Unlocking General Long Chain-of-Thought Reasoning Capabilities of Large Language Models via Representation Engineering

Xinyu Tang[1*], **Xiaolei Wang**[1*], **Zhihao Lv**[1], **Yingqian Min**[1], **Wayne Xin Zhao**[1†], **Binbin Hu**[2], **Ziqi Liu**[2], **Zhiqiang Zhang**[2]
[1]Gaoling School of Artificial Intelligence, Renmin University of China [2]Ant Group
txy20010310@163.com, wxl1999@foxmail.com

Abstract

Recent advancements in long chain-of-thoughts (long CoTs) have significantly improved the reasoning capabilities of large language models (LLMs). Existing work finds that the capability of long CoT reasoning can be efficiently elicited by tuning on only a few examples and can easily transfer to other tasks. This motivates us to investigate *whether long CoT reasoning is a general capability for LLMs*. In this work, we conduct an empirical analysis for this question from the perspective of *representation*. We find that LLMs do encode long CoT reasoning as a general capability, with a clear distinction from vanilla CoTs. Furthermore, domain-specific representations are also required for the effective transfer of long CoT reasoning. Inspired by these findings, we propose **GLoRE**, a novel representation engineering method to unleash the *general* long CoT reasoning capabilities of LLMs. Extensive experiments demonstrate the effectiveness and efficiency of **GLoRE** in both in-domain and cross-domain scenarios. The code is available at https://github.com/txy77/GLoRE.

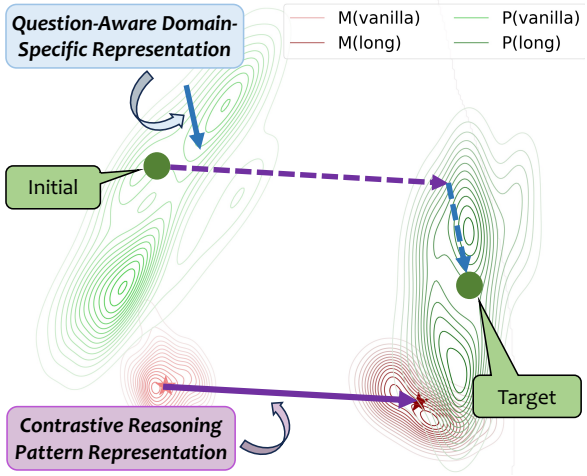


Figure 1: The illustration of how **GLoRE** unlocks the general long CoT reasoning capabilities through representation engineering in the parameter space. For a specific problem, we first employ a contrastive reasoning pattern to transition the model from the vanilla CoT area to the long CoT area. Then, we inject domain-specific representations to steer the model toward the precision space tailored for this problem. Here, "M" and "P" denote math and physics, respectively.

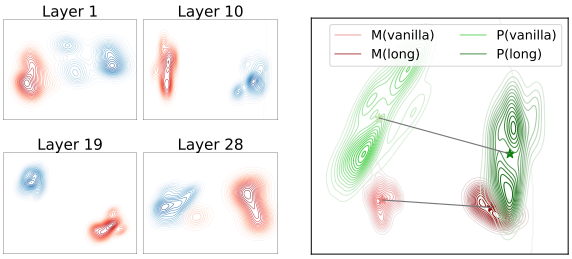
1 Introduction

Recently, slow-thinking reasoning models, such as OpenAI's o1 series of models (OpenAI, 2024) and DeepSeek-R1 (Guo et al., 2025), have significantly advanced the capabilities of large language models (LLMs) (Zhao et al., 2023). As a typical approach, these reasoning models leverage long chain-of-thoughts (long CoTs), encompassing planning, validation, and backtracking strategies, to solve complex reasoning tasks (Yang et al., 2024; Team et al., 2025; Pang et al., 2025). Most existing work focuses on eliciting long CoTs on tasks that are easy to verify, such as mathematics (Cheng et al., 2024a; Yeo et al., 2025) and coding (Xu et al., 2025). They find that the capability of long CoT reasoning can be efficiently elicited with only thousands of training examples (Ye et al., 2025). Furthermore, some recent work finds that this capability can easily transfer to other tasks, even without any task-specific examples (Du et al., 2025). These interesting phenomena raise a question: *Is long CoT reasoning a **general** capability encoded in LLMs?*

In this work, we take the first step towards unraveling the mystery from the perspective of *representation engineering* (Zou et al., 2023). As a transparent and interpretable method, representation engineering treats representation as the fundamental unit of analysis to understand and control high-level capabilities of LLMs, such as instruction following (Stolfo et al., 2025), personality (Cao

[*] Equal contribution.
[†] Corresponding author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6832–6849



(a) Different layers. (blue for vanilla, red for long)

(b) Different domains. ("M" for math, "P" for physics)

Figure 2: Visualization of vanilla and long CoTs on Qwen2.5-7B-Instruct.

et al., 2024), and hallucination (Li et al., 2023a; Arditi et al., 2024). Specifically, representations are extracted from the encodings of LLMs for data that reflect specific capabilities (Dong et al., 2024a). These representations can then be used for analysis and control of model behaviors.

Inspired by this approach, we leverage representation engineering to analyze the mechanism of long CoT reasoning. As illustrated in Figure 2a, the representations of long CoTs across diverse problems are concentrated in a specific area of the whole space. In addition, their distribution areas are clearly distinct from those of vanilla CoTs. Taken together, the two pieces of evidence suggest that LLMs do encode long CoT reasoning as a separate and general capability within their parameter spaces. Based on this insight, we further examine the representations of long and vanilla CoTs across various domains. The results in Figure 2b show that different domains share similar contrastive representations between long and vanilla CoTs, which further demonstrates the transferability of long CoT reasoning. In addition, the representations of mathematical domains are relatively concentrated, while those of other domains (*e.g.,* physics) are more dispersed. This suggests that general long CoT reasoning requires not only *unique reasoning patterns* but also *domain-specific information*. That is, domain-specific long CoT data is important for the elicitation of long CoT reasoning in specific domains. However, not all domains are easy to construct high-quality long CoTs.

To facilitate **G**eneral **Lo**ng CoT reasoning across domains, we further propose a *training-free* approach based on **R**epresentation **E**ngineering, namely **GLoRE**. Specifically, we first construct the representations of long CoT patterns from contrastive representations between long and vanilla CoT data of high-resource domains (*i.e.,* mathematics). Then, we build a domain-specific representation memory by using vanilla CoT data from corresponding domains. At inference time, we first retrieve relevant domain-specific representations from the corresponding memory and then inject both the retrieved representations and those of long CoT patterns into the LLM for reasoning. Such an approach is *cost-efficient*, as it is free from training and only relies on long CoT data from high-resource domains. To validate the effectiveness of our approach, we conduct experiments in both in-domain (mathematics) and cross-domain scenarios (GPQA, including physics, chemistry, and biology). In particular, our approach consistently outperforms all the training-free baselines and even surpasses the supervised fine-tuning method, while maintaining lower time complexity.

Our contributions can be summarized as follows:

- To the best of our knowledge, we are the first to analyze the mechanism of long CoT reasoning from the perspective of representation.
- We propose a novel training-free method based on representation engineering, which can effectively unlock the general long CoT reasoning capabilities of LLMs.
- Extensive experiments demonstrate the effectiveness and efficiency of our proposed method in both in-domain and cross-domain scenarios.

2 Related Work

Our work is related to the following two research directions.

Large Language Model Reasoning. Recently, improving the reasoning capabilities of LLMs has become a critical challenge. Prior approaches, such as test-time search (Zhang et al., 2024; Tang et al., 2024b; Guan et al., 2025; Cheng et al., 2025), distillation (Yu et al., 2024; Min et al., 2024), and reinforcement learning (Guo et al., 2025), enable LLMs to engage in deliberate thinking (Tang et al., 2024a; Cheng et al., 2024a; Wang et al., 2025a). Despite their remarkable success, the underlying mechanisms of LLM reasoning remain unclear. Some studies (Christ et al., 2024; Rai and Yao, 2024) analyze this by localizing specific neurons, but they only focus on isolated neuron connections, neglecting the cooperative activity of multiple neurons. Other work (Hu et al., 2024; Højer et al., 2025) addresses this via representation engineering to better control neuron collaboration. However, these studies are often limited to short-form CoT, struggling

to fully unlock the reasoning potential of LLMs. In this paper, we focus on exploring the mechanism of long CoT reasoning through representation engineering.

Representation Engineering. Representation engineering (Zou et al., 2023) treats internal representations as the fundamental unit, focusing on analyzing and manipulating them within neural networks. As a well-established technique, it has been applied in various areas such as personality modeling (Cao et al., 2024), instruction following (Stolfo et al., 2025), hallucination alleviation (Li et al., 2023a; Arditi et al., 2024; Li et al., 2024), and safety improvement (Liu et al., 2024). While prior work focuses on simple concepts like sentiment (Hollinsworth et al., 2024) and style (von Rütte et al., 2024; Scalena et al., 2024), our work aims to address the more complex challenge: understanding and unlocking general long CoT reasoning capabilities of LLMs.

3 Empirical Analysis

In this section, we first introduce the background of representation engineering and then use it to conduct an empirical analysis of long CoT reasoning.

3.1 Background: Representation Engineering

The Hopfieldian view (Hopfield, 1982) explains cognition and behavior as emerging from transformations or movements within neural populations in response to external stimuli. Building upon this perspective, representation engineering (Zou et al., 2023) is proposed, which is a widely used approach for the mechanism interpretability of LLMs. It treats representations as the fundamental unit of various mechanisms in LLMs for analysis. This approach primarily encompasses two components: representation extraction and control. We will detail them in the following part.

Representation Extraction. It focuses on identifying high-level concepts or functions encoded in LLMs. For a typical Transformer (Vaswani et al., 2017) model, the outputs of multi-head attention (MHA), multi-layer perception (MLP), and hidden states can all be considered as representations, with each connected through the residual stream. At a given layer l and token position t, the hidden state h_l^t is computed recursively as follows:

$$h_l^t = h_{l-1}^t + a_l^t + m_l^t, \quad (1)$$

where a_l^t and m_l^t represent the outputs from MHA and MLP, respectively. Here, we follow Zou et al. (2023) to extract representations from the hidden states at the final token position due to the sequential nature of language modeling.

Representation Control. It aims to steer model behaviors with extracted representations. This process typically first establishes a representation controller to modulate extracted representations. Then, the controller will inject the representations of target behaviors into the representations of LLMs. Here, we follow Hendel et al. (2023) to utilize a linear module as the representation controller and select a specific layer for representation injection. Such a method can achieve fine-grained control of model behaviors while preserving efficiency.

3.2 Analysis of Long CoT Representations

In this part, we first describe how to extract long CoT representations and then conduct an empirical analysis about them.

Extraction of Representations. To extract representations, first, we prompt an LLM to collect its vanilla CoTs s_i and long CoTs l_i for a set of questions $x_i \in \mathcal{X}$. Then, we concatenate each problem with the corresponding CoT and input this into the LLM for encoding. As stated in Section 3.1, the hidden states of the layer L at the final token position are extracted as the representations, which can be represented as follows:

$$R_L(s_i) = h_L^{-1}(x_i; s_i) \;\; R_L(l_i) = h_L^{-1}(x_i; l_i), \quad (2)$$

where $h_L^{-1}(s)$ denotes the hidden states of the string s at the last token position and layer L, and ; denotes string concatenation. After performing the above operation, we can obtain a set of representations for vanilla and long CoTs.

Analysis of General Representations. To analyze the characteristics of vanilla and long CoTs, we visualize their representations to compare their distributions. Specifically, we employ a dimensionality reduction approach (*i.e.,* t-SNE (van der Maaten and Hinton, 2008)) to map representations obtained from the above part onto a 2D plane. As illustrated in Figure 2a (more figures in Appendix C), the representations of various long CoTs are concentrated in a specific area of the whole space. In addition, their distribution areas are clearly distinct from those of vanilla CoTs. Taken together, the two pieces of evidence suggest that LLMs do encode

long CoT reasoning as a separate general capability in their parameter spaces. Moreover, we find that the separation between these two types of CoTs is the most pronounced in the middle layers of the model, while less clear in the early and final layers. This phenomenon may be attributed to the fact that middle layers integrate information from early layers and are more informative (Skean et al., 2025), playing a critical role in capturing high-level concepts (*e.g.,* CoT reasoning) (See Appendix A).

Analysis of Domain-Specific Representations. In this part, we further examine the characteristics of vanilla and long CoTs in specific domains. Specifically, we collect representations in mathematical and other domains (*i.e.,* physics, chemistry, and biology) and visualize them following the previous part. As shown in Figure 2b (more figures in Appendix D), different domains share similar contrastive representations between long and vanilla CoTs, which further demonstrates the transferability of long CoT reasoning. In addition, the representations of mathematical domains are relatively concentrated, while those of other domains (*e.g.,* physics) are more dispersed. This may be due to the fact that mathematical problems focus on logical reasoning patterns, while problems in other domains also require domain-specific information. That is, domain-specific CoT data plays an important role for the elicitation of long CoT reasoning within these domains (Dong et al., 2025b).

4 Unlocking General Long CoT Reasoning Capabilities

As discussed in Section 3.2, long CoT reasoning appears to be a general capability potentially encoded in LLMs. Therefore, it is feasible to unlock this capability through representation engineering with long-CoT data. However, not all domains are easy to obtain high-quality long CoT data. To solve this, our idea is to decouple long CoT reasoning into general reasoning patterns and domain-specific information. Since both reasoning patterns and domain-specific information are important for general long CoT reasoning, we design tailored methods to extract each kind of representation and inject them to control model behaviors. The overall framework is illustrated in Figure 3.

4.1 Contrastive Reasoning Pattern Representation

Recall that different domains share similar contrastive representations between long and vanilla CoTs, as stated in Section 3.2 ("Analysis of General Representations"). Motivated by this finding, we propose to facilitate the transition from vanilla CoT to long CoT patterns by leveraging contrastive representations from high-resource domains (*e.g.,* mathematics). In the following part, we introduce how to extract and inject contrastive reasoning pattern representations.

Representation Extraction. Given a set of questions $\mathcal{X}$ from high-resource domains, first, we extract their representations following the method in Section 3.2. Then, to enhance the transferability of extracted representations, we average the contrastive representations across all the data, abstracting away domain-specific information. Formally, we calculate the contrastive reasoning pattern representation p_L at the layer L as follows:

$$p_L = \frac{1}{|\mathcal{X}|} \sum_{i=1}^{|\mathcal{X}|} (R_L(l_i) - R_L(s_i)), \tag{3}$$

where $\mathcal{X}$ is the set of questions, $R_L(l_i)$ and $R_L(s_i)$ denote the representations of long and vanilla CoT at layer L, respectively.

Representation Control. After extracting the reasoning pattern representation, we inject it into specific layers of LLMs during the forward pass to guide LLMs towards deliberate reasoning. Since the reasoning pattern is a global behavior in the generation process, we choose to inject the representation into that of the first token, ensuring that each following token can attend to it. In addition, to preserve the original capabilities of LLMs as much as possible, we follow Liu et al. (2024) to normalize the updated representations. Formally, the updated representation $\tilde{h}_L^0$ of the first token at the layer L is calculated as follows:

$$\tilde{h}_L^0 = h_L^0 + \lambda_p \cdot p_L, \tag{4}$$

$$\tilde{h}_L^0 = \tilde{h}_L^0 \cdot \frac{\|h_L^0\|_2}{\|\tilde{h}_L^0\|_2}, \tag{5}$$

where λ_p is the hyperparameter controlling the strength of injection.

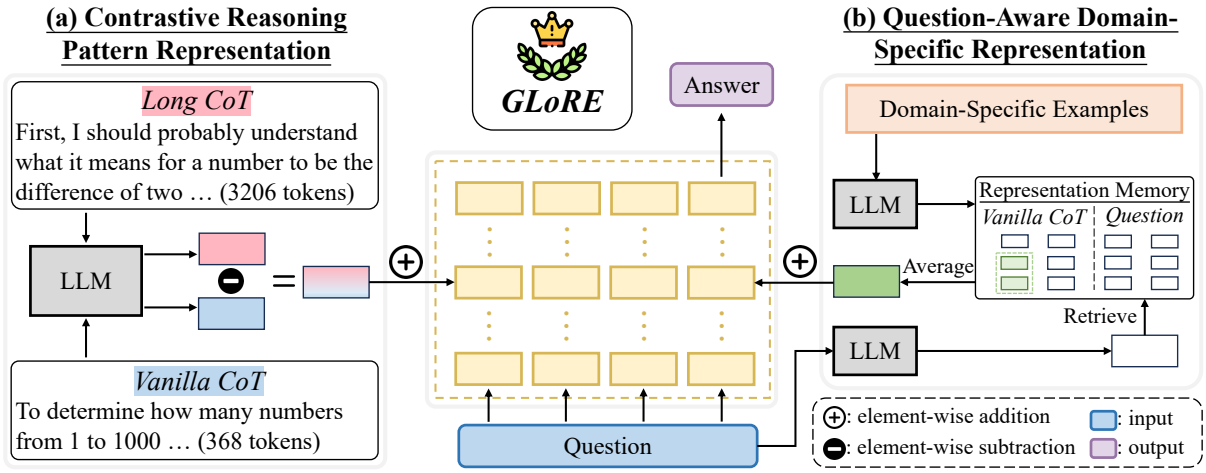


Figure 3: The overview of our approach **GLoRE**. It extracts contrastive reasoning pattern representations (part a) and question-aware domain-specific representations (part b) and injects them into LLMs. For part a, we derive the representations from the difference between long and vanilla CoTs. For part b, we construct a domain-specific representation memory from vanilla CoTs only and retrieve representations relevant to the question.

4.2 Question-Aware Domain-Specific Representation

After injecting the representation of the long CoT reasoning pattern, LLMs can be steered to generate long CoTs at inference. However, as stated in Section 3.2 ("Analysis of Domain-Specific Representations"), reasoning patterns only are not enough. For effective long CoT reasoning, domain-specific information is also important. Therefore, we propose to extract domain-specific representations and construct a representation memory. Note that the memory is constructed from vanilla CoT data, which is easy to obtain. At inference time, to provide domain-specific information for long CoT reasoning, we retrieve representations relevant to the question and inject them into LLMs. In the following part, we detail these two components.

Domain-Specific Representation Memory. Since domain-specific information is shared across long and vanilla CoTs, we propose to collect them using only vanilla CoTs, which can be easily obtained through methods like zero-shot prompting (*e.g.,*"Let's think step by step."). Then, we can construct a representation memory for relevant information retrieval at inference time. Specifically, for a question x_i and its associated vanilla CoT s_i, we extract the representation of the question $R_L(x_i)$ as the key of the memory and the representation of the question combined with the CoT $R_L(s_i)$ as the value of the memory.

Question-Aware Representation Retrieval. With the domain-specific representation memory, we can retrieve representations relevant to the specific question for better long CoT reasoning. Specifically, for a question x, we extract its representation as the query to retrieve top-k representations from the memory. The retrieval is implemented by first calculating cosine similarity between the query and keys and then extracting the corresponding values with the highest similarity values. To highlight common information, we further average the k retrieved representations. Finally, we follow the method in Section 4.1 to inject the domain-specific representation d into LLMs. Different from Section 4.1, we choose to inject into the final token position, as it can influence the generation of the next token while preserving the encoding of previous tokens. It can be represented as follows:

$$\tilde{h}_L^{-1} = h_L^{-1} + \lambda_d \cdot d, \tag{6}$$

$$\tilde{h}_L^{-1} = \tilde{h}_L^{-1} \cdot \frac{\|h_L^{-1}\|_2}{\|\tilde{h}_L^{-1}\|_2}, \tag{7}$$

where λ_d is the hyperparameter controlling the strength of injection.

5 Experiments

In this section, we first set up the experiments, then report the results and conduct a detailed analysis.

5.1 Experimental Setup

CoT Examples Construction. To obtain vanilla and long CoT examples, we utilize open-source

Scenarios		In-domain				Cross-domain			
Task		Math BenchMarks				GPQA			
		MATHOAI	AIME24	AMC23	Average	Physics	Chemistry	Biology	Overall
Qwen2.5-7B -Instruct	Zero-shot CoT	72.80	13.33	47.50	44.54	38.37	22.58	36.84	30.81
	Few-shot CoT	69.80	6.67	42.50	39.66	33.72	20.43	26.32	26.77
	BoostStep	70.80	10.00	45.00	41.93	34.88	23.66	21.05	28.28
	MathNeuro	73.60	16.67	50.00	46.76	37.21	22.58	36.84	30.30
	RoT	73.80	16.67	52.50	47.66	41.86	23.66	42.11	33.33
	SFT	74.80	23.33	**60.00**	52.71	44.19	23.66	**52.63**	35.35
	GLoRE	**76.20**	**26.67**	**60.00**	**54.29**	**46.51**	**25.81**	42.11	**36.36**
Llama3.1-8B -Instruct	Zero-shot CoT	48.20	3.33	30.00	27.18	19.77	16.13	42.11	20.20
	Few-shot CoT	45.60	6.67	27.50	26.59	22.09	15.05	36.84	20.20
	BoostStep	47.60	6.67	25.00	26.42	20.93	18.28	36.84	21.21
	MathNeuro	49.00	10.00	30.00	29.67	22.09	22.58	**47.37**	24.75
	RoT	49.40	10.00	32.50	30.63	24.42	23.66	42.11	25.76
	SFT	50.20	13.33	**35.00**	32.84	26.88	27.96	**47.37**	29.29
	GLoRE	**51.60**	**16.67**	**35.00**	**34.42**	**27.91**	**29.03**	**47.37**	**30.30**

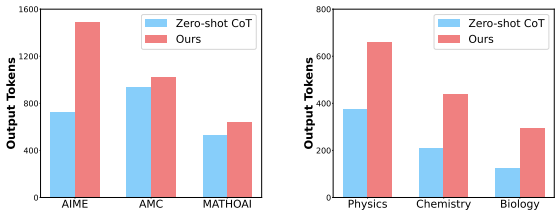
Table 1: Performance comparison in both in-domain and cross-domain scenarios using Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct. The best method in each group is marked in **bold**.

data from STILL-2 (Min et al., 2024), which is a high-quality dataset distilled from DeepSeek-R1-Lite-Preview (Guo et al., 2025). From this dataset, we randomly select 100 examples from the mathematics, physics, chemistry, and biology domains, respectively.

Datasets. To comprehensively evaluate the efficacy of our proposed method, we conduct experiments in two scenarios: in-domain and cross-domain. For the in-domain scenario, we evaluate our method on several challenging open-source mathematical benchmarks, including MATHOAI (Lightman et al., 2024), AIME2024, and AMC2023. For the cross-domain scenario, we utilize the GPQA (Rein et al., 2024) dataset, which is a challenging multiple-choice benchmark crafted by domain experts in physics, chemistry, and biology. In this paper, we use the highest quality diamond set for evaluation following Li et al. (2025).

Baselines. To facilitate a systematic comparison, we select several representative methods, including prompting-based approaches (*i.e.,* Zero-shot CoT, Few-shot CoT, and BoostStep (Zhang et al., 2025)), neuron activation method (*i.e.,* MathNeuro (Christ et al., 2024)), representation engineering method (*i.e.,* RoT (Hu et al., 2024)), and supervised fine-tuning method. Detailed descriptions of these baselines are provided in Appendix E.

Implementation Details. In our experiments, we use two representative open-source LLMs:



(a) In-domain (b) Cross-domain

Figure 4: The average number of output tokens generated by Qwen2.5-7B-Instruct.

Qwen2.5-7B-Instruct (Yang et al., 2024) and Llama3.1-8B-Instruct (Dubey et al., 2024). For the contrastive reasoning pattern representation, we set the injection strength λ_p as 0.1. For the question-aware domain-specific representation, we set the number of retrieved representations k as 8 and the injection strength λ_d as 0.1. Both representations are injected into the intermediate layer of LLMs. Following the existing work (Li et al., 2023b; Wang et al., 2024; Cheng et al., 2024b), we use the greedy decoding strategy for inference.

5.2 Experimental Results

The experimental results are presented in Table 1. As we can see, the few-shot CoT method (*i.e.,* few-shot CoT and BoostStep) performs poorly, even worse than the zero-shot CoT. The main reason is that LLMs lose their ability to learn from demonstrations after supervised fine-tuning (Wei et al., 2023). Adding examples at the discrete token level

Task	MATHOAI		GPQA	
Model	**Qwen2.5-7B -Instruct**	**Llama3.1-8B -Instruct**	**Qwen2.5-7B -Instruct**	**Llama3.1-8B -Instruct**
GLoRE	76.20	51.60	36.36	30.30
w/o CR	73.40	49.80	33.84	25.76
w VR	73.60	50.00	33.84	26.26
w LR	75.40	51.20	35.35	30.30
w/o DR	74.20	50.60	32.32	26.77
w LDR	76.40	51.60	37.88	29.80

Table 2: Ablation study on MATHOAI and GPQA datasets. "CR", "VR", "LR", "DR" and "LDR" denote contrastive reasoning pattern representation, vanilla CoT representation, long CoT representation, question-aware domain-specific vanilla, and long CoT representation.

Methods	Zero-shot	Few-shot	BoostStep	**GLoRE**
T.C.	$\mathcal{O}(p^2)$	$\mathcal{O}((d+p)^2)$	$\mathcal{O}(n(d+p)^2)$	$\mathcal{O}(p^2)$

Table 3: The efficiency analysis of **GLoRE** and previous work. Here, "T.C." is the time complexity, d, p and n denote the length of the demonstrations, the length of the problem and the number of the reasoning steps.

can increase input length, which distracts the model from the current problem and disrupts the inference process (Dong et al., 2024b; Peng et al., 2025). In contrast, MathNeuro can improve performance by identifying and scaling relevant neurons, but it focuses solely on specific neuron connections, neglecting the cooperative activity of multiple neurons. To address this limitation, RoT introduces contrastive representations of CoT and non-CoT prompts, enabling fine-grained control over the reasoning process. However, this approach is insufficient to guide the model toward deliberate thinking, resulting in limited performance improvements.

Finally, **GLoRE** significantly outperforms all the training-free baselines and even surpasses the supervised fine-tuning method. Our approach first uses a contrastive reasoning pattern representation to switch the LLM to a slow thinking pattern, enabling it to engage in deep thinking and perform step-by-step reasoning. This allows **GLoRE** to generate longer and more detailed reasoning solutions, indicating that the representation effectively guides LLMs into a slow-thinking mode, as illustrated in Figure 4. Additionally, for specific problems, we leverage question-aware domain-specific representation, which provides domain-specific information during inference to achieve fine-grained control over the reasoning process.

5.3 Detailed Analysis

In this part, we construct a detailed analysis of the effectiveness and efficiency of our approach.

5.3.1 Ablation Study

Our approach incorporates two key components to activate the long CoT reasoning capabilities of LLMs. To validate each component of our proposed method, we conduct an ablation study by removing or replacing the contrastive reasoning pattern and question-aware domain-specific representation on MATHOAI and GPQA datasets.

The results are presented in Table 2. We can see that removing any component would lead to performance degradation, indicating that all the components in our method are helpful. Specifically, for the contrastive reasoning pattern representation, we compare the effects of injecting only the representation of vanilla CoT or long CoT and observe that both lead to performance degradation. In particular, injecting only vanilla CoT representation significantly reduces performance, as the model fails to transition into a slow-thinking mode. For the question-aware domain-specific representation, we observe that injecting long CoT domain-specific thought achieves performance comparable to vanilla CoT. This indicates that our method can effectively leverage vanilla CoT from other domains, highlighting its cost efficiency.

5.3.2 The Efficiency of GLoRE

In this part, We discuss the efficiency of **GLoRE**, as shown in Table 3. First, the few-shot CoT method incorporates additional demonstrations in the input, which leads to the increased inference time. This is because the time complexity of the Transformer scales quadratically with the length of the input sequence (Dong et al., 2025a; Zhan et al., 2024; Liu et al., 2025). Additionally, BoostStep decomposes the reasoning process into multiple substeps and guides the model with relevant examples at each step. This requires the model to perform multiple reasoning iterations for a single problem, further increasing the computational overhead. In contrast, **GLoRE** maintains the same time complexity as zero-shot CoT, which is significantly lower than few-shot CoT and BoostStep. We extract representation from the model's latent space and inject them during inference, without reducing the reasoning efficiency. This demonstrates that **GLoRE** can significantly enhance reasoning capabilities through representation engineering while preserving computational efficiency.

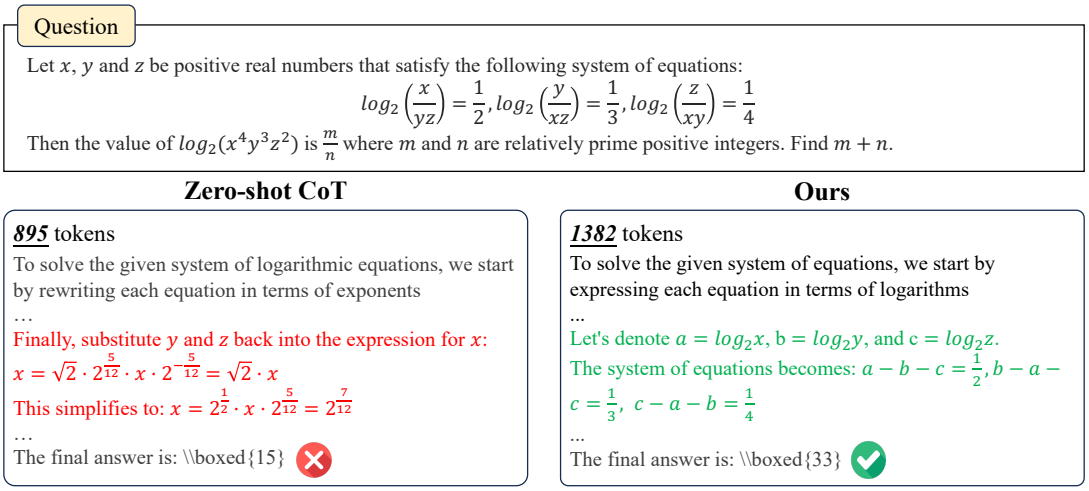
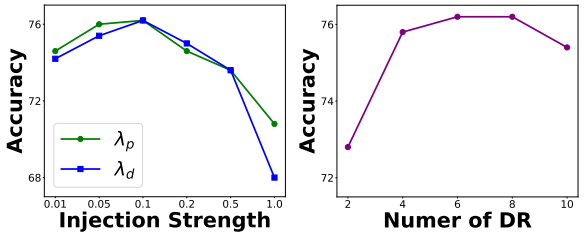
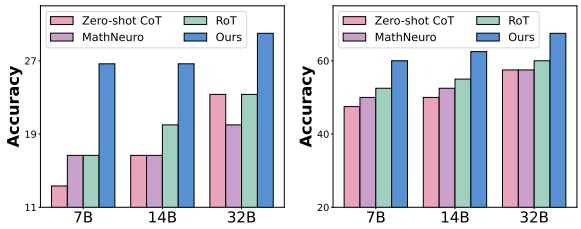


Figure 5: A specific example of how our method activates the long CoT reasoning capabilities of LLMs.



(a) MATHOAI (b) GPQA

Figure 6: Performance comparison *w.r.t.* the inject strength λ_p and λ_d, and the number of retrieved domain-specific representations k on the MATHOAI dataset using Qwen2.5-7B-Instruct. Here, "DR" denotes the retrieved question-aware domain-specific representation.



(a) AIME (b) AMC

Figure 7: Performance comparison on AIME and AMC datasets using Qwen-series LLMs.

5.3.3 Hyper-parameters Analysis

GLoRE includes a few hyper-parameters to tune. In this part, we report the tuning results of three hyper-parameters: the injection strength for contrastive reasoning pattern representation (λ_p) and question-aware domain-specific representation (λ_d), and the number of retrieved representations k. The results are shown in Figure 6.

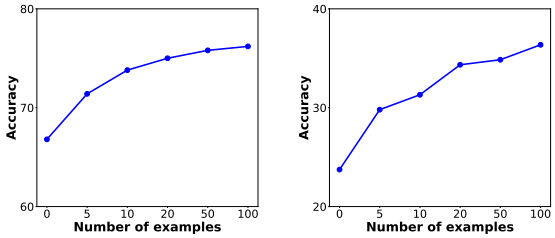
We find that the performance is optimal when both injection strengths are set to 0.1. If the injection strength is too small, the model cannot effectively perceive the intervention of the representations, preventing it from engaging in slow thinking or incorporating domain-specific information. Conversely, if the injection strength is too large, the injected representations may disrupt the original semantic information of the model, leading to performance degradation. Additionally, we observe that **GLoRE** achieves the best performance when the number of similar representations is set to 8. If the number is too small, the model cannot access sufficient domain-specific information to support reasoning. In contrast, if the number is too large, irrelevant information may be introduced, which can interfere with the reasoning process.

5.3.4 Experiments on Larger Models

In this part, we conduct experiments on Qwen-series LLMs using AIME and AMC datasets. As illustrated in Figure 7, **GLoRE** consistently outperforms all other baselines. This further demonstrates the effectiveness of our proposed method.

5.3.5 The Effect of the Representation Memory Size

In this part, we investigate the impact of scaling the representation memory size on **GLoRE**. The results are illustrated in 8. As we can see, the performance consistently improves as the number of examples in the representation memory increases. This improvement can be attributed to two main reasons. On one hand, when computing the average representation of the contrastive reasoning pattern representation, using a larger num-



(a) MATHOAI (b) GPQA

Figure 8: Accuracy with increasing numbers of examples in the representation memory on the MATHOAI and GPQA datasets using Qwen2.5-7B-Instruct.

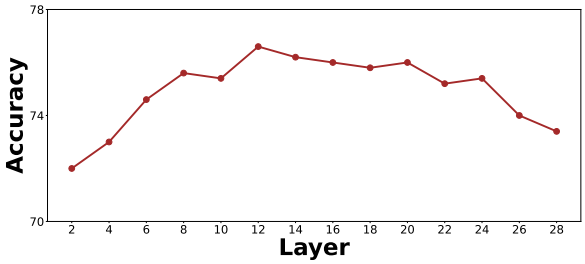


Figure 9: Performance comparison of the injection layer L on the MATHOAI dataset using Qwen2.5-7B-Instruct.

ber of demonstrations helps better isolate problem-specific information, resulting in more precise high-level long CoT pattern representations. On the other hand, when extracting and injecting domain-specific features, the increased representation memory provides the model with access to more relevant question-aware domain-specific representations, enabling fine-grained refinement of the reasoning process in specific domains.

5.3.6 The Impact of Layer Selection

In this section, we explore the impact of layer selection. We conduct experiments across different layers on the MATHOAI dataset using Qwen2.5-7B-Instruct. The results are presented in Figure 9. Our method exhibits a performance peak at the middle layer, with performance improving as the number of layers increases initially, but plateauing or declining in later layers. Additionally, our method is not significantly affected by the layer selection, which demonstrates its robustness.

5.3.7 Case Study

In this part, we demonstrate a specific example of how **GLoRE** activates the long CoT reasoning capabilities of LLMs. The case study example is illustrated in Figure 5. Overall, compared to the zero-shot CoT method, **GLoRE** can encourage LLMs to generate more intermediate reasoning steps, enabling them to engage in deliberate thinking. Specifically, the zero-shot CoT method directly converts y and z into x, leading to errors in complex variable substitution and simplification, which disrupts the reasoning chain and results in calculation mistakes. In contrast, **GLoRE** introduces intermediate variables to simplify the reasoning process and structures the problem-solving approach in a step-by-step manner. This approach helps maintain logical consistency throughout the reasoning process, significantly activating the long CoT reasoning capabilities of LLMs on complex reasoning problems.

6 Conclusion

In this work, we conduct an empirical analysis for the mechanism of long CoT reasoning from the perspective of representation. Our findings reveal that long CoT reasoning appears to be a general capability potentially encoded in LLMs. Inspired by this, we propose a novel training-free method based on representation engineering, which can effectively and efficiently unleash the general long CoT reasoning capabilities of LLMs. Overall, our work provides a deeper understanding of long CoT reasoning, paving the way for transparent and interpretable slow-thinking reasoning models.

7 Limitations

One limitation of our work is that our method requires access to the internal representations of the model, making it infeasible for closed-source LLMs. In addition, due to the constraints of our cost and resources, we only conduct experiments on representative tasks and LLMs.

Acknowledgements

This work was partially supported by National Natural Science Foundation of China under Grant No. 92470205 and 62222215, Beijing Municipal Science and Technology Project under Grant No. Z231100010323009, Beijing Natural Science Foundation under Grant No. L233008, and Ant Group. Xin Zhao is the corresponding author.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. In *NeurIPS*.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. 2024. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. In *NeurIPS*.

Xiaoxue Cheng, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024a. Chainlm: Empowering large language models with improved chain-of-thought prompting. In *LREC/COLING*, pages 2969–2983. ELRA and ICCL.

Xiaoxue Cheng, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2025. Think more, hallucinate less: Mitigating hallucinations via dual process of fast and slow thinking. *CoRR*, abs/2501.01306.

Xiaoxue Cheng, Junyi Li, Xin Zhao, Hongzhi Zhang, Fuzheng Zhang, Di Zhang, Kun Gai, and Ji-Rong Wen. 2024b. Small agent can also rock! empowering small language models as hallucination detector. In *EMNLP*, pages 14600–14615. Association for Computational Linguistics.

Bryan R. Christ, Zack Gottesman, Jonathan Kropko, and Thomas Hartvigsen. 2024. Math neurosurgery: Isolating language models' math reasoning abilities using only forward passes. *CoRR*, abs/2410.16930.

Zican Dong, Junyi Li, Jinhao Jiang, Mingyu Xu, Wayne Xin Zhao, Bingning Wang, and Weipeng Chen. 2025a. Longred: Mitigating short-text degradation of long-context large language models via restoration distillation. *CoRR*, abs/2502.07365.

Zican Dong, Junyi Li, Xin Men, Xin Zhao, Bingning Wang, Zhen Tian, Weipeng Chen, and Ji-Rong Wen. 2024a. Exploring context window of large language models via decomposed positional vectors. In *NeurIPS*.

Zican Dong, Han Peng, Peiyu Liu, Wayne Xin Zhao, Dong Wu, Feng Xiao, and Zhifeng Wang. 2025b. Domain-specific pruning of large mixture-of-experts models with few-shot demonstrations. *arXiv preprint arXiv:2504.06792*.

Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024b. BAMBOO: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *LREC/COLING*, pages 2086–2099. ELRA and ICCL.

Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. 2025. Virgo: A preliminary exploration on reproducing o1-like mllm. *arXiv preprint arXiv:2501.01904*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Alaaeldin El-Nouby, Michal Klein, Shuangfei Zhai, Miguel Ángel Bautista, Vaishaal Shankar, Alexander T. Toshev, Joshua M. Susskind, and Armand Joulin. 2024. Scalable pre-training of large autoregressive image models. In *ICML*. OpenReview.net.

Siqi Fan, Xin Jiang, Xiang Li, Xuying Meng, Peng Han, Shuo Shang, Aixin Sun, Yequan Wang, and Zhongyuan Wang. 2024. Not all layers of llms are necessary during inference. *CoRR*, abs/2403.02181.

Luis Gonzalo Sánchez Giraldo, Murali Rao, and José C. Príncipe. 2015. Measures of entropy from data using infinitely divisible kernels. *IEEE Trans. Inf. Theory*, 61(1):535–548.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. 2025. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *arXiv preprint arXiv:2501.04519*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Roee Hendel, Mor Geva, and Amir Globerson. 2023. In-context learning creates task vectors. In *EMNLP (Findings)*, pages 9318–9333. Association for Computational Linguistics.

Bertram Højer, Oliver Simon Jarvis, and Stefan Heinrich. 2025. Improving reasoning performance in large language models via representation engineering. In *The Thirteenth International Conference on Learning Representations*.

Oskar John Hollinsworth, Curt Tigges, Atticus Geiger, and Neel Nanda. 2024. Language models linearly represent sentiment. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 58–87, Miami, Florida, US. Association for Computational Linguistics.

John J Hopfield. 1982. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558.

Lijie Hu, Liang Liu, Shu Yang, Xin Chen, Zhen Tan, Muhammad Asif Ali, Mengdi Li, and Di Wang. 2024. Understanding reasoning in chain-of-thought from the hopfieldian view. *CoRR*, abs/2410.03595.

Kenneth Li, Oam Patel, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023a. Inference-time intervention: Eliciting truthful answers from a language model. In *NeurIPS*.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint arXiv:2501.05366*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023b. Evaluating object hallucination in large vision-language models. In *EMNLP*, pages 292–305. Association for Computational Linguistics.

Yifan Li, Hangyu Guo, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. 2024. Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models. In *ECCV (73)*, volume 15131 of *Lecture Notes in Computer Science*, pages 174–189. Springer.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *ICLR*. OpenReview.net.

Sheng Liu, Haotian Ye, Lei Xing, and James Y. Zou. 2024. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *ICML*. OpenReview.net.

Yurou Liu, Jiahao Chen, Rui Jiao, Jiangmeng Li, Wenbing Huang, and Bing Su. 2025. DenoiseVAE: Learning molecule-adaptive noise distributions for denoising-based 3d molecular pre-training. In *The Thirteenth International Conference on Learning Representations*.

Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, Wayne Xin Zhao, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. 2024. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *CoRR*, abs/2412.09413.

OpenAI. 2024. Learning to reason with llms.

Bo Pang, Hanze Dong, Jiacheng Xu, Silvio Savarese, Yingbo Zhou, and Caiming Xiong. 2025. Bolt: Bootstrap long chain-of-thought in language models without distillation. *arXiv preprint arXiv:2502.03860*.

Han Peng, Jinhao Jiang, Zican Dong, Wayne Xin Zhao, and Lei Fang. 2025. Cafe: Retrieval head-based coarse-to-fine information seeking to enhance multi-document qa capability. *Preprint*, arXiv:2505.10063.

Daking Rai and Ziyu Yao. 2024. An investigation of neuron activation as a unified lens to explain chain-of-thought eliciting arithmetic reasoning of llms. In *ACL (1)*, pages 7174–7193. Association for Computational Linguistics.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Daniel Scalena, Gabriele Sarti, and Malvina Nissim. 2024. Multi-property steering of large language models with dynamic activation composition. *CoRR*, abs/2406.17563.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. 2025. Layer by layer: Uncovering hidden representations in language models. *Preprint*, arXiv:2502.02013.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. 2025. Improving instruction-following in language models through activation steering. In *The Thirteenth International Conference on Learning Representations*.

Xinyu Tang, Xiaolei Wang, Wayne Xin Zhao, Siyuan Lu, Yaliang Li, and Ji-Rong Wen. 2024a. Unleashing the potential of large language models as prompt optimizers: An analogical analysis with gradient-based model optimizers. *CoRR*, abs/2402.17564.

Xinyu Tang, Xiaolei Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2024b. DAWN-ICL: strategic planning of problem-solving trajectories for zero-shot in-context learning. *CoRR*, abs/2410.20215.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Dimitri von Rütte, Sotiris Anagnostidis, Gregor Bachmann, and Thomas Hofmann. 2024. A language model's guide through latent space. In *ICML*. OpenReview.net.

Yuhao Wang, Ruiyang Ren, Junyi Li, Xin Zhao, Jing Liu, and Ji-Rong Wen. 2024. REAR: A relevance-aware retrieval-augmented framework for open-domain question answering. In *EMNLP*, pages 5613–5626. Association for Computational Linguistics.

Yuhao Wang, Ruiyang Ren, Yucheng Wang, Wayne Xin Zhao, Jing Liu, Hua Wu, and Haifeng Wang. 2025a. Reinforced informativeness optimization for long-form retrieval-augmented generation. *Preprint*, arXiv:2505.20825.

Yuhao Wang, Ruiyang Ren, Yucheng Wang, Wayne Xin Zhao, Jing Liu, Hua Wu, and Haifeng Wang. 2025b. Unveiling knowledge utilization mechanisms in llm-based retrieval-augmented generation. *arXiv preprint arXiv:2505.11995*.

Lai Wei, Zihao Jiang, Weiran Huang, and Lichao Sun. 2023. Instructiongpt-4: A 200-instruction paradigm for fine-tuning minigpt-4. *CoRR*, abs/2308.12067.

Lai Wei, Zhiquan Tan, Chenghai Li, Jindong Wang, and Weiran Huang. 2024. Large language model evaluation via matrix entropy. *CoRR*, abs/2401.17139.

Haotian Xu, Xing Wu, Weinong Wang, Zhongzhi Li, Da Zheng, Boyuan Chen, Yi Hu, Shijia Kang, Jiaming Ji, Yingying Zhang, et al. 2025. Redstar: Does scaling long-cot data unlock better slow-reasoning systems? *arXiv preprint arXiv:2501.11284*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. Limo: Less is more for reasoning. *arXiv preprint arXiv:2502.03387*.

Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. 2025. Demystifying long chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*.

Ping Yu, Jing Xu, Jason Weston, and Ilia Kulikov. 2024. Distilling system 2 into system 1. *CoRR*, abs/2407.06023.

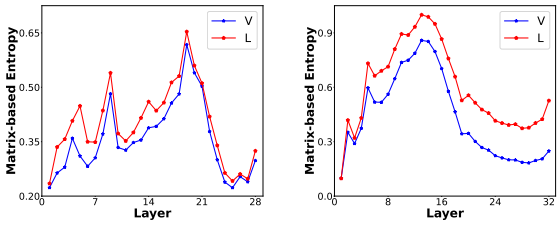
Yu-Liang Zhan, Zhong-Yi Lu, Hao Sun, and Ze-Feng Gao. 2024. Over-parameterized student model via tensor decomposition boosted knowledge distillation. In *NeurIPS*.

Beichen Zhang, Yuhong Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Haodong Duan, Yuhang Cao, Dahua Lin, and Jiaqi Wang. 2025. Booststep: Boosting mathematical capability of large language models via improved single-step reasoning. *arXiv preprint arXiv:2501.03226*.

Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. 2024. Rest-mcts*: LLM self-training via process reward guided tree search. In *NeurIPS*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *CoRR*, abs/2303.18223.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. 2023. Representation engineering: A top-down approach to AI transparency. *CoRR*, abs/2310.01405.



(a) Qwen2.5-7B-Instruct (b) Llama3.1-8B-Instruct

Figure 10: Matrix-based entropy across all layer in Qwen2.5-7B-Instruct and Llama3.1-8b-Instruct. "V" and "L" denote the vanilla and long CoT, respectively.

A Quantitative Analysis of Vanilla and Long CoT Representations.

After discovering the distinct distributions of vanilla and long CoTs within LLMs, we further conduct a quantitative analysis of their representations. Specifically, we employ matrix-based entropy (Giraldo et al., 2015; Wei et al., 2024) to investigate the information content across different layers for both CoTs. Given the representations of n samples $Z \in \mathbb{R}^{n*d}$, the matrix-based entropy $S_\alpha(\mathbf{Z})$ quantifies the diversity of features within the representations, as defined by the following equations:

$$\mathbf{K} = \mathbf{Z}\mathbf{Z}^\top, \tag{8}$$

$$S_\alpha(\mathbf{Z}) = \frac{1}{1-\alpha} \log \left(\sum_{i=1}^{r} \left(\frac{\lambda_i(\mathbf{K})}{\mathrm{tr}(\mathbf{K})} \right)^\alpha \right), \tag{9}$$

where $\mathbf{K}$ is the Gram matrix of the representation $\mathbf{Z}$, $\lambda_i(\mathbf{K})$ represents the nonnegative eigenvalues of $\mathbf{K}$, and $r = \mathrm{rank}(\mathbf{K}) \leq \min(d, n)$. Following Skean et al. (2025), we set $\alpha = 1$ for simplicity.

The matrix-based entropy metrics for vanilla and long CoTs representations across different layers in Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct are illustrated in Figure 10. We observe that the matrix-based entropy of long CoT is consistently higher than that of vanilla CoT, indicating that long CoT contains more diverse and less redundant features within the latent space. Additionally, we find that the entropy in the middle layers of the model is higher than in the final layer in both CoTs. This suggests that the middle layers are better at extracting diverse and complex features (Wang et al., 2025b), exhibiting powerful capabilities in reasoning tasks (El-Nouby et al., 2024; Fan et al., 2024).

Domain	Math	Physics	Chemistry	Biology
Average Tokens of Vanilla CoT	400.98	365.45	356.59	347.73
Average Tokens of Long CoT	2628.46	2094.35	1832.86	1607.29

Table 4: Statistics of the vanilla and long CoT examples.

B Statistics of vanilla and long CoT Examples

In this paper, we leverage open-source data from STILL-2 (Min et al., 2024), a high-quality dataset consisting of vanilla and long CoTs distilled from DeepSeek-R1-Lite-Preview (Guo et al., 2025). From this dataset, we randomly select 100 examples from the math, physics, chemistry, and biology domains, respectively. Table 4 provides detailed statistics of the vanilla and long CoT examples.

C Detailed Visualization of Short-form and Long-form Thought Representations

In this section, we present the t-SNE visualizations of Qwen2.5-7B-Instruct's and Llama3.1-8B-Instruct's representations for vanilla and long CoTs across all layers, as illustrated in Figure 11, 12, 13, and 14, respectively.

D Detailed Visualization of Representations across Different Domains

In this section, we present detailed visualizations of vanilla and long CoT representations in the middle layers of LLMs across math and other domains (*i.e.,* physics, chemistry, and biology). The results are shown in Figure 15.

E Detailed Description of Baselines.

In this part, we provide detailed descriptions of all the baselines used in our experiments. These include prompting-based approaches (*i.e.,* Zero-shot CoT, Few-shot CoT, and BoostStep (Zhang et al., 2025)), neuron activation method (*i.e.,* Math-Neuro (Christ et al., 2024)), representation engineering method (*i.e.,* RoT (Hu et al., 2024)), and supervised fine-tuning method.

- **Zero-shot CoT**: The model generates answers directly using only the problem and a CoT prompt (*i.e.,* Answer the following question step by step and put the final answer in \\boxed{}) as input, without any additional demonstrations.
- **Few-shot CoT**: The model makes predictions with long CoT examples and a CoT prompt.

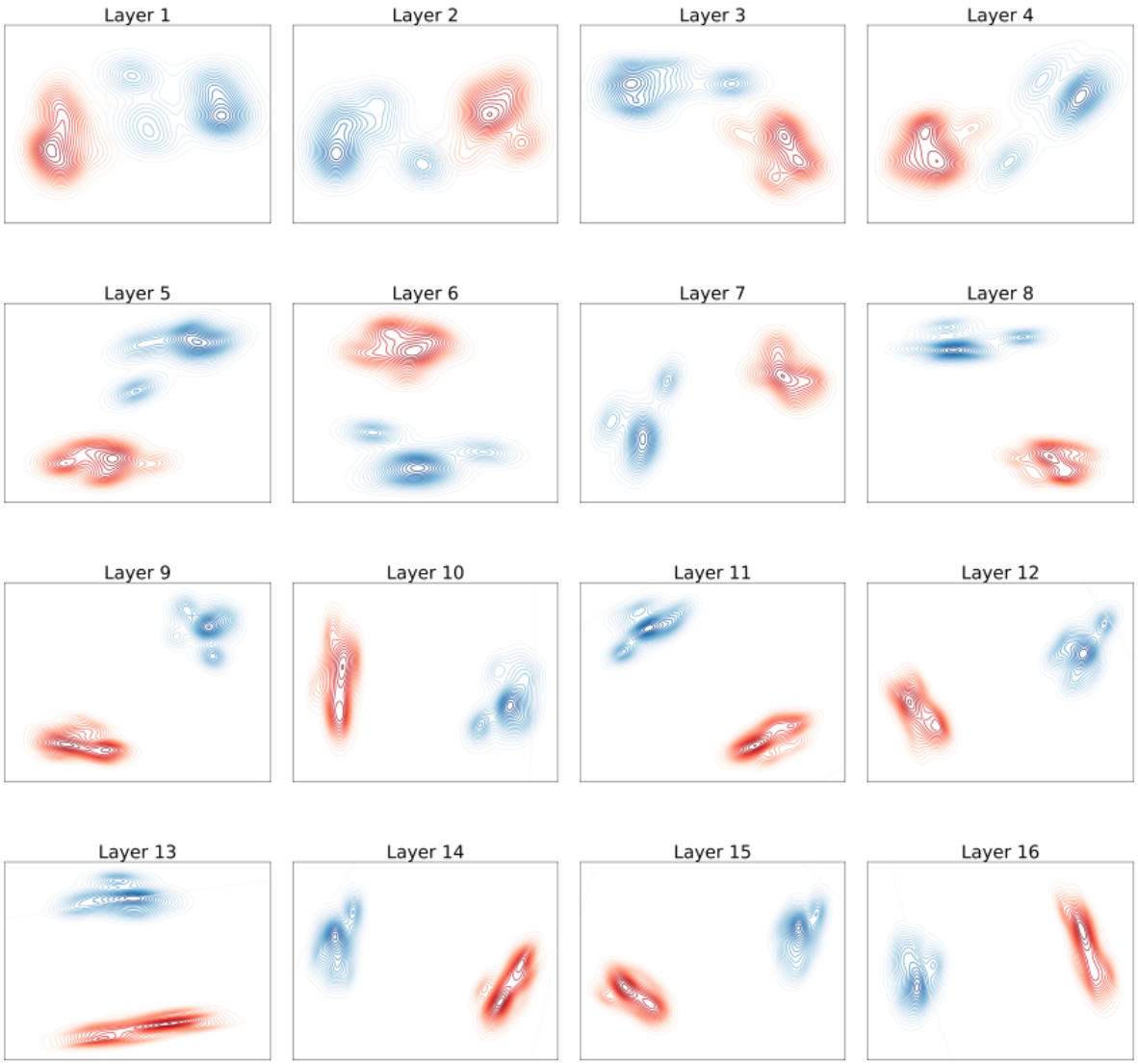


Figure 11: t-SNE plot of Qwen2.5-7B-Instruct's representations for vanilla (blue) and long CoTs (red) across 1-16 layers.

• **BoostStep** (Zhang et al., 2025): This method guides the model to perform the reasoning process incrementally and provides similar step-level examples at each reasoning step.

• **MathNeuro** (Christ et al., 2024) This method leverages weights and activations from the forward pass to identify and isolate specific parameters associated with reasoning capabilities, and enhances the model's reasoning performance through pruning and scaling of these parameters.

• **RoT** (Hu et al., 2024): This method extracts contrastive representations based on whether a CoT prompt or a non-CoT prompt is included in the input, and then injects them into the model's latent space.

• **SFT**: This method employs a supervised fine-tuning method on 100 long-form thought samples from each of four domains and performs zero-shot CoT during inference.

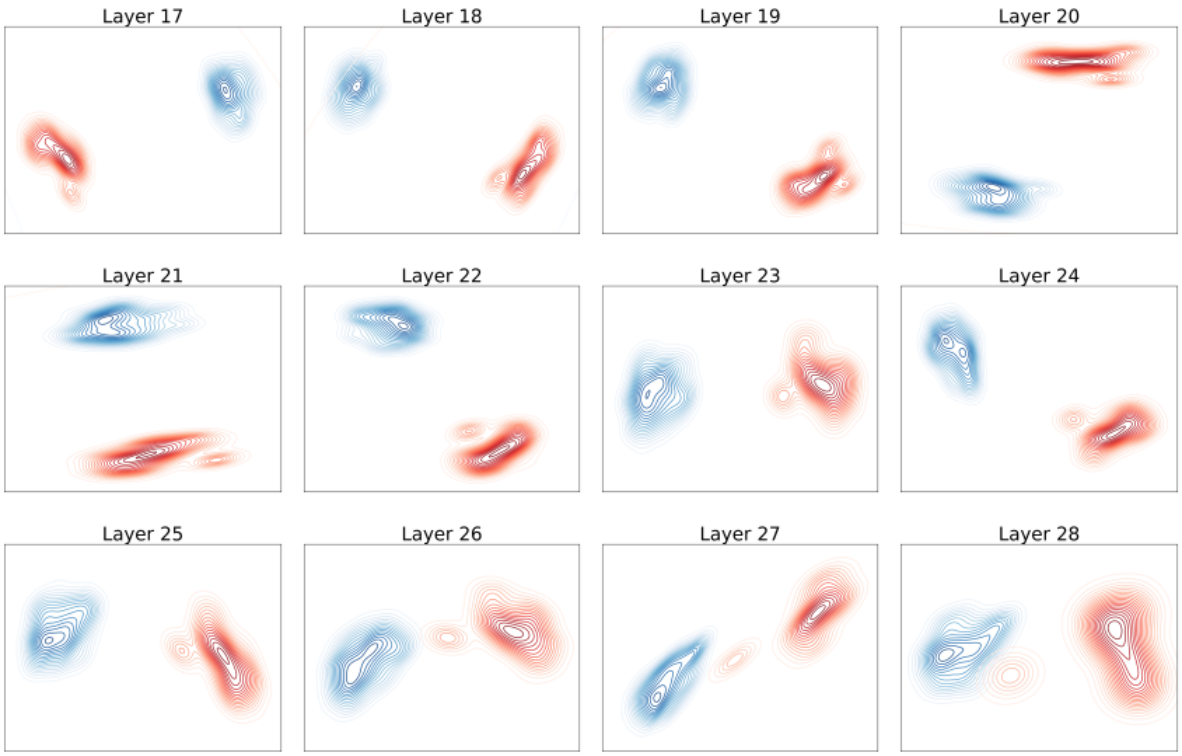


Figure 12: t-SNE plot of Qwen2.5-7B-Instruct's representations for vanilla (blue) and long CoTs (red) across 17-28 layers.

Figure 13: t-SNE plot of Llama3.1-8B-Instruct's representations for vanilla (blue) and long CoTs (red) across 1-16 layers.

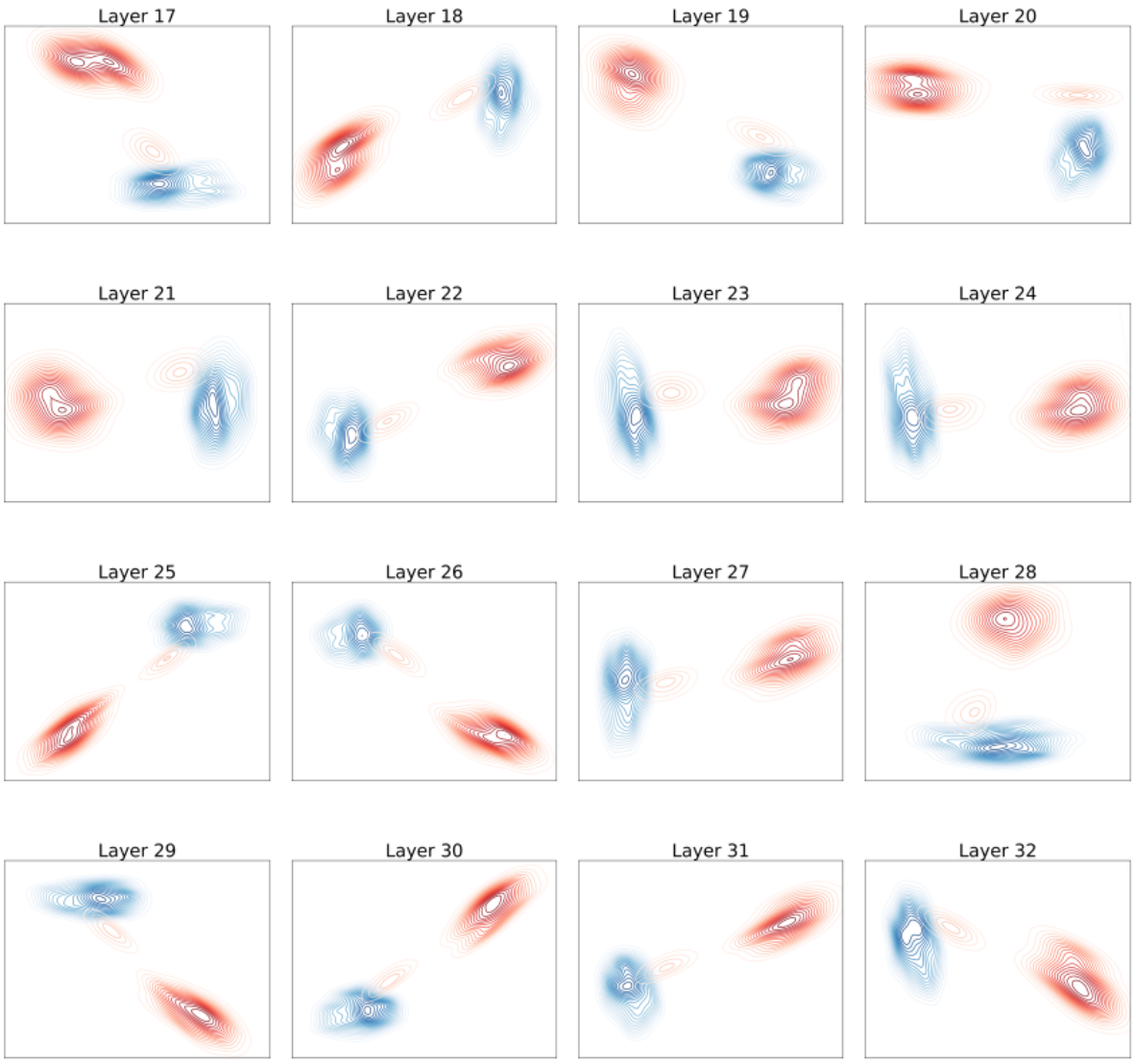
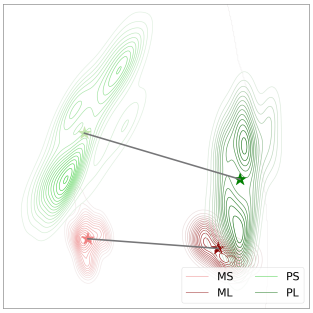
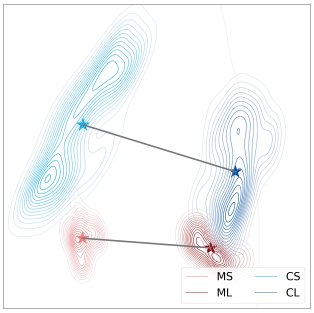


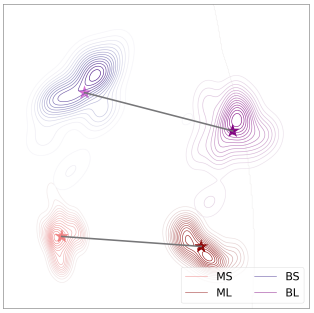
Figure 14: t-SNE plot of Llama3.1-8B-Instruct's representations for vanilla (blue) and long CoTs (red) across 17-32 layers.



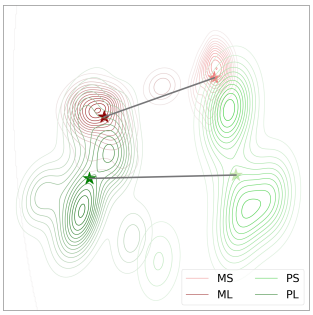
(a) Qwen (Math and Physics)



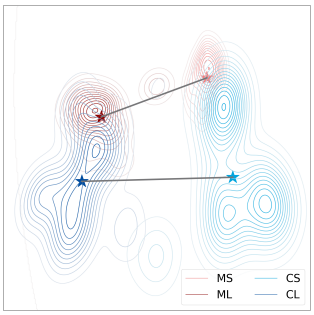
(b) Qwen (Math and Chemistry)



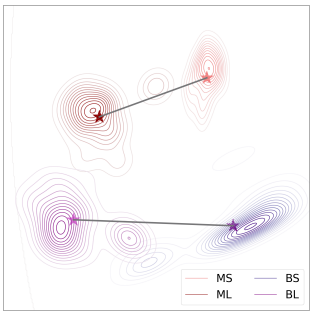
(c) Qwen (Math and Biology)



(d) Llama (Math and Physics)



(e) Llama (Math and Chemistry)



(f) Llama (Math and Biology)

Figure 15: t-SNE plot of representations from Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct for vanilla and long CoTs across math and other domains (*i.e.,* physics, chemistry and biology). "MS", "PS", "CS", and "BS" denote the vanilla CoT on the math, physics, chemistry, and biology domains, respectively. "ML", "PL", "CL", and "BL" denote the long CoT on these domains.