Socratic Style Chain-of-Thoughts Help LLMs to be a Better Reasoner

Anonymous ACL submission

Abstract

Synthetic data generation has emerged as a promising approach to enhance the reasoning capabilities of large language models. However, existing methods remain hindered by high costs—either through expensive API access or additional intermediate training—and are limited in their ability to generalize across different domains. To address these challenges, we propose a multi-agent debate framework based on the Socratic questioning strategy, abbreviated as **SoDa**. Distinguished from previous methods that prioritize data quantity, we highlight the wisdom of Socratic questioning in augmenting reasoning quality by deepening the thinking process to encourage exploration and broadening it to motivate self-reflection on each question. Combined with our efficient production pipeline, SoDa enables scaling while maintaining affordable costs. We use SoDa to generate diverse datasets for mathematics and code generation tasks with the Qwen2.5-7B-Instruct model, successfully fine-tuning a range of foundation models, from general-purpose ones to OpenAI o1-like ones. For mathematics, the experimental results show that SoDa outperforms the performance of existing datasets at the same scale, achieving improvements ranging from 1.3% to 13.5%. Remarkably, SoDa with 30K examples even surpasses the ScaleQuest dataset with 1000K samples, demonstrating significant efficiency. Our findings highlight the potential of SoDa as a universal, scalable, and cost-effective method for enhancing reasoning capabilities in large models across domains.

1 Introduction

The enhancement of reasoning capabilities in Large Language Models (LLMs) has become a recent research focus (Huang et al., 2024; Min et al., 2024). Recent studies have achieved notable progress by utilizing large language models to synthesize high-quality training data (Ding et al., 2024), resulting in substantial performance improvements on mathematical reasoning benchmarks such as GSM8K (Cobbe et al., 2021a) and OlympiadBench (He et al., 2024a). However, these methods depend on strong supervisory signals from two primary sources: i) powerful LLMs, which generally incur high API costs (Yu et al., 2024), and ii) fine-tuned smaller models, which demand extensive training and struggle with domain adaptation (Ding et al., 2024). These dependencies inherently constrain the scalability and flexibility of their data generation processes. Furthermore, existing approaches are predominantly tailored to mathematics, with limited generalizability to other domains such as code generation. Consequently, there is an urgent need for a cost-effective and adaptable method for generating high-quality data across diverse domains.

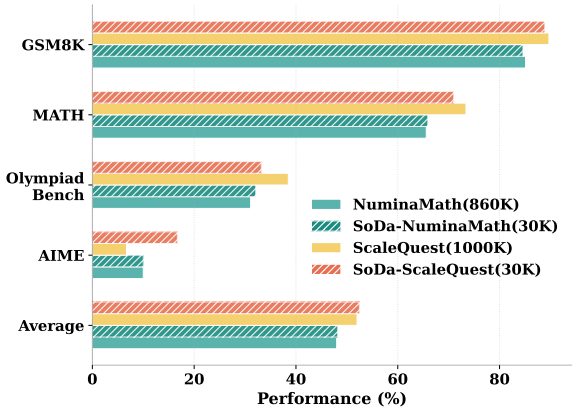

Figure 1: Performance of Qwen2-Math-7B fine-tuned on SoDa-enhanced data (30K) vs. full-scale NuminaMath (860K) and ScaleQuest (1000K) datasets on mathematical benchmarks.

We draw inspiration from the Socratic questioning method (Qi et al., 2023), a systematic approach to critical thinking that uncovers underlying assumptions, clarifies concepts, and motivates deeper understanding through iterative ques-

tioning. This iterative process mirrors the process of human learning, which involves experiencing and exploring through trial-and-error (Song et al., 2024) interactions with the environment, ultimately leading to improved solutions. If models can adopt this paradigm, it would enable a more natural extension of existing data-driven methods. This approach fundamentally differs from the direct answer-generation style of current Chain-of-thoughts (CoT) methods and offers several key advantages. First, it does not rely heavily on advanced models, making it highly efficient for scaling. Second, it requires no additional training and instead relies solely on self-debating, allowing generalization across diverse domains and even real-world problem-solving scenarios. Ultimately, the trained models would demonstrate enhanced adaptability and robustness, paving the way for broader and more versatile applications.

We propose the **SoDa** (**So**cratic **D**eb**a**te), a model debate framework based on the Socratic questioning strategy (Qi et al., 2023). For the Socratic component, we incorporate its principles to iteratively explore and reflect, while additional constraints ensure comprehensive generation of thoughts and answers. For the debate component, we simplify the process by reducing it to a fixed number of turns between two roles (*i.e.,* the Socratic and the student), enabling efficient execution without compromising quality. To further accelerate scaling, we optimize the generation pipeline by leveraging the observation that in-context learning (ICL) can effectively guide the debate process. This allows us to generate high-quality datasets using a 7B model, Qwen2.5-7B-Instruct (Yang et al., 2024b), which significantly reduces computational costs. Ultimately, our method achieves efficient dataset generation and maintains adaptability across diverse domains (see Table 1), facilitating broader applicability.

Based on the proposed SoDa, we expanded existing datasets by converting them into Socratic-dialogue formats. This includes three math datasets: DART-Math (Tong et al., 2024), NuminaMath (Li et al., 2024), and ScaleQuest (Ding et al., 2024), as well as a code dataset McEval-Instruct (Chai et al., 2024). For math tasks, training with the expanded dataset resulted in significant performance improvements on LLaMa-3.1-8B (Dubey et al., 2024), with gains ranging from 1.3% to 13.5%. These improvements were consistently observed across various models, such as Qwen2-Math-7B (Yang et al., 2024a), a math-specific model, and Skywork-o1 (Skywork-o1, 2024), an o1-like model. Notably, utilizing 30K SoDa-enhanced data achieves comparable or superior performance to full-scale datasets (see Figure 1). Additionally, our method achieved an average 2.7% improvement on code tasks. These results suggest that our approach enhances CoT reasoning, making it a versatile and efficient method for synthetic data generation across diverse domains..

Methods	Multiple Domains	Synthesis Model	Training Free
MetaMath	✗	GPT-3.5	✓
DART-Math	✗	DSMath (7B)	✓
NuminaMath	✗	GPT-4o	✓
ScaleQuest	✗	Qwen2-Math (7B)	✗
SoDa	✓	Qwen2.5 (7B)	✓

Table 1: Overview comparison between SoDa and recent advanced synthetic data generation methods.

2 Related Work

2.1 Math Reasoning

OpenAI o1 (Jaech et al., 2024) has demonstrated remarkable reasoning capabilities, primarily attributed to its use of long chain-of-thought (CoT) reasoning. This model has become a pivotal milestone in the development of reasoning methods for LLMs. Prior to o1, mathematical reasoning was predominantly driven by diverse, heuristic-based approaches, including prompting techniques (Chia et al., 2023), instruction tuning techniques (Yue et al., 2024), tool based methods (Wang et al., 2024) and preferring tuning (Lai et al., 2024). Following o1, the emphasis shifted toward long-form CoT processes, which are now widely regarded as an effective strategy for enhancing reasoning capabilities. Subsequent work, such as DeepSeekMath (Shao et al., 2024) and Qwq (Team, 2024), has built upon this paradigm, further refining the approach and fostering deeper, more systematic reasoning.

2.2 Synthetic Data Generation for LLMs

In the process of replicating o1-like reasoning, data synthesis has been recognized as a highly effective approach. Specifically, this involves expanding a seed dataset by enriching its questions or answers to generate larger and more diverse datasets. For question-centric augmentation, common methods include increasing the complexity of questions (Luo et al., 2023), modifying ques-

tions to enhance diversity (Yu et al., 2024), or generating high-quality questions through a trained model (Ding et al., 2024). While these methods are effective, high-quality models are often challenging to scale, and training-based approaches are difficult to generalize across domains. For answer-centric augmentation, existing work includes introducing additional knowledge into answers and improving their quality (Didolkar et al., 2024; Shah et al., 2024). Some approaches propose to enhance the quality of both questions and answers simultaneously. For example, Numinamath (Li et al., 2024) incorporates real-world data to achieve this dual enhancement. However, these methods are specialized for mathematics and struggle to generalize to other domains.

3 Methods

We present a general framework (see Figure 2) for efficiently generating synthetic conversational Socratic-style Chain-of-Thought (CoT) data, which includes two key components: the *Socratic module*, designed to generate insightful questions that stimulate critical thinking, and the *multi-agent debate module*, which extends the dialogue into multi-turn interactions for deeper exploration. To accelerate the data generation process, we propose an efficient pipeline leveraging in-context learning with meticulously curated high-quality demonstrations.

3.1 Socratic Style Debate Framework

Generally speaking, Socratic questioning is a method of exploration designed to inspire deeper thinking through thought-provoking questions. An intuitive way to do this is to mimic real-world conversations between a teacher and a student, with additional constraints to ensure the teacher adheres to Socratic-style principles. The key mechanism behind our method is to enrich the thought process for each individual sample, in contrast to existing methods that primarily aim to automatically increase data quantity. This enables more efficient utilization of the available data. Next, we will detail the Socratic module and the debate module.

Socratic Module. We begin with formalizing the data generation process in the context of instruction tuning. Typically, the training datasets consist of question-answer pairs, *i.e.,* $(q, a) \in \mathcal{D}_{seed}$, where $\mathcal{D}_{seed}$ denotes the seed dataset used for data synthesis. In this process, the LLM is prompted to adopt a Socratic role, generating new guiding questions Q based on the given questions. Specifically, the prompts are split into two categories: *Socratic principles (P_{So})* and *questioning requirements (P_Q)*. Then we generate the first guiding question at the initial round ($t = 0$) as follows:

$$Q^0 = LLM(q, a, P_{So}, P_Q). \quad (1)$$

For Socratic principles, we follow the study (Qi et al., 2023) and incorporate the ten common guidelines of Socratic questioning (see Table 2). For questioning requirements, we introduce three specific constraints to prevent irrelevant or endless discussions.

• *Ask questions only.* The Socratic should avoid providing direct answers, focusing instead on asking questions. This encourages the Student to think independently and make more attempts to solve the problem.

• *Encourage exploration.* The Socratic should guide the Student to explore multiple possible solutions, linking the problem to their existing knowledge and broadening their thought process.

• *Promote reflection.* The Socratic should help the Student reflect on each step to reduce uncertainty and verify correctness. This ensures a solid final solution and a deeper understanding of the problem.

Following the application of these Socratic principles, we analyze the distribution of the generated Socratic-style question types, with the results presented in Table 2. Except for the principle "Probing Evidence & Reasons ", which accounts for a slightly larger proportion (20.8%), the other principles are distributed relatively evenly. This result aligns with our expectations, as Socratic questioning is designed to approach students from diverse perspectives, facilitating broader exploration and capturing a wider range of knowledge points.

Multi-Agent Debate. Building on the foundation of Socratic principles, we extend our approach by adopting a multi-agent debate framework inspired by prior research (Liang et al., 2023). Specifically, we design two agents as debaters: the *Socratic* role R_S and the *Student* role R_u, engaging in a structured Socratic-style dialogue for a maximum number of rounds, T_{max}:

$$Q^{(t)} = R_S(q, a, P_{So}, P_Q, \{Q^i, A^i\}_{i=0}^{i<t}), \quad (2)$$
$$A^{(t)} = R_u(P_A, \{Q^i, A^i\}_{i=0}^{i<t}, Q^{(t)}),$$
$$where \quad t \in \{1, 2, \cdots, T_{max} - 1\},$$

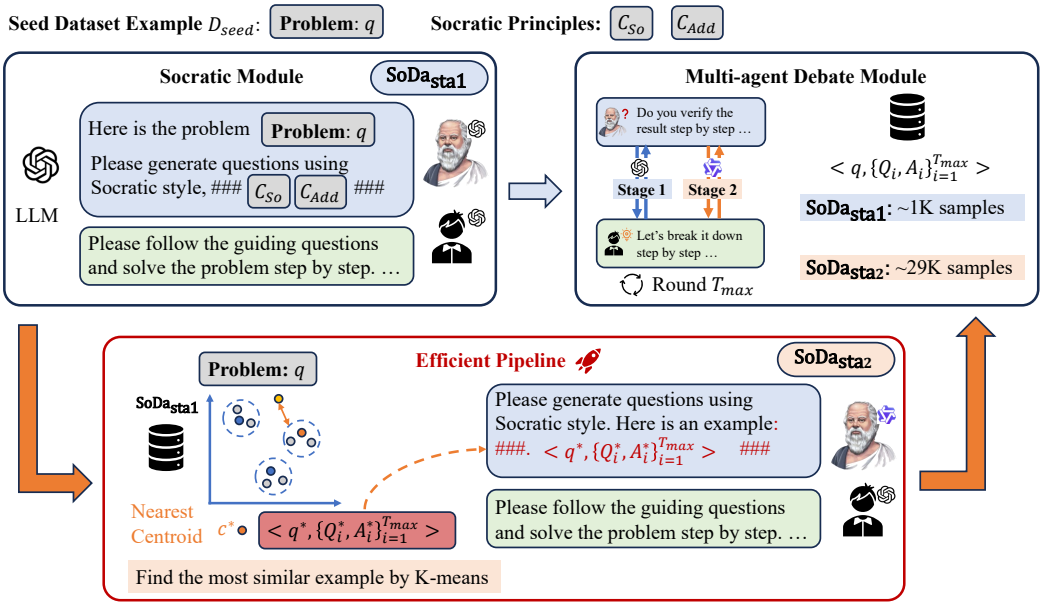

Figure 2: Overview of the SoDa framework.

Principles	Ratio (%)
Clarification Questions	9.1
Probing Assumptions	9.8
Probing Evidence & Reasons	20.8
Exploring Alternative Perspectives	10.8
Examining Implications & Consequences	8.4
Questioning the Question	6.7
Encouraging Reflection	7.7
Challenging Logic	9.4
Exploring Causes & Effects	7.1
Summarizing & Synthesizing	10.1

Table 2: Basic principles and the ratio in our dataset. Specific examples for each principle refer to Table 9 in the Appendix.

Methods	# Sample	Dataset	Olympiad Bench
Base	-	-	16.6
Tuning	$\sim$ 50K	NuminaMath*	20.0
ICL	1	random	17.6
	1	random SoDa$_{sta1}$	20.1
	1	nearest SoDa$_{sta1}$	21.0

Table 3: Investivation of the ICL experiments on the base model Qwen2-base. "SoDa$_{sta1}$" denotes the generate datasets using GPT-o1-mini as Socratic role. The "NuminaMath*" is the olympiad subset from the NuminaMath source.

where P_A represents the prompts for the student roles to answer in a step-by-step manner and $\{Q^i, A^i\}_{i=0}^{i<t}$ refers to the dialogue history up to round t. For each interaction, the Socratic guides the student step by step through insightful questions, while the student responds based on these prompts. This iterative structure allows for a progressive and dynamic exploration of the input question, fostering deeper reasoning and broader exploration. Finally, the process progressively builds the final dataset in the first stage, *i.e.,* SoDa$_{sta1}$.

Different from existing methods (Liang et al., 2023) that rely on high-quality questions and a judge mechanism to guide and control debates, our method simplifies the process by removing the judge and adopting a fixed number of rounds. In this way, we focus solely on the core discussion process, enabling faster iterations and improved scalability. This emphasis on efficiency and simplicity allows our approach to achieve effective Socratic-style reasoning while remaining cost-effective and practical for large-scale applications.

3.2 Efficient Pipeline with In-context Learning

So far, we have established an initial debate framework, but it still lacks the ability to scale efficiently. Therefore, we explore an efficient data generation approach that leverages small LLMs guided by high-quality prompts. Our key observation is that using high-quality data as prompts can effectively steer the generation process toward producing high-quality outputs. Building on this insight, we propose a clustering-based prompt generation strategy to automate and accelerate the creation of a scalable, high-quality dataset.

Observations. The previous framework has enabled a rich expansion of reasoning processes with powerful LLMs in the Socratic debate framework. It is reasonable to assume that these reasoning processes overlap with new input queries. Leveraging this overlap, we hypothesize that such data can serve not only as training material but also as effective prompts to guide the generation of new samples. If this holds true, the value of high-quality data produced by powerful LLM can be significantly amplified, as each sample can serve as an independent prompt to guide the generation of a class of related queries using small LLMs.

To validate these assumptions, we conducted an experiment where, for a given input query q_{inp}, we randomly selected a high-quality sample $< q, a, \{Q^{(t)}, A^{(t)}\}_{i=0}^{i<T_{max}} > \in SoDa_{sta1}$ to form a prompt. Then we evaluated the performance of the model on a downstream task using this prompt-based setup. The results are summarized in Table 3. We observe that randomly selected samples can effectively serve as prompts, guiding small LLMs to solve challenging problems. Notably, the accuracy achieved with random SoDa$_{sta1}$ samples (20.1%) slightly surpasses that of fine-tuning with around 50K CoT data (20.0%). Furthermore, accuracy improves when the prompt is selected based on its relatedness to the query by identifying the nearest sample (21.0%). These findings motivate the development of a clustering-based data generation pipeline to further enhance performance.

Generation by ICL. Building on the observations, we further explore how to systematically identify the most suitable example for each given query. To achieve this, we propose a clustering-based approach that efficiently leverages high-quality data samples. Specifically, we first employ a K-means clustering method to group the high-quality dataset into N clusters, $\{c_1, c_2, \cdots, c_N\}$. Each cluster i is represented by a centroid c_i and representative sample $< q_i, a_i, \{Q^{(t)}, A^{(t)}\}_{t=0}^{t<T_{max}} >$, which is the data point closest to the centroid. Second, for an input query q_{inp}, we identify the most relevant cluster c^* and form the corresponding example as prompt by locating the centroid closest to q_{inp} based on a predefined distance metric $d(\cdot)$:

$$c^* = \arg\min_{c_i} d(q_{inp}, c_i). \quad (3)$$

This clustering-based selection not only automates the process of finding suitable prompts but also ensures that each new query is guided by the most contextually appropriate example. The whole process is summarized in the Algorithm 1.

Algorithm 1 Efficient generation pipeline.

Input: High-quality dataset in first stage, SoDa$_{sta1}$, input seed dataset to process $\mathcal{D}_{seed}$.
Output : Generated dataset for the second stage SoDa$_{sta2}$
K-means Clustering
Apply K-means clustering to the queries in high-quality dataset SoDa$_{sta1}$, grouping the data into N clusters and the representative samples $\{c_1, c_2, \cdots, c_N\}$
Generation
for $q_{inp} \in \mathcal{D}_{seed}$ **do**
  / Find the closest cluster */*
  $c^* = \arg\min_{c_i} d(q_{inp}, c_i)$
  / Select the representative sample as prompts P_c */*
  $P_c = \{< q^*, a^*, \{Q^{(t)*}, A^{(t)*}\}_0^{T_{max}} >\}$
  for $t = 0 \rightarrow T_{max}$ **do**
    $Q^{(t)} = R_S(q, a, P_{So}, P_Q, P_c, \{Q^i, A^i\}_{i=0}^{i<t})$
    $A^{(t)} = R_u(P_A, \{Q^i, A^i\}_{i=0}^{i<t}, Q^{(t)})$
  end for
  SoDa$_{sta2} \leftarrow \{q_{inp}, a_{inp}, \{Q^{(t)}, A^{(t)}\}_0^{T_{max}}\}$
end for
return SoDa$_{sta2}$

4 Experiments

4.1 Experimental Setup

Data Synthesis. We evaluate our method on mathematics and code tasks. In the mathematics domain, data synthesis is performed using three datasets: DART-Math (Tong et al., 2024), NuminaMath (Li et al., 2024), and ScaleQuest (Ding et al., 2024). Code-oriented synthesis utilized the McEval-Instruct dataset (Chai et al., 2024). For each dataset, we randomly select 30,000 problem-solution pairs as the source data. GPT-o1 is used as the generator in SoDa$_{sta1}$ to synthesize 1,000 high-quality samples, with the clusters N set to 50. Based on these samples, we employ Qwen2.5-7B-Instruct in SoDa$_{sta2}$ framework to generate additional 29,000 samples. The final synthesized datasets—comprising 30,000 samples each—were designated as SoDa-DART-Math, SoDa-NuminaMath, SoDa-ScaleQuest, and SoDa-MCEval, respectively. The fine-tuning configuration is provided in the Appendix A.1.

Baseline Methods. We compare the data generated by SoDa with popular synthetic datasets, including DART-Math (Tong et al., 2024), NuminaMath (Li et al., 2024), and ScaleQuest (Ding et al., 2024). We additionally consider performance comparisons with advanced models such as GPT-4o, LLaMa-3.1-8B (Dubey et al., 2024), DeepSeekMath-7B-RL (Shao et al., 2024),

Skywork-o1-open (Skywork-o1, 2024) and models of Qwen family (Yang et al., 2024a).

For the mathematical evaluation, we consider four benchmarks: GSM8K (Cobbe et al., 2021b), MATH (Hendrycks et al., 2021), OlympiadBench (He et al., 2024b) and AIME-24 (Mathematical Association of America, 2024). Code generation capabilities were evaluated on HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2024), MBPP (Austin et al., 2021) and MBPP+ (Liu et al., 2024) problems. The zero-shot pass@1 accuracy is reported.

4.2 Main Results

Comparing with Synthetic Data Generation Methods. We fine-tune popular models using synthetic datasets and summarize the results in Table 4. At comparable dataset scales, SoDa-style datasets consistently outperform other methods, with improvements ranging from 1.3% on DART-Math data with Llama3.1-8b model to 13.5% on NuminaMath data with Llama3.1-8b model. Next, we examine the impact of seed datasets. Among the three datasets evaluated, ScaleQuest demonstrates the best average performance, followed by NuminaMath in second place. This trend is also observed in the SoDa-style datasets, highlighting the critical role of seed datasets in the synthetic data generation process. When analyzing different base models, we find that performance gains diminish as model complexity increases. For example, Skywork-o1-open, which targets o1-like abilities, shows limited improvement after fine-tuning with seed datasets. However, reasonable gains can still be observed in average metrics (58.8 *vs.* 56.5). Notably, SoDa-ScaleQuest achieves a 6.7% improvement (20.0 *vs.* 13.3) on AIME, further demonstrating the effectiveness of the SoDa approach. In Appendix Figure 4, we present the Intelligence Quality per Token (IQPT), calculated as *average performance* divided by *number of tokens*, for NuminaMath, ScaleQuest, SoDa-NuminaMath, and SoDa-ScaleQuest. As shown, our methods significantly improve IQPT, demonstrating that SoDa effectively enhances the information density.

Comparison with Advanced Models. To demonstrate the advantages of our data-trained models in mathematical reasoning, we compare them with current state-of-the-art LLMs. Among *frontier LLMs*, the closed-source model GPT4o achieves the best overall performance. However, our trained model, Skywork-o1-SoDa, achieves comparable average performance (58.5 *vs.* 58.9). Notably, on more complex benchmarks, our model outperforms GPT4o, with significant gains observed on tasks such as AIME (20.0 *vs.* 16.7) and OlympiadBench (44.9 *vs.* 43.3). These results further validate the effectiveness of the reasoning processes generated by our approach. It is particularly noteworthy that our data generation relied on a relatively simple 7B Qwen model. Despite this, fine-tuning more complex models with our generated data still yielded substantial improvements on challenging tasks, underscoring the robustness and scalability of our method.

4.3 Ablation Study

Our method comprises two key components: the Socratic module and the multi-agent debate module. To evaluate the contribution of each component, we fine-tune Qwen2-7B with each module separately and present the results in Table 5. To ablate the debate module, we modify the prompts to restrict the Socratic-style conversation to a single round. This results in a performance drop to 19.9. For the ablation of the Socratic module, we use a simplified multi-agent debate framework without a guiding judge and continue for a fixed number of rounds. This configuration produces the worst result, with performance dropping to 11.9.

These findings highlight the critical role of the Socratic module, which guides the discussion in the right direction. Without this guidance, long conversations can sometimes degrade the quality of the dataset, as observed in the results. Interestingly, the configuration "w/o both" outperforms "w/o Socratic" (19.9 *vs.* 11.9), indicating that unguided long conversations can negatively impact fine-tuning performance. This result underscores the effectiveness of our guided conversational reasoning paradigm, which ensures meaningful and focused dialogue to enhance the quality of the dataset.

4.4 Generalization on Code Domain

We use the code generation task as an example to evaluate whether SoDa can generalize to other domains. Following a similar approach to the mathematics tasks, we augmented the McEval-Instruct (Chai et al., 2024) dataset by generating 30K additional samples for instruction fine-tuning. The results are presented in Table 7. On average, SoDa-McEval outperforms McEval across all benchmarks, achieving an average improvement of

Models	SFT Data (# Samples)	Synthesis Model	GSM8K	MATH	AIME	Olympiad Bench	Average
		Frontier LLMs					
DeepSeekMath-7B-RL	-	-	88.2	52.4	0.0	19.0	39.9
Qwen2-Math-7B	-	-	80.6	55.0	3.3	19.0	39.5
Qwen2-Math-7B-Ins	-	-	89.5	73.1	6.7	37.8	52.7
Qwen2.5-7B-Ins	-	-	91.8	74.5	13.3	37.2	54.2
Skywork-o1-open	-	-	91.6	78.1	13.3	43.1	56.5
GPT-4o-2024-08-06	-	-	**92.9**	**81.1**	16.7	43.3	58.9
		General Base Model					
Llama3.1-8B	DART-Math (30K)	DSMath-7B-RL	76.5	37.0	0.0	10.7	31.0
	SoDa-DART-Math (30K)	Qwen2.5-7B-Ins	76.2	40.4	0.0	12.6	32.3 $_{\uparrow 1.3\%}$
	NuminaMath (30K)	GPT-4o	57.7	23.4	0.0	8.1	22.3
	SoDa-NuminaMath (30K)	Qwen2.5-7B-Ins	80.0	41.0	6.7	15.6	35.8 $_{\uparrow 13.5\%}$
	ScaleQuest (30K)	Qwen2-Math-7B-Ins	79.2	38.8	0.0	9.5	31.8
	SoDa-ScaleQuest (30K)	Qwen2.5-7B-Ins	79.9	46.9	0.0	20.3	36.8 $_{\uparrow 5.0\%}$
		Math-Specialized Base Model					
Qwen2-Math-7B	DART-Math (30K)	DSMath-7B-RL	88.1	60.8	0.0	24.7	43.4
	SoDa-DART-Math (30K)	Qwen2.5-7B-Ins	86.9	63.8	6.7	27.7	46.2 $_{\uparrow 2.8\%}$
	NuminaMath (30K)	GPT-4o	76.0	54.8	6.7	27.0	41.1
	SoDa-NuminaMath (30K)	Qwen2.5-7B-Ins	84.6	65.9	10.0	32.1	48.1 $_{\uparrow 7.0\%}$
	ScaleQuest (30K)	Qwen2-Math-7B-Ins	88.1	69.7	6.7	32.1	49.2
	SoDa-ScaleQuest (30K)	Qwen2.5-7B-Ins	88.9	71.0	16.7	33.2	52.5 $_{\uparrow 1.5\%}$
		o1-like Base Model					
Skywork-o1-open	DART-Math (30K)	DSMath-7B-RL	89.8	64.4	3.3	31.7	47.3
	SoDa-DART-Math (30K)	Qwen2.5-7B-Ins	88.0	73.1	13.3	38.4	53.2 $_{\uparrow 5.9\%}$
	NuminaMath (30K)	GPT-4o	90.3	75.2	6.7	42.1	53.5
	SoDa-NuminaMath (30K)	Qwen2.5-7B-Ins	90.9	76.7	13.3	43.3	56.1 $_{\uparrow 2.6\%}$
	ScaleQuest (30K)	Qwen2-Math-7B-Ins	90.1	77.4	10.0	43.6	55.2
	SoDa-ScaleQuest (30K)	Qwen2.5-7B-Ins	91.0	78.0	**20.0**	**44.9**	58.5 $_{\uparrow 3.3\%}$

Table 4: Main results on four mathematical reasoning benchmarks.

Methods	# Samples	Olympiad Bench
SoDa	1K	23.4
w/o debate	1K	19.9
w/o Socratic	1K	11.9
w/o Both	1K	18.2

Table 5: Ablation experiments of debate framework and Socratic questions.

Synthesis Model	Rounds of Debate	# Samples	Olympiad Bench
Qwen2.5-7B-Ins	10	1K	23.7
Qwen2.5-7B-Ins	5	1K	23.4
GPT-o1-mini	5	1K	25.2
Qwen2.5-32B-Ins	5	1K	23.9
Qwen2.5-7B-Ins	5	1K	23.4

Table 6: Effect of rounds and synthesis model size.

2.7%. Notably, SoDa-McEval delivers a significant performance boost of up to 6.1% (85.4 *vs.* 80.5 on HumanEval). These results demonstrate the versatility of Socratic reasoning, highlighting its potential as a universal framework for improving performance across diverse tasks.

4.5 Detailed Analysis

Comparison of Different Debate Configurations. We investigate the impact of different configurations in the debate module, including the number of rounds and the type of synthesis model used. As outlined in Section 3.1, the SoDa framework employs a fixed 5-round discussion. To explore the effect of increasing rounds, we generate 1K samples using a 10-round configuration, fine-tune the Qwen2.5-7B-Instruct model, and report the results in the first two rows of Table 6. The results show a slight improvement in performance with 10 rounds of data (23.7 vs. 23.4). However, considering the trade-off between computational cost and performance, we select the five-round configuration for our final experiments.

For the synthesis models, we select three options based on their reasoning capabilities: GPT-o1-mini, Qwen2.5-32B-Instruct, and Qwen2.5-7B-Instruct. While GPT-o1-mini demonstrates significant advantages, the 32B Qwen model only slightly outperforms its 7B counterpart. Consequently, our current choice of the 7B model remains a favorable balance between performance and efficiency. Nevertheless, leveraging more powerful LLMs to further enhance the SoDa framework represents a promising direction for future work.

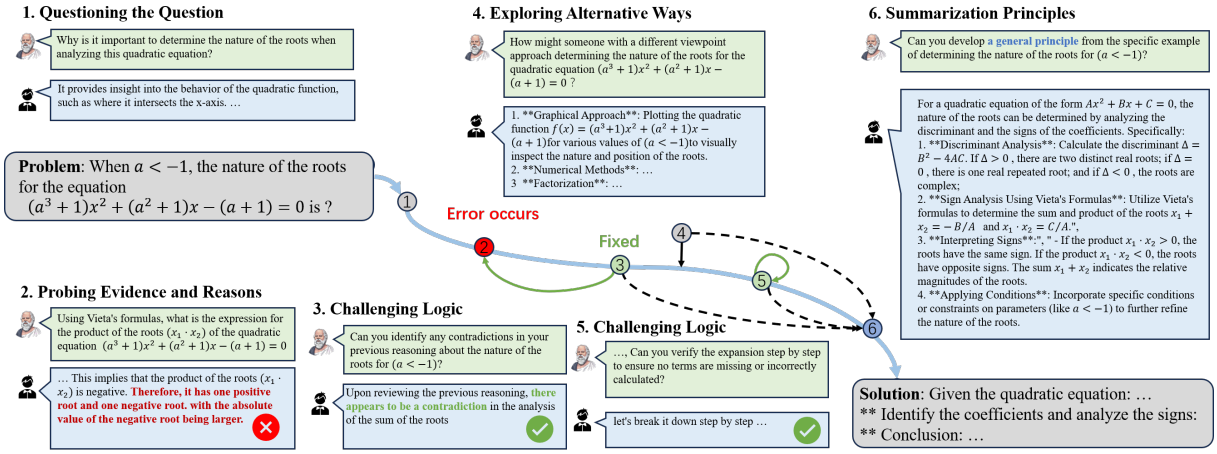

Figure 3: An example of dialogues illustrating the complete process from problem to solution. The SoDa expansion includes phases of reflection (*e.g.,* 1, 2, 3, 5), exploration (*e.g.,* 4), and summarization (*e.g.,* 3, 4, 5, 6).

SFT Data (# samples)	HumanEval	HumanEval+	MBPP	MBPP+	Average
Qwen2.5-Coder-7B(Base)	78.7	74.4	80.7	68.0	75.4
McEval-Instruct (30K)	80.5	76.8	80.7	66.9	76.2
SoDa-McEval (30K)	85.4	80.0	81.5	68.8	78.9

Table 7: Additional experiments conducted in code generation tasks.

Case Study of SoDa Dataset We analyzed the synthetic data to provide a clearer understanding of SoDa's effectiveness. SoDa leverages the debate process to introduce various questions, guiding the "student" role to explore different perspectives. This process, as illustrated in the Figure 3, involves multiple forms of reasoning, such as analyzing the problem, exploring alternative solutions, engaging in self-reflection, and revisiting previous steps. This highlights that the key to SoDa's success lies in its ability to expand these reasoning pathways, allowing the data to radiate from a single problem to cover a broader range of related areas. This radiating coverage is essential for generating diverse and comprehensive datasets, which we identify as a critical factor in SoDa's effectiveness.

Cost Analysis. Here we analyze the cost of the whole SoDa process, including high-quality data SoDa_{sta1} produced by accessing API and SoDa_{sta2} containing 30K samples generated by Qwen2.5-7B-Instruct. The results are summarized in the Table 8. We can see that the total cost is approximately 191$, with the majority of the expense coming from API access in the SoDa_{sta1} stage. This underscores the advantage of leveraging a low-cost model: with the effective data generation framework SoDa, it enables the efficient generation of high-quality datasets at a significantly reduced cost.

Stages	# Samples	GPU hours	Cost ($)
SoDa_{sta1}	1K	-	100
SoDa_{sta2}	29K	70	91
Total	30K	70	191

Table 8: Cost analysis of our method.

5 Conclusion

In this paper, we focused on enhancing data synthesis frameworks to achieve high-quality generation while maintaining low cost. To this end, we proposed *SoDa*, a multi-agent debate framework grounded in the Socratic questioning methodology. We initialized the debate roles, the Socratic and the student, using curated prompts based on fundamental Socratic principles, facilitating a structured and iterative reasoning process. To reduce generation costs, we optimized the pipeline by leveraging in-context learning with smaller models instead of relying on advanced LLMs, ensuring scalability without compromising quality. After fine-tuning a wide range of foundation models, our approach outperformed existing methods at similar data scales. Notably, it even surpassed Numina-math and ScaleQuest datasets, which contains 20x more data, on average benchmarks. This work highlights the potential of leveraging small LLMs to enhance powerful models effectively and cost-efficiently.

6 Limitations

While our method demonstrates strong empirical results across mathematical and coding domains, the current evaluation primarily focuses on synthetic benchmarks within these two fields. Its effectiveness in more diverse real-world scenarios, such as scientific reasoning or multimodal problem, remains to be explored. Additionally, our SoDa relies primarily on 7B-scale open-source models. Although this significantly reduces computational costs, scaling to larger foundation models may uncover new optimization opportunities, which warrants further exploration.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Linzheng Chai, Shukai Liu, Jian Yang, Yuwei Yin, Ke Jin, Jiaheng Liu, Tao Sun, Ge Zhang, Changyu Ren, Hongcheng Guo, et al. 2024. Mceval: Massively multilingual code evaluation. *arXiv preprint arXiv:2406.07436*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Yew Ken Chia, Guizhen Chen, Luu Anh Tuan, Soujanya Poria, and Lidong Bing. 2023. Contrastive chain-of-thought prompting. *Preprint*, arXiv:2311.09277.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021a. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021b. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

Aniket Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, Michal Valko, Timothy P. Lillicrap, Danilo J. Rezende, Yoshua Bengio, Michael Mozer, and Sanjeev Arora. 2024. Metacognitive capabilities of llms: An exploration in mathematical problem solving. *CoRR*, abs/2405.12205.

Yuyang Ding, Xinyu Shi, Xiaobo Liang, Juntao Li, Qiaoming Zhu, and Min Zhang. 2024. Unleashing reasoning capability of llms via scalable question synthesis from scratch. *CoRR*, abs/2410.18693.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024a. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3828–3850. Association for Computational Linguistics.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. 2024b. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Zhen Huang, Haoyang Zou, Xuefeng Li, Yixiu Liu, Yuxiang Zheng, Ethan Chern, Shijie Xia, Yiwei Qin, Weizhe Yuan, and Pengfei Liu. 2024. O1 replication journey–part 2: Surpassing o1-preview through simple distillation, big progress or bitter lesson? *arXiv preprint arXiv:2411.16489*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. 2024. Step-dpo: Step-wise preference optimization for long-chain reasoning of llms. *Preprint*, arXiv:2406.18629.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. 2024. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2024. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *CoRR*, abs/2308.09583.

Mathematical Association of America. 2024. American invitational mathematics examination - aime 2024. Accessed: 2025-01-27.

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. *arXiv preprint arXiv:2412.09413*.

Jingyuan Qi, Zhiyang Xu, Ying Shen, Minqian Liu, Di Jin, Qifan Wang, and Lifu Huang. 2023. The art of SOCRATIC QUESTIONING: recursive thinking with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4177–4199. Association for Computational Linguistics.

Vedant Shah, Dingli Yu, Kaifeng Lyu, Simon Park, Nan Rosemary Ke, Michael Mozer, Yoshua Bengio, Sanjeev Arora, and Anirudh Goyal. 2024. Ai-assisted generation of difficult math questions. *CoRR*, abs/2407.21009.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Skywork-o1. 2024. Skywork-o1 open series. https://huggingface.co/Skywork.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. 2024. Trial and error: Exploration-based trajectory optimization of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7584–7600, Bangkok, Thailand. Association for Computational Linguistics.

Qwen Team. 2024. Qwq: Reflect deeply on the boundaries of the unknown.

Yuxuan Tong, Xiwen Zhang, Rui Wang, Ruidong Wu, and Junxian He. 2024. Dart-math: Difficulty-aware rejection tuning for mathematical problem-solving. *CoRR*, abs/2407.13690.

Ke Wang, Houxing Ren, Aojun Zhou, Zimu Lu, Sichun Luo, Weikang Shi, Renrui Zhang, Linqi Song, Mingjie Zhan, and Hongsheng Li. 2024. Mathcoder: Seamless code integration in llms for enhanced mathematical reasoning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

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, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024a. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024b. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2024. Mammoth: Building math generalist models through hybrid instruction tuning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

A Appendix

A.1 Training Configuration

All instruction fine-tuning experiments are conducted on 8 NVIDIA A800 GPUs. We train the model using the AdamW optimizer for 2 epochs with a batch size of 16, a learning rate of 2×10^{-5}, a 3% warm-up ratio, a weight decay of 0.1, and gradient clipping with a maximum norm of 1.0.

A.2 Socratic Principles and Examples

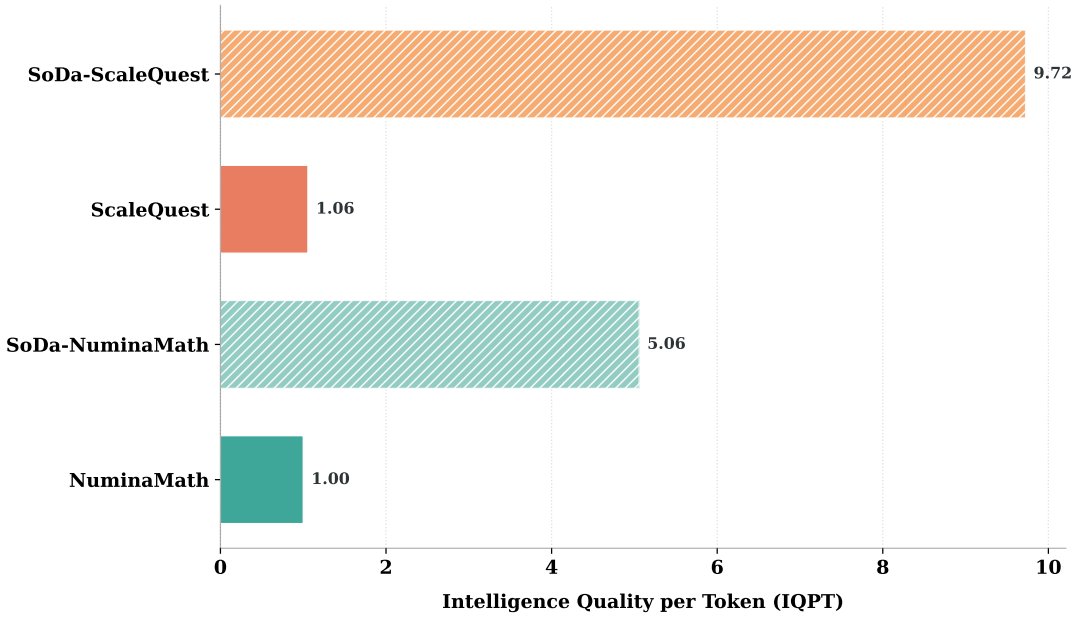

Figure 4: Intelligence Quality per Token (IQPT) of different datasets in fine-tuning Qwen2-Math-7B. IQPT is calculated as *average performance* divided by *number of tokens*, with the IQPT of NuminaMath normalized to 1 as the baseline.

Principles	Examples
Clarification Questions	What do you mean by [specific term]? Can you explain that further?
Probing Assumptions	What are you assuming here? Why do you think that assumption holds true?
Probing Evidence & Reasons	What evidence supports your conclusion? Are there alternative explanations?
Exploring Alternative Perspectives	What is another way to look at this problem? How might someone with a different viewpoint approach this?
Examining Implications & Consequences	What are the implications of this conclusion? What might be the consequences if this is implemented?
Questioning the Question	Is the question properly framed, or is there a better way to ask it? What are we missing by focusing on this question?
Encouraging Reflection	How has your understanding changed during this discussion? What questions remain unanswered?
Challenging Logic	Does this reasoning follow logically? Can you identify any contradictions?
Exploring Causes & Effects	What are the root causes of this issue? How does one factor influence another?
Summarizing & Synthesizing	Can you summarize the main points? Can you develop a general principle from these specifics?

Table 9: Basic principles and examples.