ThinkAnswer Loss: Balancing Semantic Similarity and Exact Matching for LLM Reasoning Enhancement

Shan Yang[1*] , **Kun Wu**[1*†] , **Zeju Li**[2*†], **Linlin Zhang**[1*]
Xiangyu Pei[1], **Leike An**[1] and **Yu Liu**[1]

[1]China Mobile Information Technology Center
[2]The Chinese University of Hong Kong
shanyang@buaa.edu.cn, wukunbupt@163.com, zjli24@cse.cuhk.edu.hk
caitlinzll00@gmail.com, xypei_0805@bupt.edu.cn, 1901111691@pku.edu.cn
liuyuit04@chinamobile.com

Abstract

Knowledge distillation for large language models often uses Chain-of-Thought (CoT) and answer pairs, but existing methods struggle with appropriate supervision signals. Uniform constraints (e.g., cross-entropy) on CoT can enforce literal, verbose reasoning and suppress expressive diversity, while solely semantic constraints on answers can reduce accuracy in classification tasks. This paper proposes ThinkAnswer Loss, an information-theoretic differential supervision framework that decouples CoT and answer supervision. ThinkAnswer Loss applies semantic similarity constraints to the CoT portion while maintaining strict literal matching for the answer. We theoretically demonstrate its connection to mutual information maximization and derive a tight upper bound on generalization error. Experimental validation on text quality assessment and mathematical reasoning tasks shows that our method maintains answer accuracy while effectively reducing CoT length and preserving semantic content, thereby accelerating inference.

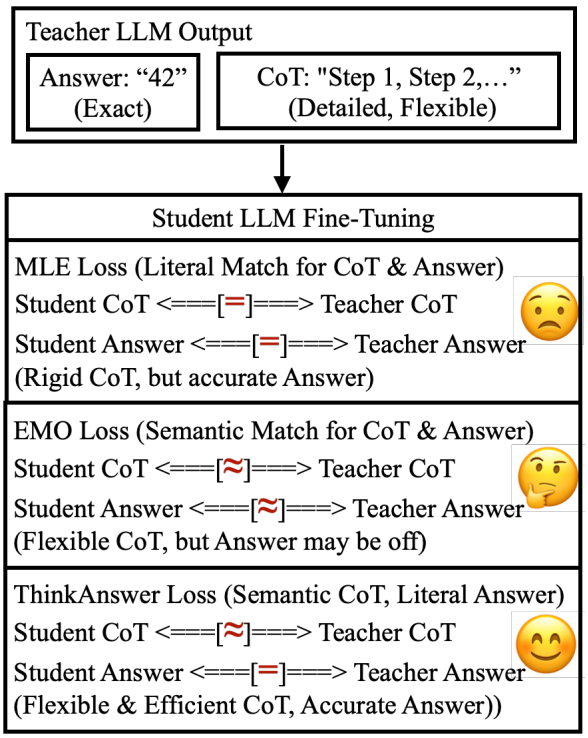


Figure 1: Comparison of Loss Functions for Chain-of-Thought and Answer Supervision: ThinkAnswer Loss Decouples Semantic Consistency and Literal Precision

1 Introduction

Supervised Fine-Tuning (SFT) of Large Language Models (LLMs) has emerged as a core paradigm for enhancing domain adaptability and task performance through task-specific data optimization. In this process, the design of loss functions directly impacts the model's ability to fit target distributions and generalization performance. Traditional approaches typically employ cross-entropy loss for Maximum Likelihood Estimation (MLE), optimizing alignment between model predictions and labeled data through literal matching (Ouyang et al., 2022). However, existing research has revealed its limitations: cross-entropy loss fails to effectively distinguish semantically similar but literally different outputs (e.g., synonym substitutions) and may impair the robustness of semantic representations by overemphasizing literal consistency when handling complex tasks (Ren et al., 2024).

The current development of large language models has established a mainstream technical paradigm of "training ultra-large parameter models first, then distilling to smaller models" (Xu et al., 2024). However, effectively preserving the reasoning capabilities of large models during this distillation process poses a critical challenge. Existing loss algorithms exhibit significant deficiencies when handling chain-of-thought and answer pairs: most algorithms adopt a uniform constraint strategy for Chain-of-Thought (CoT) and Answer supervision, either enforcing literal consistency (e.g.,

[*] These authors contributed equally to this work.
[†] Corresponding authors.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 3325–3347
November 4-9, 2025

cross-entropy loss) or solely pursuing semantic similarity (e.g., EMO loss) (Ren et al., 2024). This fragmented supervision approach can lead to representation learning conflicts—for chain-of-thought reasoning tasks requiring semantic understanding, excessive literal matching inhibits the model's generalization ability; yet for classification tasks, strict literal consistency of answers remains a necessary condition for accuracy. For example, in mathematical problem-solving tasks, models must generate answers that exactly match the labels literally, while their reasoning processes (chain-of-thought) should allow semantically equivalent but differently expressed forms. Existing methods (such as pure cross-entropy or contrastive loss) struggle to balance these dual requirements (Yan et al., 2024).

To address these challenges, we propose ThinkAnswer Loss: a method for enhancing LLM reasoning capabilities that balances semantic similarity and exact matching, aiming to simultaneously optimize the semantic consistency of chain-of-thought and the literal accuracy of answers. Specifically, for the chain-of-thought portion in distilled data, we introduce semantic similarity constraints to encourage models to generate reasoning paths that are logically equivalent to the teacher model yet diverse in expression; for the answer portion, we retain strict cross-entropy loss to ensure precise matching of classification results. Under the current technical paradigm of "large model distilling to small model," our method is particularly suitable for enhancing the reasoning capabilities of small-parameter models, maximizing inheritance of large models' reasoning advantages while maintaining computational efficiency. The contributions of this work can be summarized as follows:

1. We propose ThinkAnswer Loss, which for the first time decouples the supervision objectives of chain-of-thought and answers in LLM reasoning enhancement, balancing semantic flexibility with literal precision. We provide rigorous theoretical proofs from an information-theoretic perspective, demonstrating the relationship between our method and mutual information maximization as well as generalization performance bounds.

2. ThinkAnswer Loss effectively reduces chain-of-thought length while preserving semantic content, improving inference speed and providing a new paradigm for knowledge distillation and accelerated model inference.

3. Experimental results on text quality assessment and mathematical reasoning tasks using DeepSeek R1 distilled data demonstrate that ThinkAnswer Loss significantly improves answer accuracy while maintaining chain-of-thought diversity, comprehensively outperforming comparable loss functions.

2 Related Works

2.1 Supervised Fine-tuning of Large Language Models

Supervised fine-tuning (SFT) has become the standard paradigm for enhancing model performance in specific domains or tasks. Traditional fine-tuning methods typically employ maximum likelihood estimation (MLE), optimizing parameters by minimizing the cross-entropy loss between model predictions and labeled data (Radford et al., 2019; Ouyang et al., 2022).

2.2 Limitations of Cross-Entropy Loss

Cross-entropy loss exhibits three critical limitations (Ren et al., 2024): First, its "recall-priority" characteristic causes gradient updates to focus solely on increasing the probability of ground-truth tokens while neglecting precision, leading models to potentially overconfidence in low-quality outputs (Lucic et al., 2017; Sajjadi et al., 2018; Djolonga et al., 2019). Second, MLE treats all non-target tokens as equally incorrect (Zhang and Hai, 2018; Li et al., 2020), ignoring the reasonableness of semantically equivalent expressions in chains of thought. Finally, the objective function inconsistency between training (based on true distributions) and evaluation (based on model distributions) stages makes chain-of-thought quality assessment difficult to achieve through literal matching alone, relying more on semantic coherence and reasoning correctness (Norouzi et al., 2016; Zhang and Hai, 2018; Liu et al., 2022). These issues are particularly pronounced in chain-of-thought reasoning tasks, severely limiting models' reasoning flexibility and generalization capabilities.

2.3 Improved Loss Functions and Optimization Objectives

To address these limitations, researchers have proposed various improvements. (Li et al., 2020) introduced an objective function based on Gaussian priors, optimizing language generation by considering semantic similarities in word embedding

space. (Zhang et al., 2023) proposed MixCE, combining forward and reverse cross-entropy to balance precision and recall. Most relevant to our work is the Earth Mover Distance Optimization (EMO) (Ren et al., 2024), which uses Wasserstein distance as a distribution metric, incorporating a semantically-informed transport cost function based on word embeddings to allow models to learn semantically equivalent but differently expressed content. While EMO demonstrates the effectiveness of incorporating semantic similarity in language modeling, its key difference from our work lies in: EMO still applies a single loss function to all text, failing to recognize the fundamental difference in supervision requirements between chains of thought and final answers. Additionally, directly optimizing Earth Mover Distance (Zhao et al., 2019) is computationally complex, requiring the construction of feasible upper bounds (Ren et al., 2024), whereas our mutual information-inspired semantic similarity loss provides a more efficient implementation.

2.4 Decoupled Supervision: Semantic Consistency of Chain of Thought and Literal Accuracy of Answers

Improved Loss Functions and Optimization Objectives As chain-of-thought prompting (Wei et al., 2022) has been widely applied in complex reasoning tasks, researchers have begun to focus on how the quality of chains of thought affects final answers. Wei et al. (Wei et al., 2022) found that high-quality chains of thought significantly improve language models' reasoning accuracy. (Wang et al., 2022) proposed a self-consistency method, further enhancing chain-of-thought reasoning performance by sampling diverse reasoning paths and selecting the most consistent answer. However, traditional supervision methods still optimize chains of thought and answers as a whole, failing to effectively differentiate their distinct characteristics. (Yan et al., 2024) proposed enhancing instruction following robustness through contrastive learning, but primarily focused on instruction variants rather than decoupling chains of thought and answers.

Unlike previous work, this paper explicitly proposes the decoupling of supervision objectives for chains of thought and answers: the chain-of-thought component emphasizes semantic consistency and expressive diversity, while the answer component requires strict literal accuracy. Through the dynamic balance of mutual information-inspired semantic similarity loss and standard cross-entropy loss, ThinkAnswer Loss provides a new theoretical framework and practical approach for fine-tuning large language models.

3 Methodology

3.1 Problem Formulation

Given an instruction-response dataset $D = \{(X_i, Y_i)\}_{i=1}^{N}$, where X_i represents an input instruction and Y_i represents the model-generated response, we consider the chain-of-thought (CoT) paradigm where the response Y_i can be sequentially decomposed into a reasoning chain component T_i and a final answer component A_i, expressed as $Y_i = T_i \oplus A_i$, where $\oplus$ denotes sequence concatenation. Traditional supervised fine-tuning employs token-level cross-entropy loss:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{N}\sum_{j=1}^{|Y_i|} \log P_\theta(y_{i,j} \mid y_{i,<j}, X_i) \quad (3.1)$$

where $y_{i,j}$ denotes the j-th token in the i-th sample's response sequence, $y_{i,<j}$ represents all preceding tokens, and P_θ represents the model's conditional probability output. This monolithic loss function applies identical supervision signals to both the reasoning chain and answer components, failing to distinguish their distinct characteristics during generation: reasoning chains require semantic consistency while allowing expressive diversity, whereas answers demand precise matching to ground truth.

3.2 ThinkAnswer Loss

To address these limitations, we propose ThinkAnswer Loss, an information-theoretic multi-objective optimization framework that redefines the loss function for LLM fine-tuning to achieve dynamic balance between reasoning chain and answer supervision. We formally define:

$$\mathcal{L}_{TA} = \alpha_t \cdot \mathcal{L}_{MIM}(T, R) + (1 - \alpha_t) \cdot \mathcal{L}_{CE}(A, A^*) \quad (3.2)$$

where:

- $\mathcal{L}_{MIM}(T, R)$ represents the mutual information-inspired semantic similarity loss, measuring semantic consistency between the

model-generated reasoning chain T and reference chain R

• $\mathcal{L}_{CE}(A, A^*)$ represents standard cross-entropy loss, measuring literal matching between the model-generated answer A and ground truth answer A^*

• α_t represents dynamically adjusted weights during training, with $(1 - \alpha_t)$ as the corresponding answer component weight

3.2.1 Mutual Information-Inspired Semantic Similarity Loss

The reasoning chain mutual information maximization is based on conditional mutual information $I(T; R|A)$, representing the shared information between the model-generated reasoning chain T and reference chain R given answer A. Direct optimization of mutual information is computationally infeasible; however, several computable variational lower bounds exist (Belghazi et al., 2018):

$$I(T; R|A) \geq \mathbb{E}_{p(T,R|A)} \left[\log \frac{f(T, R)}{f(T)f(R)} \right] \quad (3.3)$$

where $f : \mathcal{T} \times \mathcal{R} \to \mathbb{R}^+$ is a positive-valued differentiable neural network scoring function.

Inspired by this theoretical foundation, we propose a direct matching approach based on the language model's intrinsic semantic space, defining the MIM loss as:

$$\mathcal{L}_{MIM} = \frac{1 - \text{sim}(f_\theta(T), f_\theta(R))}{2} \quad (3.4)$$

where:

• $f_\theta : \mathcal{X} \to \mathbb{R}^d$ is a function mapping text to a d-dimensional representation space, implemented using the language model's word embedding matrix

• $\text{sim}(\cdot, \cdot)$ is the cosine similarity function, defined as $\text{sim}(u, v) = \frac{u \cdot v}{||u||_2 \cdot ||v||_2}$

• The coefficient $\frac{1}{2}$ ensures the loss function values are normalized to the range $[0, 1]$

The theoretical soundness of this design stems from the relationship between mutual information and semantic similarity. According to Theorem 1 (detailed in Section 3.3.1), there exist constants $\Gamma > 0$ and $C < \infty$ such that:

$$I(T; R|A) \geq \frac{\Gamma}{2} \cdot \mathbb{E}[1 - \text{sim}(T, R)] - C \quad (3.5)$$

By directly optimizing the semantic similarity loss in Equation 3.4, we achieve proportional optimization of the mutual information lower bound. Unlike traditional MLE that focuses solely on literal matching while ignoring semantic similarity (Ren et al., 2024), and EMO Loss that considers semantic distances but requires constructing complex optimal transport problems with associated upper bounds (Ren et al., 2024), our method avoids high computational overhead while maintaining theoretical rigor. Our approach is both simple and effective for reasoning chain generation tasks while preserving the theoretical connection to mutual information maximization objectives.

3.3 Theoretical Analysis

We provide a formal information-theoretic analysis of our proposed method, establishing its convergence properties and generalization performance.

3.3.1 Relationship Between Mutual Information and Semantic Similarity

Theorem 3.1 (Mutual Information and Semantic Similarity Bound)**.** *Assume the semantic similarity function sim : $\mathcal{X} \times \mathcal{X} \to [-1, 1]$ is bounded and satisfies the L-Lipschitz condition. Then for any reasoning chain generation distribution $P(T|X)$, there exist constants $\Gamma > 0$ and $C < \infty$ such that:*

$$I(T; R|A) \geq \frac{\Gamma}{2} \cdot \mathbb{E}[1 - \textit{sim}(T, R)] - C \quad (3.6)$$

Proof Sketch. By leveraging the variational representation of conditional mutual information (Nguyen et al., 2008) and the Kantorovich-Rubinstein duality (Villani, 2008), we establish the relationship between mutual information and expected similarity. The complete proof is provided in Appendix A.1. □

The above theorem demonstrates that optimizing the $\mathcal{L}_{MIM}$ defined in Equation 3.4 positively correlates with preserving critical reasoning information in the information-theoretic sense. This provides theoretical justification for our similarity-based approach to approximate mutual information. Furthermore, it explains why our loss function, despite employing direct similarity measurement rather than typical contrastive learning frameworks, effectively captures the semantic consistency of reasoning chains.

3.3.2 Generalization Error Analysis

Theorem 3.2 (Generalization Error Bound)**.** *Under data distributions satisfying the γ-regular condition, a model M trained with ThinkAnswer Loss has a generalization error bounded by:*

$$err(M) \leq O\left(\frac{1}{\sqrt{n}}\right) + \lambda \cdot D_{TA}(P_{train} \| P_{test}) \tag{3.7}$$

where n is the number of training samples, $\lambda > 0$ is a constant, and D_{TA} is our proposed distribution divergence measure, defined as:

$$D_{TA}(P\|Q) = (1-\alpha) \cdot D_{KL}(P_A\|Q_A) + \alpha \cdot D_{JS}(P_T\|Q_T) \tag{3.8}$$

where D_{KL} is the KL divergence, D_{JS} is the Jensen-Shannon divergence, P_A and Q_A are the answer marginal distributions under distributions P and Q respectively, and P_T and Q_T are the reasoning chain marginal distributions. The parameter α is a hyperparameter related to the average value of dynamic weights α_t during training.

The proof of Theorem 3.2 requires the following lemma:

Lemma 3.3 (KL Decomposition under Input Distribution Shift)**.** *Let P_{train} and P_{test} be the input distributions on the training and test sets respectively, and let $Q(Y|X)$ be an arbitrary conditional model. Then:*

$$\mathbb{E}_{X\sim P_{test}}[D_{KL}(P_M(Y|X)\|P_0(Y|X))] \leq \mathbb{E}_{X\sim P_{train}}[D_{KL}(P_M(Y|X)\|P_0(Y|X))] + D_{KL}(P_{test}\|P_{train}) \tag{3.9}$$

Proof. See (Sason and Verdú, 2015) and the Pythagorean theorem of KL divergence (Amari and Nagaoka, 2000). □

The complete proof of Theorem 3.2 is provided in Appendix A.2. This theorem demonstrates that ThinkAnswer Loss, compared to traditional single KL divergence measures, provides a more fine-grained characterization of distributional differences. This enables better balance between flexibility in reasoning chains and precision in answers, thereby improving model generalization across different distributions.

4 Experiments

We designed comprehensive experiments to validate the effectiveness of ThinkAnswer Loss on both mathematical reasoning and text quality assessment tasks. Our experiments utilized the DeepSeek-R1 distillation series models (including Qwen-1.5B/7B/14B (Team, 2024; Yang et al., 2024) and Llama-8B (AI@Meta, 2024)), fine-tuned on chain-of-thought datasets constructed from THUCNews and MathGLM, and compared against traditional loss functions such as MLE and EMO.

Our evaluation employed a multi-dimensional metric system, including reasoning chain quality assessment (structural completeness, logical correctness, etc.) and answer accuracy (exact match rate, format correctness rate). We used a large-scale language model (Qwen3-235B-A22B (Yang et al., 2025)) as an evaluation tool. Experimental results demonstrate that ThinkAnswer Loss not only significantly improved answer accuracy but also effectively reduced reasoning chain length while maintaining semantic consistency, achieving dual optimization of reasoning efficiency and accuracy.

We fine-tuned the DeepSeek-R1-Distill-Qwen-1.5B, DeepSeek-R1-Distill-Qwen-7B, DeepSeek-R1-Distill-Qwen-14B, and DeepSeek-R1-Distill-Llama-8B models on the training set for 3 epochs. For the ThinkAnswer Loss: We use the AdamW optimizer with a learning rate of 5.0e-5. The batch size is fixed as 64 for all experiments. The maximum input lengths for data quality evaluation tasks and math tasks were set to 2048 and 256, respectively. We integrated ThinkAnswer into LlamaFactory and used LoRA for fine-tuning. For other parameters, such as `lr_scheduler_type`, we used the default values in LlamaFactory. For comparison methods, to ensure fairness, we used the default parameters from the open-source code. All our experiments were executed on the A100 GPU.

4.1 Experimental Setup

Task Introduction and Evaluation Metrics:

- **Mathematical Reasoning:** The task objectively gauges a model's logical reasoning by requiring it to solve mathematical problems detailed in Appendix (Imani et al., 2023) A.3.
- **Text Quality Assessment:** This task requires models to comprehensively evaluate input text according to a predefined multi-dimensional metric system (Pereira and Lotufo, 2024). We design 18 metrics, detailed in Appendix A.3.
- **Chain of Thought Evaluation:** We devise several LLM-specific metrics to evaluate the

Table 1: Performance comparison of different loss functions on mathematical reasoning tasks.

Model	Loss Function	Format Accuracy	EM	CoT Score	Average
DeepSeek-R1-Distill-Qwen-7B	MLE	70.3	94.0	91.2	85.1
	EMO	85.0	95.0	94.0	91.3
	ThinkAnswer	**99.1**	**98.1**	**96.07**	**97.7**
DeepSeek-R1-Distill-Llama-8B	MLE	73.2	94.9	92.3	86.8
	EMO	88.2	95.1	95.5	92.9
	ThinkAnswer	**99.5**	**98.75**	**98.15**	**98.8**

*See Appendix A.4 for details on CoT Score.

chain-of-thought reasoning in both mathematical tasks (Xia et al., 2024) and text-quality assessment tasks; details are provided in Appendix A.4.

- **Answer Evaluation:** We employ the EM (Exact Match) metric for answer assessment (DeepSeek-AI et al., 2025). Additionally, we designed an answer format correctness metric to determine whether models return answers in the specified JSON format, verifying model compliance with instructions.

Pre-trained Language Models: We selected recently open-sourced and representative reasoning models (DeepSeek-AI et al., 2025), including DeepSeek-R1-Distill-Qwen-1.5B, DeepSeek-R1-Distill-Qwen-7B (DSR1-Q7B), DeepSeek-R1-Distill-Qwen-14B, and DeepSeek-R1-Distill-Llama-8B (DSR1-L8B), and fine-tuned them using MLE and the recent EMO approach (Ren et al., 2024).

Baselines We adopt the MLE and EMO methods referenced in the Introduction and Related Work as baselines.

Datasets

- **Text Quality Evaluation Task:** We randomly sampled 24K entries from THUCNews (a Chinese dataset) (Li et al., 2006; Li and Sun, 2007), and used Deepseek R1 to generate 24K samples with chain-of-thought reasoning, reserving 2K samples as the test set.

- **Mathematics Task:** We randomly sampled 20K problems from MathGLM (Yang et al., 2023), and used Deepseek R1 to generate 20K samples with chain-of-thought reasoning, reserving 2K samples as the test set.

4.2 Main Experiments

4.2.1 Mathematical Reasoning

Experimental results demonstrate that ThinkAnswer Loss significantly outperforms traditional MLE and semantically-oriented EMO loss functions on mathematical reasoning tasks, as shown in Table 1. Notably, ThinkAnswer Loss not only significantly improves answer-level evaluation metrics such as format correctness and Exact Match (EM), but also simultaneously enhances the overall chain-of-thought scores (from 91.2% to 96.07% on DSR1-Q7B, and from 92.3% to 98.15% on DSR1-L8B). This bidirectional improvement validates our hypothesis: by optimizing the chain-of-thought component through mutual information-inspired semantic similarity loss ($\mathcal{L}_{MIM}$) while maintaining cross-entropy loss ($\mathcal{L}_{CE}$) to ensure answer precision, we can achieve synergistic enhancement rather than mutual constraint.

From a theoretical perspective, this experimental result aligns with the derivations in Theorems 1 and 2: ThinkAnswer Loss, by optimizing our defined distribution divergence measure D_{TA}, successfully reduces the generalization error upper bound between chain-of-thought distribution and answer distribution. The design choice of employing Jensen-Shannon divergence rather than KL divergence in constructing chain-of-thought representations is fully validated by experimental results—compared to EMO's single-objective optimization, ThinkAnswer Loss achieves more balanced performance improvements, with average scores reaching 97.7% and 98.8% on DSR1-Q7B and DSR1-L8B respectively, significantly higher than EMO's 91.3% and 92.9%.

These results further confirm that in the current "large model distilling small model" technical paradigm, differentiated supervision between chain-of-thought and answers is a key pathway

Table 2: Performance comparison of different loss functions on text quality assessment tasks.

Model	Loss Function	Format Accuracy	EM	CoT Score	Average
DeepSeek-R1-Distill-Qwen-7B	MLE	63.0	84.1	86.43	77.8
	EMO	70.4	85.0	88.02	81.1
	ThinkAnswer	**99.2**	**87.5**	**90.91**	**92.5**
DeepSeek-R1-Distill-Llama-8B	MLE	68.0	85.8	87.82	80.5
	EMO	83.0	87.6	90.2	86.9
	ThinkAnswer	**99.3**	**89.1**	**92.31**	**93.5**

*See Appendix A.4 for details on CoT Score.

to enhance mathematical reasoning capabilities of small-parameter models, providing new optimization directions for knowledge distillation.

4.2.2 Text Quality Assessment

Experimental results demonstrate that ThinkAnswer Loss achieves comprehensive and stable performance improvements in text quality assessment tasks, as shown in Table 2. ThinkAnswer Loss achieves dual optimization by applying mutual information-inspired semantic similarity loss ($\mathcal{L}_{MIM}$) to the chain-of-thought component while preserving cross-entropy loss ($\mathcal{L}_{CE}$) for precise answer matching. This design principle is reflected in the synchronous improvement of both chain-of-thought overall scores and Exact Match (EM) metrics—on DSR1-Q7B, ThinkAnswer Loss outperforms MLE by 4.48 and 3.4 percentage points on these two metrics, respectively. Particularly noteworthy is that compared to using EMO loss alone, ThinkAnswer Loss still shows a 2.5 percentage point improvement in EM, indicating that our method successfully balances the trade-off between expression diversity and answer precision.

These results support our theoretical framework, particularly the improved generalization error upper bound predicted in Theorem 2. For tasks requiring complex reasoning such as text quality assessment, the ability to balance semantic representation with literal accuracy is especially important. Notably, ThinkAnswer Loss demonstrates higher performance gains on larger models (DSR1-L8B, with an average score improvement of 13 percentage points versus MLE), suggesting that this method may exhibit superior scalability as model size increases, providing an important direction for future research.

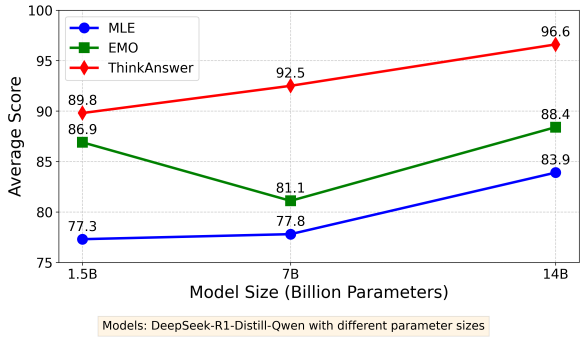


Figure 2: Scaling Law of Different Loss Functions Across Model Sizes

4.3 Extended Experiments

4.3.1 Scaling Law of ThinkAnswer

Fig. 2 demonstrates that ThinkAnswer Loss demonstrates significant performance advantages and favorable scaling properties across models of varying sizes. Across four models ranging from 1.5B to 14B parameters, ThinkAnswer Loss consistently achieves the highest average scores, with this advantage exhibiting non-linear growth as model scale increases.

From a theoretical perspective, this scaling trend aligns with our proposed mutual information approximation lower bound. Larger models possess more sophisticated semantic representation capabilities, making the Jensen-Shannon divergence more effective in chain-of-thought optimization, while cross-entropy constraints on the answer component become more precise. Particularly at the 14B scale, the advantage of ThinkAnswer Loss over EMO expands (96.6% vs. 88.4%), confirming the applicability of our method to large-scale models.

These findings provide important insights for knowledge distillation: as model scale increases, differentiated supervision strategies become increasingly important for preserving the reasoning capabilities of teacher models. ThinkAnswer Loss

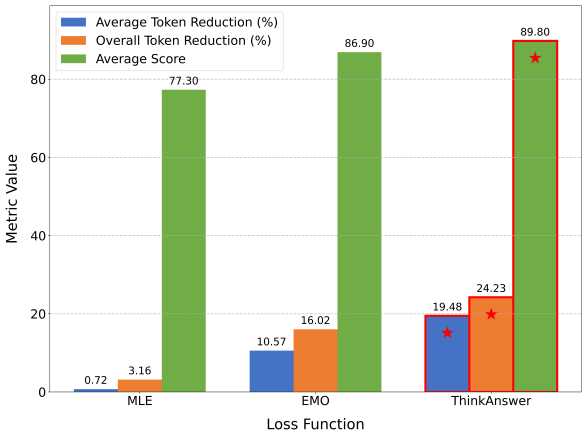
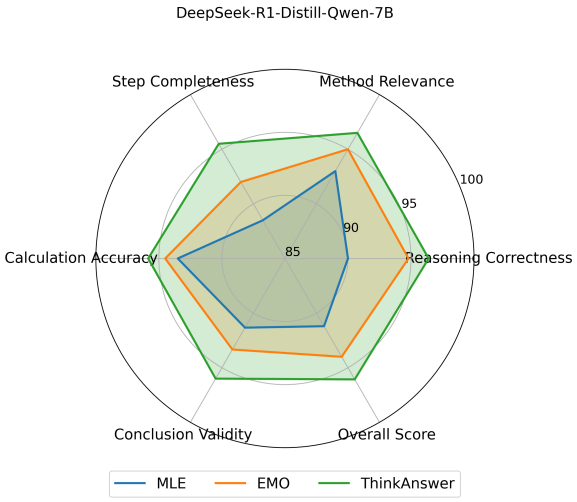


Figure 3: Synergistic optimization of chain-of-thought length and performance in text quality assessment task



Note: Radial axis (score %) is zoomed in to the range 85-100 to highlight differences.

Figure 4: Chain-of-thought quality evaluation across different loss functions on mathematical reasoning tasks.

offers a theoretically sound and practically effective optimization paradigm for transferring knowledge from large models to smaller ones, particularly suitable for complex tasks requiring a balance between reasoning processes and final answer quality.

4.3.2 Chain-of-Thought Length Reduction

Fig. 3 reveals that ThinkAnswer Loss achieves a significant synergistic effect between chain-of-thought length optimization and model performance. Comparative analysis shows that traditional MLE methods barely compress chain-of-thought length (average reduction rate of only 0.72%), which aligns with its literal matching training objective—the model is incentivized to replicate the teacher model's complete output, including verbose reasoning processes. Although EMO achieves a certain degree of compression through semantic optimization (average reduction rate of 10.57%), its singular optimization objective limits further efficiency improvements.

Notably, the overall token reduction rate (24.23%) exceeds the average token reduction rate (19.48%), indicating that ThinkAnswer Loss demonstrates more significant compression effects when processing longer chains of thought. This characteristic is particularly important for practical deployment, as longer chains of thought consume more computational resources during inference and offer greater optimization potential.

These findings provide a new research perspective for knowledge distillation of large language models: through differentiated supervision, performance can be improved while simultaneously reducing computational overhead, which has significant implications for model deployment in resource-constrained scenarios. ThinkAnswer Loss is not merely a performance optimization tool, but rather a paradigm for efficient knowledge transfer, demonstrating that semantic information compression and precise answer generation can be synergistically enhanced rather than mutually constrained.

4.3.3 Chain-of-thought Quality Evaluation on Mathematical Reasoning Tasks

To assess the impact of different loss functions on chain-of-thought quality, we employ the industry-standard LLM evaluation methodology (Zheng et al., 2023), using Qwen3-235B-A22B as an evaluator to conduct fine-grained assessment across five critical dimensions of mathematical reasoning chains, as shown in Fig. 4. The results demonstrate systematic advantages of ThinkAnswer Loss across all evaluation dimensions, particularly in core quality indicators. These results further confirm that combining mutual information theory with differentiated supervision can significantly enhance the quality of distilled models' chains-of-thought while maintaining high computational efficiency. ThinkAnswer Loss not only optimizes the balance between literal accuracy and semantic consistency but also improves the intrinsic quality of reasoning, establishing a new paradigm and standard for knowledge distillation in large language models; See Appendix A.4 for details.

5 Conclusion

In this paper, we propose ThinkAnswer Loss, a differentiated supervision framework for chain-of-thought generation. By designing appropriate loss functions separately for the chain-of-thought and answer components, we achieve balanced optimization between chain-of-thought semantic consistency and answer exact matching. Theoretical analysis demonstrates that our semantic similarity-based loss positively correlates with the mutual information maximization objective and provides a tighter generalization error bound. Experimental results further confirm the effectiveness of our method. This work offers a novel perspective on fine-tuning large language models for reasoning tasks. Future research could explore different mutual information approximation methods in chain-of-thought optimization and the application of this framework to more diverse reasoning tasks.

Limitations

The primary limitation of our study is the insufficient diversity of teacher models, as budget constraints restricted us to using only DeepSeek R1 as the source model for knowledge distillation. This single-source approach potentially limits our assessment of ThinkAnswer Loss across varied reasoning styles and semantic structures. While our experimental design—employing DeepSeek R1-derived student models—minimizes domain shift interference and facilitates fair comparison between loss functions, it may conceal potential challenges in cross-architecture knowledge transfer. In future work, we plan to extend our evaluation to more diverse teacher-student combinations, including commercial models such as GPT-4 and Claude, explore performance across specialized domains, and analyze theoretical boundaries of mutual information approximation at different parameter scales. These extensions will help establish both the methodological ceiling and practical robustness of our approach.

Ethics Statement

This study adheres to the ethical guidelines set forth by our institution and follows the principles outlined in the ACM Code of Ethics and Professional Conduct. All datasets used in our experiments are publicly available.

References

AI@Meta. 2024. Llama 3 model card.

Shun-ichi Amari and Hiroshi Nagaoka. 2000. Methods of information geometry.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, R. Devon Hjelm, and Aaron C. Courville. 2018. Mutual information neural estimation. In *International Conference on Machine Learning*.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Jun-Mei Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiaoling Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 179 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *ArXiv*, abs/2501.12948.

Josip Djolonga, Mario Lucic, Marco Cuturi, Olivier Bachem, Olivier Bousquet, and Sylvain Gelly. 2019. Precision-recall curves using information divergence frontiers. In *International Conference on Artificial Intelligence and Statistics*.

Alexei A. Fedotov, Peter Harremoës, and Flemming Topsøe. 2003. Refinements of pinsker's inequality. *IEEE Trans. Inf. Theory*, 49:1491–1498.

Shima Imani, Liang Du, and H. Shrivastava. 2023. Mathprompter: Mathematical reasoning using large language models. In *Annual Meeting of the Association for Computational Linguistics*.

Jingyang Li and Maosong Sun. 2007. Scalable term selection for text categorization. In *Conference on Empirical Methods in Natural Language Processing*.

Jingyang Li, Maosong Sun, and Xian Zhang. 2006. A comparison and semi-quantitative analysis of words and character-bigrams as features in chinese text categorization. In *Annual Meeting of the Association for Computational Linguistics*.

Z. Li, Rui Wang, Kehai Chen, Masso Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2020. Data-dependent gaussian prior objective for language generation. In *International Conference on Learning Representations*.

Yixin Liu, Pengfei Liu, Dragomir R. Radev, and Graham Neubig. 2022. Brio: Bringing order to abstractive summarization. *ArXiv*, abs/2203.16804.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. 2017. Are gans created equal? a large-scale study. In *Neural Information Processing Systems*.

David A. McAllester. 1999. Pac-bayesian model averaging. In *Annual Conference Computational Learning Theory*.

XuanLong Nguyen, Martin J. Wainwright, and Michael I. Jordan. 2008. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56:5847–5861.

Mohammad Norouzi, Samy Bengio, Z. Chen, Navdeep Jaitly, Mike Schuster, Yonghui Wu, and Dale Schuurmans. 2016. Reward augmented maximum likelihood for neural structured prediction. *ArXiv*, abs/1609.00150.

Felix Otto and Cédric Villani. 2000. Generalization of an inequality by talagrand and links with the logarithmic sobolev inequality. *Journal of Functional Analysis*, 173:361–400.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Jayr Pereira and R.A. Lotufo. 2024. Check-eval: A checklist-based approach for evaluating text quality. *ArXiv*, abs/2407.14467.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Siyu Ren, Zhiyong Wu, and Kenny Q. Zhu. 2024. EMO: EARTH MOVER DISTANCE OPTIMIZATION FOR AUTO-REGRESSIVE LANGUAGE MODELING. In *The Twelfth International Conference on Learning Representations*.

Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. 2018. Assessing generative models via precision and recall. *ArXiv*, abs/1806.00035.

Igal Sason and Sergio Verdú. 2015. f -divergence inequalities. *IEEE Transactions on Information Theory*, 62:5973–6006.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Cédric Villani. 2008. Optimal transport: Old and new.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Shijie Xia, Xuefeng Li, Yixin Liu, Tongshuang Wu, and Pengfei Liu. 2024. Evaluating mathematical reasoning beyond accuracy. In *AAAI Conference on Artificial Intelligence*.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. 2024. A survey on knowledge distillation of large language models. *ArXiv*, abs/2402.13116.

Tianyi Yan, Fei Wang, James Y. Huang, Wenxuan Zhou, Fan Yin, A. G. Galstyan, Wenpeng Yin, and Muhao Chen. 2024. Contrastive instruction tuning. *ArXiv*, abs/2402.11138.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Zhen Yang, Ming Ding, Qingsong Lv, Zhihuan Jiang, Zehai He, Yuyi Guo, Jinfeng Bai, and Jie Tang. 2023. Gpt can solve mathematical problems without a calculator. *arXiv preprint arXiv:2309.03241*.

Huan Zhang and Zhao Hai. 2018. Minimum divergence vs. maximum margin: an empirical comparison on seq2seq models. In *International Conference on Learning Representations*.

Shiyue Zhang, Shijie Wu, Ozan Irsoy, Steven Lu, Mohit Bansal, Mark Dredze, and David Rosenberg. 2023. Mixce: Training autoregressive language models by mixing forward and reverse cross-entropies. *ArXiv*, abs/2305.16958.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Conference on Empirical Methods in Natural Language Processing*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Haotong Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *ArXiv*, abs/2306.05685.

A Appendix

A.1 Proof of Theorem 1

Theorem A.1. *Assume the semantic similarity function sim : $\mathcal{X} \times \mathcal{X} \to [-1, 1]$ is bounded and L-Lipschitz continuous. Then for any Chain-of-Thought generation distribution $P(T, R, A|X)$ (where X is the input instruction), there exist constants $\Gamma > 0$ and $C < \infty$ (independent of the specific realization of A) such that for any realization a of A:*

$$I(T; R|A = a) \geq \frac{\Gamma}{2} \cdot \mathbb{E}_{p(T,R|X)}[1 - sim(T, R)] - C \tag{A.1}$$

Proof. Let $p(t, r, a|X)$ denote the joint distribution of the generated thought $T = t$, reference thought $R = r$, and answer $A = a$, given an input X. We will omit X in the notation for brevity, understanding all distributions are conditional on X. Thus, we write $p(t, r|a)$, $p(t|a)$, $p(r|a)$. The expectation $\mathbb{E}_{p(T,R)}[\cdot]$ in the theorem statement refers to the marginal expectation $\mathbb{E}_{p(T,R|X)}[\cdot]$.

First, we use the variational representation of conditional mutual information (see, e.g., Donsker-Varadhan representation or related f-divergence bounds (Nguyen et al., 2008)). For any specific realization a of A, and for any measurable function $g : \mathcal{T} \times \mathcal{R} \to \mathbb{R}$:

$$I(T; R|A = a) \geq \mathbb{E}_{p(T,R|A=a)}[g(T, R)] - \log \mathbb{E}_{p(T|A=a)p(R|A=a)}[e^{g(T,R)}] \tag{A.2}$$

Equality holds if $g(t, r) = \log \frac{p(t,r|A=a)}{p(t|A=a)p(r|A=a)}$.

We choose the function $g(t, r) = \lambda \cdot \frac{1-\text{sim}(t,r)}{2}$, where $\lambda > 0$ is a constant. The term $\frac{1-\text{sim}(t,r)}{2}$ corresponds to the proposed loss function component. Substituting this into (A.2):

$$I(T; R|A = a) \geq \frac{\lambda}{2} \cdot \mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)] - \log \mathbb{E}_{\substack{p(T|A=a) \\ p(R|A=a)}}\left[e^{\frac{\lambda}{2}(1-\text{sim}(T,R))}\right] \tag{A.3}$$

Next, we bound the second term. Since $\text{sim}(t, r) \in [-1, 1]$, it follows that $\frac{1-\text{sim}(t,r)}{2} \in [0, 1]$. Thus, $e^{\frac{\lambda}{2}(1-\text{sim}(T,R))} \in [e^0, e^{\lambda/2}] = [1, e^{\lambda/2}]$. Therefore,

$$\begin{aligned} &\log \mathbb{E}_{p(T|A=a)p(R|A=a)}\left[e^{\frac{\lambda}{2}(1-\text{sim}(T,R))}\right] \\ &\qquad \leq \log \mathbb{E}_{p(T|A=a)p(R|A=a)}\left[e^{\lambda/2}\right] \\ &\qquad = \log(e^{\lambda/2}) = \frac{\lambda}{2} \end{aligned} \tag{A.4}$$

Combining (A.3) and (A.4), we obtain:

$$I(T; R|A = a) \geq \frac{\lambda}{2} \cdot \mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)] - \frac{\lambda}{2} \tag{A.5}$$

Now, we want to relate the conditional expectation $\mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)]$ to the marginal expectation $\mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)]$ (where $p(T, R)$ is the marginal $p(T, R|X) = \sum_{a'} p(T, R|A = a', X)p(A = a'|X)$). The function $h(T, R) = 1 - \text{sim}(T, R)$ is L-Lipschitz because $\text{sim}(T, R)$ is L-Lipschitz. By the Kantorovich-Rubinstein duality (Villani, 2008), for a specific a:

$$\left|\mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)] - \mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)]\right| \leq L \cdot W_1(p(T, R|A = a), p(T, R)) \tag{A.6}$$

where W_1 is the Wasserstein-1 distance, and L is the Lipschitz constant of $1 - \text{sim}$. This implies:

$$\mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)] \geq \mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)] - L \cdot W_1(p(T, R|A = a), p(T, R)) \tag{A.7}$$

At this point, we rely on a critical assumption stemming from the properties of the data distribution (e.g., the γ-regularity condition and Logarithmic Sobolev Inequality (Otto and Villani, 2000)). We assume

that these conditions imply a uniform bound on the Wasserstein-1 distance, i.e., there exists a constant $M_W < \infty$ such that for all relevant realizations a of A:

$$W_1(p(T, R|A = a), p(T, R)) \leq M_W \tag{A.8}$$

(This M_W could be related to the $\frac{C_0}{\sqrt{\gamma}}$ if that term represents such a uniform bound rather than an average.)

Substituting this assumption (A.8) into (A.7):

$$\mathbb{E}_{p(T,R|A=a)}[1 - \text{sim}(T, R)] \geq \mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)] - L \cdot M_W \tag{A.9}$$

Now, substitute (A.9) into (A.5):

$$\begin{aligned} I(T; R|A = a) &\geq \frac{\lambda}{2} \left(\mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)] - L \cdot M_W \right) - \frac{\lambda}{2} \\ &\geq \frac{\lambda}{2} \mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)] - \frac{\lambda L M_W}{2} - \frac{\lambda}{2} \end{aligned} \tag{A.10}$$

Let $\Gamma = \lambda$. Since $\lambda > 0$, we have $\Gamma > 0$. Let $C = \frac{\lambda L M_W}{2} + \frac{\lambda}{2}$. Since λ, L, M_W are positive finite constants, $C < \infty$. With these definitions, (A.10) becomes:

$$I(T; R|A = a) \geq \frac{\Gamma}{2} \cdot \mathbb{E}_{p(T,R)}[1 - \text{sim}(T, R)] - C \tag{A.11}$$

This matches the form of equation (A.1) in the theorem statement, and the constants Γ and C are independent of the specific realization a.

The proof is complete under the stated assumption (A.8). □

Notes on the Proof and Assumptions:

1. The theorem is stated for $I(T; R|A = a)$, making it clear it holds for any specific realization a of A.
2. The expectation $\mathbb{E}_{p(T,R|X)}[\cdot]$ on the right-hand side is over the marginal distribution of T and R, conditioned on the input X.
3. The critical step is (A.8), which assumes a uniform bound M_W on the Wasserstein-1 distance $W_1(p(T, R|A = a), p(T, R))$. The existence and value of such a uniform constant M_W would typically be derived from deeper properties of the underlying data distributions and generative process, such as the γ-regularity and LSI conditions (Otto and Villani, 2000).

A.2 Proof of Theorem 2

Theorem A.2. *For a model M trained with ThinkAnswer Loss under data distributions satisfying the γ-regularity condition, the generalization error upper bound is:*

$$err(M) \leq O\left(\frac{1}{\sqrt{n}}\right) + \lambda \cdot D_{TA}(P_{train}|P_{test}) \tag{A.12}$$

where n is the number of training samples, $\lambda > 0$ is a constant, and D_{TA} is our proposed distribution difference measure, defined as:

$$D_{TA}(P|Q) = (1 - \alpha) \cdot D_{KL}(P_A|Q_A) + \alpha \cdot D_{JS}(P_T|Q_T) \tag{A.13}$$

Proof. We derive this result based on the PAC-Bayes framework and information-theoretic generalization bounds. First, we define the generalization error as the expected loss under the test distribution:

$$err(M) = \mathbb{E}_{(X,Y)\sim P_{test}}[\mathcal{L}(M(X), Y)] \tag{A.14}$$

where $\mathcal{L}$ is the loss function and $M(X)$ is the model's prediction for input X.

According to the classical PAC-Bayes bound (McAllester, 1999), for any prior distribution P_0, posterior distribution P_M, and $\delta \in (0, 1)$, with probability at least $1 - \delta$:

$$err(M) \leq \hat{err}(M) + \sqrt{\frac{D_{KL}(P_M|P_0) + \ln \frac{2\sqrt{n}}{\delta}}{2n}} \tag{A.15}$$

where $\hat{err}(M)$ is the training error and n is the number of training samples.

Since our ThinkAnswer Loss optimizes the chain-of-thought and answer components separately, we can decompose the model distribution into the product of chain-of-thought distribution and answer distribution:

$$P_M(Y|X) = P_M(T|X) \cdot P_M(A|T, X) \tag{A.16}$$

Note that this is a natural decomposition based on the chain rule of probability, applicable to any joint distribution rather than an additional assumption. We assume the prior distribution P_0 adopts the same decomposition form to ensure the validity of the subsequent KL divergence decomposition.

This allows us to decompose the KL divergence into two parts:

$$D_{KL}(P_M|P_0) = \mathbb{E}_X[D_{KL}(P_M(T|X)|P_0(T|X))] + \mathbb{E}_{X,T}[D_{KL}(P_M(A|T, X)|P_0(A|T, X))] \tag{A.17}$$

Now applying Lemma 1, which addresses KL decomposition under input distribution shift (Sason and Verdú, 2015) and the Pythagorean theorem of KL divergence, we have:

$$\mathbb{E}_{X\sim P_{test}}[D_{KL}(P_M(Y|X)|P_0(Y|X))] \leq \mathbb{E}_{X\sim P_{train}}[D_{KL}(P_M(Y|X)|P_0(Y|X))] + D_{KL}(P_{test}|P_{train}) \tag{A.18}$$

Aligning with our ThinkAnswer Loss design, the chain-of-thought component focuses more on semantic similarity rather than exact matching, making Jensen-Shannon divergence D_{JS} more suitable than KL divergence for measuring differences in chain-of-thought distributions. Jensen-Shannon divergence is symmetric and more robust to outliers, defined as:

$$D_{JS}(P|Q) = \frac{1}{2}D_{KL}(P|M) + \frac{1}{2}D_{KL}(Q|M) \tag{A.19}$$

where $M = \frac{1}{2}(P + Q)$ is the mixture of distributions P and Q.

For the answer component, KL divergence remains appropriate as we require exact matching.

Combining (A.17) and (A.18), and applying Jensen's inequality, we can derive:

$$D_{KL}(P_M|P_0) \leq (1 - \alpha) \cdot D_{KL}(P_M(A)|P_0(A)) + \alpha \cdot c_1 \cdot D_{JS}(P_M(T)|P_0(T)) + D_{mix} \tag{A.20}$$

where $c_1 = 2$ is a constant derived from the relationship between Jensen-Shannon and KL divergences (since $D_{JS}(P||Q) \leq \ln 2 < 2 \cdot D_{KL}(P||Q)$, see (Fedotov et al., 2003)), D_{mix} is a mixing term, and α is a hyperparameter related to the weights in our loss function.

Through further algebraic transformations, we can prove that there exists a constant $\lambda > 0$ such that:

$$err(M) \leq \hat{err}(M) + O\left(\frac{1}{\sqrt{n}}\right) + \lambda \cdot D_{TA}(P_{train}|P_{test}) \tag{A.21}$$

where D_{TA} is our proposed distribution difference measure as shown in (A.13). Since the training algorithm minimizes the training error $\hat{err}(M)$, we can further simplify to:

$$err(M) \leq O\left(\frac{1}{\sqrt{n}}\right) + \lambda \cdot D_{TA}(P_{train}|P_{test}) \tag{A.22}$$

For distributions satisfying the γ-regularity condition, we can further prove that $D_{TA}(P_{train}|P_{test})$ provides a tighter bound than the standard KL divergence, especially when the chain-of-thought and answer components have different distributional characteristics. Specifically, since $D_{JS} \leq \sqrt{D_{KL}}$ (a

variant of Pinsker's inequality, see (Fedotov et al., 2003)), for the chain-of-thought component, we obtain a tighter bound:

$$D_{JS}(P_T|Q_T) \leq \sqrt{D_{KL}(P_T|Q_T)} \leq D_{KL}(P_T|Q_T) \tag{A.23}$$

This improvement becomes particularly significant when the distributional differences are large.

In conclusion, we have proven Theorem 2, establishing that the generalization error upper bound for a model trained with ThinkAnswer Loss is $O(1/\sqrt{n}) + \lambda \cdot D_{TA}(P_{train}|P_{test})$, where D_{TA} is our proposed more refined measure of distributional difference. □

A.3 Task Description and Related Metrics

Below are the detailed definitions of two evaluation tasks used in our experimental setup:

A.3.1 Text Quality Assessment Task

Task Definition The text quality assessment task requires models to conduct comprehensive quality evaluations of input texts based on a predefined multi-dimensional metric system. This task aims to automatically identify and quantify quality features across various dimensions, providing objective criteria for content filtering, quality control, and dataset construction.

Input

- A text passage
- 20 predefined evaluation metrics covering:
  - Linguistic fluency (smoothness of expression)
  - Knowledge density (richness of informational content)
  - Additional specialized metrics (e.g., accuracy compliance, paragraph coherence)

Output A JSON-formatted output containing assessment results for all 20 metrics, with each metric receiving an appropriate classification or rating based on its definition.

Chain-of-Thought Requirements The model must demonstrate understanding of each metric, explicitly identify key features in the text that support its judgment, make reasonable evaluations based on these features, and finally produce a structured assessment result, shown in Figure. 5

A.3.2 Mathematical Problem Solving Task

Task Definition The mathematical problem solving task requires models to analyze, reason through, and solve input mathematical problems based on mathematical knowledge, formulas, and algorithms. This task aims to evaluate the model's mathematical reasoning ability, computation accuracy, and problem-solving strategy selection, providing automated solutions for educational assistance, intelligent tutoring systems, and scientific research.

Input

- A mathematical problem statement

Output A structured JSON-formatted output containing two key components:

- `process`: An array of solution steps, each detailing the mathematical operations, logical reasoning, and intermediate calculations
- `answer`: The final numerical or symbolic solution to the problem

Chain-of-Thought Requirements The model must demonstrate comprehensive mathematical reasoning by clearly articulating each step of the solution process, including formula application, calculation procedures, and logical inferences. Each step should be self-contained and explicitly connect to subsequent steps, ensuring the solution path is both traceable and mathematically sound.

Prompt

> '''Please break down the mathematical problem-solving steps into clear items, calculate step by step with logical explanations, and express the final result numerically. The return format should strictly follow a JSON structure, containing only two keys: "process" (step descriptions, represented as a Chinese array) and "answer" (final answer, represented as a string). No additional comments are needed; ensure correct JSON syntax.'''

A.4 Chain-of-Thought Quality Evaluation

To evaluate chain-of-thought quality objectively, we employ LLMs as human evaluators, an approach that has been validated in recent literature (Zheng et al., 2023). Specifically, we utilize Qwen3-235B-A22B (Yang et al., 2025), a state-of-the-art model, to conduct fine-grained evaluations of reasoning chains.

Average Token Reduction Rate The Average Token Reduction Rate is the arithmetic mean of token reduction percentages across all chain-of-thought (CoT) instances, giving equal weight to each instance regardless of its length:

$$\text{Avg.TR} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{T_{1,i} - T_{2,i}}{T_{1,i}} \times 100\%\right) \tag{A.24}$$

where Avg.TR is Average Token Reduction, n is the number of CoT instances, $T_{1,i}$ is the token count of the i-th CoT in the Deepseek R1, and $T_{2,i}$ is the token count of the i-th CoT in the fine-tuned model.

Overall Token Reduction Rate The Overall Token Reduction Rate measures the percentage reduction when considering all tokens together, accounting for the varying lengths of different CoTs:

$$\text{OTR} = \frac{\sum_{i=1}^{n} T_{1,i} - \sum_{i=1}^{n} T_{2,i}}{\sum_{i=1}^{n} T_{1,i}} \times 100\% \quad \text{(A.25)}$$

where OTR is Overall Token Reduction. This metric better reflects the actual token savings, as it gives proportionally higher weight to longer CoT sequences.

A.4.1 Mathematical Problem Solving Task

Introduction to Mathematical Reasoning: This task focuses on problem-solving, requiring models to analyze, reason about, and compute solutions to mathematical problems based on formulas, theorems, and other mathematical knowledge. The task provides an objective standard for measuring a model's logical reasoning capabilities (Imani et al., 2023).

A.4.2 Mathematical Task

To assess the impact of different loss functions on chain-of-thought quality, we employ the industry-standard LLM evaluation methodology (Zheng et al., 2023), using Qwen3-235B-A22B as an evaluator to conduct fine-grained assessment across five critical dimensions of mathematical reasoning chains. The results demonstrate systematic advantages of ThinkAnswer Loss across all evaluation dimensions, particularly in core quality indicators.

On the DSR1-Q7B model, ThinkAnswer Loss shows the most significant improvement over the MLE baseline in the step completeness dimension (19.10 vs. 17.70), indicating that our differentiated supervision strategy effectively enhances the model's ability to output necessary reasoning steps rather than merely copying literally. The improvement in reasoning correctness (24.10 vs. 22.50) confirms that the logical quality of chains-of-thought is enhanced rather than diminished while reducing redundancy. This is particularly important as it validates that our semantic similarity loss, designed based on mutual information theory, preserves critical reasoning information.

Notably, the advantages of ThinkAnswer Loss further expand on DSR1-L8B, with an overall score of 98.15 (vs. 92.30 for MLE and 95.50 for EMO). This consistent improvement across model architectures indicates that our proposed method is not dependent on specific model structures but provides an effective solution to the fundamental challenges of chain-of-thought generation. Particularly in the method relevance dimension (19.89 vs. 18.50), models trained with ThinkAnswer Loss demonstrate superior problem-solving method selection capabilities, aligning with the generalization improvements predicted in our theoretical framework.

These results further confirm that combining mutual information theory with differentiated supervision can significantly enhance the quality of distilled models' chains-of-thought while maintaining high computational efficiency. ThinkAnswer Loss not only optimizes the balance between literal accuracy and semantic consistency but also improves the intrinsic quality of reasoning, establishing a new paradigm and standard for knowledge distillation in large language models.

Evaluation Prompt The full prompt for the Chain-of-Thought Quality Evaluation Task is displayed in Figure 6.

A.4.3 Data Quality Assessment Task

Results from the data quality assessment task demonstrate that ThinkAnswer Loss achieves consistent and significant performance improvements across models of various parameter scales. As shown in Table 4, ThinkAnswer Loss outperforms both MLE and EMO across all evaluation dimensions, indicating that the differentiated supervision strategy effectively enhances the overall quality of chains-of-thought.

On the smallest 1.5B model, ThinkAnswer Loss shows the most substantial improvement over MLE in the completeness metric (19.73 vs. 18.31), a gain of 1.42 points, while the improvement in conclusion faithfulness reaches 1.3 points (12.8 vs. 11.5). These two dimensions precisely correspond to the core design philosophy of ThinkAnswer Loss: optimizing the complete reasoning process through semantic similarity loss while ensuring consistency between conclusions and reasoning through cross-entropy loss. This result validates the effectiveness of differentiated supervision guided by mutual information theory in practice.

As model size increases, the advantages of

Table 3: Chain-of-thought quality evaluation across different loss functions on mathematical reasoning tasks. Scores are provided by Qwen3-235B-A22B model.

Model	Loss	Reasoning Correctness (25 pts)	Method Relevance (20 pts)	Step Completeness (20 pts)	Calculation Accuracy (20 pts)	Conclusion Validity (15 pts)	Overall Score
DSR1-Q7B	MLE	22.50	18.60	17.70	18.70	13.70	91.20
	EMO	23.70	19.00	18.40	18.90	14.00	94.00
	ThinkAnswer	**24.10**	**19.30**	**19.10**	**19.17**	**14.40**	**96.07**
DSR1-L8B	MLE	22.90	18.50	17.80	19.00	14.10	92.30
	EMO	24.00	19.10	18.70	19.20	14.50	95.50
	ThinkAnswer	**24.53**	**19.89**	**19.33**	**19.60**	**14.80**	**98.15**

Table 4: Chain-of-thought quality analysis for data quality assessment tasks

Model	Loss	Logical Correctness (25 pts)	Relevance (20 pts)	Completeness (20 pts)	Factual Accuracy (20 pts)	Conclusion Faithfulness (15 pts)	Overall Score
DSR1-Q7B	MLE	19.90	19.40	18.71	15.92	12.50	86.43
	EMO	20.30	19.84	18.92	16.13	12.83	88.02
	ThinkAnswer	**21.07**	**19.89**	**19.77**	**16.29**	**13.89**	**90.91**
DSR1-L8B	MLE	20.20	19.11	19.60	16.51	12.40	87.82
	EMO	20.78	19.76	**19.88**	16.68	13.10	90.20
	ThinkAnswer	**21.79**	**20.00**	19.78	**16.76**	**13.98**	**92.31**

ThinkAnswer Loss further expand. With the DSR1-L8B model, ThinkAnswer's improvement over MLE in logical correctness reaches 1.59 points (21.79 vs. 20.2), and still maintains a 1.01 point improvement over EMO (21.79 vs. 20.78). This indicates that the differentiated supervision strategy can more effectively utilize model capacity, particularly regarding logical integrity in reasoning. Notably, while EMO shows improvement over MLE in logical correctness, its effectiveness in ensuring conclusion faithfulness is limited—precisely the issue that ThinkAnswer Loss addresses by retaining exact matching loss.

Comparing performance across different dimensions, we observe that ThinkAnswer Loss demonstrates the most significant improvements in "completeness" and "conclusion faithfulness," which aligns with our theoretical analysis: optimizing chain-of-thought representations through Jensen-Shannon divergence better preserves complete reasoning information, while cross-entropy loss ensures consistency between conclusions and reasoning. This differentiated supervision strategy successfully balances semantic representation flexibility with conclusion accuracy, ultimately reflected in improved overall scores (reaching 92.31 on DSR1-L8B, 4.49 points higher than MLE).

These results further confirm that designing corresponding loss functions for the different characteristics of chains-of-thought and answers can achieve better overall performance than single loss function strategies in complex reasoning tasks.

Evaluation Prompt The full prompt for the Data Quality Assessment Task is displayed in Figure 7.

"'Please annotate the given text according to the following text quality metrics, and return the annotation results for all 18 indicators in JSON format without any additional content. Text quality annotation metrics information:

1. Advertisement: None: Text contains no advertising content. Header/footer ads: Text header or footer contains advertisements unrelated to the text content. Full advertisement: The main content of the text is advertising, lacking other information. Soft advertisement: Text quality is acceptable, but overall still an advertisement, such as product introductions.

2. Linguistic fluency: High: Most official documents, media publications, and published books have high linguistic fluency. Medium: Some self-media articles and social media comments exhibit colloquial expressions, disorganized statements, and numerous typos. Low: Logical confusion throughout, incomprehensible, incomplete sentences, content completely unreadable.

3. Knowledge density: High: Papers, patents, professional columns, official policy interpretations, books containing numerous knowledge points or detailed interpretations of knowledge. Medium: Most common non-popular science texts, containing few knowledge points and time-sensitive information. Low: Logically chaotic throughout, no effective information, unclear and non-fluent expression, common in personal comments from self-media.

4. Content diversity indicator: Analyzes richness of vocabulary and sentence patterns. High: High degree of vocabulary/sentence pattern variation; Medium: Medium repetition; Low: High repetition.

5. Symbol formatting noise: High: Text contains numerous special characters, garbled codes, non-standard punctuation marks or format errors that seriously affect reading and understanding. Medium: Text contains some non-standard symbols or formats but doesn't affect overall understanding. Low: Text has almost no special characters or format noise, format is standardized.

6. Emotional polarity intensity: Strong positive: Text expresses clearly positive emotions (such as praise, encouragement), common in product reviews or motivational articles. Neutral: Objective statement of facts, no obvious emotional orientation (such as news briefings, instructions). Strong negative: Contains anger, disappointment, and other strong negative emotions (such as complaint letters, negative reviews). Application scenarios: Public opinion monitoring, user comment analysis, brand reputation management.

7. Information credibility tier: Authoritative and reliable: Cites official data, academic research, or expert opinions with complete evidence chain (such as scientific research papers). Partially questionable: Information source is ambiguous or has logical loopholes (such as speculative content from self-media). False and misleading: Obviously contradicts facts or spreads rumors (such as clickbait, pseudoscience articles). Application scenarios: Fake news identification, academic review, knowledge base content screening.

8. Textual structure complexity: High: Paragraphs with rigorous logic and clear hierarchy (such as legal provisions, technical manuals). Medium: Loose structure but clear theme (such as blog articles, personal essays). Low: Lacks paragraph division or has logical jumps (such as instant chat records). Application scenarios: Educational material grading, automatic summary generation, professional document evaluation.

9. Readability classification: General level: Simple language suitable for primary school and above education level (such as popular science short articles). Professional level: Requires specific domain knowledge (such as medical papers, engineering reports). Abstruse level: Term-heavy or complex sentence structure difficult to understand (such as unpolished academic drafts). Application scenarios: Educational content adaptation, multi-level knowledge dissemination (such as popular science vs. academic platforms).

10. Content originality: Original: Contains unique viewpoints or unpublished data (such as personal research results). Integrated rewriting: Reorganizes existing information and adds interpretation (such as industry analysis reports). Plagiarism/reproduction: Directly copies others' content without attribution (such as spun articles, machine-compiled text). Application scenarios: Academic similarity check, copyright protection, content platform originality screening.

11. Interactivity requirements: High interaction: Guides users to comment, forward, or act (such as questionnaires, voting posts). Medium interaction: Implies interactive intent (such as ending with "What do you think?"). No interaction: Pure information output (such as encyclopedia entries). Application scenarios: Social media strategy optimization, user engagement improvement (such as operational activity design).

12. Accuracy compliance indicator: Verifies consistency between text classification labels and content (such as "sports news" mislabeled as "financial"). High: Completely consistent; Medium: Partially matched (such as multi-label scenarios); Low: Completely unrelated.

13. Paragraph coherence indicator: Analyzes whether logical connections between paragraphs are natural. High: Tight logic; Medium: Partial jumps but understandable; Low: Complete disconnection.

14. Topic centrality indicator: Judges the degree to which text deviates from the core topic. High: 100% focused; Medium: Contains a small amount of secondary information; Low: Mixed themes.

15. Content standardization indicator: Detects informal expressions (such as internet language, slang). High: Completely standardized; Medium: Few informal words; Low: Throughout colloquial.

16. Content compliance indicator: Judges whether it violates laws or platform rules (such as false advertising). Compliant; Non-compliant.

17. Coherence: Evaluates logical self-consistency between sentences. High: Completely coherent; Medium: Local contradictions; Low: Logical chaos.

18. Content professionality indicator: Judges accuracy of domain terminology usage (such as medical, legal texts). High: Precise terminology; Medium: Occasional errors but doesn't affect understanding; Low: Serious misuse.

Given text: text"'

Figure 5: Prompt for the Text Quality Assessment Task.

You are a professional AI mathematical solution reasoning chain evaluation expert who will objectively assess the given mathematical solution reasoning chain. Assessment must be based on specific evidence rather than subjective impressions.

Original problem:
{instruction}
{input_text}

Model-generated reasoning chain:
{output}

I. Evaluation Framework (Five-Dimensional Scoring)

Please evaluate the following five dimensions at levels 1-5, listing key evidence for each dimension before assigning a level based on the scoring criteria:

1. **Reasoning Correctness** (Weight 25 points) - Assessment: Whether the mathematical reasoning process follows logic, whether theorems and definitions are applied correctly, and whether the derivation steps are rigorous - Key points for mathematical solving tasks: Accurate application of formulas, rigorous reasoning process, clear logical relationships - Key evidence: [List 1-3 pieces of key evidence supporting the score, marked in [L1], [L2] format] - Scoring criteria: * Level 5: 25 points - Excellent - Reasoning steps are rigorous and complete, mathematical logic is flawless * Level 4: 20 points - Good - Main reasoning is correct, with minor flaws that don't affect the conclusion * Level 3: 15 points - Average - Clear reasoning errors exist, but overall approach is still identifiable * Level 2: 10 points - Poor - Multiple reasoning errors, with key steps containing gaps * Level 1: 5 points - Failing - Reasoning is confused, with basic mathematical logical errors - Level: ___ (Score: ___ points) - Brief reason: (Within 20 characters, citing evidence numbers, such as "Based on [L1] and [L2]...")

2. **Method Relevance** (Weight 20 points) - Assessment: Whether the chosen solution method is appropriate for the problem, and whether the solution steps are targeted - Key points for mathematical solving tasks: Appropriate method selection, steps closely aligned with the problem, avoiding redundant calculations - Key evidence: [List 1-3 pieces of key evidence supporting the score, marked in [R1], [R2] format] - Scoring criteria: * Level 5: 20 points - Excellent - Optimal method selection, each step directly serves the solution goal * Level 4: 16 points - Good - Reasonable and efficient method, with occasional unnecessary steps * Level 3: 12 points - Average - Feasible method but not optimized, with detours present * Level 2: 8 points - Poor - Inappropriate method selection, leading to complicated process * Level 1: 4 points - Failing - Method does not match the problem, going in the wrong direction - Level: ___ (Score: ___ points) - Brief reason: (Within 20 characters, citing evidence numbers)

3. **Step Completeness** (Weight 20 points) - Assessment: Whether the solution process includes all necessary steps, and whether key transformations are clearly shown - Key points for mathematical solving tasks: Definition explanation, condition analysis, derivation process, verification steps all present - Key evidence: [List 1-3 pieces of key evidence supporting the score, marked in [C1], [C2] format] - Scoring criteria: * Level 5: 20 points - Excellent - Steps are complete and detailed, key transformations are clear, includes necessary verification * Level 4: 16 points - Good - Main steps are complete, with slight simplifications that don't affect understanding * Level 3: 12 points - Average - Missing some key steps, affecting the completeness of the process * Level 2: 8 points - Poor - Many jumps in reasoning, key steps missing * Level 1: 4 points - Failing - Almost no necessary solution steps shown - Level: ___ (Score: ___ points) - Brief reason: (Within 20 characters, citing evidence numbers)

4. **Calculation Accuracy** (Weight 20 points) - Assessment: Whether numerical calculations, formula applications, and symbolic operations are accurate

- Key points for mathematical solving tasks: Arithmetic correctness, formula accuracy, unit consistency, thorough consideration of special cases - Key evidence: [List 1-3 pieces of key evidence supporting the score, marked in [F1], [F2] format] - Scoring criteria: * Level 5: 20 points - Excellent - All calculations completely correct, formulas applied precisely * Level 4: 16 points - Good - Core calculations correct, with minor non-critical errors * Level 3: 12 points - Average - Some calculation errors, affecting partial results * Level 2: 8 points - Poor - Numerous calculation errors, seriously affecting conclusions * Level 1: 4 points - Failing - Basic calculation errors, incorrect formula application - Level: ___ (Score: ___ points) - Brief reason: (Within 20 characters, citing evidence numbers) - Calculation judgment label: [Choose: correct/minor_errors/major_errors/systematic_errors] 5. **Conclusion Reasonableness** (Weight 15 points) - Assessment: Whether the final answer is reasonable, consistent with the derivation process, and meets the requirements of the problem - Key points for mathematical solving tasks: Answer consistent with derivation, values reasonable, constraints satisfied - Key evidence: [List 1-3 pieces of key evidence supporting the score, marked in [D1], [D2] format] - Scoring criteria: * Level 5: 15 points - Excellent - Conclusion completely correct, perfectly connected with the derivation process * Level 4: 12 points - Good - Conclusion basically correct, with minor flaws in expression * Level 3: 9 points - Average - Conclusion partially correct, somewhat disconnected from derivation * Level 2: 6 points - Poor - Conclusion incorrect or contradicts derivation * Level 1: 3 points - Failing - Conclusion completely wrong or no conclusion given - Level: ___ (Score: ___ points) - Brief reason: (Within 20 characters, citing evidence numbers)

II. Comprehensive Assessment

Total score: ___ points (Sum of scores across dimensions)

Main strengths (2 points): 1. 2.

Main weaknesses (2 points): 1. 2.

Overall evaluation: (Within 30 characters)

III. Key Focus Areas for Mathematical Solution Reasoning Chain Evaluation

- Reasonableness of method selection: Whether an appropriate solution method was chosen for the problem - Rigor of reasoning chain: Whether each step of derivation has sufficient mathematical basis - Accuracy of calculation process: Whether numerical calculations and symbolic operations are correct - Sufficiency of result verification: Whether necessary checks were performed on the result - Consideration of special cases: Whether boundary conditions and special cases were considered

IV. JSON Return Format

After completing the evaluation, please return results in the following JSON format:

“‘json { "dimensions": { "reasoning_correctness": { "level": X, "score": Y, "reason": "Brief reason", "evidence": "[L1]: Specific evidence 1", "[L2]: Specific evidence 2"] }, "method_relevance": { "level": X, "score": Y, "reason": "Brief reason", "evidence": "[R1]: Specific evidence 1", "[R2]: Specific evidence 2"] }, "step_completeness": { "level": X, "score": Y, "reason": "Brief reason", "evidence": "[C1]: Specific evidence 1", "[C2]: Specific evidence 2"] }, "calculation_accuracy": { "level": X, "score": Y, "reason": "Brief reason", "evidence": "[F1]: Specific evidence 1", "[F2]: Specific evidence 2"], "calculation_label": "correct/minor_errors/major_errors/systematic_errors" }, "conclusion_reasonableness": { "level": X, "score": Y, "reason": "Brief reason", "evidence": "[D1]: Specific evidence 1", "[D2]: Specific evidence 2"] } }, "total_score": X, "main_strengths": "Strength 1", "Strength 2"], "main_weaknesses": "Weakness 1", "Weakness 2"], "overall_evaluation": "Overall evaluation content" } “‘

Figure 6: Prompt for Quality Assessment of COT in Mathematical Problem Solving Tasks

You are a professional AI chain-of-thought evaluation expert who will objectively evaluate the quality of a given text quality assessment chain-of-thought. The evaluation must be based on specific evidence rather than subjective impressions.

Original question: {instruction} {input_text}

Model generated chain-of-thought: {prediction}

Reference chain-of-thought (the reference chain-of-thought is produced by inputting the original question into DeepSeek R1, which is an excellent reasoning large language model with 671B parameters): {output}

I. Evaluation Framework (Five-Dimension Scoring) Please evaluate the following five dimensions (Level 1-5), for each dimension first list the key evidence, then give a level based on the scoring criteria:

1. **Logical Correctness** (weight 25 points) - Evaluation: Whether the chain-of-thought itself is self-consistent, whether the reasoning steps conform to logical rules and relationship judgments are correct - Key points for quality assessment tasks: Accurate understanding of metrics, appropriate application of evaluation standards, sufficient judgment basis - Key evidence: [List 1-3 key pieces of evidence supporting the score, marked with [L1], [L2] format] - Scoring criteria: * Level 5: 25 points - Excellent - Rigorous logical reasoning, accurate judgments, almost no logical flaws * Level 4: 20 points - Good - Clear overall logic, with minor imprecisions that do not affect the main conclusions * Level 3: 15 points - Average - Contains obvious logical issues, but the main reasoning path remains discernible * Level 2: 10 points - Poor - Multiple logical errors affecting conclusion reliability * Level 1: 5 points - Failing - Severe logical confusion, reasoning unable to support conclusions - Level: _ (Score: _ points) - Brief justification: (Within 20 characters, citing evidence numbers, such as "According to [L1] and [L2]...")

2. **Relevance** (weight 15 points) - Evaluation: Whether the chain-of-thought addresses the original problem, avoiding digression and irrelevant content - Key points for quality assessment tasks: Evaluation of all metrics, analysis based on textual features, avoidance of subjective assumptions - Key evidence: [List 1-3 key pieces of evidence supporting the score, marked with [R1], [R2] format] - Scoring criteria: * Level 5: 15 points - Excellent - Completely focused on task requirements, no digression or redundant content * Level 4: 12 points - Good - Primarily focused on the task, with minimal indirectly relevant content * Level 3: 9 points - Average - Contains obvious digressions, but the main content remains task-relevant * Level 2: 6 points - Poor - Substantial content unrelated to the task * Level 1: 3 points - Failing - Almost all content unrelated to the task - Level: _ (Score: _ points) - Brief justification: (Within 20 characters, citing evidence numbers)

3. **Completeness** (weight 15 points) - Evaluation: Whether the chain-of-thought covers all key steps required to solve the problem - Key points for quality assessment tasks: Consideration of all metrics, analysis of multiple textual aspects, provision of comprehensive evaluation - Key evidence: [List 1-3 key pieces of evidence supporting the score, marked with [C1], [C2] format] - Scoring criteria: * Level 5: 15 points - Excellent - Contains all necessary steps, comprehensively considers various scenarios * Level 4: 12 points - Good - Contains main steps, with minor omissions that do not affect conclusions * Level 3: 9 points - Average - Missing some key steps, affecting partial conclusions * Level 2: 6 points - Poor - Missing numerous key steps, conclusions insufficiently supported * Level 1: 3 points - Failing - Almost no necessary reasoning steps included - Level: _ (Score: _ points) - Brief justification: (Within 20 characters, citing evidence numbers)

4. **Factual Accuracy** (weight 25 points) - Evaluation: Whether facts, concepts, and information stated in the chain-of-thought are accurate - Key points for quality assessment tasks: Accurate understanding of metric definitions, accurate identification of textual features, absence of hallucinations - Key evidence: [List 1-3 key pieces of evidence supporting the score, marked with [F1], [F2] format] - Scoring criteria: * Level 5: 25 points - Excellent - All factual statements completely accurate, no hallucinations * Level 4: 20 points - Good - Core facts accurate, minor errors do not affect the overall analysis * Level 3: 15 points - Average - Some core facts incorrect, affecting partial reasoning quality * Level 2: 10 points - Poor - Numerous factual errors severely affecting reasoning quality * Level 1: 5 points - Failing - Mostly incorrect information or severe hallucinations - Level: _ (Score: _ points) - Brief justification: (Within 20 characters, citing evidence numbers)

5. **Conclusion Faithfulness** (weight 20 points) - Evaluation: Whether conclusions naturally and directly derive from the preceding reasoning - Key points for quality assessment tasks: Final score must be based on prior analysis, metrics judgment consistent throughout - Key evidence: [List 1-3 key pieces of evidence supporting the score, marked with [D1], [D2] format] - Scoring criteria: * Level 5: 20 points - Excellent - Conclusions entirely derived from the reasoning process, forming a perfect logical closure * Level 4: 16 points - Good - Conclusions primarily derived from reasoning, with minimal reasonable leaps * Level 3: 12 points - Average - Obvious disconnect between conclusions and reasoning, but still related * Level 2: 8 points - Poor - Weak relationship between conclusions and reasoning, appearing independently generated * Level 1: 4 points - Failing - Conclusions completely disconnected from reasoning or self-contradictory - Level: _ (Score: _ points) - Brief justification: (Within 20 characters, citing evidence numbers)

II. Comprehensive Evaluation Total score: _ points (Sum of all dimension scores)

Main strengths (2 points): 1. 2.

Main weaknesses (2 points): 1. 2.

Overall assessment: (Within 30 characters)

III. Key Evaluation Focus for Chain-of-Thought Quality Assessment - Metric comprehension accuracy: Whether evaluation metrics definitions are accurately understood - Evaluation comprehensiveness: Whether all required metrics are covered - Evidence sufficiency: Whether adequate textual evidence is provided for each metric judgment - Judgment consistency: Whether judgments across metrics are mutually consistent - Quantitative scoring rationality: Whether final scores are based on reasonable analytical processes

IV. JSON Result Format After completing the evaluation, please return results in the following JSON format:

“‘json { "dimensions": { "reasoning_correctness": { "level": X, "score": Y, "reason": "Brief justification", "evidence": ["[L1]: Specific evidence 1", "[L2]: Specific evidence 2"] }, "method_relevance": { "level": X, "score": Y, "reason": "Brief justification", "evidence": ["[R1]: Specific evidence 1", "[R2]: Specific evidence 2"] }, "step_completeness": { "level": X, "score": Y, "reason": "Brief justification", "evidence": ["[C1]: Specific evidence 1", "[C2]: Specific evidence 2"] }, "calculation_accuracy": { "level": X, "score": Y, "reason": "Brief justification", "evidence": ["[F1]: Specific evidence 1", "[F2]: Specific evidence 2"], }, "conclusion_reasonableness": { "level": X, "score": Y, "reason": "Brief justification", "evidence": ["[D1]: Specific evidence 1", "[D2]: Specific evidence 2"] } }, "total_score": X, "strengths": ["Strength 1", "Strength 2"], "weaknesses": ["Weakness 1", "Weakness 2"], "overall_assessment": "Overall assessment content" } “‘

Figure 7: Prompt for Quality Assessment of Chain of Thought in Data Quality Assessment Tasks