Grade-Aware Controllable Text Generation for Math Education via Hierarchical Knowledge Graphs

Anonymous ACL submission

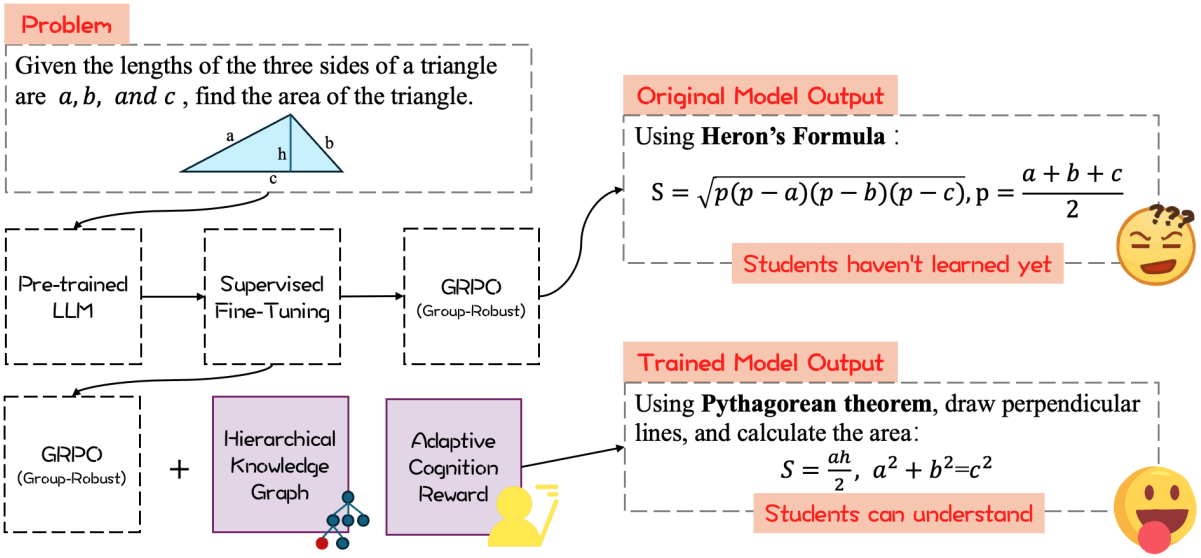


Figure 1: In some mathematical problems, large models tend to use more complex knowledge to solve them. Our method can guide large models to generate answers that are more conducive to teaching.

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in solving mathematical problems, yet their solutions often rely on knowledge beyond the cognitive level of target student groups, limiting their educational value. In this paper, we propose a novel training framework that enables LLMs to generate mathematically correct yet pedagogically appropriate solutions aligned with students' grade-level knowledge. By integrating a hierarchical knowledge graph(HKG) annotated with textbook-aligned difficulty levels and designing a multi-turn dialogue-based reward function, we extend Controllable Text Generation (CTG) to control the knowledge difficulty of generated content. Our adaptive cognition reward mechanism evaluates solutions based on their alignment with target-grade knowledge, guiding model optimization through a customized Group Relative Policy Optimization (GRPO) algorithm. Experimental results on a stratified subset of the OpenR1-Math-220k dataset demonstrate that our approach effectively reduces knowledge difficulty in generated solutions while maintaining correctness, offering a significant step toward grade-aware and instruction-friendly educational AI.

1 Introduction

In recent years, LLMs have made significant progress in mathematical reasoning and problem-solving (Ahn et al., 2023; Liang et al., 2023), demonstrating relatively high accuracy in handling mathematical problems (Liu et al., 2023; Guo et al., 2023). However, despite their excellent performance in outputting correct answers, these models overlook the difficulty of the knowledge involved in the reasoning process, which directly affects the applicability and effectiveness of the generated content in educational settings, a factor crucial in the field of education.

In real educational scenarios, although models provide correct answers, the problem-solving process does not adequately correspond to the knowledge and comprehension levels of students in the target grade. The main reason for this issue is that existing mathematical problem datasets primarily focus on the correctness of answers, neglecting the difficulty levels of the problem-solving processes. Most datasets only include rough difficulty levels (Miao et al., 2021; Hendrycks et al., 2021), lacking detailed annotations on the difficulty of the knowl-

edge points involved in the problem-solving steps, leading models to potentially learn more complex problem-solving methods during training. Moreover, when models generate solutions, they lack consideration for students' knowledge levels, making it difficult to provide solution processes that match students' comprehension abilities.

Our goal is to develop a model capable of generating problem-solving processes that match students' knowledge levels. To achieve this objective, CTG technology offers a potential solution, allowing control over the attributes, content, or style of text during the generation process (Liang et al., 2024b; Upadhyay et al., 2022). However, current CTG research mainly focuses on aspects such as the safety, legality, and style of generated text, lacking the ability to match the knowledge level of the target user group (Lorandi and Belz, 2023; Wang et al., 2024; Zhang et al., 2023a). This results in generated text that often fails to adapt to the knowledge background of different audiences, especially in educational scenarios, where it has limited educational significance for younger students.

Inspired by Reinforcement Learning (RL) techniques, particularly Reinforcement Learning from Human Feedback (RLHF) (Bai et al., 2022), which optimize model generation quality through reward signals (Zhang et al., 2023b; Rafailov et al., 2023; Yang et al., 2023), we propose a new training method. This method aims to address the limitations of existing RL and RLHF methods in the educational field, which lack consideration for the knowledge level of the target user group, leading to generated text that does not match the audience's comprehension ability. By constructing a hierarchical knowledge graph(HKG) and closely integrating it with LLMs, we achieve precise evaluation of the difficulty levels of the knowledge points in the model-generated answers and automatically add fine-grained difficulty labels to existing datasets. Our reward function design involves multi-turn dialogues with LLMs, simulating human thinking processes, summarizing problem-solving steps, and comparing extracted knowledge with the knowledge graph. This method enhances the accuracy of extracted knowledge points and grade levels, thus enabling generated text to better adapt to the knowledge levels and comprehension abilities of students across different grades, addressing the issue of overly complex knowledge points used by large models.

Our main innovations are, as illustrated in Figure 1:

- **Expanding the scope of Controllable Text Generation**: We extend the traditional goals of CTG from controlling text style and safety to **controlling knowledge difficulty**, filling the gap in existing CTG research concerning knowledge level matching.
- **Designing a reward function based on multi-turn dialogue**: By engaging in multi-turn automated dialogues with the LLM, we simulate human thinking processes to summarize the problem-solving steps and compare them with the knowledge graph, proposing a method to design reward functions based on the difficulty difference of knowledge points.
- **Integrating knowledge graphs with multi-turn interaction mechanisms**: By constructing a local knowledge graph and tightly integrating it with multi-turn interactions with the LLM, we achieve precise evaluation and dynamic adjustment of the difficulty in the model's generated answers.

2 Related Work

2.1 Mathematical Problem-Solving Abilities of LLMs and Their Datasets and Evaluations

LLMs have demonstrated remarkable capabilities in mathematical reasoning and problem solving (Ahn et al., 2023)(Liang et al., 2023). Studies indicate that LLMs achieve high accuracy when handling mathematical problems (Liu et al., 2023) (Guo et al., 2023), showcasing strong understanding and reasoning abilities, and providing a solid foundation for educational applications.

2.2 Controllable Text Generation (CTG)

CTG focuses on controlling the attributes, content, or style of text during the generation process and has become a key research area in natural language generation (Liang et al., 2024b)(Upadhyay et al., 2022). Current CTG methods include model retraining, fine-tuning, reinforcement learning, prompt engineering, latent space manipulation, and decoding time interventions. Each method has its advantages and limitations, suitable for various applications and requirements.

The controllable text generation method based on dynamic attribute graphs (Liang et al., 2024a) introduces a pluggable Dynamic Attribute Graphs-based controlled text generation (DATG) frame-

work, combining dynamic attribute graphs with LLMs. This approach provides a novel and flexible attribute-driven text generation method, achieving fine-grained control over text generation by dynamically adjusting attribute weights during the generation process.

2.3 Reinforcement Learning from Human Feedback (RLHF)

Reinforcement Learning (RL) techniques optimize model generation quality based on reward signals, effectively fine-tuning models towards specific goals (Zhang et al., 2023b)(Rafailov et al., 2023)(Yang et al., 2023). Feedback in reinforcement learning can be automatic or from human input, with the latter known as Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017)(Ouyang et al., 2022). RLHF enables LLMs to better align with human styles and ethical values(Bai et al., 2022).

On this basis, we propose a method that integrates a fig module with a multi-round dialogue mechanism. This approach enables the dynamic assessment of the difficulty levels of knowledge points in generated solutions, ensuring precise alignment with the cognitive and comprehension capabilities of lower-grade students.

3 Methods

We propose a method to reduce the difficulty level of knowledge points used by LLMs in mathematical problem solving. The method comprises fig construction, adaptive cognition reward mechanism, and GRPO training (Guo et al., 2023)(Shao et al., 2024). These components collectively guide the model to generate simpler and grade-appropriate solutions. The overall architecture is depicted in Figure 2.

3.1 Task Definition

Although LLMs have achieved high accuracy in solving mathematical problems, most existing research focuses on enhancing solution correctness, often overlooking the educational significance of the problem-solving process. This tendency leads models to utilize advanced concepts beyond the knowledge scope of lower-grade students to solve problems, resulting in solutions that are difficult for these students to understand and lacking in instructional value.

In this study, our objective is to train a large language model such that, for mathematical problems q in the dataset annotated with grade information d_{data}, the model-generated answer a involves knowledge points K whose corresponding grade levels d_k in the fig G have a maximum value d_{model} that is as low as possible, thereby enhancing the answer's educational value.

To avoid redundant computations, we preprocess the dataset's questions q and reference answers a_{ref} by applying the same knowledge point extraction and difficulty evaluation algorithms. This process yields the annotated grade or difficulty information d_{data} of the questions for subsequent use.

3.2 Hierarchical Knowledge Graph Construction

To accurately assess the difficulty of knowledge points and guide the model to generate simpler and more comprehensible answers, we construct a fig $G = (V, E)$ as visualized in Figure 3 that incorporates grade-level information. The difficulty levels within this knowledge graph are aligned with standard textbooks, ensuring consistency, while its hierarchical structure facilitates access and understanding by LLMs. This design enables the model to produce responses appropriate to the cognitive levels of students.

The knowledge graph consists of three layers. The first layer nodes represent the main categories of mathematics, such as discrete mathematics, geometry, and algebra, forming the set L_1 and covering a wide range of mathematical fields. The second layer nodes are subfields under these main categories, such as triangles, solid geometry, set theory, and combinatorics, forming the set L_2 and providing a more detailed subdivision of each discipline. The third layer comprises specific knowledge points forming the set K, where each knowledge point $k \in K$ is associated with a difficulty level $d_k \in \mathbb{N}^+$ corresponding to a specific textbook grade or chapter. This difficulty level is defined directly based on the grade levels or chapter difficulties in textbooks, ensuring accuracy and consistency when the model evaluates the difficulty of answers.

The design of this hierarchical structure simplifies the search space for the model when selecting knowledge points, reducing the number of candidates it needs to consider during each selection. This simplification enhances the model's efficiency and accuracy in choosing appropriate knowledge points and ensures that it can accurately determine the grade-level difficulty associated with each

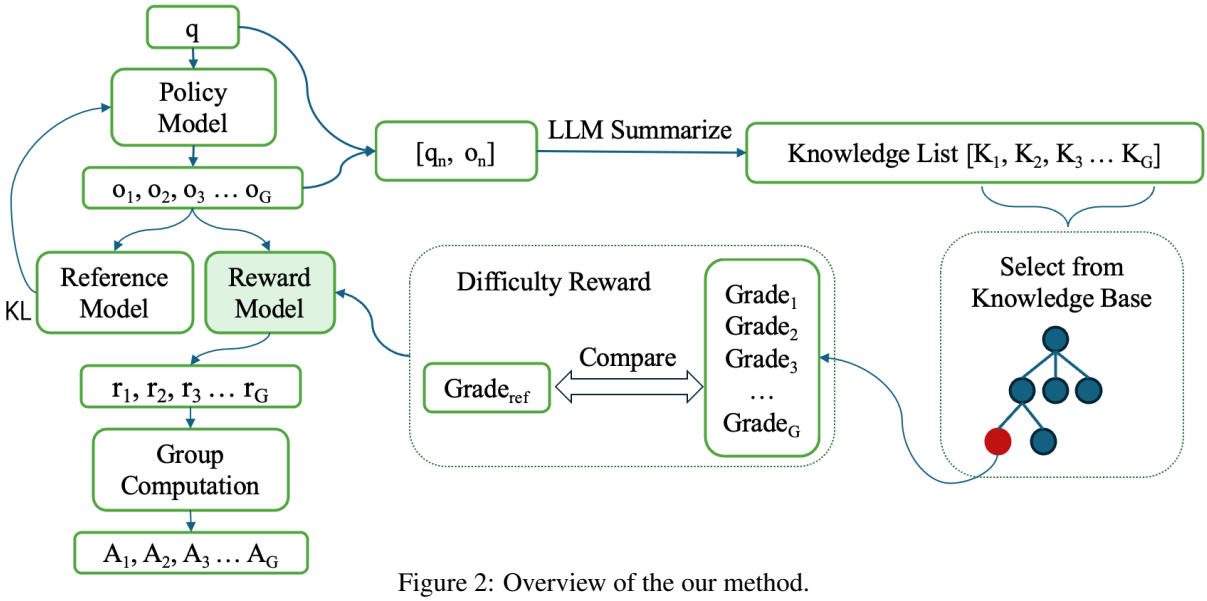


Figure 2: Overview of the our method.

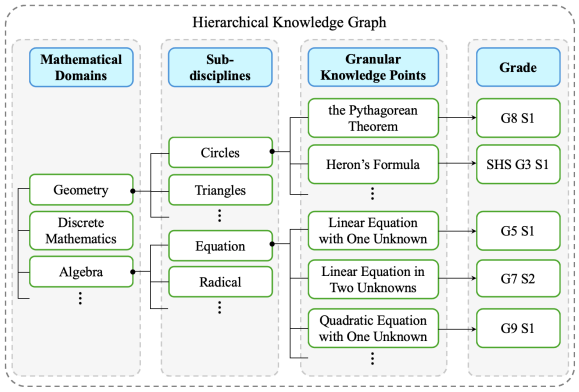


Figure 3: Hierarchical structure of the knowledge base showing three know levels and

point. Consequently, the model can generate answers that align with the target students' cognitive levels, thereby enhancing the educational value of its responses.

When processing an answer a, the model can map the involved knowledge points to nodes within the knowledge graph, forming a set of knowledge points $K_a \subseteq K$. Utilizing the difficulty information provided by the graph, the model can compute the overall difficulty level of the answer:

$$d_{\text{model}} = \max_{k \in K_a} d_k \tag{1}$$

Through this computation, the model can assess whether its generated answer is suitable for students at the target grade level. If necessary, it can adjust the content and expression of the answer, making it more aligned with the students' cognitive abilities. This approach ensures that the model's responses are not only correct but also pedagogically appropriate, thereby maximizing their instructional value.

3.3 Adaptive Cognition Reward

To train LLMs to solve mathematical problems accurately while utilizing lower-level knowledge points, we introduce an adaptive cognition reward mechanism. This mechanism guides the model to generate correct answers that are simpler and more appropriate for the target grade level, enhancing both correctness and educational suitability.

We define a reward function $R(a, q)$ that evaluates the model-generated answer a by comparing its difficulty level d_{model} with the annotated difficulty level d_{data} of the question q. The difference in difficulty levels is calculated as:

$$\Delta d = d_{\text{data}} - d_{\text{model}} \tag{2}$$

To map this difference to a reward value and ensure a smooth transition between positive and negative rewards, we apply a sigmoid function:

$$R_{\text{difficulty}} = \frac{1}{1 + e^{-\Delta d}} \tag{3}$$

This difficulty reward encourages the model to produce answers using knowledge points that are at or below the difficulty level of the question, aligning with educational practices that prioritize accessibility for students. By leveraging the fig, the model gains awareness of knowledge point levels, simulating human-like evaluation and reasoning without lacking cognition of knowledge point difficulty.

The total reward function incorporates additional components to ensure the quality and correctness of the generated answers:

- **Format Reward** $R_{\textbf{format}}$: Ensures that the answer is enclosed within the specified tags (e.g., `<think>` and `<answer>`), maintaining output standardization.
- **Tag Quantity Reward** $R_{\textbf{tag}}$: Verifies that the number of tags used meets expectations, preserving the structural integrity of the response.
- **Accuracy Reward** $R_{\textbf{accuracy}}$: Assesses the correctness of the answer, encouraging the generation of accurate solutions.

Combining these components, the final total reward function is defined as:

$$\begin{aligned} R =& \lambda_{\text{difficulty}} R_{\text{difficulty}} + \lambda_{\text{format}} R_{\text{format}} \\ &+ \lambda_{\text{tag}} R_{\text{tag}} + \lambda_{\text{accuracy}} R_{\text{accuracy}} \end{aligned} \tag{4}$$

Here, λ represents the weighting parameters for each component, allowing for adjustment of their influence on the overall reward according to specific training objectives.

By integrating this adaptive cognition reward mechanism, we effectively guide the LLM to generate answers that are not only correct but also appropriately simplified for the target audience. This approach enhances both the educational value and accessibility of the model's responses, simulating human feedback and reasoning processes in an automated manner.

3.4 Training and Inference

To reduce the difficulty level of knowledge points utilized by LLMs in mathematical problem-solving, we implement GRPO with formalized objective. For each question q, sampling G outputs $\{o_1, o_2, ..., o_G\}$ from old policy $\pi_{\theta_{old}}$, we optimize the policy model by maximizing:

$$\begin{aligned} \mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)] \\ \frac{1}{G}\sum_{i=1}^{G}(\min(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \\ \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right)A_i\Big) \\ - \beta\mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{\text{ref}})\Big)\Big], \end{aligned} \tag{5}$$

$$\begin{aligned} \mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{\text{ref}}) = \mathbb{E}_{o_i}\Big[\frac{\pi_{\text{ref}}(o_i|q)}{\pi_\theta(o_i|q)} \\ - \log\frac{\pi_{\text{ref}}(o_i|q)}{\pi_\theta(o_i|q)} - 1\Big], \end{aligned} \tag{6}$$

$$A_i = \frac{r_i - \text{mean}(\{r_j\})}{\text{std}(\{r_j\})},\ j = 1, \ldots, G. \tag{7}$$

where the KL divergence constraint is defined as:

$$D_{KL}(\pi_\theta||\pi_{ref}) = \mathbb{E}_{o_i}\left[-\log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1\right] \tag{8}$$

The advantage function A_i is computed through standardized reward differences within each group:

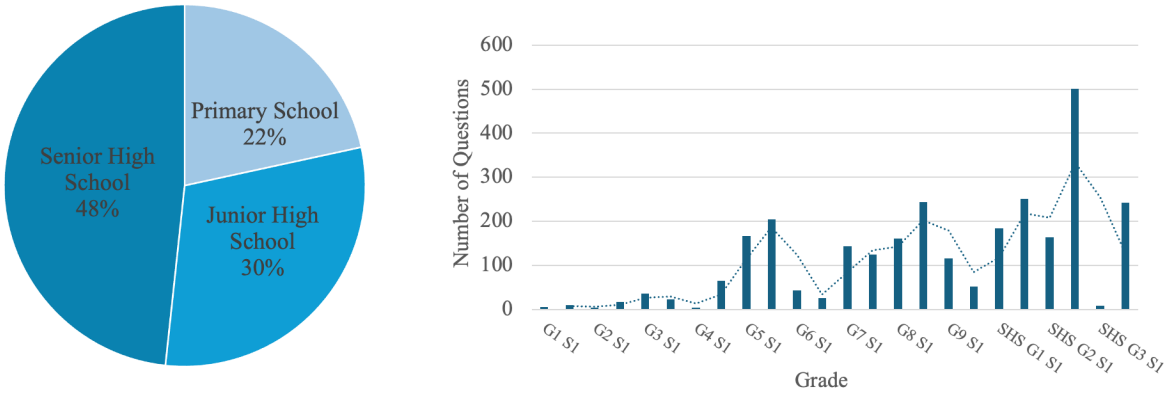
$$A_i = \frac{r_i - \mu(\{r_j\}_{j=1}^{G})}{\sigma(\{r_j\}_{j=1}^{G})} \tag{9}$$

The hyperparameters ϵ control the clip threshold for policy updates, while β adjusts the KL regularization strength. This mechanism automatically establishes dynamic baselines using group reward statistics, with relative advantage evaluation guiding the model to generate solutions adhering to difficulty constraints, demonstrating superior adaptability to multi-objective reward scenarios compared to fixed baseline approaches.

4 Experimental Setup

4.1 Datasets

The dataset employed in this study is derived from the OpenR1-Math-220k dataset (Guo et al., 2025). We extracted a stratified subset from this dataset that aligns with the textbook syllabus, ensuring that the distribution of question grades corresponds to real-world teaching scenarios. This subset encompasses three educational stages: primary school (21.6%), junior high school (30.1%), and senior high school (48.3%), as illustrated in Figure 4(a). By leveraging a fig and manual verification, we annotated the dataset with fine-grained difficulty labels corresponding to the textbook. The specific distribution of these labels is shown in Figure 4(b). The distribution exhibits a fluctuating pattern of initially rising and then declining across different learning stages, which is consistent with the pedagogical system. Notably, the scarcity of new knowledge in the final year of senior high school results



a b

Figure 4: (a) Distribution of educational stages (primary, junior high, senior high); (b) Distribution of difficulty labels.

in a limited number of questions being designated for this grade. The significant number of questions attributed to the second semester of senior high school is due to the categorization of these questions under more challenging knowledge points, thereby validating the accuracy of our grade extraction method. The dataset was divided into training, validation, and test sets through stratified sampling, preserving the grade distribution.

4.2 Baselines

In this study, we selected DeepSeek-R1-Distill-Qwen-1.5B and Qwen2.5-1.5B-Instruct (Yang et al., 2024) as the baseline models to systematically evaluate the performance enhancement of the improved model. Both models are based on the Qwen architecture and represent two key technical pathways: knowledge distillation and instruction tuning. DeepSeek-R1-Distill-Qwen-1.5B achieves model lightweighting through knowledge distillation, enhancing inference efficiency while maintaining performance. Qwen2.5-1.5B-Instruct, on the other hand, has been specifically tuned for instructions, demonstrating outstanding performance in task adaptability and instruction following.

The selection of these two models not only considers their technical representativeness, covering the main directions of current language model optimization, but also takes into account the feasibility of the experiment and resource efficiency. Since both models share the same architectural foundation, they can minimize the interference of model differences on the experimental results, ensuring the fairness and reliability of the performance comparison. These two widely validated baseline models provide a scientific reference standard for the study, which helps accurately assess the actual effects of the improvement strategies.

4.3 Evaluation Metrics

To comprehensively evaluate the model's performance, we employed a combination of automatic and human evaluations. These evaluation metrics are designed to measure the model's control over difficulty and educational adaptability in generating answers, closely aligning with the experimental results.

Automatic Evaluation:

- **Delta Difficulty**: This metric measures the difference in grade values of the knowledge involved in the model-generated answers compared to the dataset answers. Specifically, it is calculated by determining the difference between the difficulty labels of the generated answers and the corresponding answers in the dataset. Our goal is to have the trained model generate answers with lower grade values, hence a lower Delta Difficulty is desired. This metric directly reflects the model's effectiveness in reducing the difficulty of answers.
- **Accuracy**: The proportion of correct answers generated by the trained model out of the total number of answers.

Human Evaluation:

Given the current lack of a widely accepted automated method for assessing answer difficulty, we opted for human expert evaluation. We enlisted 10 experienced math teachers to evaluate 100 randomly selected test samples. To ensure objectivity, each question presented both the original pre-trained model's and the improved model's answers, with model identifiers concealed and an-

swers randomly ordered. The evaluation dimensions included:

- **Knowledge Point Difficulty**: Scored on a 1-5 scale, compared to the pre-trained model's answers. This assesses whether the knowledge points used in the new model's answers are simpler, with 0 indicating significantly more difficult, 3 indicating comparable difficulty, and 5 indicating significantly simpler.
- **Reasoning Complexity**: This evaluates whether the new model's answers are easy to understand and suitable for the target grade students' cognitive abilities. Using a 1-5 scale: 1 indicates complex reasoning that is hard to understand; 3 indicates moderate reasoning that some students might need additional explanation for; and 5 indicates clear and easy-to-understand reasoning.

5 Experimental Results

5.1 Evaluation Results

The experimental results demonstrate that the trained model has achieved significant improvements in multiple key metrics, especially in terms of difficulty control. Specifically, as shown in Table 1, the following observations were made:

Accuracy Our model achieved an accuracy of 0.17, which remains competitive in the context of the current study. This accuracy is slightly lower than that of the DeepSeek-R1-Distill-Qwen-1.5B model (0.19), which may be attributed to our deliberate focus on controlling the difficulty of knowledge points during the training process. It is important to note that our model was not trained on the full dataset. This indicates that despite the limited data, our model is still capable of effectively learning and generating accurate answers.

Compared with the pre-trained model Qwen2.5-1.5B-Instruct (accuracy of 0.12, difficulty score of 69.5), our model not only made significant progress in difficulty control but also achieved an improvement in accuracy. This demonstrates that our training method can effectively enhance model performance even with limited data and specific training objectives.

Difficulty Control Our model excelled in difficulty control. Compared with the pre-trained model Qwen2.5-1.5B-Instruct, our model not only reduced the difficulty score but also ensured that the generated answers are easier to understand and do not exceed the syllabus.

The consistency between the automatic evaluation and human assessment of the difficulty of the answers generated by our model further validates the effectiveness of our approach.

Educational Significance Through a blind test evaluation by 10 mathematics teachers, the new model demonstrated significant advantages in core teaching dimensions. Compared with the original pre-trained model, the new model generated answers using simpler knowledge points in 61% of cases, and in 83% of the assessed data, the reasoning steps of the new model were found to be more in line with students' cognitive development patterns and easier to understand. This indicates that the tutoring role of our model in the field of subject education has been significantly enhanced.

In summary, although our model's accuracy is slightly lower than that of models trained on the full dataset and focused solely on accuracy, it has made significant progress in difficulty control and educational significance, bringing more educationally meaningful outputs. This demonstrates that our method has important application value in balancing accuracy and difficulty control.

5.2 Ablation Experiment

To evaluate the impact of each component on model performance, we conducted ablation experiments. We compared the fine-tuned model with the pre-trained model using prompt engineering and the distilled model.

Our approach reduces the complexity of the knowledge points used by the model without compromising the accuracy of the model's answers. In the test data, the average difficulty of the knowledge points used per data point was reduced by 0.85 and 1.75 grade levels compared to the pre-trained and distilled models, respectively. Given that many problems may not have simpler solutions at lower grade levels, this reduction is quite significant. For those problems that do have simpler solutions, our approach demonstrates substantial superiority.

These results highlight the effectiveness of our training methodology. By fine-tuning the model with a fig-based reward mechanism for educational scenarios, we ensure that the model provides accurate answers while explaining them in a manner that is easier for students to understand and accept. This high alignment with educational goals is crucial for developing models that can truly support and enhance the learning experience.

Model	Accuracy	Difficulty
Qwen2.5-1.5B-Instruct	0.12	69.5
DeepSeek-R1-Distill-Qwen-1.5B	0.19	78.5
Our Model	0.17	61.0

Table 1: Comparative performance of different models. Accuracy represents the model's correctness rate (higher is better), while Difficulty indicates the average problem difficulty level the model can handle (lower means handles more difficult problems).

5.3 User Study

To evaluate the effectiveness of the model in educational settings, we conducted a user study with 10 experienced math teachers and 20 students (7 from primary, 7 from junior high, and 6 from senior high school). The participantsants reviewed 20 math problems, each paired with two anonymsolutions, one, one from the original model and one from our imprmodel, withoutithout knowing their sources.

Teachers assessed each solution pair based on grade-level suitability, clarity of reasoning, and appropriate use of knowledge. Students chose the solution they found easier to understand and explained their choice in plain language.

Results showed that teachers preferred the improved model's solutions in 61% of cases, citing clearer reasoning and better curriculum alignment. Similarly, 64% of students favored the improved solutions, often describing them as "more familiar" and "easier to follow." Teachers also noted that the improved model avoided advanced concepts and better reflected textbook-style explanations, increasing its classroom relevance.

5.4 Case Study

To further demonstrate the effectiveness of our new model,we have selected a specific case for detailed analysis.

As shown in Figure 2, our model effectively generates problem solving processes that match students' knowledge levels. Our model employs fundamental concepts such as the Pythagorean theorem and systems of equations,which are typically introduced in earlier grades,while avoiding more advanced knowledge like Heron's formula that is usually taught in high school. This problem-solving approach significantly reduces the cognitive load on students,facilitating their grasp of key concepts and ensuring alignment with their learning progress.

Problem:
Given the lengths of the three sides of a triangle are $4\sqrt{2}, 5,$ and 7, find the area of the triangle

Original Model Output:
Using Heron's Formula:
Calculate the semi-perimeter p:

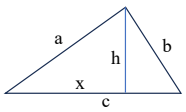
$$p = \frac{a+b+c}{2} = \frac{4\sqrt{2}+5+7}{2} = 2\sqrt{2}+6$$

Apply Heron's formula:

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$
$$= \sqrt{(2\sqrt{2}+6)(2\sqrt{2}+6-4\sqrt{2})(2\sqrt{2}+6-5)(2\sqrt{2}+6-7)}$$
$$= 14$$

Our Model Output:
Using Altitude and Pythagorean Theorem:
Construct the altitude: Draw an altitude from the vertex opposite the longest side ($c = 7$) to the base c. Let the altitude be h and the segments of the base be x and $7-x$.
Apply the Pythagorean theorem:

$$\begin{cases} 32 = x^2 + h^2 \\ 25 = (7-x)^2 + h^2 \end{cases}$$



Solve the equations and calculate the area:

$$\begin{cases} h = 4 \\ x = 4 \end{cases}$$

$$S = \frac{c * h}{2} = 4$$

Table 2: The figure presents a case study comparing the outputs of the original and trained models on a sample problem. The original model provides a concise answer, while our model employs simpler knowledge points, making it more comprehensible for lower-grade students.

6 Conclusion

This paper demonstrates that integrating HKGs into the training of large language models significantly improves their ability to generate mathematically sound and grade-appropriate solutions. By capturing hierarchical dependencies among knowledge points, HKGs impose structural constraints that guide models toward logical and curriculum-aligned reasoning. When incorporated into the reward model, HKGs also enable fine-grained control over the educational quality of generated responses, making them more suitable for instructional use.

7 Limitations

While our method shows clear benefits in producing grade-appropriate mathematical solutions, its broader application remains constrained. The current design is tightly coupled with the structure of mathematics education, relying on manually crafted HKGs that reflect well-defined curricular progressions. Transferring this approach to other subjects may prove challenging, especially in domains lacking similarly structured knowledge or where educational content is more fluid. Additionally, the reward mechanism is curriculum-specific and may require adaptation when applied to different academic areas or educational standards.

References

Junbum Ahn, Rishabh Verma, Ruiqi Lou, and 1 others. 2023. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2302.00157*.

Yuntao Bai, Andy Jones, Kamilė Ndousse, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Paul F. Christiano, Jan Leike, Tom Brown, and 1 others. 2017. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 30.

Daya Guo, Duyu Yang, Hongbo Zhang, and 1 others. 2023. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2301.12948*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, and 1 others. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Xin Liang, Shiji Song, Zhiyong Zheng, and 1 others. 2023. Internal consistency and self-feedback in large language models: A survey. *arXiv preprint arXiv:2307.14507*.

Xin Liang, Hong Wang, Shiji Song, and 1 others. 2024a. Controlled text generation for large language model with dynamic attribute graphs. *arXiv preprint arXiv:2402.11218*.

Xin Liang, Hong Wang, Yue Wang, and 1 others. 2024b. Controllable text generation for large language models: A survey. *arXiv preprint arXiv:2408.12599*.

Weijian Liu, Haipeng Hu, Jinfeng Zhou, and 1 others. 2023. Mathematical language models: A survey. *arXiv preprint arXiv:2312.07622*.

Marco Lorandi and Anya Belz. 2023. How to control sentiment in text generation: A survey of the state-of-the-art in sentiment-control techniques. In *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 341–353.

Shuaichen Miao, Chen-chi Liang, and Keh-Yih Su. 2021. A diverse corpus for evaluating and developing english math word problem solvers. *arXiv preprint arXiv:2106.15772*.

Long Ouyang, Jeffrey Wu, Xu Jiang, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Rumen Rafailov, Abhinav Sharma, Eric Mitchell, and 1 others. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Z Shao, P Wang, Q Zhu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Bhagvant Upadhyay, Ashwin Sudhakar, and Aravind Maheswaran. 2022. Efficient reinforcement learning for unsupervised controlled text generation. *arXiv preprint arXiv:2204.07696*.

Jiachang Wang, Chen Zhang, Dinghan Zhang, and 1 others. 2024. A recent survey on controllable text generation: A causal perspective. *Fundamental Research*.

A Yang, B Yang, B Zhang, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Kevin Yang, Dan Klein, Asli Celikyilmaz, and 1 others. 2023. Rlcd: Reinforcement learning from contrastive distillation for language model alignment. *arXiv preprint arXiv:2307.12950*.

He Zhang, Haoran Song, Shujian Li, and 1 others. 2023a. A survey of controllable text generation using transformer-based pre-trained language models. *ACM Computing Surveys*, 56(3):1–37.

Li Zhang, Qi Zhang, Linxi Shen, and 1 others. 2023b. Evaluating model-free reinforcement learning toward safety-critical tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15313–15321.