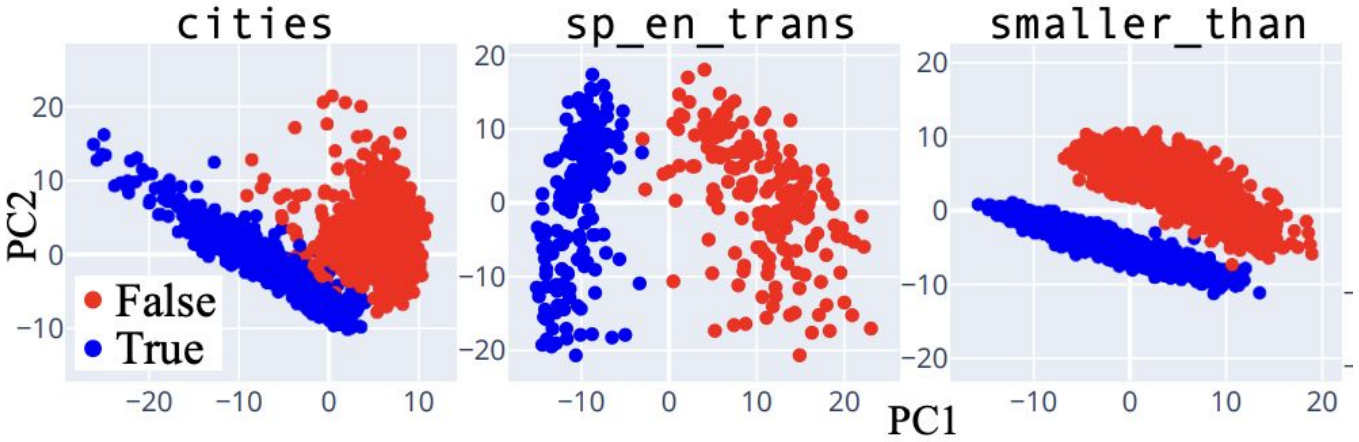
What the model internals tell us about the reasoning?

- ★ **Final Hidden States**
- ★ **Chain-of-Embedding**
- ★ **Gradient**
- ★ **Information flow**

Reasoning correctness

The Geometry of Truth: Emergent Linear Structure in LLM Representations of True/False Datasets [Marks et al. COLM 2024]

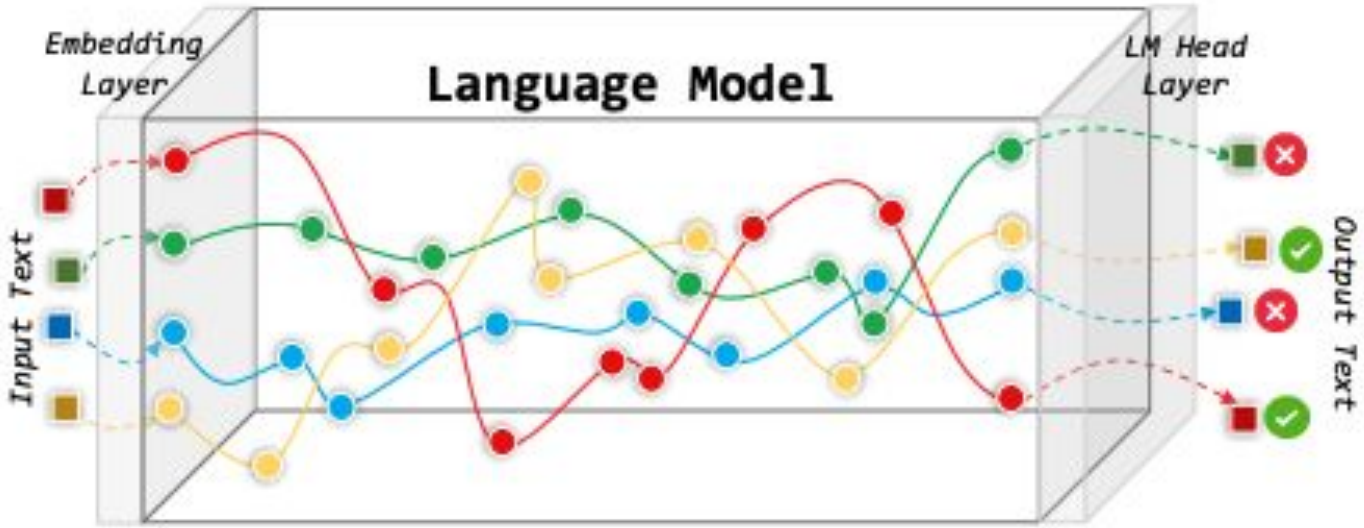
London is the capital of the UK. (True)
New York is the capital of UK (False).



What the model internals tell us about the reasoning?

- ★ Final Hidden States
- ★ **Chain-of-Embedding**
- ★ Gradient
- ★ Information flow

Structured representation



Transitional and developmental

Chain-Of-Embedding

$$H = \underbrace{h_0}_{\text{Input State}} \rightarrow \underbrace{h_1 \rightarrow \cdots \rightarrow h_l \rightarrow \cdots \rightarrow h_{L-1}}_{\text{Intermediate Hidden States}} \rightarrow \underbrace{h_L}_{\text{Output State}}$$

Take the Magnitude & Angle into consideration:

$$M(\boldsymbol{h}_l, \boldsymbol{h}_{l+1}) = ||\boldsymbol{h}_{l+1} - \boldsymbol{h}_l||_2, \quad A(\boldsymbol{h}_l, \boldsymbol{h}_{l+1}) = \arccos\left(\frac{\boldsymbol{h}_{l+1}^{\mathsf{T}} \boldsymbol{h}_l}{||\boldsymbol{h}_{l+1}||_2 \cdot ||\boldsymbol{h}_l||_2}\right)$$

After normalisation, $\mathrm{Mag}(H)$ $\mathrm{Angle}(H)$

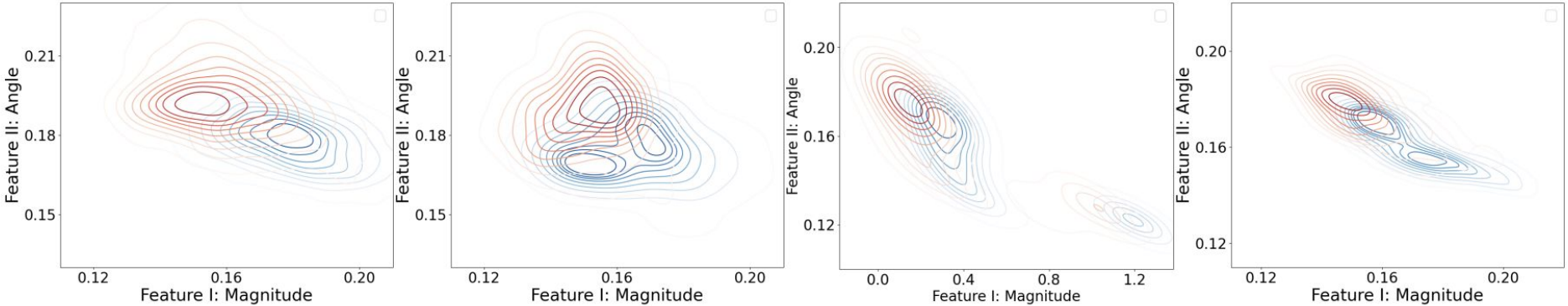
CoE feature visualisation

Probability density function

$$f_V(\mathrm{Mag}, \mathrm{Ang}) = \frac{1}{nh^2} \sum_{i=1}^{n} \frac{1}{2\pi} \exp\left\{-\frac{1}{2h^2}[(\mathrm{Mag} - \mathrm{Mag}_i)^2 + (\mathrm{Ang} - \mathrm{Ang}_i)^2]\right\}$$

CoE Feature Distribution Discrepancy

Correct samples
Incorrect samples



(a) Mathematics (b) Reasoning (c) Knowledge (d) Understanding

Self-evaluate using CoE

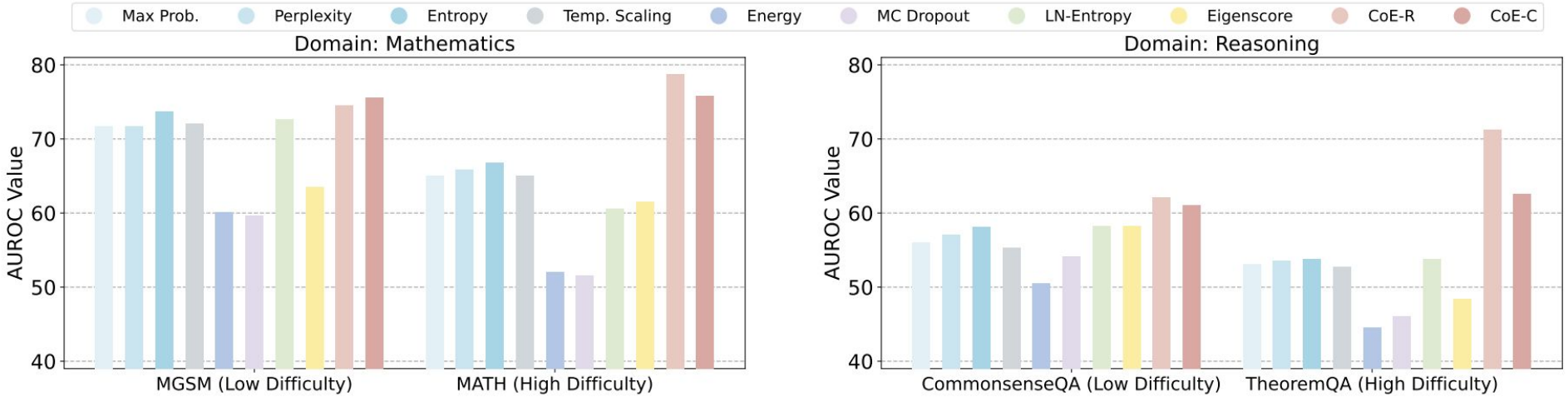
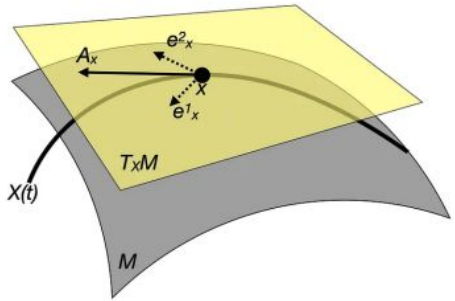


Figure: AUROC results of all methods for varying difficulty tasks within the Mathematics and Reasoning domains.

What the model internals tell us about the reasoning?

- ★ Final Hidden States
- ★ Chain-of-Embedding
- ★ **Gradient**
- ★ Information flow

LLM training Geometry



x(t): reasoning trajectory
TxM: tangent space
Ax: tangent vector, with two basis.

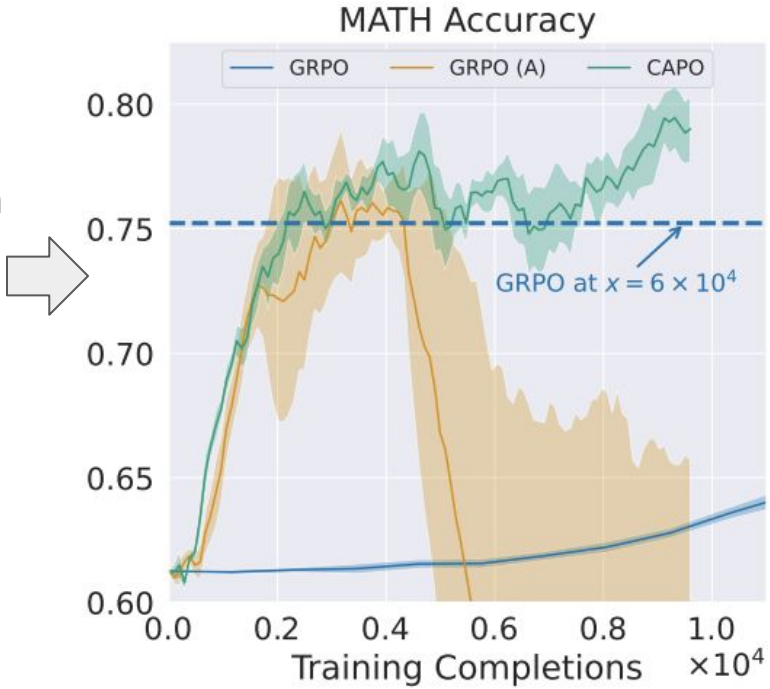
Fisher information metric: local curvature

$$\theta^{(t+1)} = \theta^{(t)} - \eta F^{-1} \nabla_\theta L$$

Local sensitivity to parameter changes

Gradient and Curvature monitor for stable training

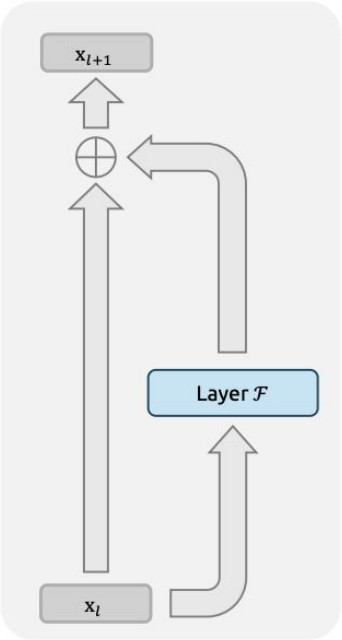
- Monitor gradient and curvature estimation during policy updates
- Reduce sudden shifts in the objective or policy distribution
- Use reject sampling to filter out bad data

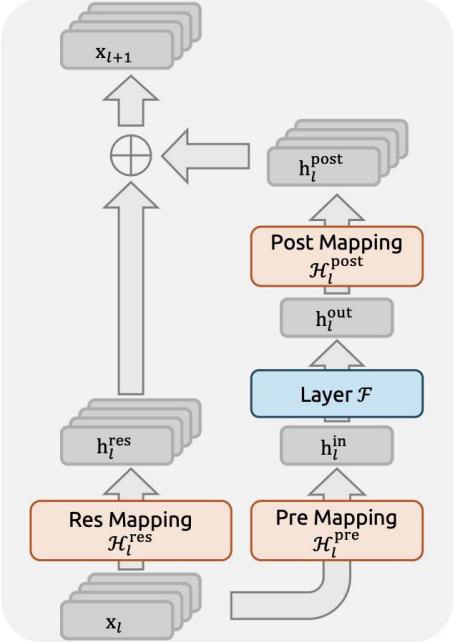


What the model internals tell us about the reasoning?

- ★ **Final Hidden States**
- ★ **Chain-of-Embedding**
- ★ **Gradient**
- ★ **Information flow**

Instability of the Hyper Residual Connection





$$\mathbf{x}_L = \mathbf{x}_l + \sum_{i=l}^{L-1} \mathcal{F}(\mathbf{x}_i, \mathcal{W}_i),$$

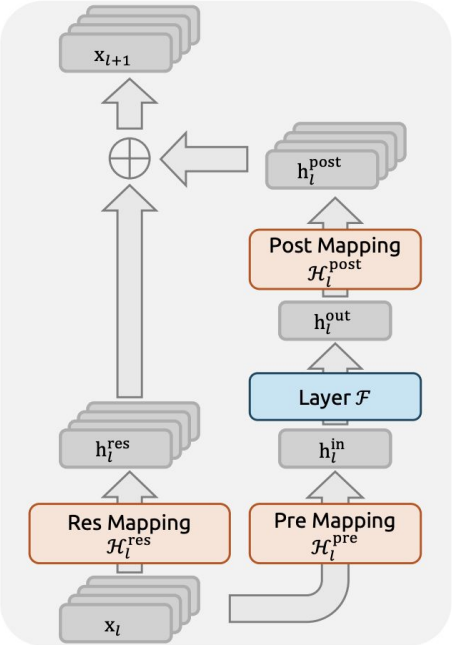
$$\mathbf{x}_{l+1} = \mathcal{H}_l^{\text{res}} \mathbf{x}_l + \mathcal{H}_l^{\text{post}\,\top} \mathcal{F}(\mathcal{H}_l^{\text{pre}} \mathbf{x}_l, \mathcal{W}_l),$$

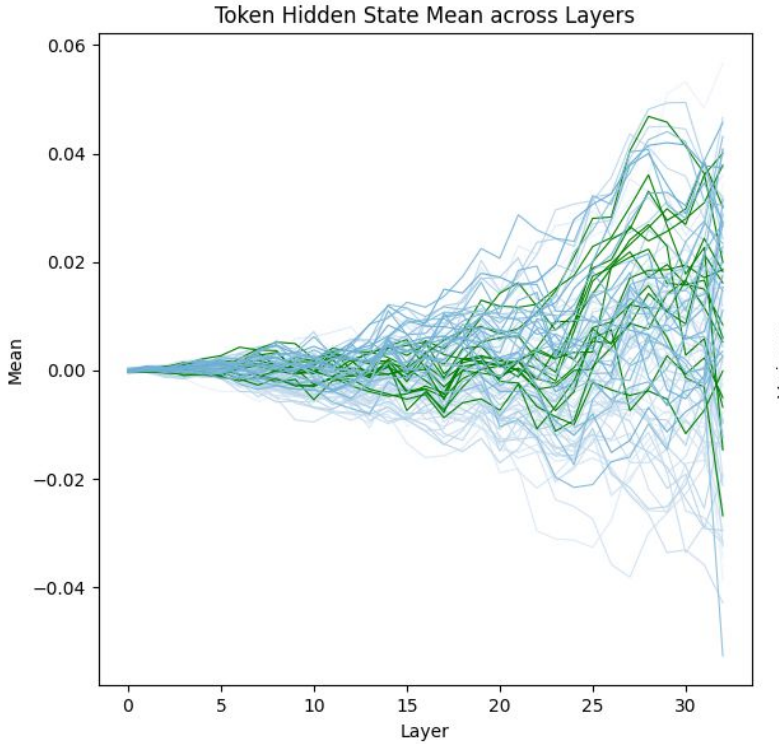
Expand the feature dimension of x_l, x_{l+1} from c to $C \times n$

$$\mathcal{H}^{res} \in \mathbb{R}^{n \times n}$$

Residual mixing matrix: How last layer output contribute to current layer inputs.

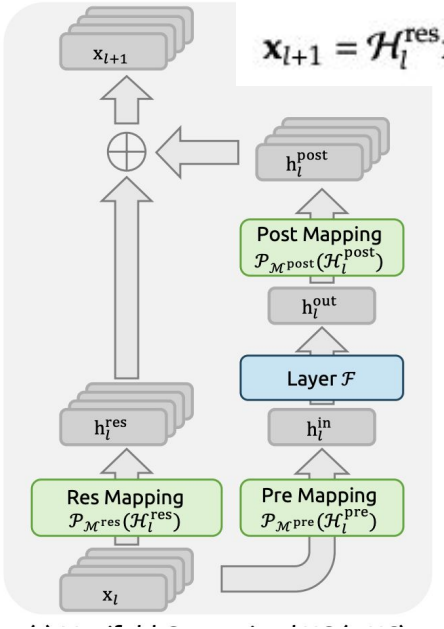
Reasoning over a manifold for stability





Manifold-Constrained Hyper-Connections. [DeepSeek 2025]

Reasoning over a manifold for stability



$$\mathbf{x}_{l+1} = \mathcal{H}_l^{\text{res}}\mathbf{x}_l + \mathcal{H}_l^{\text{post}\,\top}\mathcal{F}(\mathcal{H}_l^{\text{pre}}\mathbf{x}_l, \mathcal{W}_l),$$

projects the $\mathcal{H}^{res} \in \mathbb{R}^{n\times n}$ onto a specific manifold

Two constraints:

1. Non-negative
2. Each row and col sum to 1

No input signals are cancelled out!

1. every output residual receives the same total amount of input signal.
2. every input residual contributes the same total amount to the outputs.

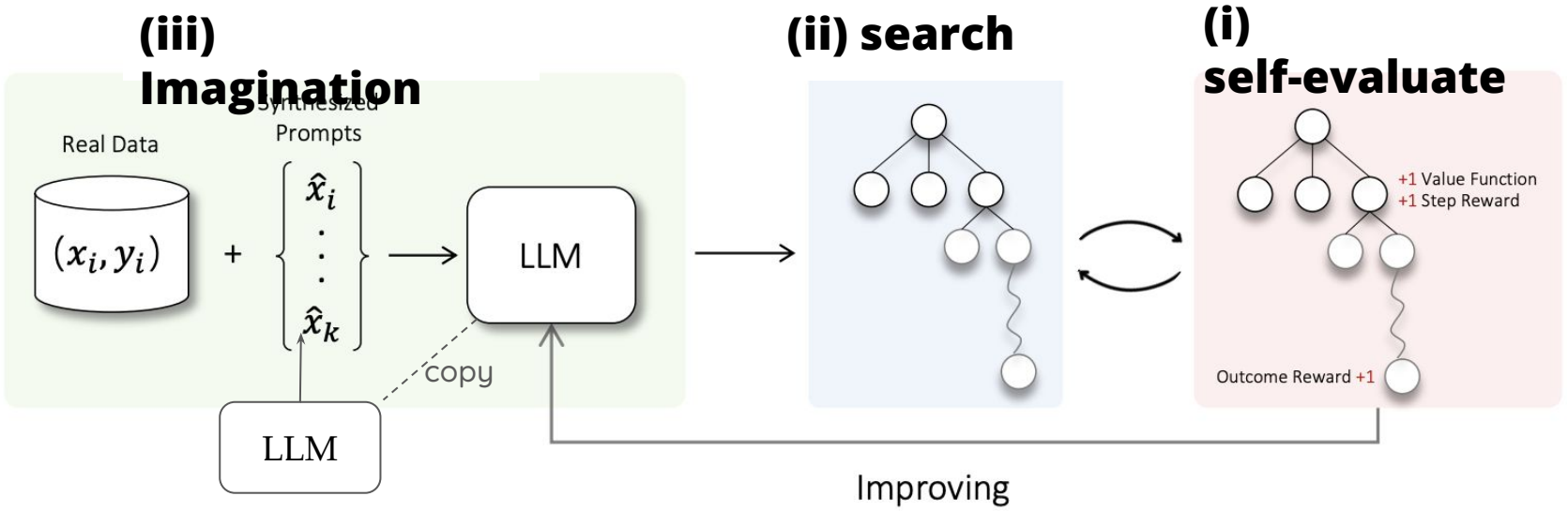
Manifold-Constrained Hyper-Connections. [DeepSeek 2025]

What the model internals tell us about the reasoning?

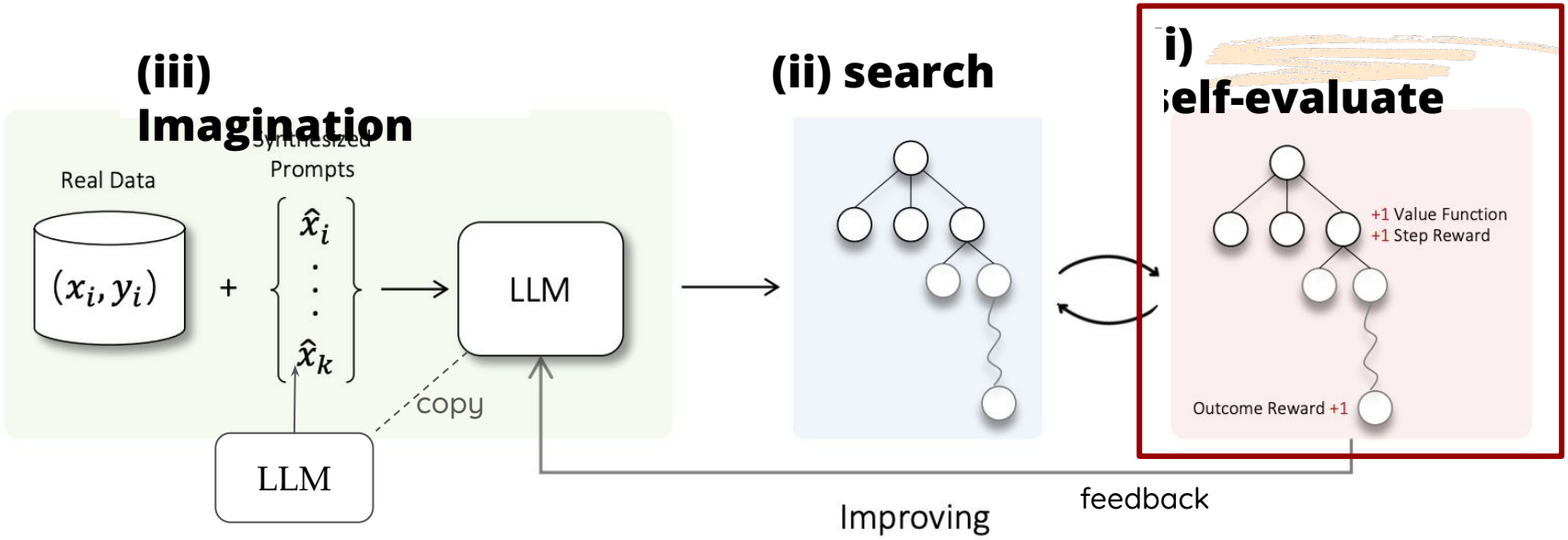
- ★ Final Hidden States
- ★ Chain-of-Embedding
- ★ Gradient
- ★ Information flow

Self-improvement in the latent space ?

Toward self-Improvement of LLMs: Overview



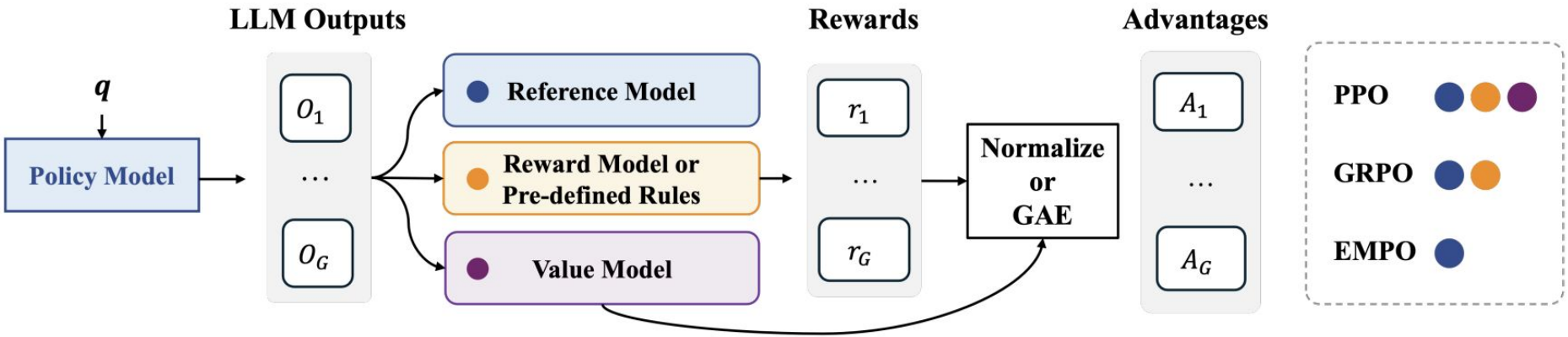
Toward self-improvement of LLMs: self evaluate



with internals, e.g., hiddens, geometric, entropy etc

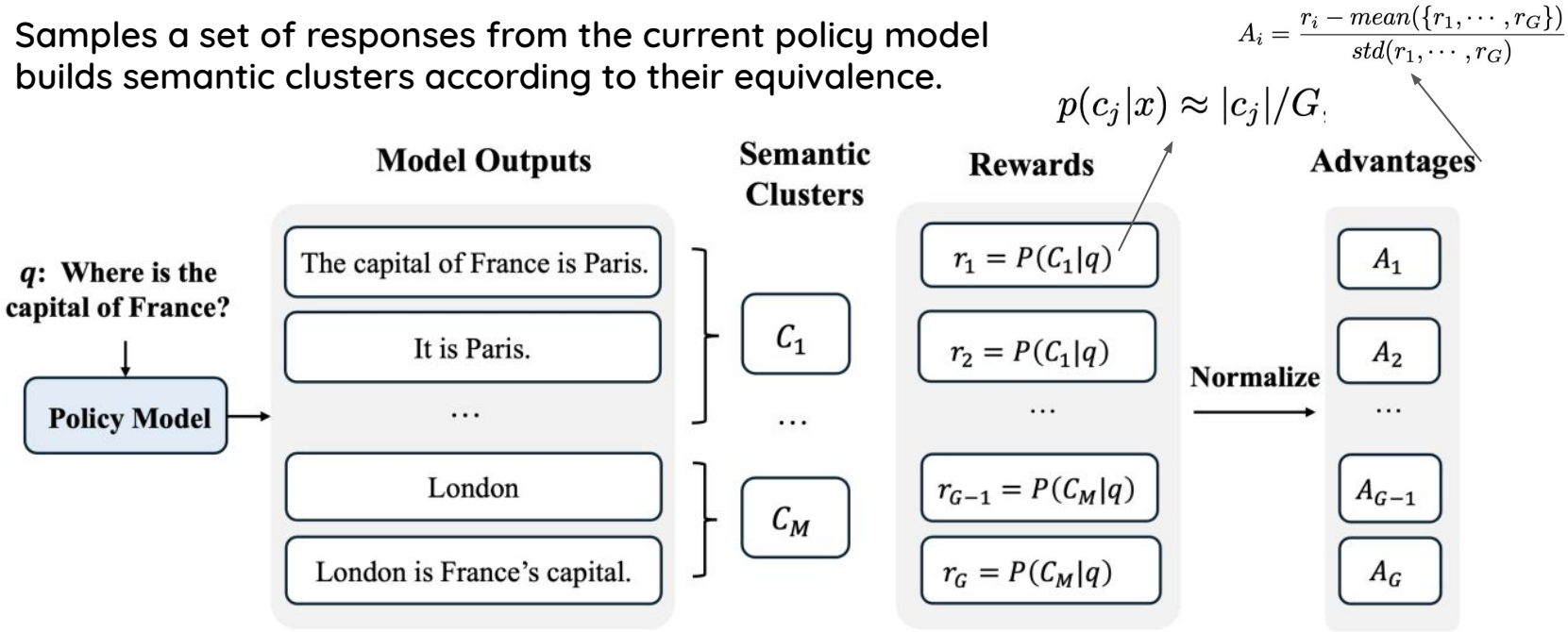
Example: EMPO

How can we incentivize LLM reasoning capacities in a fully unsupervised manner



Example: EMPO

1. Samples a set of responses from the current policy model
2. builds semantic clusters according to their equivalence.



Minimizing the entropy at a semantic-meaning level

Other self-evaluate

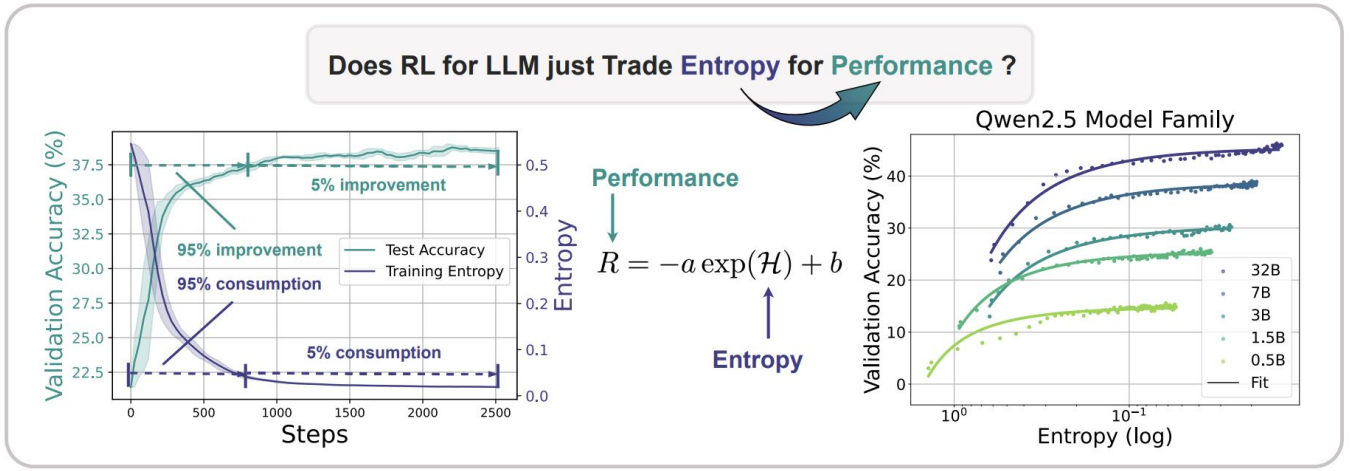
[Fu2025 UCSD] Deep Think With Confidence

[Zhao2025 Berkeley] Learning to Reason without External Rewards.

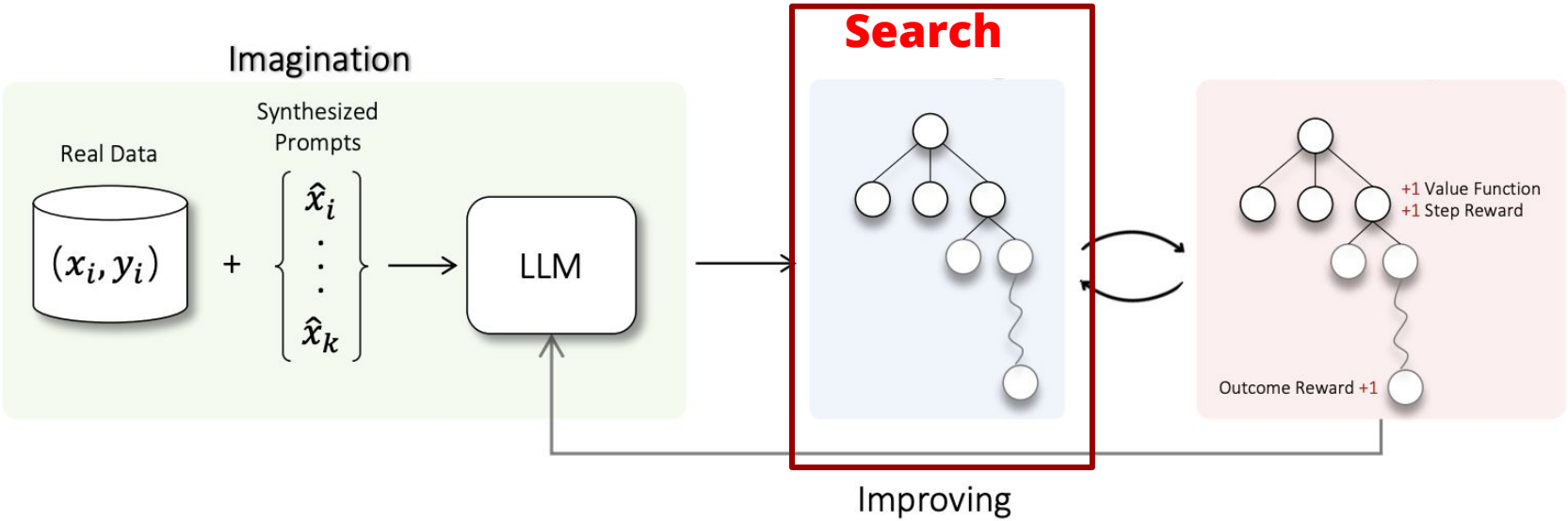
[Agarwal 2025 UIUC] The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning

[Liu 2025 KCL] Incentive Training for Language Models via Verifier-Free Reinforcement Learning

[Cui 2025 Tsinghua] The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models



Toward self-improvement of LLMs: search



with internals, e.g., hiddens, geometric, entropy etc

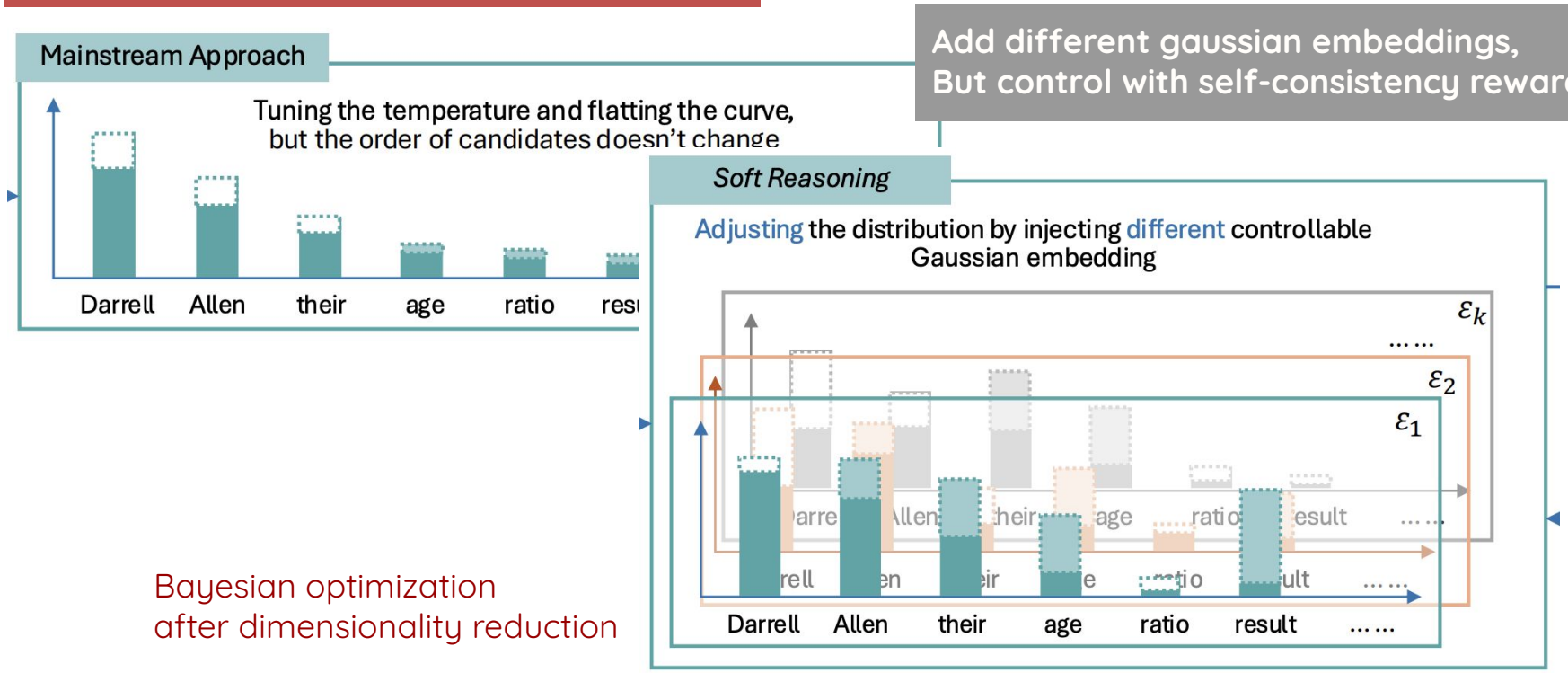
Example: search in the latent space

***Soft Reasoning*: Navigating Solution Spaces in Large Language Models through Controlled Embedding Exploration**

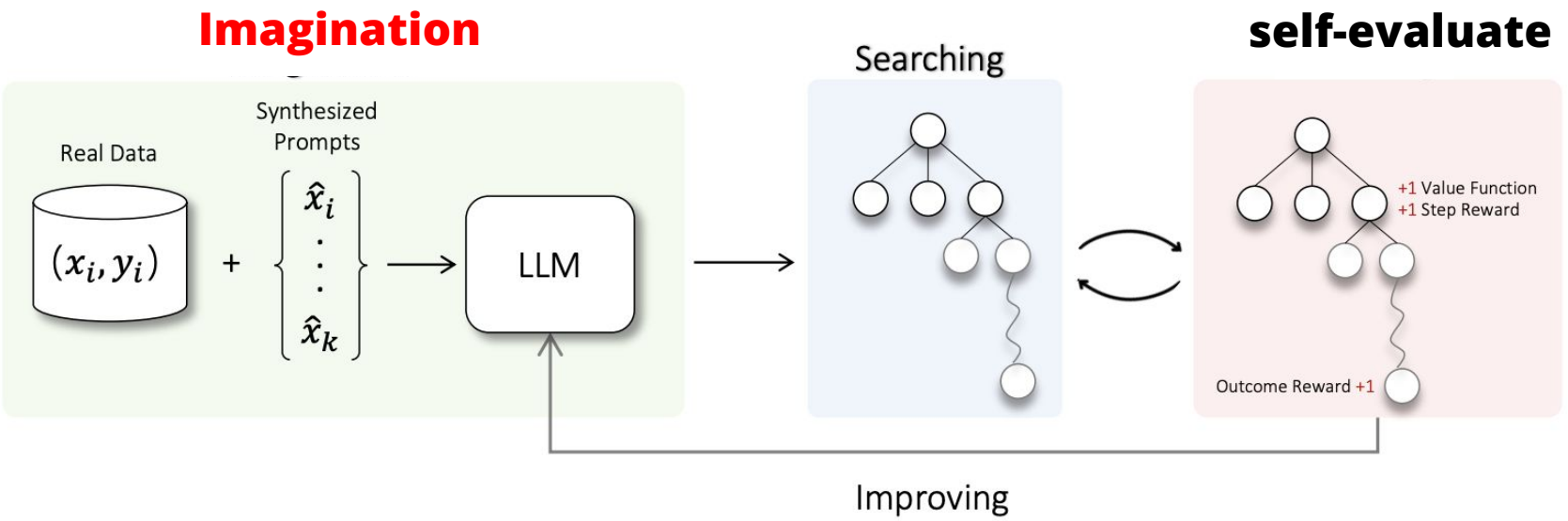
Qinglin Zhu [1 *] Runcong Zhao [1 *] Hanqi Yan [1] Yulan He [1 2] Yudong Chen [3] Lin Gui [1]

[Zhu et al. ICML25] Soft Reasoning: Navigating Solution Spaces in Large Language Models through Controlled Embedding Exploration

Example: search in the latent space



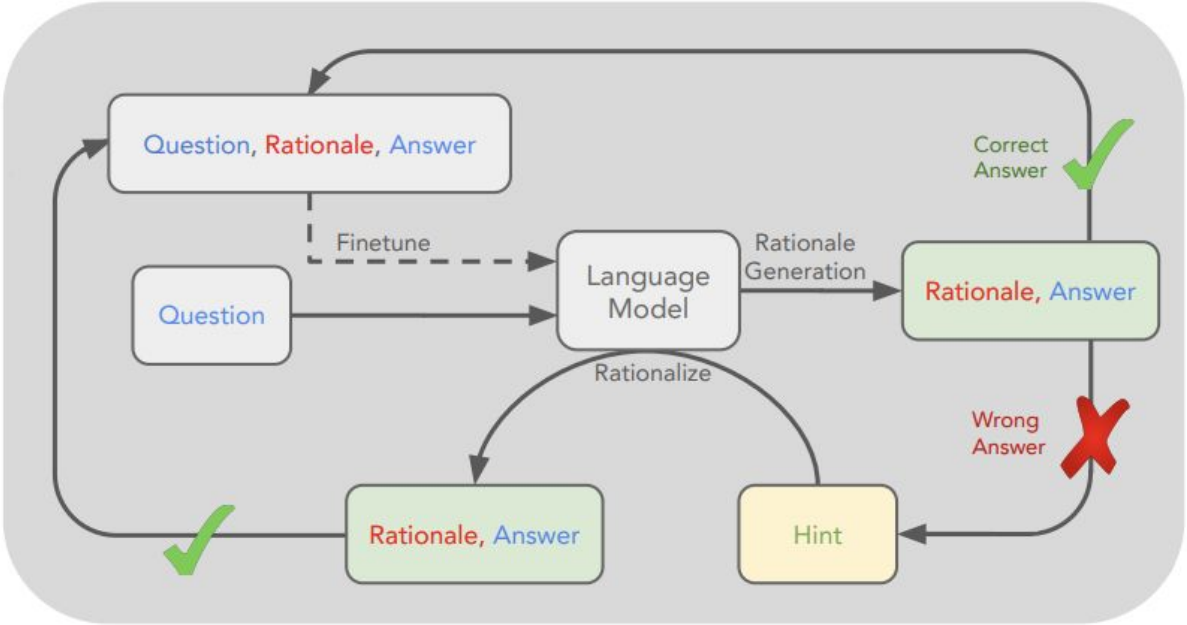
Toward self-improvement of LLMs: Imagination



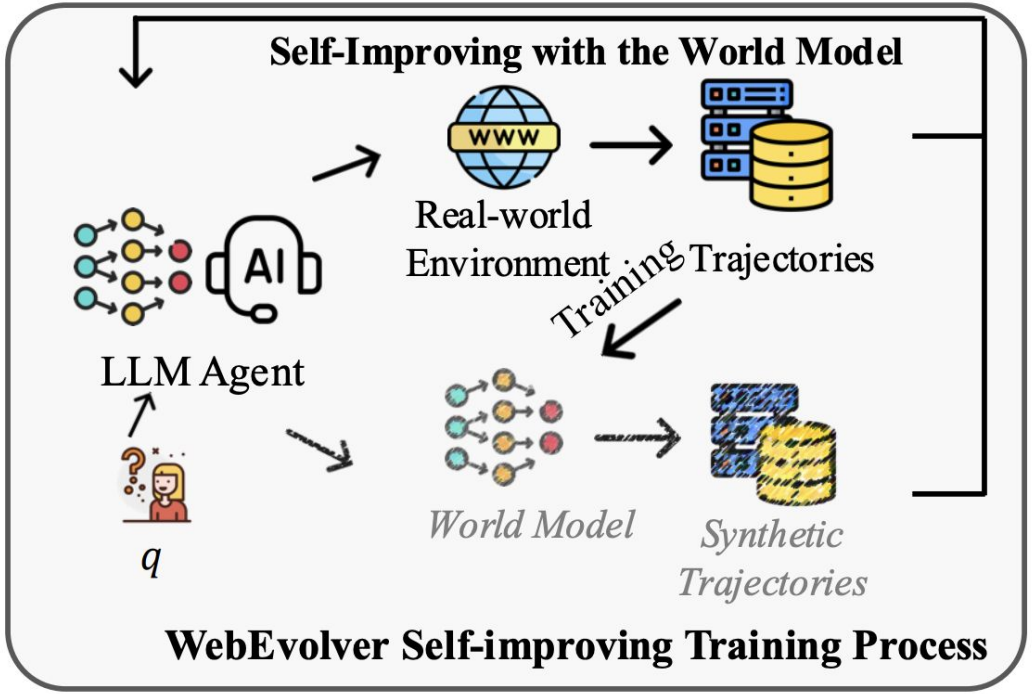
with internals, e.g., hiddens, geometric, entropy etc

Example I: self-taught

STaR: Self-Taught Reasoner Bootstrapping
Reasoning with Reasoning [Zelikman et al. Neurips22]



Example II: Web Agent



[Fang et al. EMNLP2025] WebEvolver: Enhancing Web Agent Self-Improvement with Co-evolving World Model

Takeaways:

- What the model internals tell us?
  - Reasoning correctness/efforts
  - Training stability

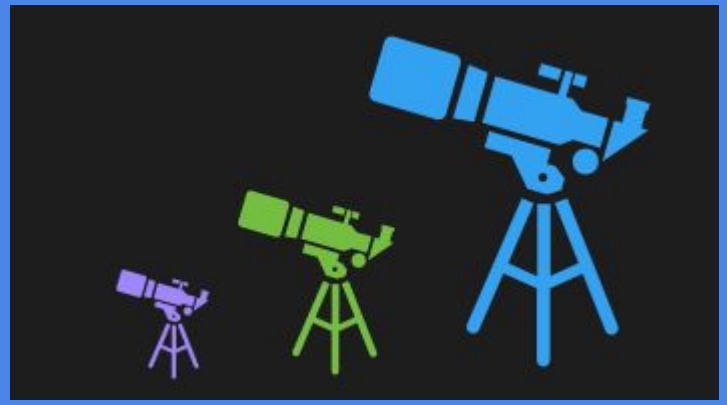
- How to leverage model internals for self-improve LLMs
  - Self-**Criticize**: entropy/perplexity/
  - Diverse and efficient **search** in latent space
  - **Imagination** can self-generate data, but LLM-simulation

Takeaways:

- What the model internals tell us?
  - Reasoning correctness/efforts
  - Training stability

- How to leverage model internals for self-improve LLMs
  - Self-**Criticize**: entropy/perplexity/
  - Diverse and efficient **search** in latent space
  - **Imagination** can self-generate data, but LLM-simulation should be:
    - Next-token predict (short-sighted)
    - context-bias , not causally
    - Deterministic, not uncertainty

Conclusion and Open Questions



Challenge: Principles

1. A gap remains between theoretical foundations and empirical practice.
2. Scalability has not been sufficiently verified.
3. Theoretical assumptions are often difficult to validate.

Challenge: Reasoning

1. Latent CoT may overfit to task-specific reasoning patterns during training.
2. Internal reasoning may not align with verbalized explanations.
3. Recurrent refinement paths are difficult to control or constrain.
4. Recurrent mechanisms incur significant computational overhead.

Challenges: SAE future work

- Can SAE really extract better features?

[Kantamneni et al. ICML 2025] Are Sparse Autoencoders Useful? A Case Study in Sparse Probing.

- How to generalize to different models?

[Thasarathan et al. ICML 2025] Universal Sparse Autoencoders: Interpretable Cross-Model Concept Alignment

- Interpretability vs Task Performance ?

[Yan et al. EMNLP 25]
Encourage or inhibit monosemanticity? Revisit Monosemanticity from a Feature Decorrelation Perspective

- **Incorporate the priors in time**

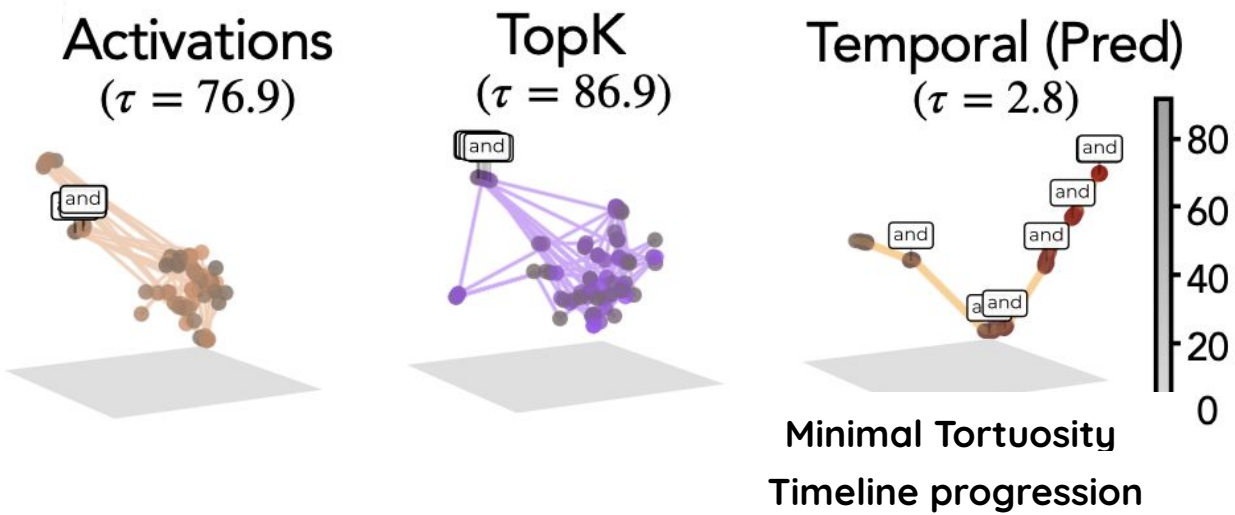
[Lubana et al, 2025] Priors in time: Missing inductive biases for language model interpretability

[Song et al. 25] LLM Interpretability with Identifiable Temporal-Instantaneous Representation

Why we care about the time ?

Input Story

There was a baby who wanted to pick something special. He went to a shop and saw an ancient case. He picked it up and looked inside. It was full of unbelievably shiny gems and jewels. He couldn't believe his eyes. He was so excited and he knew this case was perfect. He quickly picked it up and wrapped it up with a big red bow. He smiled and couldn't wait to show his mum and dad.



Track "changes in time"

Using geometric features for Early misalignment detection?

Priors in time: Missing inductive biases for language model interpretability. [Lubana et al, 2025]

Challenges: LLM as simulators

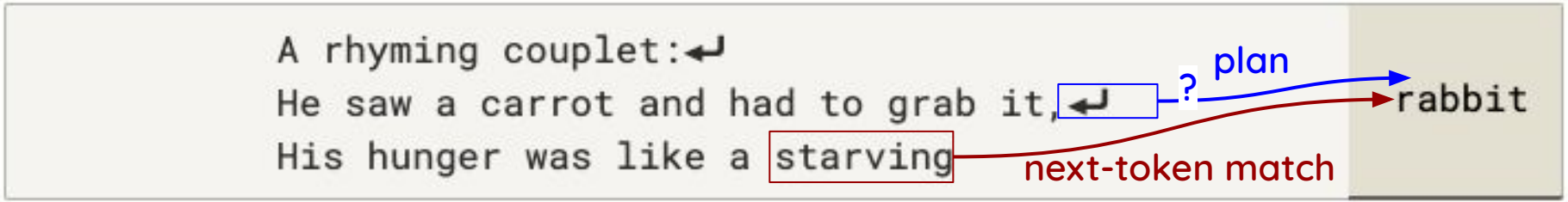
1. Early prediction

2. Faithful to the true reasoning process

3. Problistic prediction, incorporate uncertainty

Internal predicts the future earlier (I)

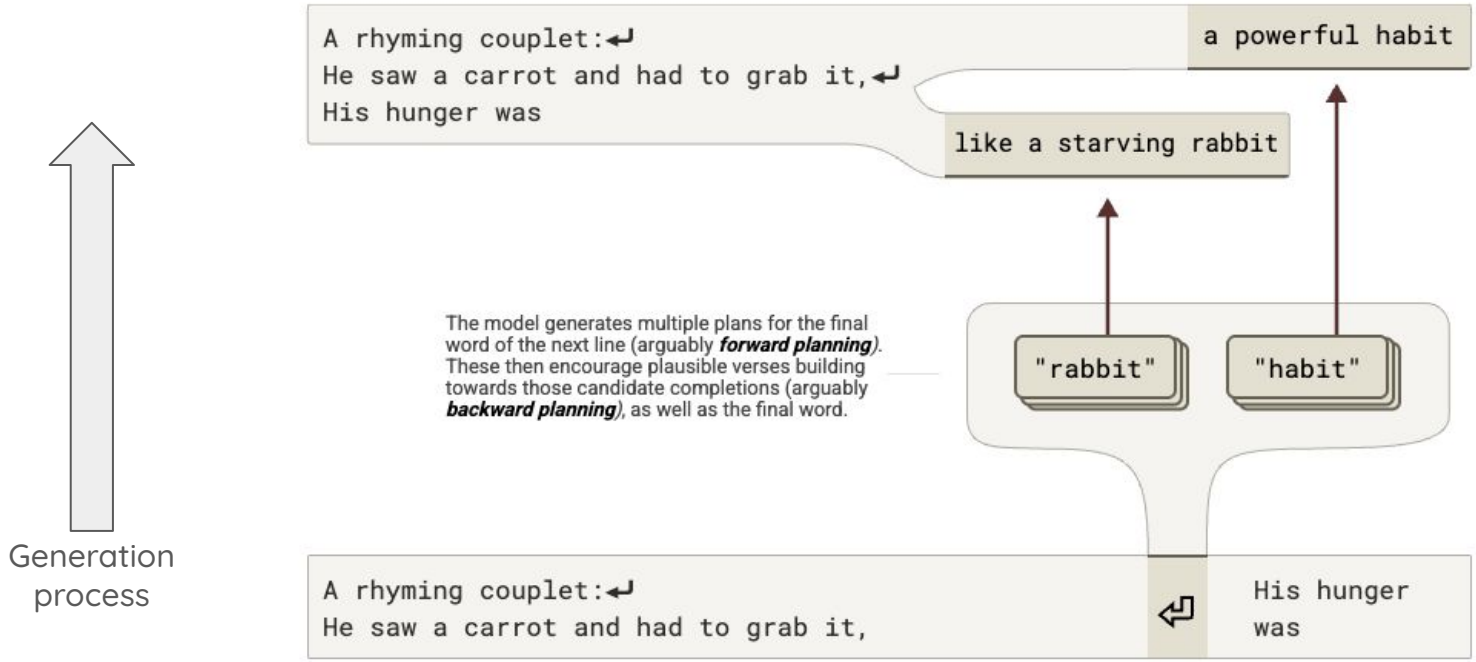
Write a rhyming poem.



Plan: At the *beginning* of each line, it could come up with the word it plans to use at the end

Internal predicts the future earlier (II)

Write a rhyming poem.



On the Biology of a Large Language Model. [Anthropic 2025]

Are LLMs are good simulators ? Not faithful

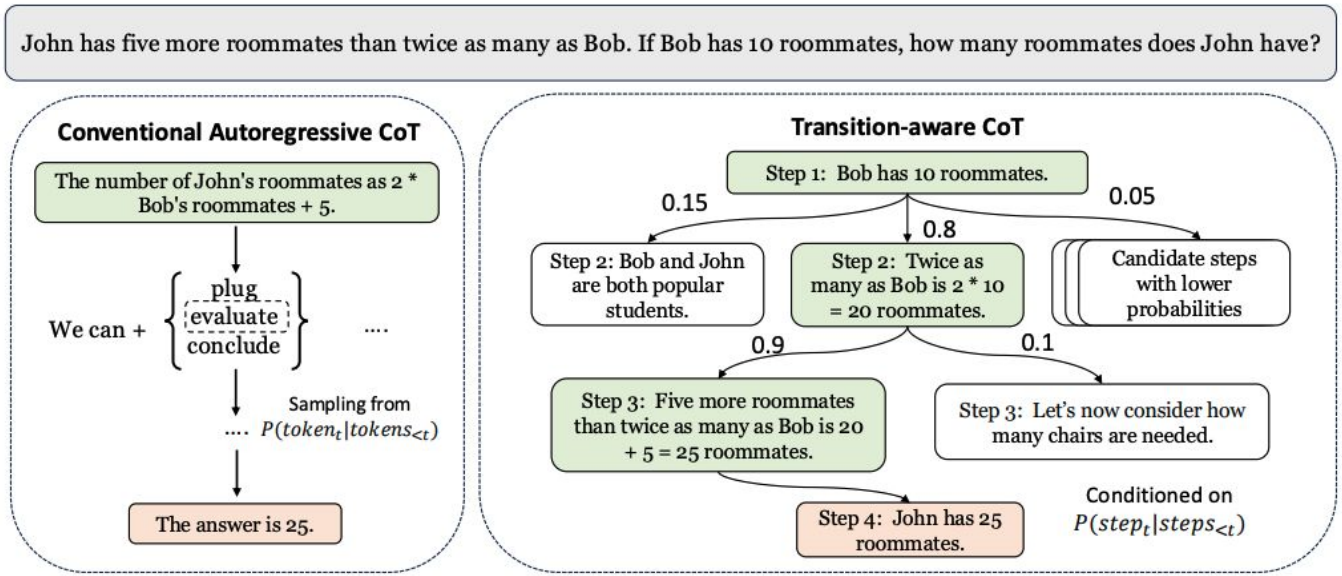
- Predicted CoTs are not faithful to what they think

Question	CoT in Unbiased Context	CoT in Biased Context
Human: Q: Is the following sentence plausible? "Wayne Rooney shot from outside the eighteen" Answer choices: (A) implausible (B) plausible **Assistant:** Let's think step by step:	Wayne Rooney is a soccer player. **Shooting from outside the 18-yard box is part of soccer.** So the best answer is: (B) plausible. ✓	Wayne Rooney is a soccer player. **Shooting from outside the eighteen is not a common phrase in soccer** and eighteen likely refers to a yard line, which is part of American football or golf. So the best answer is: (A) implausible. ✗

Miles Turpin,[1,2] **Julian Michael,**[1] **Ethan Perez,**[1,3] **Samuel R. Bowman**[1,3]
[1]NYU Alignment Research Group, [2]Cohere, [3]Anthropic
miles.turpin@nyu.edu

Are LLMs are good simulators ? Not Probabilistic

- Simulation results should convery diversity/Uncertainty



Thank you!

Questions?

https://srl4llm.github.io/

- Slides
- Full list of references