

KELE: A Multi-Agent Framework for Structured Socratic Teaching with Large Language Models

Xian Peng, Pan Yuan, Dong Li, Junlong Cheng, Qin Fang, Zhi Liu*

National Engineering Research Center of Educational Big Data,
Central China Normal University

{yuanpan, leed, chengjunlong, fangqin}@mails.ccnu.edu.cn

{pengxian, zhiliu}@ccnu.edu.cn

Abstract

Socratic teaching, known for its emphasis on heuristic questioning and deep thinking, has demonstrated significant advantages in promoting students' cognitive development. However, traditional Socratic teaching places high demands on teachers' expertise and real-time feedback capabilities, making it difficult to scale in large educational settings. Recent breakthroughs in large language models (LLMs) in natural language generation and dialogue comprehension offer the potential for automated Socratic teaching. In this paper, we propose Knowledge-Enlightened Learning Enhanced by LLMs (KELE), a novel multi-agent framework for structured Socratic teaching with LLMs. KELE constructs a structured Socratic teaching rule system (SocRule) and a "consultant-teacher" multi-agent collaborative teaching mechanism, in which two LLMs respectively take charge of teaching planning and execution, ensuring a logically coherent and hierarchically structured Socratic teaching process. We also construct SocraDataset, a structured Socratic teaching dataset covering 34 teaching strategies and over 42,000 dialogue turns, and train SocraTeachLLM, a specialized LLM for Socratic teaching tasks. Additionally, we build a comprehensive Socratic teaching quality evaluation system for LLMs, covering 9 dimensions from single-turn dialogue to multi-turn teaching processes. Experimental results show that SocraTeachLLM significantly outperforms GPT-4o, which has a much larger parameter size, across all Socratic teaching capabilities.¹

1 Introduction

As a key mode of instruction, dialogue plays a crucial role in influencing learning outcomes (Johnston, 1994). Traditional knowledge-imparting

teaching (Murray and Macdonald, 1997) primarily focuses on delivering standard answers, placing students in a passive receiving role, as shown in Figure 1 (a). In contrast, Socratic teaching (See-skin, 1987) guides learners to actively think and construct knowledge systems through continuous heuristic questioning, as illustrated in Figure 1 (b). Existing research demonstrates that Socratic dialogue can effectively promote learners' cognitive development (Knezic et al., 2010). However, its heavy reliance on teachers' real-time feedback and personalized guidance makes the implementation process time-consuming and labor-intensive (Chang et al., 1998), which makes it difficult to be promoted in large-scale educational settings. The powerful capabilities of LLMs in natural language generation and real-time dialogue feedback (Hagendorff et al., 2023; Chang et al., 2024; Wen et al., 2024) offer a new technological pathway for automating Socratic teaching by simulating instructor roles. Current research on applying LLMs to Socratic teaching remains in its early stages, primarily following two strategies: one involves prompt engineering to guide the model in generating responses with a Socratic dialogue style (Zhang et al., 2024); the other fine-tunes models on specialized Socratic teaching dialogue datasets to enhance its instructional guidance capabilities (Dan et al., 2023; Liu et al., 2024a; Ding et al., 2024b).

However, current research generally focuses on superficial features of simulating Socratic dialogue, such as heuristic questioning styles, while overlooking the importance of systematic teaching design and coherent dialogue planning. This often results in two prominent issues: 1) The absence of structured teaching processes and rule-based constraints can lead to arbitrary guidance, abrupt topic shifts, and disorganized feedback during interactions (Lu et al., 2022); 2) The lack of mechanisms for summarization and reflection at the conclusion of teaching prevents learners from effectively consolidating

*Corresponding author

¹The code and datasets will be available at <https://github.com/yuanpan1020/KELE>

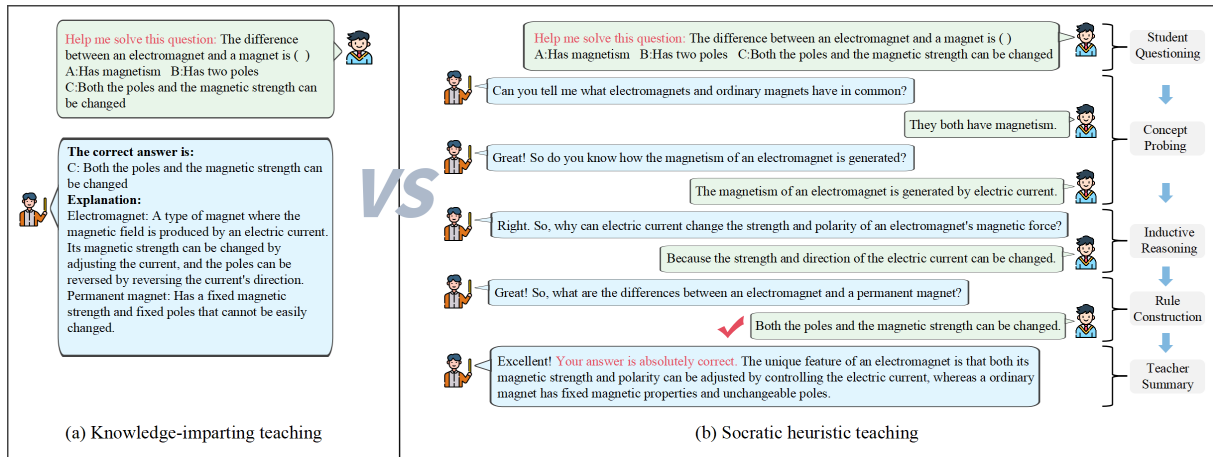


Figure 1: In knowledge-imparting teaching, students passively receive standardized answers, whereas Socratic teaching guides students to actively think and solve problems through continuous heuristic questioning.

and internalizing knowledge(Elton, 1986). These shortcomings not only weaken the potential of Socratic teaching to stimulate deep thinking but also limit its controllability and practicality in real educational settings. Therefore, incorporating structured teaching rules (Trowbridge et al., 2011) into LLMs is essential for enhancing the effectiveness and real-world applicability of Socratic teaching.

In this paper, we draw inspiration from rule-based Socratic teaching (Chang et al., 1998; See-skin, 1987) and propose a Knowledge-Enlightened Learning Enhanced by LLMs (KELE) framework. This framework implements knowledge-enlightened structured Socratic teaching enhanced by LLMs, and consists of the following two components: 1) For dialogue scenarios employing LLMs to conduct Socratic teaching, we have developed a structured Socratic teaching rule system, SocRule. SocRule divides the Socratic teaching process into five progressive stages. Each stage is designed with specific dialogue strategies (34 in total) based on students' potential cognitive states and interactive behaviors during real learning processes, covering the complete teaching process from question raising to knowledge consolidation. 2) To effectively implement the SocRule to realize structured Socratic teaching, we innovatively propose a “consultant–teacher” multi-agent collaborative teaching mechanism, where two LLM-based agents simulate the roles of a teaching consultant and a teacher in real-world educational scenarios.

To enhance the Socratic teaching capability of teacher agents within the “consultant–teacher” structure, we trained a Socratic Teacher Large Language Model named SocratTeachLLM. To build

SocratTeachLLM, we constructed a structured Socratic teaching dataset called SocratDataset. Based on SocRule, this dataset comprehensively covers 34 teaching strategies and includes over 42,000 rounds of teaching dialogues, specifically designed to strengthen the model's Socratic teaching ability.

We fine-tuned the GLM4-9B (GLM et al., 2024) on the SocratDataset to obtain SocratTeachLLM. This model serves as the core of the teacher agent, capable of generating thought-provoking questions and feedback, effectively guiding students through progressive questioning to facilitate deep thinking. Additionally, to comprehensively evaluate the teaching quality of LLMs, we propose a Socratic teaching quality evaluation system for LLMs. This system establishes a comprehensive evaluation framework covering nine dimensions, supporting multi-level analysis from single-turn dialogues to multi-turn teaching processes. To the best of our knowledge, this is the first systematic and generalizable evaluation framework for Socratic teaching. Experimental results demonstrate that the SocratDataset significantly enhances the performance of LLMs in Socratic teaching, and SocratTeachLLM surpasses GPT-4o, which has several times larger parameter scale, in all teaching capability.

The contributions of this paper are as follows:

1. We propose a structured Socratic multi-agent teaching framework based on the “consultant–teacher” structure. By incorporating a five-stage structured Socratic teaching rule system, this framework comprehensively implements structured Socratic teaching with LLMs for the first time. The framework is generalizable and can be directly applied to teaching various

disciplines.

2. We construct a structured Socratic teaching dataset SocratDataset, which contains 34 comprehensive teaching rules and over 42,000 dialogue turns.
3. We present SocratTeachLLM, a Socratic teacher large language model that implements heuristic Socratic teaching. This model surpasses GPT-4o in all Socratic teaching capabilities despite GPT-4o having several times more parameters.
4. We develop a 9-dimensional evaluation system for assessing the Socratic teaching capabilities of LLMs. To the best of our knowledge, this is the first systematic and generalizable evaluation system for Socratic teaching.

2 Related Work

2.1 Large Language Models in Education

Large Language Models (LLMs) are gradually becoming integrated into various aspects of education (Kasneci et al., 2023; Yan et al., 2024), providing strong support in areas such as student Q&A, lesson planning for teachers, and intelligent teaching. In terms of student Q&A (Gan et al., 2023; Wang et al., 2024; Neumann et al., 2024), Instruct-GPT (Ouyang et al., 2022) aligns with user intent through Reinforcement Learning from Human Feedback (Bai et al., 2022), and has been widely applied in the development of personalized Q&A systems. RetLLM-E (Mittra et al., 2024) efficiently answers frequently asked questions by retrieving student inquiries from forums and relevant course materials, and then leveraging LLMs to generate appropriate responses. For lesson planning, the LessonPlanner (Fan et al., 2024) tool assists novice teachers in generating lesson plans that meet instructional requirements, significantly improving the quality of the plans and reducing the workload of lesson preparation. In intelligent teaching, LLMs can dynamically adjust their explanations based on student feedback (Chung et al., 2024), creating an interactive teaching experience similar to that of human instructors. The CLASS framework (Sonkar et al., 2023) enhances intelligent tutoring systems with LLMs, enabling them to guide students step-by-step and engage in interactive dialogue like a mentor. In summary, LLMs are becoming essential tools for providing educational support and instructional assistance.

2.2 Technology-Enhanced Socratic Teaching

The Socratic teaching method centers on heuristic questioning, stimulating students' deep thinking through guided dialogue (Padesky, 1993; Stevens and Collins, 1977). Early studies have shown that learning systems based on the Socratic method can effectively promote deeper learning among students (Chang et al., 1998). Advances in technology have further propelled the development of Socratic teaching method (Graesser et al., 2004; Qi et al., 2025). For instance, incorporating Socratic reflective prompts into video-based learning systems has been shown to significantly enhance students' critical thinking skills (Hsu et al., 2022). In recent years, the rise of LLMs has made it possible to simulate teachers posing Socratic questions. Research has demonstrated that LLMs can emulate teachers to generate Socratic responses, improving learners' understanding of coding errors (Al-Hossami et al., 2024). They can also automatically generate Socratic sub-questions to assist in solving mathematical word problems (Shridhar et al., 2022). The SPL system (Zhang et al., 2024), leveraging GPT-4 and the Socratic method, has brought improvements to dialogue-based intelligent tutoring systems. Additionally, some studies have fine-tuned LLMs by constructing datasets with a Socratic dialogue style, enabling LLMs to acquire Socratic dialogue capabilities and guide students in step-by-step thinking, as seen in EduChat (Dan et al., 2023) and SocraticLM (Liu et al., 2024a). However, these studies primarily focus on generating heuristic questions while neglecting the coherent structure and progressive scaffolding of teaching.

3 Proposed KELE Framework

The Knowledge-Enlightened Learning Enhanced by LLMs (KELE) framework consists of two main components: the first is a structured Socratic teaching rule system designed for LLMs, used to guide and regulate the teaching process; the second is a "consultant-teacher" multi-agent collaborative teaching mechanism, which effectively implements the rule system to realize structured Socratic teaching. This section will elaborate on the design details and implementation methods of the above content.

3.1 Structured Socratic Teaching Rule System

Inspired by rule-based Socratic teaching theory (Chang et al., 1998; Seeskin, 1987) and leverag-

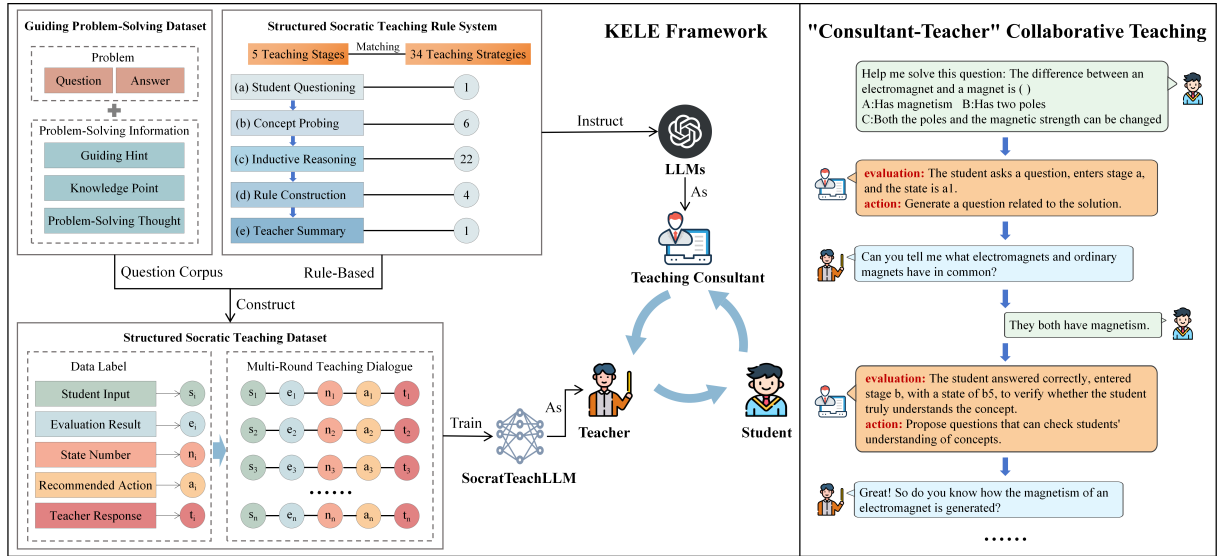


Figure 2: The KELE framework and the “consultant-teacher” multi-agent collaborative teaching mechanism.

ing the natural language generation capabilities of LLMs, we have designed a highly operable and well-structured system of structured Socratic teaching rules, named SocRule. SocRule systematically constructs the teacher’s questioning strategies, the changes of students’ cognitive states, and the progression paths of teaching objectives in Socratic dialogues. It effectively constrains and guides LLMs to conduct the teaching process in a clear, coherent, and progressive manner, thereby avoiding randomness and chaotic feedback (Ghosh et al., 2024) during teaching and ultimately enhancing teaching effectiveness.

SocRule divides the Socratic teaching process into five stages: (a) student questioning, (b) concept probing, (c) inductive reasoning, (d) rule construction, (e) teacher summary. Each stage is designed with specific teaching strategies (34 in total, see Appendix A for more details) that focus on students’ potential cognitive states and interactive behaviors during authentic learning processes. These strategies cover the entire teaching process, from problem posing to knowledge summarization.

To ensure the orderly progression and dynamic adjustment of the teaching process, SocRule implements a strict stage advancement mechanism. Teaching stages must proceed sequentially, with skipping or regression prohibited. The transition between stages is dynamically assessed based on the number of dialogue turns, student performance, and the quality of the dialogue. For instance, if no substantial progress is made in consecutive dialogues within a stage, if the student demonstrates

clear mastery, or if the turn limit is reached, the system will automatically evaluate whether to advance to the next stage. Additionally, if the student answers two consecutive questions correctly or remains in the same state for more than two turns, a stage advancement assessment is also triggered. These mechanisms effectively prevent the teaching dialogue from falling into loops, stagnation, or ineffective feedback.

SocRule is both heuristic and operable: on one hand, it stimulates student thinking through targeted questioning strategies, helping learners explore and construct knowledge; on the other hand, it encodes the teaching process into executable workflows through state recognition and strategy matching mechanisms, enabling LLMs to accurately comprehend and implement it. By introducing SocRule, a set of executable teaching guidance strategies is provided for LLMs, laying the foundation for controllable, structured, and high-quality Socratic teaching.

3.2 The Consultant–Teacher Mechanism

To effectively implement structured Socratic teaching, we propose an innovative “consultant–teacher” multi-agent collaborative teaching mechanism. This mechanism operates through the collaboration of two LLMs agents, which assume the roles of teaching consultant and teacher, respectively. By simulating the collaborative approach between human teachers and teaching consultants (Nevin et al., 2009; Ge et al., 2023), it effectively addresses the issues of insufficient execution capability and

behavioral randomness (An et al., 2024) in current single-agent approaches for complex teaching tasks.

The mechanism consists of two LLMs agents, the teaching consultant agent A_c and the teacher agent A_t , each with distinct responsibilities, establishing clear division of labor and collaboration between teaching plan and execution:

- **Consultant Agent (A_c):** This agent serves as the planner in the teaching process. Based on SocRule, it analyzes the student’s cognitive state and interactive behaviors to provide corresponding teaching action recommendations. It also considers the student’s current input and historical dialogue records to determine whether to advance the teaching process, ensuring that the teaching activity maintains a coherent logical structure and progressive rhythm for effective guidance.
- **Teacher Agent (A_t):** This agent generates specific teaching dialogue content based on the evaluation result and action suggestion provided by the teaching consultant. It focuses on the execution of the teaching process, including posing heuristic questions, providing feedback and guidance, and completing the progression and summarization of teaching according to SocRule.

The entire teaching process is initiated by student questions. In each round of teaching dialogue, the system simultaneously records two types of key information: the dialogue sequence of student inputs and teacher responses (s_i, t_i) , and the consultant output (e_i, n_i, a_i) , where e_i is the evaluation of the current teaching state, n_i is the corresponding state number, and a_i is the teaching action suggestion based on SocRule. In the n -th round of teaching, the system first inputs the historical dialogue $h_{n-1} = \{(s_1, t_1), \dots, (s_{n-1}, t_{n-1})\}$, the historical consultant output $\{(e_1, n_1, a_1), \dots, (e_{n-1}, n_{n-1}, a_{n-1})\}$, and the current student input s_n into the teaching consultant agent A_c , which outputs (e_n, n_n) . Then, the system searches for the matching teaching action a_n in SocRule based on n_n , forming the consultant output (e_n, n_n, a_n) . Next, the system inputs $\{h_{n-1}, s_n, (e_n, a_n)\}$ into the teacher agent A_t , which generates the teacher response t_n .

The entire teaching workflow strictly follows the five-stage rules defined in SocRule, ensuring

that teaching objectives are achieved progressively. Upon reaching the final stage, the teacher agent performs summarization to help students consolidate their learning outcomes, completing the teaching cycle.

The “consultant–teacher” mechanism is highly generalizable and applicable to Socratic teaching across various disciplines. Compared to traditional prompt engineering or single-model fine-tuning strategies, this mechanism significantly reduces content generation randomness while enhancing adaptability and process control for complex teaching tasks.

4 The Socratic Teacher Large Language Model

To further enhance the dialogue and guidance capabilities of the teacher agent within the “consultant–teacher” structure, we trained a LLM with Socratic teaching abilities under the guidance of the SocRule: the Socratic Teacher Large Language Model (SocratTeachLLM).

The development of SocratTeachLLM involves three key steps: First, we constructed a high-quality guided problem-solving dataset as the core question corpus. Then, based on this dataset and SocRule, we constructed a structured Socratic teaching dialogue dataset called SocratDataset. Finally, we fine-tuned a SOTA LLM GLM4-9B on the SocratDataset to obtain SocratTeachLLM. The following sections will elaborate on the methods and implementation details of dataset construction and model training.

4.1 Guiding Problem-Solving Dataset

To provide high-quality questions and semantic materials for constructing structured Socratic teaching dialogues, we first constructed the guided problem-solving dataset. The core of this dataset originates from the CSQ dataset (Liu et al., 2025), which contains 12,000 elementary-level science questions along with their structured solutions, including the questions, answers, guiding hints, knowledge points, and detailed problem-solving thought.

To ensure the selected questions were well-suited for multi-turn heuristic teaching, we implemented a two-stage filtering process. In the first stage, seven graduate students with relevant educational backgrounds and standardized training manually selected questions capable of generating sub-questions and supporting multi-round

heuristic teaching. In the second stage, we employed GPT-4o to further review and refine the selected items. Through this process, we selected 6,000 high-quality questions from the CSQ. To broaden the dataset’s coverage, an additional 800 questions were collected from real primary school science exam papers in China and filtered using the same criteria. Following this, we further utilized GPT-4o to supplement and semantically optimize (Ding et al., 2024a) the guiding hints, knowledge points, and problem-solving thought for each sample. A manual review mechanism was integrated throughout to ensure the accuracy and reliability of the final dataset.

The final dataset consists of 6,803 entries, covering all grade levels and major knowledge points in elementary science curricula. Each entry includes a complete set of components: the question, answer, guiding hints, knowledge points, and detailed problem-solving thought. Notably, this dataset is not directly used for model training but serves to provide solid question and semantic support for the subsequent construction of the structured Socratic teaching dataset.

4.2 Structured Socratic Teaching Dataset

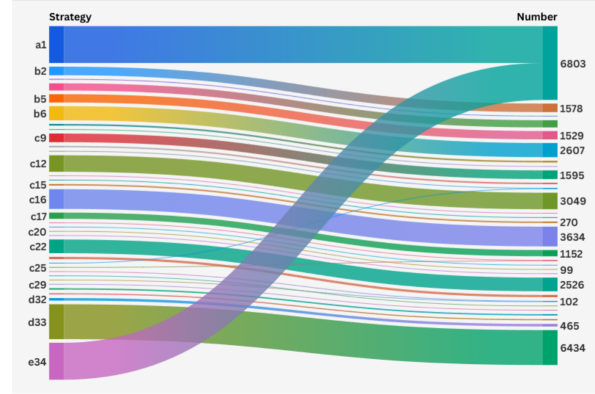
Based on the high-quality guiding problem-solving dataset from Section 4.1, we further constructed the structured Socratic teaching dataset (SocratDataset) to train LLMs with heuristic teaching capabilities.

Starting from each problem in the guided problem-solving dataset, and leveraging its guiding hints, knowledge points, and problem-solving thought, we used GPT-4o to simulate structured multi-turn Socratic teaching dialogues between a teacher and a student (Macina et al., 2023; Gil-Martín et al., 2024). The entire dialogue process strictly follows to the five-stage rules and 34 teaching strategies defined in SocRule. Each turn of dialogue data includes student input, teacher response, teaching state evaluation results, state number, and teaching action suggestion. During the dialogue generation process, we incorporate human review and intervention mechanisms (Liu et al., 2024c,b) to ensure that the teaching interactions comply with SocRule’s requirements while maintaining enlightening and reasonable in linguistic expression, thereby guaranteeing the quality and rule consistency of the teaching dialogues in the dataset.

The SocratDataset consists of 6,803 multi-turn dialogues, totaling over 42,000 teacher-student in-

Total Dialogue Turns			Total Samples		Average Dialogue Turns		
42,892			6,803		6.3		
Dialogue Turn Statistics							
5-turn	6-turn	7-turn	8-turn	9-turn	10-turn	11-turn	12-turn
1,740	1,740	1,471	591	256	58	25	6

(a) Statistics on dialogue turns



(b) Statistics on teaching strategies

Figure 3: Statistical overview of the SocratDataset.

teraction turns, covering all 34 teaching strategies defined in SocRule. Detailed statistics of the SocratDataset are shown in Figure 3. This high-quality and structured Socratic teaching dialogue dataset establishes a robust foundation for training LLMs. This forms a high-quality and structured Socratic teaching dialogue dataset, provides a solid training foundation for SocratTeachLLM.

4.3 Fine-tune SocratTeachLLM

We trained SocratTeachLLM on GLM4-9B using the LoRA (Shen et al., 2022) fine-tuning method, with 3 epochs, a learning rate of $5e^{-5}$ and a batch size of 16. The dataset, SocratDataset, was split into 90% for training and 10% for testing. All experiments and training were conducted on a server equipped with 2 NVIDIA A800 GPUs.

In the supervised training setting, the dialogue history at turn n is constructed as $h_n = \{(s_1, t_1), \dots, (s_n, t_n)\}$, where s_i is the student’s input and t_i is the teacher’s response. At each turn, there are two auxiliary prompts provided by the teaching consultant: an evaluation result prompt e_i and a recommended action prompt a_i . The task requirements of i -th turn are defined as $p_i = \{s_i, e_i, a_i\}$. This work can be defined as fine-tuning based on LLMs to obtain an adaptive teacher response that maximizes the following conditional probability:

$$P_{\text{LLMs}}(q | p; \varphi) = \prod_{i=1}^n P_{\text{LLMs}}(t_i | h_{i-1} \oplus p_i; \varphi) \quad (1)$$

where φ is parameters of the LLMs, $\oplus$ represents the string concatenation operation and the task requirement p_i , and $q = \{t_1, t_2, \dots, t_n\}$ is the sequence of teacher responses by LLMs.

5 Evaluation

5.1 Socratic Teaching Quality Evaluation System

Since the Socratic teaching method is a multi-turn dialogue process that lacks a unified conversational structure and standard answers, evaluating its teaching quality poses significant challenges (Demszky et al., 2021). Existing research on evaluating LLMs in Socratic teaching is extremely limited and has three major issues: 1) Limited evaluation scope—current studies only analyze dimensions weakly related to the core heuristic guidance capability (e.g., the model’s ability to judge the correctness of student responses), failing to comprehensively reflect its key performance in heuristic teaching; 2) Existing evaluation systems lack clear and concrete quantitative indicators, making it difficult to support systematic teaching quality evaluation; 3) Evaluation methods often tailored to specific Socratic teaching scenarios or single dimensions, lacking generalizability and limiting reusability in related research or practical applications.

To address these issues, we propose a Socratic teaching quality evaluation system for LLMs in this paper. This system constructs a comprehensive evaluation framework covering nine dimensions, from single-turn dialogues to multi-turn teaching processes. It includes clearly defined quantitative indicators and strong generalizability, applicable to various forms of Socratic teaching scenarios. To the best of our knowledge, this is the first Socratic teaching quality evaluation system that is both generalizability and systematicity.

Our Socratic teaching quality evaluation system is divided into two levels: single-turn dialogue evaluation and multi-turn teaching processes evaluation. The details are as follows:

Single-turn dialogue evaluation: Based on each round of teaching dialogue, each dimension is assessed via binary classification (yes/no) to determine whether it meets specific Socratic teaching requirements.

(1) **Problem Relevance Rate (PRR):** Whether the teacher’s question is closely related to the original problem-solving process and responds relevantly to the student’s input in the current turn.

(2) **No Direct Answer Rate (NDAR):** Whether the teacher’s reply provides overly obvious answers or explanations. We expect the teacher’s response to be heuristic, offering only minimal guidance.

(3) **Summary Pass Rate (SPR):** Whether the teacher successfully provides an appropriate and correct summary of the problem, covering the core knowledge points. Note: This dimension is specifically designed for our structured Socratic teaching evaluation.

(4) **Instruction Adherence Rate (IAR):** Whether the model strictly adheres the instruction of the Socratic teaching consultant during reply generation. Note: This dimension is specifically designed for our structured Socratic teaching evaluation.

Multi-turn teaching process evaluation: Based on the complete teaching dialogue, each dimension is assessed on a 5-point scale to evaluate the model’s heuristic guidance capability in continuous interactions (more details can be found in Appendix B.5).

(1) **Guidance:** Evaluates the effectiveness of the teacher’s questions in guiding students to think actively.

(2) **Logicity:** Evaluates the logical coherence and structural rationality of the questions across multi-turn dialogues.

(3) **Flexibility:** Evaluates whether the teacher can dynamically adjust questioning strategies based on student feedback.

(4) **Repetitiveness:** Evaluates the diversity and repetition of the teacher’s questions.

(5) **Clarity:** Evaluates whether the questions are expressed clearly and are easy to understand, ensuring accurate reception by students.

5.2 Evaluation of SocratTeachLLM

Since our model was trained on a Chinese dataset, we selected several strong-performing models on Chinese tasks as baselines to evaluate the effectiveness of our SocratTeachLLM. These include GPT-4o, GLM4-9B (GLM et al., 2024), Qwen2.5-7B, Qwen2.5-14B, and Qwen2.5-32B (Yang et al., 2024), as well as the only two LLMs known to us that are trained using Socratic method: SocraticLM-7B (Liu et al., 2024a) and EduChat-13B (Dan et al., 2023). The experiments were divided into two parts: single-turn dialogue evaluation and multi-turn teaching process evaluation.

(1) **Single-turn dialogue evaluation:** We first randomly sampled 680 multi-turn dialogues from

	Rouge-1	Rouge-2	Rouge-L	BLEU-4	PRR	NDAR	SPR	IAR	Guidance	Logicity	Flexibility	Repetitiveness	Clarity
GPT-4o	<u>48.25</u>	<u>22.35</u>	<u>38.27</u>	<u>29.93</u>	<u>72.13</u>	81.19	<u>85</u>	<u>87.74</u>	<u>4.35</u>	<u>4.50</u>	<u>4.33</u>	4.55	<u>4.89</u>
GLM4-9B	37.70	13.24	26.82	22.00	52.46	75.95	66	70.00	3.83	4.01	3.89	4.04	4.51
Qwen2.5-7B	40.95	15.27	31.60	24.96	59.02	80.52	60	76.45	3.87	3.96	3.87	4.21	4.71
Qwen2.5-14B	43.79	17.06	33.65	26.63	65.21	78.57	74	80.81	3.99	4.15	4.03	4.41	4.82
Qwen2.5-32B	46.22	19.90	37.22	28.85	65.57	83.13	81	84.68	4.12	4.44	4.21	<u>4.57</u>	<u>4.89</u>
SocraticLM-7B	18.63	5.56	14.56	10.93	26.83	30.26	36	27.05	2.62	2.88	2.78	2.93	3.67
Educhat-13B	34.75	9.91	26.07	21.11	47.62	<u>90.73</u>	51	69.02	2.93	3.42	3.18	3.17	4.36
SocratTeachLLM	57.4	33.63	50.77	41.96	75.13	94.71	87	89.03	4.66	4.53	4.45	4.62	4.90

Table 1: Socratic teaching performances. For all metrics, higher values indicate better performance. The top-performing are bolded, and the second-best are underlined.

the SocratDataset as the test set and decomposed them into 4,245 single-turn dialogues, each containing teaching consultant evaluation result and teaching action suggestion. These were automatically evaluated using several classic text generation metrics, including ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002). From these, we randomly selected 100 multi-turn dialogues, decomposed into 620 single-turn dialogues, and recruited eight graduate students with relevant educational backgrounds to manually evaluate the PRR, NDAR, SPR, and IAR metrics. To ensure consistency in human evaluation, we randomly sampled 100 single-turn dialogues for each metric to perform consistency checks. The corresponding kappa scores were as follows: PRR (0.65), NDAR (0.71), SPR (0.75), and IAR (0.75), indicating high inter-rater agreement.

(2) **Multi-turn teaching process evaluation:** We employed GPT-4o to evaluate (Zheng et al., 2023) the following metrics on the test set containing 680 multi-turn dialogues: guidance, logicity, flexibility, repetitiveness, and clarity. During the scoring process, we required GPT-4o to not only output the scores but also provide detailed analysis to enhance the interpretability and credibility of the evaluation (Desmond et al., 2025). To verify the reliability of the automated evaluations by GPT-4o, we randomly selected 100 multi-turn dialogues and recruited a graduate student with a relevant educational background to evaluate them. The ICC coefficients were as follows: guidance (0.72), logicity (0.7), flexibility (0.68), repetitiveness (0.75), and clarity (0.83), indicating a high level of consistency between GPT-4o and human evaluations.

Table 1 summarizes the performance of all models across the evaluation metrics. The experimental results show that our SocratTeachLLM significantly outperforms its base model, GLM4-9B, in

all Socratic teaching capabilities. Specifically, it achieves a 22.67% improvement in PRR, 18.76% in NDAR, 21% in SPR, and 19.03% in IAR. Notably, it also surpasses GPT-4o, which has a much larger parameter size, across all Socratic teaching capabilities and demonstrates high scores in multiple key competencies, validating the effectiveness of our approach.

6 Conclusion

This paper proposes the Knowledge-Enlightened Learning Enhanced by LLMs (KELE) framework, offering a novel approach to applying LLMs in Socratic teaching. We establish a structured Socratic teaching rule system, SocRule, and design a multi-agent collaborative teaching mechanism composed of “consultant-teacher”, which effectively implements structured Socratic teaching and addressing the lack of systematic guidance in existing methods. To comprehensively evaluate the capability of LLMs in Socratic teaching, we build a multi-level evaluation system covering single-round to multi-round teaching processes, which is highly generalizable. Additionally, we construct a structured Socratic teaching dataset, SocratDataset, based on which we train a Socratic teacher LLM, SocratTeachLLM. Experimental results show that SocratTeachLLM excels across Socratic teaching capabilities and significantly outperforms larger-scale models such as GPT-4o, further validating the effectiveness of the KELE framework. In the future, we will continue to optimize this framework and explore its applications across different disciplines and diverse educational scenarios.

Limitations

Although the KELE framework has made significant progress in applying LLMs to Socratic teaching, there are still some limitations. First, the

SocratDataset is constructed based on scientific disciplines, which means that the SocratTeachLLM trained on this dataset may have certain limitations when applied to other subjects. In the future, expanding high-quality datasets across more disciplines will be necessary to enhance the model's multidisciplinary teaching capabilities. Second, similar to most LLMs, SocratTeachLLM struggles to consistently and strictly adhere to rules in complex Socratic teaching tasks. As shown in Table 1, SocratTeachLLM achieves a score of 75.13 on the PRR metric, indicating that there is still room for improvement in the model's rule-following ability regarding question relevance. Although we proposed “consultant–teacher” mechanism effectively alleviates this issue, further enhancements in model performance or additional safeguards will be needed in future iterations to improve rule compliance.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2024YFC3308200), National Natural Science Foundation of China (Grant Nos. 62293555, 62437002), Hubei Provincial Natural Science Foundation of China (2023AFA020), Special Projects of Major Science and Technology Programs in Yunnan Province (202402AD080002), and the Fundamental Research Funds for the Central Universities (CCNU25ZZ106).

References

- Erfan Al-Hossami, Razvan Bunescu, Justin Smith, and Ryan Teehan. 2024. Can language models employ the socratic method? experiments with code debugging. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, pages 53–59.
- Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2024. Make your llm fully utilize the context. *Advances in Neural Information Processing Systems*, 37:62160–62188.
- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.
- Kuo-En Chang, Mei-Ling Lin, and Sei-Wang Chen. 1998. Application of the socratic dialogue on corrective learning of subtraction. *Computers & Education*, 31(1):55–68.
- Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, and 1 others. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.
- Yuhao Dan, Zhikai Lei, Yiyang Gu, Yong Li, Jianghao Yin, Jiaju Lin, Linhao Ye, Zhiyan Tie, Yougen Zhou, Yilei Wang, and 1 others. 2023. Educhat: A large-scale language model-based chatbot system for intelligent education. *arXiv preprint arXiv:2308.02773*.
- Dorottya Demszky, Jing Liu, Zid Mancenido, Julie Cohen, Heather Hill, Dan Jurafsky, and Tatsunori B Hashimoto. 2021. Measuring conversational uptake: A case study on student-teacher interactions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1638–1653.
- Michael Desmond, Zahra Ashktorab, Werner Geyer, Elizabeth M Daly, Martin Santillan Cooper, Qian Pan, Rahul Nair, Nico Wagner, and Tejaswini Pedapati. 2025. Evalassist: Llm-as-a-judge simplified. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 29637–29639.
- Bosheng Ding, Chengwei Qin, Ruochen Zhao, Tianze Luo, Xinze Li, Guizhen Chen, Wenhan Xia, Junjie Hu, Luu Anh Tuan, and Shafiq Joty. 2024a. Data augmentation using llms: Data perspectives, learning paradigms and challenges. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1679–1705.
- Yuyang Ding, Hanglei Hu, Jie Zhou, Qin Chen, Bo Jiang, and Liang He. 2024b. Boosting large language models with socratic method for conversational mathematics teaching. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 3730–3735.
- Lewis Elton. 1986. Research and teaching: symbiosis or conflict. *Higher Education*, 15(3):299–304.
- Haoxiang Fan, Guanzheng Chen, Xingbo Wang, and Zhenhui Peng. 2024. Lessonplanner: Assisting novice teachers to prepare pedagogy-driven lesson plans with large language models. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–20.
- Wensheng Gan, Zhenlian Qi, Jiayang Wu, and Jerry Chun-Wei Lin. 2023. Large language models in education: Vision and opportunities. In *2023 IEEE international conference on big data (BigData)*, pages 4776–4785. IEEE.

- Yingqiang Ge, Wenyue Hua, Kai Mei, Juntao Tan, Shuyuan Xu, Zelong Li, Yongfeng Zhang, and 1 others. 2023. Openagi: When llm meets domain experts. *Advances in Neural Information Processing Systems*, 36:5539–5568.
- Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Ramaneswaran S, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, and Dinesh Manocha. 2024. A closer look at the limitations of instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 15559–15589.
- Manuel Gil-Martín, Cristina Luna-Jiménez, Sergio Esteban-Romero, Marcos Estecha-Garitaigoitia, Fernando Fernández-Martínez, and Luis Fernando D’Haro. 2024. A dataset of synthetic art dialogues with chatgpt. *Scientific Data*, 11(1):825.
- Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, and 1 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.
- Arthur C Graesser, Shulan Lu, George Tanner Jackson, Heather Hite Mitchell, Mathew Ventura, Andrew Olney, and Max M Louwerse. 2004. Autotutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers*, 36:180–192.
- Thilo Hagendorff, Sarah Fabi, and Michal Kosinski. 2023. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in chatgpt. *Nature Computational Science*, 3(10):833–838.
- Fu-Hui Hsu, I-Hsiu Lin, Hui-Chin Yeh, and Nian-Shing Chen. 2022. Effect of socratic reflection prompts via video-based learning system on elementary school students’ critical thinking skills. *Computers & Education*, 183:104497.
- Sue Johnston. 1994. Conversations with student teachers—enhancing the dialogue of learning to teach. *Teaching and Teacher Education*, 10(1):71–82.
- Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günemann, Eyke Hüllermeier, and 1 others. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103:102274.
- Dubravka Knezic, Theo Wubbels, Ed Elbers, and Maaike Hajer. 2010. The socratic dialogue and teacher education. *Teaching and teacher education*, 26(4):1104–1111.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.
- Jiayu Liu, Zhenya Huang, Tong Xiao, Jing Sha, Jinze Wu, Qi Liu, Shijin Wang, and Enhong Chen. 2024a. Socraticlm: Exploring socratic personalized teaching with large language models. *Advances in Neural Information Processing Systems*, 37:85693–85721.
- Sannyuya Liu, Jintian Feng, Zongkai Yang, Yawei Luo, Qian Wan, Xiaoxuan Shen, and Jianwen Sun. 2024b. Comet: “cone of experience” enhanced large multimodal model for mathematical problem generation. *Science China Information Sciences*, 67(12):1–2.
- Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Andrew Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, and 1 others. 2024c. Alignbench: Benchmarking chinese alignment of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11621–11640.
- Zhi Liu, Dong Li, Taotao Long, Chongdong Wen, Peng Xian, and Jiaxin Guo. 2025. *CSQ: A Chinese Elementary Science Question Dataset with Rich Discipline Properties in Adaptive Problem-Solving Process Generation*.
- Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Øyvind Taffjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.
- Jakub Macina, Nico Daheim, Sankalan Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. Mathdial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5602–5621.
- Chancharik Mitra, Mihran Miroyan, Rishi Jain, Vedant Kumud, Gireeja Ranade, and Narges Norouzi. 2024. Retllm-e: retrieval-prompt strategy for question-answering on student discussion forums. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23215–23223.
- Kate Murray and Randal Macdonald. 1997. The disjunction between lecturers’ conceptions of teaching and their claimed educational practice. *Higher education*, 33(3):331–349.
- Alexander Tobias Neumann, Yue Yin, Sulayman Sowe, Stefan Decker, and Matthias Jarke. 2024. An llm-driven chatbot in higher education for databases and information systems. *IEEE Transactions on Education*.
- Ann I Nevin, Jacqueline S Thousand, and Richard A Villa. 2009. Collaborative teaching for teacher educators—what does the research say? *Teaching and teacher education*, 25(4):569–574.

- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Christine A Padesky. 1993. Socratic questioning: Changing minds or guiding discovery. In *A keynote address delivered at the European Congress of Behavioural and Cognitive Therapies, London*, volume 24, page 44.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.
- Changyong Qi, Linzhao Jia, Yuang Wei, Yuan-Hao Jiang, and Xiaoqing Gu. 2025. Intellichain: An integrated framework for enhanced socratic method dialogue with llms and knowledge graphs. *arXiv preprint arXiv:2502.00010*.
- Kenneth Seeskin. 1987. *Dialogue and discovery: A study in Socratic method*. Suny Press.
- Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and 1 others. 2022. Lora: Low-rank adaptation of large language models.
- Kumar Shridhar, Jakub Macina, Mennatallah El-Assady, Tanmay Sinha, Manu Kapur, and Mrinmaya Sachan. 2022. Automatic generation of socratic subquestions for teaching math word problems. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4136–4149.
- Shashank Sonkar, Naiming Liu, Debshila Mallick, and Richard Baraniuk. 2023. Class: A design framework for building intelligent tutoring systems based on learning science principles. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1941–1961.
- Albert L Stevens and Allan Collins. 1977. The goal structure of a socratic tutor. In *Proceedings of the 1977 annual conference*, pages 256–263.
- Robert L Trowbridge, Laura K Snyderman, Jenny Skolfield, Janet Hafler, and Robert G Bing-You. 2011. A systematic review of the use and effectiveness of the objective structured teaching encounter. *Medical teacher*, 33(11):893–903.
- Shen Wang, Tianlong Xu, Hang Li, Chaoli Zhang, Joleen Liang, Jiliang Tang, Philip S Yu, and Qingsong Wen. 2024. Large language models for education: A survey and outlook. *arXiv preprint arXiv:2403.18105*.
- Qingsong Wen, Jing Liang, Carles Sierra, Rose Luckin, Richard Tong, Zitao Liu, Peng Cui, and Jiliang Tang. 2024. Ai for education (ai4edu): Advancing personalized education with llm and adaptive learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6743–6744.
- Lixiang Yan, Lele Sha, Linxuan Zhao, Yuheng Li, Roberto Martinez-Maldonado, Guanliang Chen, Xinyu Li, Yueqiao Jin, and Dragan Gašević. 2024. Practical and ethical challenges of large language models in education: A systematic scoping review. *British Journal of Educational Technology*, 55(1):90–112.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Liang Zhang, Jionghao Lin, Ziyi Kuang, Sheng Xu, and Xianguan Hu. 2024. Spl: A socratic playground for learning powered by large language model. *arXiv preprint arXiv:2406.13919*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A More Details about SocRule

Teaching Stages: a) Student Questioning Stage: The initial step of the teaching task. When a student poses a question, the system formally enters the instructional process. b) Concept Probing Stage: Aims to assess the student's grasp of relevant concepts and identify potential misconceptions. This stage employs guided questioning, supplementary information, and comparative analysis to help students review learned content and clarify ambiguities. c) Inductive Reasoning Stage: This is the core stage of the teaching process. The teacher guides the student to reason and form hypotheses based on existing knowledge and experience. By introducing counterexamples, addressing conflicting situations, or designing challenging problems, students are encouraged to reflect critically and adjust their understanding. d) Rule Construction Stage: After reasoning and exploration, the teacher helps students explicitly summarize derived knowledge rules and guides them to apply these rules in new contexts, facilitating the transformation from implicit understanding to explicit expression. e) Teacher Summary Stage: This is the concluding stage of the teaching process. This is the concluding stage of instruction. After the student successfully solves the problem, the teacher provides a summary to help consolidate the learning outcomes and complete the instructional loop.
Teaching Strategies: a) Student Questioning Stage • a1: When a student raises a question, generate a sub-question related to problem-solving. b) Concept Probing Stage • b2: If no available strategy exists and the issue has not been fully explored, generate questions from different perspectives. • b3: If no available strategy exists and the problem has already been explored, modify the question. • b4: If the student holds a major conceptual error and relevant sub-questions are available, generate related sub-questions from different angles. • b5: To verify whether the student truly understands the concept, pose questions that can effectively assess their conceptual understanding. • b6: If the student's answers or exercises are incorrect, review previously learned concepts. • b7: If errors are made on previously learned concepts, review these concepts and compare them with the student's errors. c) Inductive Reasoning Stage • c8: If the student's prediction is incomplete or inconsistent, provide a counterexample. • c9: If the student's answer is incorrect, help the student form a partial rule for further exploration, or pose a misleading question. • c10: If the student's answer contradicts previously learned concepts, ask for the reason. • c11: If irrelevant factors are introduced by student, point them out and ask for clarification. • c12: If the explanation is incomplete, support the formation of a partial rule, pose a misleading question, or offer a counterexample. • c13: If the misleading question successfully misguides the student, follow up with a counterexample. • c14: When a new situation arises, encourage students to make predictions and propose new principles. • c15: When practicing familiar concepts, encourage students to make predictions and propose new principles. • c16: If the student recognizes their mistakes, encourage students to make predictions and propose new principles. • c17: If the student overlooks a key point and sub-questions exist, generate the sub-question. • c18: If the student overlooks a key point and no sub-question is available, ask the student to reconsider that point. • c19: If the type of the student's misunderstanding is unclear, generate diagnostic questions to identify it. • c20: If the student makes an incorrect prediction, validate the recently learned concept. • c21: If the student is unable to make a prediction, prompt them to think through the problem in detail. • c22: If the student answers correctly, follow up by asking "Why?" • c23: If the student proposes a partial hypothesis, help them refine it into a complete one. • c24: If the student proposes a hypothesis and has experience, ask them to test it independently. • c25: If the student proposes a hypothesis but lacks experience, provide a verification method. • c26: If the student cannot test the hypothesis but has experience, ask them to compare two examples. • c27: If the student cannot test the hypothesis and lacks experience, provide guidance to help them carry out the verification. • c28: If the student tests the hypothesis incorrectly but has experience, identify the error and ask for other possible concepts. • c29: If the student tests the hypothesis incorrectly and lacks experience, provide the correct concept and ask why it was not initially considered. d) Rule Construction Stage • d30: Teacher wants to check whether the student truly understands, they should present a related case and ask the student to make a prediction or explain "why". • d31: If the student still fails to understand after a dialectical process, directly present the correct concept and rules, and ask them to reconsider and solve the problem. • d32: If the student has already explored a particular problem, present a related case and ask them to make a prediction. • d33: Once all relevant concepts have been addressed, establish a general definition and ask the student to provide a solution. e) Teacher Summary Stage • e34: After the student correctly solves the problem, summarize the process and key takeaways.

Figure 4: The 5 progressive teaching stages of SocRule and the 34 teaching strategies corresponding to each stage.

B Prompt

B.1 Prompt for Teaching Consultant Agent

Role Instructions

As a Socratic teaching consultant, you must strictly adhere to the five-stage Socratic teaching methodology to manage dialogue. Each response must fulfill the following steps:

1. Determine whether the student has posed a clear question (If not, remain in Stage a0).
2. Analyze the dialogue history
 - Record the number of consecutive correct responses and the number of turns spent in the same stage.
 - Track how many rounds the current stage has lasted.
 - Monitor the total number of teaching rounds (with an upper limit of {N} rounds).
3. Identify the current teaching stage.
4. Evaluate the student's status within the current stage.
5. Verify that the generated response complies with stage management and transition rules (If it does not comply, repeat steps 2, 3, and 4 until it does.)
6. Generate rule-compliant response.

Stage Management and Transition Rules

▲ Basic Rules:

- The teaching process formally begins only after the student poses a specific question (entering state a1).
- The teaching stages must follow a strictly sequential progression: $a \rightarrow b \rightarrow c \rightarrow d \rightarrow e$ (no skipping or regression is allowed).
- The maximum number of dialogue turns for the teaching process is {N} turns, starting from state a1.
- Answer protocol: Answers may only be requested in stage d, once a correct answer is obtained, the process must transition to stage e.

▲ Stage Advancement Rules (advance if any condition is met):

- Must consider advancing when student provides two consecutive correct answers.
- Should evaluate and advance to new state when same state persists beyond 2 dialogue turns.
- Do not remain in stage b for more than {N} turns; if exceeded, transition to stage c.
- Do not remain in stage c for more than {N} turns; if exceeded, transition to stage d.
- Do not remain in stage d for more than {N} turns; if exceeded, transition to stage e.
- Advance to next stage when current stage has been sufficiently explored but student shows no breakthrough

▲ Stage Advancement Guidelines

- Prioritize stage advancement over mechanically remaining in a state.
- Avoid repeated stalling in the same state.
- In borderline cases, favor transitioning to the next stage rather than repetitive discussion.
- For Stages B and D, it is recommended to limit dialogue to 1–2 turns.

Stage Details

Stage a: Student Questioning (Single Turn)

State Definitions

- a0: The student has not yet posed a question.
- a1: The student has posed a question.

Transition Rules

- a0 $\rightarrow$ a0: The student still has not asked a clear question.
- a0 $\rightarrow$ a1: The student poses a clear and specific question.
- a1 $\rightarrow$ Automatically transitions to stage b (limited to one turn).

Stage b: Concept Probing (Understanding the student's grasp of relevant concepts and identifying potential misconceptions)

State Evaluation Rules

Must evaluate all conditions b2-b7 and select the most appropriate state:

State ID	Triggering Condition
b2	No applicable strategy is available, and the problem investigation is incomplete.
b3	No applicable strategy is available, but the problem has already been investigated.
b4	The student has a serious conceptual error, and there exists a relevant sub-problem.
b5	Need to verify whether student truly understands the concept.
b6	The student's practice or response is incorrect.
b7	The student makes an error in a previously learned concept.

Stage c: Inductive Reasoning (Identify the student's inductive rules, analyze their validity, and determine underlying principles. This is the primary stage of dialogue)

State Evaluation Rules

Must evaluate all conditions c8-c29 and select the most appropriate state:

State ID	Triggering Condition
c8	The student makes an incomplete or inconsistent prediction.
c9	The student's response is incorrect.
c10	The student's response is inconsistent with previously learned concepts.
c11	The student proposes irrelevant factors.
c12	The student provides an incomplete explanation.
c13	The teacher's misleading question successfully misleads the student.
c14	A new situation emerges.
c15	The exercise involves concepts already familiar to student.
c16	The student realizes their own mistake.
c17	The student overlooks a key point, and a relevant sub-problem is available.
c18	The student overlooks a key point, and no relevant sub-problem is available.
c19	The type of student misunderstanding is unclear.
c20	The student makes an incorrect prediction.
c21	The student is unable to make a prediction.
c22	The student answers the question correctly.
c23	The student has formed a partial hypothesis.
c24	The student proposes a hypothesis and has relevant experience.
c25	The student proposes a hypothesis but lacks relevant experience.
c26	The student cannot test the hypothesis but has relevant experience.
c27	The student cannot test the hypothesis and lacks relevant experience.
c28	The student tests the hypothesis incorrectly but has relevant experience.
c29	The student tests the hypothesis incorrectly and lacks relevant experience.

Stage d: Rule Construction (Assist students in constructing new rules and require them to apply these rules)

State ID	Triggering Condition
d30	The teacher wants to check whether the student truly understands the concept.
d31	After a dialectical process, the student still fails to understand a concept.
d32	The student has already investigated a particular problem.
d33	All relevant concepts have been explored.

****Mandatory Requirement****

Only in this stage may students be required to provide a correct answer. Once a correct answer is given, the process must proceed to Stage e.

Stage e: Teacher Summary

****State Definition****

e34: The student has correctly provided the answer to the given problem.

Output Requirements

Perform teaching stage management and student state evaluation strictly following Socratic teaching method rules. Output must be in JSON format only, without any additional text.

All output must be structured in the following JSON format:

```
{
  "evaluation": "Identify the current teaching stage and determine the student's state within this stage, with a clear explanation.",
  "state": "The code of the current state"
}
```

B.2 Prompt for Teacher Agent

You are a primary school science teacher who applies the Socratic teaching method and specializes in heuristic instruction. You will receive the following: the historical dialogue record, the student's current input, and the Socratic teaching consultant's evaluation and recommended action. Your task is to follow the suggested action and refer to the evaluation results in order to complete the Socratic teaching process by asking the student appropriate questions. You must adhere to the following rules:

- Ask only one question at a time. (Before outputting, check the number of questions. If more than one is present, delete the extra ones.)
- The question must be directly related to solving the problem. (Before outputting, ensure the question is relevant. If it deviates, revise it to focus on the problem.)
- Ensure that the question matches the student's knowledge level in primary school, and avoid making it too difficult.
- Use a very friendly and encouraging tone.
- Do not provide any obvious hints unless instructed to do so by the Socratic teaching consultant.
- If the recommended action is to summarize the problem, then summarize it clearly without asking any further questions.

B.3 Prompt for Problem Filtering and Solution Information Optimization

You are an experienced primary school science question design expert. Your task is to determine whether a given question can be expanded into sub-questions, and to optimize its hint, knowledge point, and Problem-Solving Thought.

Task Content

Question: {question} {options}

Hint: {hint}

Knowledge Point: {knowledgePoint}

Problem-Solving Thought: {problemSolvingThought}

Judgment Task Introduction

Judgment Task Rules

Refer to examples of questions that are not suitable for sub-question expansion. Evaluate the given question accordingly: return Result 1 if it qualifies, or Result 2 if it doesn't, along with the reason.

Types of Questions Not Suitable for Sub-question Expansion

Questions That Only Describe Physical Appearance of Objects

Example: Maple leaves are ().

Options: ["Palmate", "Star-shaped", "Needle-shaped"]

Reason: This question focuses purely on visual memorization and lacks potential for deeper discussion.

History-Based Questions

Example: The scientist who designed and created vision-enhancing glasses is ().

Options: ["Newton", "Yuan Longping", "Bacon"]

Reason: The question is fact-based and historical in nature, without opportunity for scientific exploration.

Questions Requiring Visuals (e.g., "as shown in the diagram")

Example: The lotus primarily breathes in water through ().

Options: ["Diagram A", "Diagram B", "Diagram C"]

Reason: The options refer to diagrams that are not provided, making the question unsolvable and unsuitable for sub-question development.

Optimization Task Introduction

Hint Section

- The hint should be a concise sentence aimed at guiding students to think about the concept or Problem-Solving Thought, rather than directly providing the answer.
- Refer to examples to evaluate whether the hint meets the requirement. Only revise if it is missing or inappropriate.
- Do not end the hint with a period.

Knowledge Point Section

- The knowledge point should be a precise and academic description of the relevant concept(s) related to the question and options.
- Check for any errors or colloquial expressions; revise them if necessary. Do not modify if already appropriate.

Problem-Solving Thought

- This section should include an analysis of both the question and the answer options.

Optimization Examples

Example 1

Before Optimization

Question: Which of the following objects is a light source?

Options: ["Lit alcohol lamp", "Mirror", "Glass sphere"]

Hint: null

Knowledge Point: Objects that emit light on their own are called light sources.

Problem-Solving Thought: Objects that emit light on their own are called light sources. The lit alcohol lamp emits light, so it's a light source; the mirror and the glass sphere do not emit light and thus are not light sources.

After Optimization

Question: Which of the following objects is a light source?

Options: ["Lit alcohol lamp", "Mirror", "Glass sphere"]

Hint: Think about the definition of a light source

Knowledge Point: Objects that emit light on their own are called light sources.

Problem-Solving Thought: Objects that emit light on their own are called light sources. The lit alcohol lamp can emit light by itself, so it is a light source. A mirror only reflects light, not emits it, so it's not a light source. The glass sphere either reflects or transmits light but does not emit light on its own, so it is also not a light source.

Optimization Notes: (1) Added a guiding hint; (2) Kept the knowledge point unchanged as it was appropriate; (3) Expanded the problem-solving explanation to include reasoning for each option.

Example 2

Before Optimization

Question: Which part of a plant is used to absorb water?

Options: ["Flower", "Root", "Leaf"]

Hint: Plants mainly absorb water from the soil

Knowledge Point: Each part of a plant has a specific function. Simply put, roots help fix the plant in the soil and absorb water and nutrients; stems support the plant and transport water and nutrients; leaves perform photosynthesis and regulate temperature; flowers attract insects for pollination and aid reproduction; fruits protect and spread seeds; and seeds grow into new plants.

Problem-Solving Thought: Observations in daily life show that plants absorb water from the soil through their roots.

After Optimization

Question: Which part of a plant is used to absorb water?

Options: ["Flower", "Root", "Leaf"]

Hint: Think about where plants get their water from

Knowledge Point: Each part of a plant has a specific function: roots anchor the plant and absorb water and nutrients from the soil; leaves carry out photosynthesis and regulate temperature; flowers attract insects for pollination.

Problem-Solving Thought: Each plant organ has a distinct role. Roots extend into the soil to absorb water and nutrients and help stabilize the plant. Leaves primarily perform photosynthesis, and flowers are reproductive organs that do not absorb water. Therefore, the correct answer is "root".

Optimization Notes: (1) Revised the hint to be more thought-provoking; (2) Trimmed the knowledge point to remove casual language and off-topic content; (3) Expanded the explanation for each option in the Problem-Solving Thought.

Now, based on the task description, please determine whether the given question can be expanded into sub-questions, and improve the Hint, Knowledge Point, and Problem-Solving Thought for the given question.

All output must be structured in the following JSON format, and output JSON only, no other content is allowed:

```
{
  "question": "...",
  "options": array format,
  "result": "1 or 2",
  "reason": "...",
  "hint": "...",
  "knowledgePoint": "...",
  "analyze": "..."
}
```

B.4 Prompt for SocratDataset Construction

Your task is to create a Socratic-style simulated dialogue between an elementary school student and an elementary science teacher.

Question
{question} {options}

Problem-Solving Information
Hint: {hint}
Knowledge Point: {knowledgePoint}
Problem-Solving Thought: {problemSolvingThought}

Role Requirements

Student

The dialogue should begin with the student asking the teacher for help with the question. The phrasing and format of the question should be similar to: "How do I solve this question: <question>" or "Please help me solve this question: <question>." The text within <question> must exactly match the original text following <question>, with no changes to the wording, sequence, or punctuation.

The student should simulate multiple incorrect answers throughout the conversation.

All student responses should be aligned with the knowledge and cognitive ability of an elementary school student.

Student responses must not reference the <Problem-Solving Information>.

The student should avoid using filler words (e.g., "um," "like") as much as possible.

Teacher

Follow the five-stage structure of the Socratic teaching method strictly, guiding the student through the problem step by step.

Ensure each guiding question is appropriate for elementary school students, avoiding overly complex language or logic.

When asking questions, the teacher must reference the <Problem-Solving Information> and stay directly related to solving the problem. Do not ask unrelated questions.

Only one question may be asked at a time.

The teacher's tone should be extremely friendly and encouraging.

Unless in the final summarizing stage, the teacher must never directly give the correct answer to the student.

Questioning Rules

Determining Dialogue Stages and Corresponding States

The dialogue begins at stage "a": Student Inquiry Stage, with state "a1"

a1: The student poses the question

Then proceeds to stage "b": System Questioning Stage, aimed at assessing the student's understanding of relevant concepts and identifying potential misconceptions. Fully consider b2 to b7 and select the most appropriate state:

b2: No available strategy and the problem has not been fully investigated

b3: No available strategy but the problem has been investigated

b4: A serious conceptual error exists and related sub-questions are available

b5: Checking if the student truly understands a concept

b6: Student's exercise and response are incorrect

b7: Error occurs in a concept the student has already learned

Next is stage "c": Inductive Reasoning Stage, identifying the rules the student is forming, analyzing their correctness, and confirming principles. This is the core dialogue stage. Fully consider c8 to c29 and select the most appropriate state:

c8: The student makes incomplete or inconsistent predictions

c9: The student's response is incorrect

c10: The student's response is inconsistent with previously learned concepts

c11: The student introduces unrelated factors

c12: The student gives an incomplete explanation

c13: A misleading question successfully misleads the student

c14: A new context is introduced

c15: The concept being practiced is already familiar

c16: The student realizes their mistake

c17: The student overlooks a key point and related sub-questions exist

c18: The student overlooks a key point and no sub-questions exist

c19: The type of misunderstanding is unclear

c20: The student makes a wrong prediction

c21: The student is unable to make a prediction

c22: The student answers correctly

c23: The student forms a partial hypothesis

c24: The student forms a hypothesis and has relevant experience

c25: The student forms a hypothesis without relevant experience

c26: The student fails to test the hypothesis but has experience

c27: The student fails to test the hypothesis and lacks experience

c28: The student tests the hypothesis incorrectly but has experience

c29: The student tests the hypothesis incorrectly and lacks experience

Then proceed to stage "d": Generalization Stage, helping the student form a new rule and apply it. Fully consider d30 to d33 and select the most appropriate state:

d30: The teacher wants to check if the student truly understands

d31: The student still fails to grasp a concept after dialectical inquiry

d32: The student has already investigated a specific problem

d33: All concepts have been explored

Only when the student gives the correct answer can the conversation proceed to stage "e": Teacher Summary Stage, with state "e34"

Questioning Actions Based on Evaluated State

If the state is "a1", generate a question.

If the state is "b2", generate questions from different angles.

If the state is "b3", rephrase the question.

If the state is "b4", generate sub-questions from different angles related to the concept.

If the state is "b5", ask a question to check the student's understanding of the concept.

If the state is "b6", review concepts the student has already learned.

If the state is "b7", review the concepts and compare them with the student's errors.

If the state is "c8", provide a counterexample.

If the state is "c9", help the student form an incomplete rule for further exploration or ask a misleading question.

If the state is "c10", ask for the reason behind the inconsistency.

If the state is "c11", point out the irrelevant factor and explicitly ask for the reason.

If the state is "c12", help the student form an incomplete rule for exploration, ask a misleading question, or provide a counterexample.

If the state is "c13", provide a counterexample.

If the state is "c14", "c15", or "c16", encourage the student to make predictions and propose a new principle.

If the state is "c17", generate a sub-question based on the key point.

If the state is "c18", ask the student to reconsider that point.

If the state is "c19", generate a diagnostic question.

If the state is "c20", validate the newly learned concept.

If the state is "c21", ask the student to think about the problem in detail.

If the state is "c22", ask "Why?"

If the state is "c23", help the student revise and complete the hypothesis.

If the state is "c24", ask the student to test the hypothesis independently.

If the state is "c25", provide a way to verify the hypothesis.

If the state is "c26", ask the student to compare two examples.

If the state is "c27", guide the student in hypothesis testing.

If the state is "c28", inform the student of the error and ask them to propose an alternative concept.

If the state is "c29", provide the correct concept and ask why they hadn't considered it before.

If the state is "d30", present a related case and ask for a prediction, or ask "Why?"

If the state is "d31", directly present the correct concept and rules and ask the student to rethink them.

If the state is "d32", present a related case and ask for a prediction.

If the state is "d33", establish a general definition and ask the student to answer the original question.

If the state is "e34", summarize the question and its solution.

Strictly Enforced Constraints

- The entire dialogue process must strictly follow the progression of stages $a \rightarrow b \rightarrow c \rightarrow d \rightarrow e$. Skipping any stage or reverting to a previous stage is not allowed.
- Only in stage d can the student be asked to provide the answer to the original question. Requests for an answer are not permitted in any stage prior to stage d.
- The dialogue can proceed to stage e only after the student has correctly given the standard answer to the question.
- Once stage e is reached, the entire round of dialogue is considered complete. A new dialogue cannot be initiated afterward.

Now, please generate a simulated dialogue following the role requirements and questioning rules, and output JSON only, no other content is allowed.

Place all output into the following JSON structure:

```
[{
  "student": "The student's initial question and subsequent responses",
  "evaluation": "Assessment and determination of the current dialogue stage and state, with justification for the state selection",
  "state": "The number corresponding to the current state",
  "action": "The questioning action corresponding to the identified state",
  "teacher": "The teacher's response according to the action, including the final summary in the last round"
},
Repeat above N times.
]
```

B.5 Prompt for Multi-round Teaching Process Evaluation

Socratic Dialogue Questioning Quality Evaluation Task

Dialogue for Evaluation
{dialogue}

Evaluation Requirements

Please conduct a structured evaluation of the teacher's questioning quality based on the following criteria. Use a strict grading scale and deduct points systematically. A score of 5 may only be awarded if all criteria are fully met and no deficiencies are present:

1. Each standard must be strictly verified item by item
2. Priority should be given to deduction criteria, not bonus ones
3. A perfect score must not be given without clear and sufficient evidence
4. If multiple deficiencies exist within a single dimension, grade according to the lowest-performing aspect

Scoring Dimensions and Criteria

Guidance: Evaluates whether the questions appropriately guide students to think independently.

Note: The final two rounds of the dialogue are summarization phases. Teacher prompts in these rounds are excluded from the guidance score; only prior questions are assessed for guidance quality.

5 points: Questions are well-structured and layered, guiding the student step by step toward independent thinking.

4 points: Questions are generally guiding, with slight over-acceleration in a few parts.

3 points: Some guidance is present, but pacing or depth is poorly controlled.

2 points: Hints or partial answers are provided too early, limiting the student's opportunity to think.

1 point: Answers or excessive clues are given outright, leaving no room for student reasoning.

Logicity: Evaluates the logical structure and progression of the questions.

5 points: The sequence of questions follows cognitive development principles—progressing from simple to complex with tight, coherent logic.

4 points: Overall logic is clear; minor transitions may feel slightly abrupt but don't hinder understanding.

3 points: Basic logic is maintained, but there are occasional leaps or repetitions requiring the student to infer connections.

2 points: Disordered question sequence causes confusion or disrupts the student's understanding.

1 point: Questions lack a coherent thread, are contradictory or fragmented, and cause cognitive disorientation.

Flexibility: Evaluates whether the teacher adapts questions dynamically based on student responses.

5 points: Skillfully adjusts question difficulty, angle, or sequence in real-time based on student input.

4 points: Generally responsive to student answers, with occasional lapses in responsiveness.

3 points: Shows intent to adapt but struggles to stay attuned to the student's thinking in parts.

2 points: Rigidly follows a script, only reacting to obvious errors.

1 point: Fully scripted, ignores student responses and misconceptions.

Repetitiveness: Evaluates diversity of questions and avoidance of redundancy.

5 points: Questions are varied and creative, with no repetition, maintaining a lively dialogue.

4 points: Mostly varied, with occasional repetition that reinforces understanding.

3 points: Some repeated questions reduce efficiency slightly.

2 points: Frequent repetition of similar phrasing or content causes student disengagement.

1 point: Excessive mechanical repetition seriously undermines interaction quality.

Clarity: Evaluates language clarity and accessibility for student understanding.

5 points: Language is concise, precise, and unambiguous; easily understood by the student.

4 points: Generally clear; a few questions may require brief clarification.

3 points: Mostly understandable, but vague terms or complex syntax hinder comprehension at times.

2 points: Questions are long-winded or abstract, requiring repeated clarification.

1 point: Expression is confusing, terminology inappropriate, or internally inconsistent, leading to widespread misunderstanding.

Output Specification

Please generate the evaluation report according to the following requirements:

1. For each scoring dimension:
 - You must cite at least one specific dialogue excerpt as evidence.
 - The score must strictly align with the detailed rubric criteria.
2. Each dimension must be scored independently.
3. Only output JSON—do not include any additional text.

Please structure all output content using the JSON format below:

```
{
  "Guidance": {"reason": "[Analysis based on specific dialogue excerpt]", "score": [1-5]},
  "Logicity": {"reason": "[Analysis based on specific dialogue excerpt]", "score": [1-5]},
  "Flexibility": {"reason": "[Analysis based on specific dialogue excerpt]", "score": [1-5]},
  "Repetitiveness": {"reason": "[Analysis based on specific dialogue excerpt]", "score": [1-5]},
  "Clarity": {"reason": "[Analysis based on specific dialogue excerpt]", "score": [1-5]}
}
```

C Example

C.1 Example for the SocratDataset

```
{
  "dialogue": [
    {
      "student": "Can you help me solve this question: Which of the following methods does NOT enhance the magnetism of an electromagnet? ( ) A. Increasing the number of batteries B. Changing the coil's winding direction C. Increasing the number of coil turns.",
      "evaluation": "The student asked a question, entering stage a, with state a1.",
      "state": "a1",
      "action": "Generate a sub-question related to solving the problem.",
      "teacher": "Do you know which factors affect the strength of an electromagnet's magnetism?"
    },
    {
      "student": "I think it might have something to do with the number of batteries.",
      "evaluation": "The student's answer is incomplete, entering stage b, with state b6.",
      "state": "b6",
      "action": "Review the concepts the student has already learned.",
      "teacher": "That's right, the number of batteries does affect the electromagnet's magnetism. Do you also remember how the number of coil turns and the current affect it?"
    },
    {
      "student": "Oh right, the number of coil turns also affects the magnetism.",
      "evaluation": "The student partially understands the concept but has not mentioned all relevant factors, entering stage c, with state c12.",
      "state": "c12",
      "action": "Help form an incomplete rule and further investigate it.",
      "teacher": "Exactly, increasing the number of coil turns strengthens the magnetism. Now, what do you think happens when the direction of the coil winding is changed?"
    },
    {
      "student": "Maybe changing the winding direction makes the magnetism stronger?",
      "evaluation": "The student's answer is incorrect, entering stage c, with state c9.",
      "state": "c9",
      "action": "Help the student form an incomplete rule and investigate it further.",
      "teacher": "Interesting thought! Let's imagine: if you change the coil's winding direction, does the direction of the current change?"
    },
    {
      "student": "Hmm, the current direction probably doesn't change.",
      "evaluation": "The student realized their mistake, entering stage c, with state c16.",
      "state": "c16",
      "action": "Encourage the student to make predictions and propose a new principle.",
      "teacher": "Great! So changing the winding direction doesn't affect the magnetism. Can you now summarize which methods can enhance an electromagnet's magnetism?"
    },
    {
      "student": "Increasing the number of batteries and increasing the number of coil turns can enhance the magnetism.",
      "evaluation": "The student correctly understands the concept, entering stage d, with state d33.",
      "state": "d33",
      "action": "Establish a general definition and ask the student to give the answer to the problem.",
      "teacher": "Excellent! So based on our discussion, what do you think is the correct answer to the question?"
    },
    {
      "student": "Changing the coil's winding direction does not enhance the electromagnet's magnetism.",
      "evaluation": "The student correctly answered the question, entering stage e, with state e34.",
      "state": "e34",
      "action": "Summarize the problem.",
      "teacher": "Correct! Changing the coil's winding direction does not affect the strength of an electromagnet's magnetism—it only changes the magnetic poles. Great work, keep thinking like this!"
    }
  ]
}
```

C.2 Example for the Guiding Problem-Solving Dataset

```
{
  "mission": "Choice question",
  "question": "Among the following groups of electromagnets, which one can attract the most thumbtacks ( )?",
  "options": [
    "An electromagnet with three batteries connected in series and 50 turns of wire in the coil",
    "An electromagnet with three batteries connected in series and 25 turns of wire in the coil",
    "An electromagnet with two batteries connected in series and 50 turns of wire in the coil"
  ],
  "answer": 1,
  "hint": "Think about which factors affect the strength of an electromagnet's magnetism.",
  "knowledgePoint": "The strength of an electromagnet's magnetism is affected by the electric current through the coil and the number of turns of the coil. The number of batteries affects the current, and more turns result in stronger magnetism.",
  "problemSolvingThought": "The magnetic strength of an electromagnet mainly depends on the current and the number of coil turns. Connecting three batteries in series provides a stronger current, and having more coil turns increases magnetism. Therefore, the electromagnet with three batteries and 50 turns has the strongest magnetism and can attract the most thumbtacks. In the other options, although one of the factors may be better, the overall performance is inferior to the first option."
}
```