

From Objectives to Questions: A Planning-based Framework for Educational Mathematical Question Generation

Cheng Cheng^{1,2}, Zhenya Huang^{1,2,3*}, Guanhao Zhao^{1,2}, Yuxiang Guo^{1,2}, Xin Lin^{1,2},
Jinze Wu^{2,4}, Xin Li^{2,4}, Shijin Wang^{2,4}

¹ University of Science and Technology of China

² State Key Laboratory of Cognitive Intelligence

³Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

⁴iFLYTEK Research

{doublecheng, ghzhao0223, guoyx18, linx}@mail.ustc.edu.cn

{huangzhy, leexin}@ustc.edu.cn

Abstract

{jzwu4, sjwang3}@ifytek.com

Automatically generating high-quality mathematical problems that align with educational objectives is a crucial task in NLP-based educational technology. Traditional generation methods focus primarily on textual quality, but they often overlook educational objectives. Moreover, these methods address only single-dimensional, simple question generation, failing to meet complex, multifaceted educational requirements. To address these challenges, we constructed and annotated EduMath, a dataset of 16k mathematical questions with multi-dimensional educational objectives. Based on this dataset, we developed EQGEVAL, which incorporates three evaluation dimensions and is designed to assess the ability of models to generate educational questions. Drawing inspiration from teachers' problem design processes, we propose the Educational Question Planning with self-Reflection (EQPR) method for educational mathematical question generation, following a "plan-evaluate-optimize" approach. Specifically, by combining planning algorithm based on Monte Carlo Tree Search with the generative capabilities of Large Language Models, we continuously optimize questions through iterative feedback. This self-optimization mechanism ensures that the generated questions both fit the educational context and strategically achieve specific basic educational objectives. Through extensive experiments based on EQGEVAL, we have demonstrated that EQPR achieves significant improvements in generating questions that meet multi-dimensional educational objectives.¹

1 Introduction

Mathematical questions are fundamental elements of educational assessment and cognitive development, playing an irreplaceable role in cultivating

*Corresponding author

¹Code and data are publicly available at <https://github.com/doublecheng12/QuestionAgent>

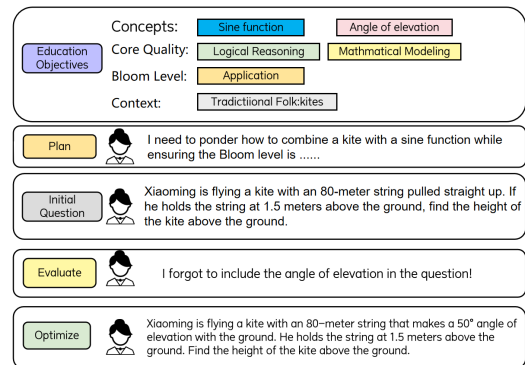


Figure 1: A simple example of question design based on educational objectives

students' logical thinking and problem-solving abilities (Kurdi et al., 2020; Liu et al., 2019). When designing such questions, educators coordinate multiple educational objectives (e.g., concepts, mathematical qualities), not only to accurately evaluate students' degree of knowledge mastery and application skills but also to enhance their comprehensive problem-solving capabilities through progressive challenges. Figure 1 illustrates the systematic process of how teachers design test questions (Kliebard, 1970; Wiggins, 2005). Starting with educational objectives, teachers carefully consider multiple dimensions including core concepts, competencies to be assessed, and real-world scenarios. They strategically combine these elements to create engaging questions that connect theoretical knowledge with practical applications. Throughout this process, teachers continuously evaluate whether their questions adequately cover all intended objectives and make necessary refinements.

However, in the literature, the generation of mathematics questions that consider multiple educational objectives has not received sufficient attention. In widely studied mathematical question generation, previous work has relied solely on manually crafted templates for generation (Polozov et al.,

2015; Khodeir et al., 2018), unable to generate content based on educational objectives expressed in natural language. Conversely, most more advanced works employ seq2seq models for generation (Zhou and Huang, 2019; Liu et al., 2020), but due to model limitations, their applications remain confined to question generation based on mathematical formulas such as arithmetic operations and equation systems. With the emergence of large language models like ChatGPT (Kojima et al., 2022) and LLaMA (Dubey et al., 2024), which significantly enhance the ability to generate diverse and complex content, researchers have begun using few-shot learning or Chain-of-Thought templates for question generation, framing it as a goal-based reasoning task. However, question generation is a complex task, and single-step reasoning may lead to failures, as seen in the teacher’s first attempt in Figure 1. Consequently, the limitations of single-step reasoning—its inherent inability to handle complex problems requiring multi-step inferences and its lack of reflective capabilities to check for errors—significantly hinder the full potential of large language models in question generation.

To validate the use of large language models for creating educational questions, we empirically assess whether these models can match human teachers’ ability to design problems that achieve specific educational objectives. However, existing mathematics question generation datasets present significant limitations. Most datasets focus primarily on elementary-level mathematics (e.g., LMWP (Liu et al., 2020), HMWP (Qin et al., 2020)), offering a narrow scope of assessment. Furthermore, a lack of comprehensive annotation is prevalent; the majority of these datasets (e.g., Gaokao-bench (Zhang et al., 2023), GSM8K (Cobbe et al., 2021)) lack annotations for educational objectives, hindering a thorough evaluation of LLMs’ capabilities in mathematical question generation.

In this paper, to advance the field further, we present two datasets, EduMath-SQ and EduMath-CQ, which are filtered from real high school test papers and annotated with educational objectives. Specifically, for each question, we annotate four to five categories of educational objectives based on Tyler’s rationale (Kliebard, 1970) and two-way specification table (Odiagbe, 2016), aiming to comprehensively evaluate models’ objective-based mathematical question generation capabilities.

Along this direction, several technical challenges remain. First, there is a lack of evaluation metrics.

Past question generation work has adopted text quality metrics such as BLEU and ROUGE. For instance, in Figure 1, while the initial and modified questions show high textual similarity, the initial question lacks a concept compared to the latter. However, these traditional text generation evaluation metrics often fail to effectively measure mathematical question generation quality, particularly in terms of educational objective alignment. Second, questions with multidimensional educational objectives typically cannot be generated in a single attempt, as the complexity of multiple objectives often leads to certain objectives being overlooked or missed in a single generation attempt, requiring repeatedly evaluated and optimized by educators. Additionally, a single dimension of an educational objective may encompass multiple components—as illustrated in Figure 1, where the "concepts" dimension includes both sine functions and elevation angles—further increasing the challenge of accurately generating questions that satisfy multidimensional educational objectives.

To address these challenges, we first propose EQGEVAL, a novel evaluation benchmark based on the EduMath dataset that comprehensively evaluates question generation quality through LLM using three metrics: solvability (question feasibility), Pass Rate (objective fulfillment), and Win Rate (comparison with gold standards). Furthermore, in real-world educational settings, teachers typically design mathematical questions through an iterative process, repeatedly evaluating and optimizing questions based on educational objectives and student feedback. Inspired by this practice, we develop a methodology that mirrors this systematic refinement process. Specifically, we develop the Educational Question Planning with self-Reflection (EQPR) method, which combines Monte Carlo Tree Search (MCTS) with large language models to simulate diverse question design strategies, achieve educational objectives, and systematically evaluate and refine the question creation process through continuous feedback.

In summary, our contributions are:

- We introduce a comprehensive dataset derived from authentic educational assessments, together with EQGEVAL, a novel benchmark framework designed to evaluate large language models’ proficiency in generating questions that are precisely aligned with specified educational objectives.

- Our framework, EQPR, integrates Monte Carlo Tree Search (MCTS) for strategic planning with a reflection mechanism that evaluates question quality and proposes improvements, enabling iterative refinement of the generated questions.
- Empirical validation through extensive experimentation underscores our framework’s efficacy. When implemented with two distinct LLMs and evaluated on the EQGEVAL benchmark, EQPR achieved superior performance across nearly all evaluation metrics, establishing new state-of-the-art results.

2 Related Works

2.1 Question Generation

Question generation is a significant research direction in educational technology (Kurdi et al., 2020; Zhao et al., 2024a,b). Its core function lies in automatically generating questions from structured or natural language text, such as deriving questions from dialogue content (Guo et al., 2024; Liu et al., 2024) or extracting them from story texts (Li and Zhang, 2024). The key value of this technology is that it can substantially reduce the time and cost of manual design and construction of questions, while also dynamically generating questions of varying difficulty and type based on the content.

In mathematics education, question generation has been a significant area of research. Early work focused on generating mathematical questions based on mathematical formulas and scenarios, relying on predefined templates and rule-based methods. Researchers developed language knowledge bases and rhetorical structure rules to aid in question generation (Khodeir et al., 2018). As natural language processing evolved, the field shifted toward neural network-based approaches. Sequence-to-sequence frameworks for mathematical question generation (Zhou and Huang, 2019) integrated thematic and formulaic information using attention mechanisms. Building on pre-trained language models, later work improved topic word selection and introduced question-solving modules to enhance the solvability of generated questions (Wang et al., 2021). MapKG (Qin et al., 2023a), inspired by educational experts’ test design experience, advanced the field with a "plan-then-generate" strategy that incorporated dual attention mechanisms and knowledge graphs to improve question solvabil-

ity and diversity. Recent developments in large language models have opened new research avenues, with methods such as gradient-based techniques being used to generate questions with controlled difficulty levels (Lin et al., 2024). However, existing studies have largely overlooked the generation of questions that address multi-dimensional educational objectives. Our proposed EQPR model aims to fill this gap.

2.2 LLM Planning and Reflection

Recent research on planning with large language models (LLMs) has witnessed remarkable advancement, significantly impacting domains such as common-sense reasoning (Zhu et al., 2023; Xue et al., 2024) and embodied intelligence (Sun et al., 2024). A central focus has been enhancing LLMs’ capacity for systematic, step-by-step problem solving. Foundational techniques like Chain-of-Thought (CoT) prompting (Wei et al., 2022) have established the paradigm of incremental problem decomposition, while more sophisticated approaches such as Tree-of-Thoughts (ToT)(Yao et al., 2024) and Reasoning via Planning (RAP)(Hao et al., 2023) explore solution spaces through hierarchical tree structures, leveraging methodologies like Monte Carlo Tree Search to systematically expand the search space. However, extensive iterative planning processes can introduce cumulative error propagation. To address this limitation, Self-Refine (Madaan et al., 2024) incorporates iterative feedback mechanisms that enable models to reflect upon and refine their reasoning processes, thereby mitigating error accumulation. Distinguished from these planning-focused methodologies, our work introduces EQPR, which synergistically combines iterative planning with reflective mechanisms to optimize question generation processes.

3 Problem Formulation

As outlined in Section 1, our objective is to leverage large language models for generating questions that align with specified educational objectives O , encompassing fundamental goals (e.g., conceptual understanding, Bloom’s taxonomy levels) and their contextual frameworks (e.g., traditional cultural narratives, sports scenarios), formalized as $q = LLM(O)$.

However, single-pass question generation often yields limited quality and inadequate alignment

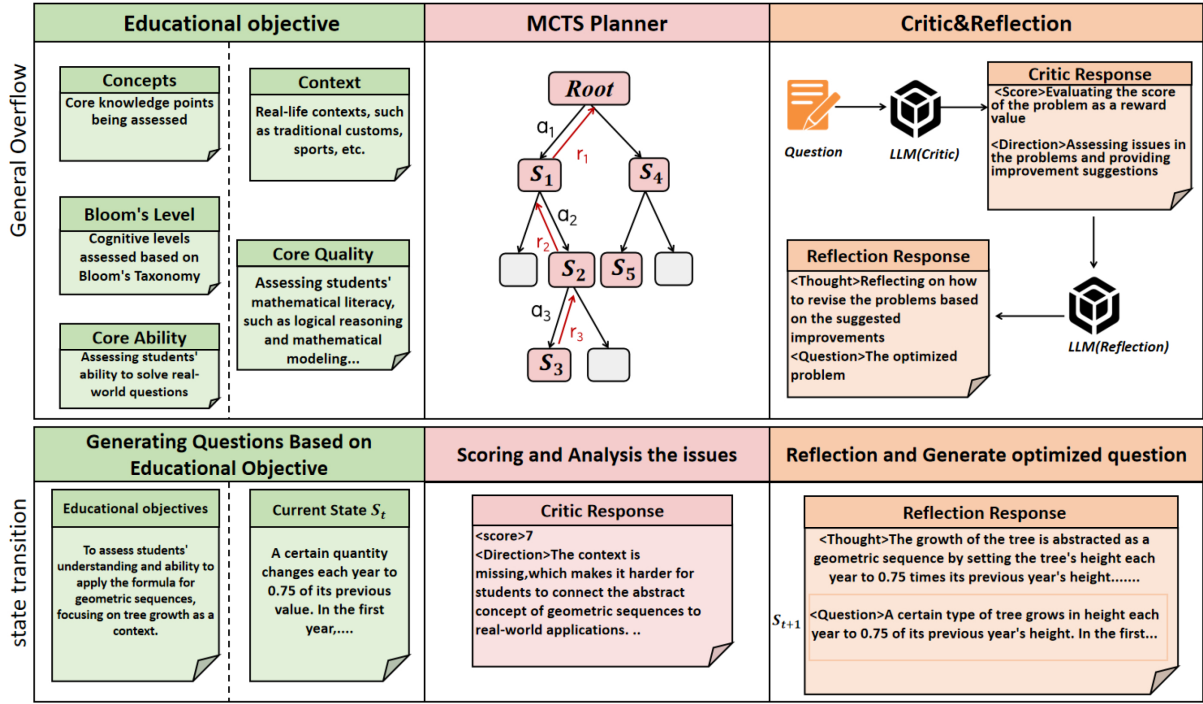


Figure 2: The overall framework of EQPR. The upper part of the figure illustrates the entire EQPR process, which uses an MCTS structure for deep reasoning and iteratively improves question quality via the Critic and Reflection modules. The lower part shows a simplified state transition example, where a previous question is modified based on feedback and progresses to the next state.

with educational objectives. To address this limitation, we draw inspiration from teachers' iterative improvement processes, modeling this as a sequential decision-making problem and formalizing it as a Markov Decision Process (MDP).

We formalize this question improvement process as an MDP defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, R)$, where:

- **State space \mathcal{S} :** Each state $s_t \in \mathcal{S}$ represents the current version of an educational question at time step t , denoted as q_t . States may include metadata such as alignment with educational objectives O , revision history, or evaluative metrics.
- **Action space \mathcal{A} :** Each action $a_t \in \mathcal{A}$ corresponds to an improvement operation targeting specific dimensions of the question, such as “enhancing conceptual clarity” or “increasing real-world relevance.” As shown in the upper part of Figure 2, these actions are formulated as structured feedback that guides iterative improvement, offering detailed analysis of current shortcomings paired with actionable improvement strategies.
- **Reward function $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$:** After executing action a_t in state s_t , the system re-

ceives a reward based on the revised question's quality. This reward reflects multiple educational criteria including conceptual clarity, cognitive depth, and contextual appropriateness, assessed by a *critic module* providing both numerical scores and formative feedback (detailed in Section 4.2).

- **Transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$:** The system transitions to a new state s_{t+1} after applying action a_t to the current question s_t . This process is implemented by a *reflection module*, which interprets the critic's feedback and generates an improved question accordingly (detailed in Section 4.3).

As illustrated in Figure 2, the process operates as follows: Given a current state s_t and educational objectives O , the critic module analyzes the question and samples an improvement action $a_t \sim \text{Critic}(a | s_t, O)$. Simultaneously, it assigns a reward r_t and identifies specific improvement areas. This feedback is passed to the reflection module, which translates it into actionable revision instructions and generates an enhanced question better aligned with educational objectives. For instance, if the action is to “increase cognitive depth,” the reflection module may introduce more abstract or higher-order reasoning components. The newly

generated question becomes the next state s_{t+1} , and the process continues iteratively until optimal educational alignment is achieved.

4 Method

4.1 Overview

Drawing inspiration from educators’ systematic approach to question development (Wang et al., 2022) - which encompasses planning, writing, evaluating, and optimizing - we introduce an innovative framework for generating educational questions that ensures both quality and alignment with educational objectives. The framework consists of three modules: the Critic module, which evaluates each generated question (state) based on multiple dimensions of educational objectives and provides directions for modification; the Reflection module, which analyzes the feedback from the Critic to determine optimization directions and refine the question generation process, leading to the creation of new questions; and the MCTS-based Planning module, directed by the Critic and Reflection modules, which provide the necessary guidance and constraints to enable it to systematically and efficiently navigate the vast and multifaceted search space, thereby exploring a wide range of potential question structures and formulations.

4.2 Critic

We employ the Critic module to systematically evaluate the alignment between generated questions and educational objectives. Through Large Language Models (LLMs) equipped with Critic prompts (see Table 8) and Chain-of-Thought reasoning, the module performs comprehensive assessment across multiple dimensions. Specifically, for a state s_t at phase t and educational objective O , the Critic generates question modification directions a_t and $score_t$ as follows:

$$(score_t, a_t) = Critic(s_t, O), \quad (1)$$

Here, $score_t$ represents the quantitative evaluation score, while a_t denotes the suggested modification directions. The evaluation examines concept coverage, contextual relevance, conceptual coherence, etc., ensuring comprehensive assessment of educational requirements, and logical interconnections. This structured evaluation framework enables precise identification of gaps between generated questions and desired educational objectives, facilitating targeted improvements in subsequent iterations.

4.3 Reflection

Cognitive science research demonstrates that humans continuously iterate and reflect upon their thinking based on new information (Frederick, 2005), allowing them to correct errors and deepen understanding. This reflective process is particularly evident in educational settings, where teachers systematically refine questions through iterative evaluation and improvement to align with educational objectives.

Consider the example illustrated in the bottom part of Figure 2, where the initial question generation lacks contextual materials. Through reflection, this limitation is identified, leading to an improved iteration that incorporates a practical tree-planting scenario, making the mathematical concept more accessible and applicable.

To formalize this reflective iteration process, we introduce a Reflection module powered by large language models. This module employs specialized reflection prompts (detailed in the Table 9) to analyze the historical trajectory of question states and improvement suggestions, ensuring alignment with educational objectives. In the iteration step t , we construct the historical trajectory τ as:

$$\tau = \{(s_0, a_0), (s_1, a_1), \dots, (s_{t-1}, a_{t-1})\}, \quad (2)$$

where s_0 represents the initial question state generated based on the initial generation prompt (see Table 7) at the root node, a_0 denotes the Critic module’s modification suggestions for initial question s_0 , s_t indicates the preceding state of the current node, and a_t represents the Critic module’s modification suggestions for question s_t . This trajectory captures the complete history of question states and their corresponding improvement suggestions up to the current iteration. The next state generation is accomplished through the Reflection module:

$$s_{t+1} = Reflection(O, \tau, s_t, a_t), \quad (3)$$

where O represents the educational objectives.

This formalized reflection mechanism enables the question generation process to emulate human teachers’ approach, continuously iterating to enhance problem quality.

4.4 MCTS-based Planning

We combine large language models (LLMs) with Monte Carlo Tree Search (MCTS) to better meet

our educational objectives. MCTS efficiently explores the question-generation space, balancing exploration and exploitation to improve the educational value of the resulting questions. As shown in the top panel of Figure 2, we model the process as a search tree: each node represents a partially generated question, and each edge denotes an editing action—such as extending or rewriting the text. The overall planning loop follows the standard MCTS phases of selection, expansion, simulation, and backpropagation, detailed in Algorithm 1 in the Appendix.

Selection. The selection phase chooses the most promising node from the tree’s branches for further exploration. Starting from the root node s_0 (initial question), the selection phase iteratively chooses the most promising nodes for exploration. To balance between known high-quality question structures (exploitation) and exploring new directions for improvement (exploration), we use the well-known Upper Confidence Bounds applied to Trees (UCT) (Kocsis and Szepesvári, 2006) for node selection, as shown below:

$$UCT(s_t, a_t) = Q(s_t, a_t) + c\sqrt{\frac{\ln N(s_t)}{N(ch(s_t, a_t))}}, \quad (4)$$

where $Q(s_t, a_t)$ is the potential future reward of applying action a_t at time t , $N(s_t)$ is the number of visits to node s_t , $ch(s_t, a_t)$ is the child node reached after executing action a_t in state s_t , and c is a constant used to adjust exploration. At each level of the tree, the child node with the highest UCT value is selected.

Expansion. During the expansion phase, we utilize the Reflection module to generate new question states. This module leverages the historical optimization trajectory τ to analyze patterns from previous modifications, generating new candidate questions according to: $s_{t+1} = Reflection(O, \tau, s_t, a_t)$. To explore a wider range of question designs, multiple nodes are generated as child nodes in each expansion step.

Simulation. From the expanded nodes, the simulation phase explores potential question optimization paths via simulations. In each simulation, the system evaluates options based on estimated cumulative rewards, selecting the highest-reward path for further exploration. This continues until a terminal state, yielding a comprehensive simulation of potentially effective question optimization schemes.

Backpropagation. When a simulation reaches a terminal state, backpropagation begins, using the

cumulative future reward obtained at the terminal node (illustrated by the red arrows in the central MCTS tree section of Figure 2) to update the Q-values of each state-action pair, with the aim of refining future question selection.

5 Educational QG Dataset & Benchmark

Existing mathematical question generation datasets primarily focus on elementary-level content and often lack comprehensive educational annotations. To address these limitations, we developed EduMath, aiming to create a high-quality dataset with annotation accuracy exceeding 90% across all educational dimensions. We developed this multi-dimensional annotation framework based on the following considerations: concept mapping reflects knowledge coverage, ability assessment captures problem-solving requirements, Bloom’s Taxonomy represents cognitive levels, mathematical literacy evaluation aligns with educational objectives, and real-world context identification demonstrates practical value. These dimensions characterize the educational attributes of mathematical questions from distinct perspectives. We sourced 16k high-school level mathematics problems from mock exams and college entrance examinations, implementing strict quality filters to exclude problems with images or incomplete solutions. Using DeepSeek-V3, we conduct an iterative three-round annotation process. Initial annotations are reviewed for accuracy and consistency. Specifically, we employ Chain-of-Thought prompting to guide multiple large language models in evaluating annotation correctness through a voting mechanism. Annotations flagged as inaccurate are re-annotated to resolve identified issues. This rigorous, multi-stage process results in an annotation accuracy of 95.2% across all dimensions. The final dataset includes two variants: EduMath-SQ (Standard Questions) and EduMath-CQ (Contextual Questions), with EduMath-SQ focusing exclusively on non-contextual problems.

Building on this, we define controllable educational question generation and corresponding evaluation metrics. Previous question generation approaches typically rely on text quality metrics such as BLEU and ROUGE for evaluation, but these are far from sufficient. Truly effective questions must be solvable and meet teachers’ instructional and assessment needs. Following TOOLEVAL (Qin et al., 2023b) and based on DeepSeek-V3, we propose EQGEVAL, which includes the following metrics

(see Appendix A.4 for details): Solvability as a fundamental requirement that the generated problems must have valid solutions; Pass Rate measuring the proportion of generated problems that meet the educational objectives; and Win Rate where we present educational objectives and two problems to DeepSeek-V3 evaluators, asking them to determine which problem better serves the intended purpose. To ensure the reliability of our evaluation, all metrics are determined through a majority voting mechanism, where multiple independent evaluations are conducted to derive the final assessment results.

6 Experiment

6.1 Experimental Settings

Datasets. As outlined in Section 5, we conducted experimental evaluations using the EduMath-SQ and EduMath-CQ datasets. We randomly selected 10% of the data to serve as the test set. The EduMath-CQ dataset comprises 589 educational objectives paired with their corresponding gold-standard questions, while EduMath-SQ contains 1,034 educational objectives along with their respective gold-standard questions. For more detailed information about these datasets, please refer to Appendix A.1.

Baselines. Since this is a novel task, we approach it as a reasoning problem and benchmark it against established reasoning methods, including Chain-of-Thought (CoT) (Wei et al., 2022), CoT-BON, ReAct (Yao et al., 2023), and the tree-structured DEAR method (Xue et al., 2024). DeepSeek-V3 (Deepseek, 2024) and GPT-4o-Mini (OpenAI, 2024a) serve as our primary backbone models. Additionally, we evaluated Claude-3.5 (Anthropic, 2024) and GPT-4o (OpenAI, 2024b) on the EduMath-CQ dataset; for those results, please refer to Appendix B.

Metrics. We evaluate our approach using both automatic metrics and human assessment. For automatic evaluation, we primarily employ three metrics from the EQGEVAL framework, where the Win Rate metric compares generated questions against gold-standard problems. To comprehensively assess text quality, we also incorporate widely-adopted natural language generation metrics, including Rouge-L (Lin and Och, 2004), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and BERTSCORE (Zhang et al., 2020). For human evaluation, we design two

metrics: Fluency and Human-rated Win Rate. The Fluency metric uses a three-level scoring system (0-2), with detailed scoring criteria provided in the Appendix A.7.

6.2 Main Results

We conduct experiments to verify the effectiveness of our framework EQPR, and report the results in Table 1. We get the following observations. Primarily, our method demonstrates superior performance in Win Rate and Pass Rate across both EduMath-CQ and EduMath-SQ datasets, as well as across both GPT-4o-Mini and DeepSeek models. This comprehensive outperformance validates the effectiveness of our approach. Notably, when testing with DeepSeek on EduMath-CQ, our method achieved a Win Rate of 46.23%, surpassing the next-best method DEAR (41.8%) by a substantial margin of 4.42%. This improvement clearly demonstrates the efficacy of our iterative refinement strategy. Nevertheless, we observed that Pass Rates consistently decreased when transitioning from standard questions (EduMath-SQ) to contextual scenarios (EduMath-CQ), suggesting that large language models still face challenges in seamlessly incorporating contextual elements while addressing multiple educational objectives.

Regarding Solvability, our method achieves near-optimal performance, ranking second across both sub-datasets. The slight reduction in solvability scores can be attributed to the nature of our iterative refinement process, where questions naturally evolve to become more sophisticated through multiple iterations.

In terms of traditional text quality metrics, our method maintains competitive performance while prioritizing educational effectiveness. For instance, in the EduMath-SQ dataset with DeepSeek, the modest difference between our method’s BERTScore (74.79) and REACT’s (76.05) is acceptable, considering that these metrics primarily assess lexical and semantic similarities rather than educational value. Our superior Win Rate and Pass Rate scores underscore our method’s success in achieving its primary objective: generating educationally meaningful and high quality questions, even if this occasionally leads to slightly lower linguistic metric scores.

6.3 Human Evaluation Results

To thoroughly assess the effectiveness of our methodology, we carried out human evaluation

Table 1: Evaluation results on datasets EduMath-CQ and EduMath-SQ(%).

Dataset	Method	BLEU	METEOR	ROUGE-L	BERTSCORE	WIN RATE	SOLVABLE	PASS RATE
GPT-4o-Mini								
EduMath-CQ	COT	11.60	29.64	24.45	68.52	35.78	90.12	27.88
	COT-BON	8.52	22.59	10.56	69.19	34.50	90.23	21.25
	REACT	18.90	53.20	36.72	72.55	38.40	89.32	33.66
	DEAR	13.31	23.09	21.74	70.34	35.18	85.77	27.35
	EQPR	24.80	52.70	48.13	70.89	39.92	90.63	35.81
	DeepSeek-V3							
EduMath-CQ	COT	12.11	42.90	22.51	70.64	38.20	90.84	31.45
	COT-BON	13.58	43.60	28.93	70.85	38.99	90.25	29.63
	REACT	21.76	47.98	47.06	70.32	39.75	90.64	38.91
	DEAR	20.71	49.06	42.11	74.48	41.81	92.59	37.88
	EQPR	20.33	46.86	44.57	71.02	46.23	91.73	43.11
	GPT-4o-Mini							
EduMath-SQ	COT	1.76	21.04	7.64	67.30	32.20	83.65	64.60
	COT-BON	2.63	30.44	11.76	68.43	36.64	85.23	75.12
	REACT	36.72	52.19	36.72	72.55	36.81	82.90	83.22
	DEAR	8.51	20.52	4.71	66.34	36.70	81.89	82.36
	EQPR	32.77	52.75	52.78	68.83	37.18	84.40	84.37
	DeepSeek-V3							
EduMath-SQ	COT	28.80	56.91	50.33	75.38	37.40	87.15	85.46
	COT-BON	30.66	61.50	52.07	76.05	42.70	86.88	88.82
	REACT	26.46	60.47	46.05	75.45	44.45	85.95	89.76
	DEAR	24.52	58.23	40.07	73.48	40.36	87.14	82.74
	EQPR	29.71	57.29	49.63	74.79	45.65	91.31	91.50

Method	Clarity	Win Rate-Human(%)
COT	1.83	27.00
COT-BON	1.78	33.67
DEAR	1.84	31.00
REACT	1.87	35.00
EQPR	1.93	36.67
Fleiss' kappa	0.71	0.47

Table 2: Human evaluation results on EduMath-CQ(DeepSeek-V3).

experiments. We randomly selected 100 samples from the EduMath-CQ dataset and enlisted three highly educated evaluators, each with at least a bachelor’s degree, to conduct the assessments. The evaluation focused on two key dimensions: question clarity (whether the questions are easily readable and understandable) and quality comparison. Given the potential biases in large language models when determining win rates, the evaluators were asked to compare the quality of the generated questions against the gold-standard questions. As depicted in Table 2, our method demonstrated superior performance in both clarity scores and human-evaluated win rates. Furthermore, we evaluated the reliability of the annotations using Fleiss’ Kappa coefficient. The Kappa values for both clarity and quality comparison exceeded the credibility thresh-

old of 0.41, thereby confirming the reliability of our evaluation results.

6.4 Ablation Results

Effect of Reflection Module. To validate the effectiveness of the Reflection Module, we conducted a comparative experiment. After removing the Reflection Module, the model no longer optimizes based on iterative feedback but directly generates questions according to the target (referred to as the "w/o Reflection" experiment). The results showed significant performance degradation across both models: using DeepSeek as an example, illustrated in Figure 3, the Pass Rate decreased from 46.23% to 43.51%, and the Win Rate declined from 43.11% to 40.74%. These observations highlight the vital importance of the Reflection Module. By engaging in multiple rounds of iterative optimization, the system adeptly integrates knowledge from previously encountered questions, enabling it to produce content that more effectively aligns with educational objectives.

Effect of MCTS-based Planning. To quantify the contribution of MCTS-based planning within our proposed framework, we conducted an ablation study by removing the MCTS component and employing greedy search exclusively for action selection. As demonstrated in Figure 3, the model’s per-

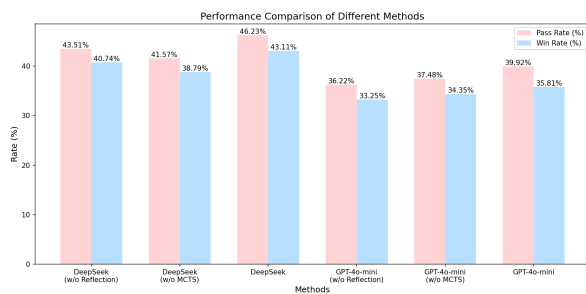


Figure 3: The results of ablation studies. We test different methods on EduMath-CQ datasets

formance deteriorated substantially without MCTS-based planning. Specifically, DeepSeek experienced a notable decline in both pass rate (from 46.23% to 41.57%) and win rate (from 43.11% to 38.79%) when operating without Monte Carlo Tree Search. We attribute this performance degradation to MCTS-based planning’s superior capability in navigating the question optimization space through its dual mechanism of prospective outcome prediction and retrospective evaluation. These empirical findings validate the critical role of MCTS integration in our framework’s effectiveness.

7 Conclusion

In this paper, we introduced EQPR (Educational Question Planning with self-Reflection), an innovative framework for generating high-quality mathematical questions that align with educational objectives. EQPR integrates a "plan-evaluate-optimize" process, combining Monte Carlo Tree Search with the generative power of LLMs, enabling continuous refinement through feedback optimization. We also introduced EduMath, a high-quality dataset of 16k mathematics problems, and EQGEVAL, a comprehensive framework for evaluating the educational value of generated questions. Extensive experiments demonstrate that EQPR outperforms existing reasoning methods on key educational metrics across multiple large language models.

Acknowledgement

This research was partially supported by the National Science and Technology Major Project(No.2022ZD0117103), the National Natural Science Foundation of China (Grants No.62477044), Anhui Provincial Natural Science Foundation (No. 2308085QF229), the Fundamental Research Funds for the Central Universities (No.WK2150110038). Zhenya Huang gratefully acknowledges the support of the Young Elite

Scientists Sponsorship Program by CAST (No. 2024QNRC001)

Limitations

Our research primarily focused on mathematical question generation and has not yet been extended to other subject areas. This presents an important direction for future research. Additionally, we face certain challenges in establishing educational objectives, particularly in the assessment of question difficulty. Since difficulty evaluation is largely subjective and challenging to standardize, this remains a significant hurdle in the field that requires further investigation. Furthermore, while we utilize large language models for evaluation, these models may exhibit certain biases, and their assessment results do not always align perfectly with the professional judgment of human educators. Consequently, exploring effective methods to align human evaluation with LLM-based assessments and establishing a more accurate evaluation system remains a crucial direction for future research.

References

- Anthropic. 2024. Introducing claude 3.5 sonnet. <https://www.anthropic.com/news/claude-3-5-sonnet>.
- Satanjeev Banerjee and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005*, pages 65–72.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Deepseek. 2024. DeepSeek-V3. <https://huggingface.co/deepseek-ai/DeepSeek-V3>.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Shane Frederick. 2005. Cognitive reflection and decision making. *Journal of Economic perspectives*, 19(4):25–42.
- Shasha Guo, Lizi Liao, Jing Zhang, Cuiping Li, and Hong Chen. 2024. Pcpqr: Proactive conversational question planning with reflection. *arXiv preprint arXiv:2410.01363*.

- Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*.
- Nabila Ahmed Khodeir, Hanan Elazhary, and Nayer Wanas. 2018. Generating story problems via controlled parameters in a web-based intelligent tutoring system. *The International Journal of Information and Learning Technology*, 35(3):199–216.
- Herbert M Kliebard. 1970. The tyler rationale. *The School Review*, 78(2):259–272.
- Levente Kocsis and Csaba Szepesvári. 2006. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.
- Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, 30:121–204.
- Kunze Li and Yu Zhang. 2024. Planning first, question second: An llm-guided method for controllable question generation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4715–4729.
- Chin-Yew Lin and FJ Och. 2004. Looking for a few good metrics: Rouge and its evaluation. In *Ntcir workshop*, pages 1–8.
- Fan Lin, Shuyi Xie, Yong Dai, Wenlin Yao, Tianjiao Lang, Zishan Xu, Zhichao Hu, Xiao Xiao, Yuhong Liu, and Yu Zhang. 2024. Idgen: Item discrimination induced prompt generation for llm evaluation. *arXiv preprint arXiv:2409.18892*.
- Jiayu Liu, Zhenya Huang, Tong Xiao, Jing Sha, Jinze Wu, Qi Liu, Shijin Wang, and Enhong Chen. 2024. Socraticlm: Exploring socratic personalized teaching with large language models. *Advances in Neural Information Processing Systems*, 37:85693–85721.
- Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):100–115.
- Tianqiao Liu, Qiang Fang, Wenbiao Ding, Hang Li, Zhongqin Wu, and Zitao Liu. 2020. Mathematical word problem generation from commonsense knowledge graph and equations. *arXiv preprint arXiv:2010.06196*.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.
- Sunny Ighalo Odiagbe. 2016. Table of specification: Ensuring content validity of teacher-made-test among senior secondary schools teachers in kwali area council. *Journal of the Nigerian Academy of Education Vol*, 13(1):69.
- OpenAI. 2024a. GPT-4o mini: Advancing cost-efficient intelligence. <https://openai.com/index/gpt-4o-mini>.
- OpenAI. 2024b. Hello gpt-4o. <https://openai.com/index/hello-gpt-4o/>.
- 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, ACL 2002*, pages 311–318.
- Oleksandr Polozov, Eleanor O’Rourke, Adam M Smith, Luke Zettlemoyer, Sumit Gulwani, and Zoran Popović. 2015. Personalized mathematical word problem generation. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
- Jinghui Qin, Lihui Lin, Xiaodan Liang, Rumin Zhang, and Liang Lin. 2020. Semantically-aligned universal tree-structured solver for math word problems. *arXiv preprint arXiv:2010.06823*.
- Longhu Qin, Jiayu Liu, Zhenya Huang, Kai Zhang, Qi Liu, Binbin Jin, and Enhong Chen. 2023a. A mathematical word problem generator with structure planning and knowledge enhancement. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1750–1754.
- Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. 2023b. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.
- Haotian Sun, Yuchen Zhuang, Lingkai Kong, Bo Dai, and Chao Zhang. 2024. Adaplanner: Adaptive planning from feedback with language models. *Advances in Neural Information Processing Systems*, 36.
- Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. 2023. Promptagent: Strategic planning with language models enables expert-level prompt optimization. *arXiv preprint arXiv:2310.16427*.
- Xu Wang, Simin Fan, Jessica Houghton, and Lu Wang. 2022. Towards process-oriented, modular, and versatile question generation that meets educational needs. *arXiv preprint arXiv:2205.00355*.

- Zichao Wang, Andrew S Lan, and Richard G Baraniuk. 2021. Math word problem generation with mathematical consistency and problem context constraints. *arXiv preprint arXiv:2109.04546*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- Grant Wiggins. 2005. Understanding by design. *Association for Supervision and Curriculum Development*.
- Shangzi Xue, Zhenya Huang, Jiayu Liu, Xin Lin, Yuting Ning, Binbin Jin, Xin Li, and Qi Liu. 2024. Decompose, analyze and rethink: Solving intricate problems with human-like reasoning cycle. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.
- Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020*.
- Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. 2023. Evaluating the performance of large language models on gaokao benchmark. *arXiv preprint arXiv:2305.12474*.
- Guanhao Zhao, Zhenya Huang, Yan Zhuang, Haoyang Bi, Yiyan Wang, Fei Wang, Zhiyuan Ma, and Yixia Zhao. 2024a. A diffusion-based cognitive diagnosis framework for robust learner assessment. *IEEE Transactions on Learning Technologies*.
- Hongke Zhao, Likang Wu, Yuqing Shan, Zonghan Jin, Yuanpei Sui, Zipeng Liu, Nan Feng, Minqiang Li, and Wei Zhang. 2024b. A comprehensive survey of large language models in management: Applications, challenges, and opportunities. *Challenges, and Opportunities (August 14, 2024)*.
- Qingyu Zhou and Danqing Huang. 2019. Towards generating math word problems from equations and topics. In *Proceedings of the 12th international conference on natural language generation*, pages 494–503.
- Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2023. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations*.

A More Experiment Details

A.1 Dataset Details

As shown in the Table 4, we present the number of questions for each dataset along with the average number of concepts covered per question. On average, each question involves more than two concepts, indicating that we intend to use these datasets to evaluate the ability of large language models to generate questions that address complex educational objectives. To this end, we randomly sampled 10% of the questions from each dataset to serve as a test set.

A.2 Explanation of the Educational Objectives

- **Concept:** Understand and master core mathematical concepts such as trigonometric functions, sequences, and probability.
- **Core Quality:** Develop essential skills such as logical reasoning, mathematical modeling, and problem-solving to tackle complex mathematical tasks.
- **Core Ability:** Develop the ability to choose and apply appropriate mathematical techniques—for instance, recognizing when to use identities like $\sin^2 x + \cos^2 x = 1$ in solving problems.
- **Bloom Level:** Design questions that intentionally target specific cognitive levels in Bloom’s Taxonomy, such as application, analysis, or creation.
- **Context:** Enable students to interpret and solve mathematical problems within real-world scenarios and authentic contexts.

A.3 Dataset Comparison

We conducted a comprehensive comparison between our dataset and existing mathematics question generation datasets. As shown in Table 3, Our EduMath dataset demonstrates several significant advantages. First, the scale of our dataset substantially surpasses other comparable datasets in terms of problem quantity. Second, our dataset specifically focuses on high school mathematics problems that require deeper cognitive reasoning, whereas datasets like LMWP and HMWP primarily target elementary-level mathematics question generation. The generation of high school mathematics problems presents considerably greater challenges due

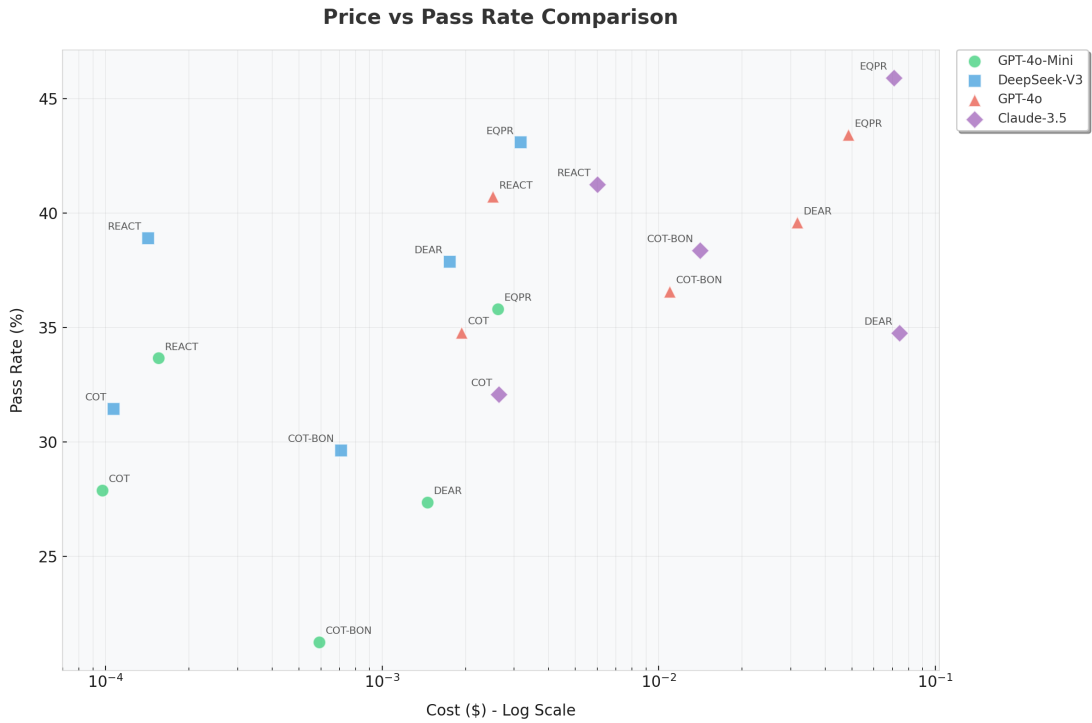


Figure 4: The cost refers to the average cost required to generate a question on the EduMath-CQ dataset.

to their complexity and reasoning requirements. Furthermore, our dataset features meticulous annotations performed by large language models (LLMs). We implemented a rigorous two-stage annotation process: after the initial annotation phase, we employed LLMs to verify the accuracy of the annotations, with any identified inaccuracies undergoing a re-annotation process. This makes EduMath the first and only open-source high school mathematics dataset that incorporates multi-dimensional educational objective annotations, setting a new standard for educational resource development.

A.4 Educational Metric Details

Solvable We employed the state-of-the-art large language model, DeepSeek-V3, to solve the generated questions. We then determined the solvability of each question using a majority voting approach based on self-consistency (five samples).

Pass Rate We employ Chain of Thought (COT) reasoning by inputting both educational objectives and generated questions into the large language model, enabling step-by-step analysis of whether the question meets each educational objective. Using the Self-Consistency approach, we generate 5 independent judgment samples and determine the

final result through majority voting. If a question fails to meet any educational objective, it is marked as failing.

Win Rate The Win Rate metric evaluates question quality by inputting pairs of questions along with their educational objectives into the large language model to determine which is superior. The evaluation criteria include adherence to educational objectives, natural language flow, and seamless integration of context. As shown in Table 10, we provide a specific case study demonstrating this judgment process.

A.5 Implementation Details

Parameter Details. For LLM parameter settings, we maintained a consistent temperature of 0.7 across all four models to ensure output diversity. The model versions used in our testing include gpt-4o-mini-2024-07-18, Claude-sonnet-3.5-0622, GPT-4o-2024-11-20, and DeepSeek-V3. Regarding method parameters, for COT-BOT (Best of N), we generated 5 candidate outputs and selected the best result; for our proposed EPQR method, the Monte Carlo Tree Search (MCTS) parameters were configured as follows: 4 iterations, maximum depth of 3, and an exploration parameter c of 2.5 in the

Table 3: Comparison of Different Mathematics Problem Datasets

Datasets	EduMath (Ours)	GAOKAO (Zhang et al. (2023))	LMWP (Liu et al. (2020))	HMWP (Qin et al. (2020))	GSM8k (Cobbe et al. (2021))
Deep Reasoning	✓	✓	✗	✗	✗
Objectives Annotation	✓	✗	✓	✓	✗
Multi-Edu objectives	✓	✗	✗	✗	✗
#Problems	16348	300	5447	5470	8500

Table 4: Number of Questions and Average Number of Concepts for Two Datasets

Dataset	# Questions	Avg. # Concepts
EduMath-SQ	10763	2.57
EduMath-CQ	5585	2.39

UCT formula.

Output strategy. Each iteration of the Monte Carlo Tree Search (MCTS) yields a path from the root node to a leaf node. Following PromptAgent (Wang et al., 2023), we select the path with the highest average question reward and then choose the question with the highest reward from that path as the final output. This strategy ensures that we identify the best question from the overall optimal search trajectory.

A.6 Baseline Details

- **Chain-of-thoughts (CoT) (Wei et al., 2022):** Prompts language models to think step-by-step before reaching final conclusions, incorporating deliberate reasoning and systematic thinking to generate more powerful and insightful answers.
- **Chain-of-thoughts Best of N (Wei et al., 2022):** Samples multiple CoT outputs and selects the best one from the generated candidates.
- **ReAct (Yao et al., 2023):** Simulates human problem-solving patterns through reasoning and action steps, enabling large language models to better understand tasks, gather information, execute operations, and correct errors, thereby significantly improving their performance on complex tasks. For the education question generation task, we adopt a thought + action approach, where the model generates a thought before producing the final question.
- **Dear (Xue et al., 2024):** A human cognition-inspired reasoning framework that builds a reasoning tree through a three-stage cycle.

Table 5: Additional Evaluation results on datasets EduMath-CQ(%).

Method	WIN RATE	PASS RATE
GPT-4o		
COT	38.16	34.78
COT-BON	34.34	36.58
REACT	42.68	40.72
DEAR	41.30	39.59
Ours	44.07	43.43
Claude-3.5		
COT	37.95	32.08
COT-BON	41.61	38.37
REACT	45.27	41.26
DEAR	44.19	34.75
Ours	47.11	45.92

It decomposes complex problems into sub-problems in the Decompose stage, generates and self-checks reasoning processes for each sub-problem in the Analyze stage, and updates parent node reasoning based on child node results in the Rethink stage, thereby enhancing large language models’ complex reasoning capabilities.

A.7 Human Evaluation Criteria

- 0: Incomprehensible - The question is confusing and impossible for students to understand, making it impossible to answer.
- 1: Partially Clear - Students can grasp the core idea of the question and attempt to answer, but the question still needs improvement.
- 2: Completely Clear - The question is concise, clear, easy to understand, and allows students to answer smoothly.

B Additional Results

B.1 Evaluation of Methods on More LLMs

To validate the model-agnostic nature of our framework, we conducted experiments on EduMath-CQ using GPT-4o and Claude-3.5 as backbone models,

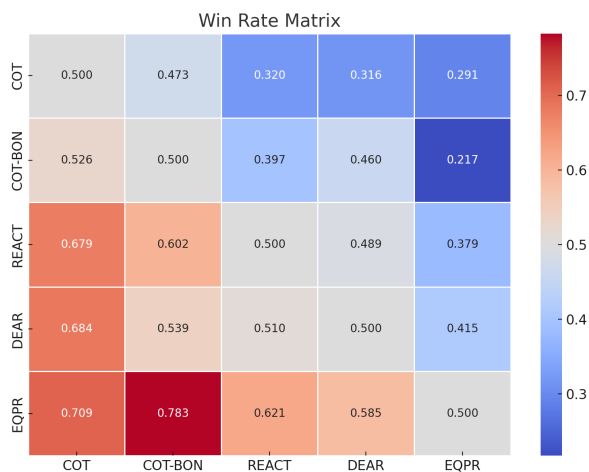


Figure 5: Win Rate Comparison Matrix Across Different Methods

with results presented in Table 5. The experimental results demonstrate that our framework consistently outperforms the baseline across both models in terms of Win Rate and Pass Rate metrics. This consistent superior performance indicates that our framework’s effectiveness is model-agnostic and can be successfully applied across various large language models. Additionally, Claude-3.5 demonstrates the strongest performance among the four models, likely attributed to its enhanced reasoning capabilities and superior instruction-following abilities. This finding further indicates that the performance of the foundation model has a significant impact on the overall effectiveness of the framework.

B.2 Cost Analysis

We presented a comparative analysis of four models on the EduMath-CQ dataset, examining both the average cost per question generation and the corresponding pass rates. The Figure 4 demonstrates that all models performed better under the EQPR method compared to the baseline. Notably, DeepSeek achieved exceptional cost-effectiveness, attaining the third-highest pass rate at less than \$0.01 per question, significantly lower than the costs of Claude-3.5 and GPT-4o. While the Claude-3.5-based EQPR solution had the highest cost, it achieved the best pass rate, with expenses still remaining under \$0.1 per question. Given that the cost of human-generated questions would be considerably higher than this amount, we consider this investment to be justified.

B.3 Win Rate Comparison of Methods

To evaluate the performance of these five methods in a more fine-grained manner, we conducted pairwise comparisons on the EduMath-CQ dataset based on DeepSeek-V3’s results and calculated their respective Win Rates, as detailed in the Figure 5. The results demonstrate that our proposed method, EQPR, outperforms other methods in terms of Win Rate, highlighting its superiority. Specifically, EQPR exhibits significant advantages even when compared to strong baseline methods. For instance, EQPR achieved Win Rates of 58.5% and 62.1% against DEAR and REACT, respectively. This suggests that EQPR is more effective in guiding the model to generate high-quality questions that are both coherent and aligned with educational objectives.

B.4 case study

To better illustrate the process of question generation, critic, and revision, Table 6 presents a case study based on the learning objective of "categorizing random events." The system first generates an initial question based on predefined education objectives : "Which of the following describes a random event?..." The Critic module then evaluates the question, assigning a score of 6 and identifying several issues, such as the lack of a real-world context, failure to address impossible events, and limited cognitive depth. In response, the Reflection module proposes concrete revisions—for example, ensuring that the answer choices include random, certain, and impossible events, maintaining a single-choice format, and keeping the difficulty level moderate. Based on these suggestions, an improved version of the question and answer options is generated. This case study demonstrates how the question quality can be progressively refined to better align with instructional goals and cognitive requirements.

Table 6: Case study of an example generated by our method (Claude-3.5)

Education Objectives	
Education Objectives:	<ul style="list-style-type: none"> • Concept: Random events • Core Quality: Mathematical abstraction, logical reasoning • Core Ability: Distinguish between random, certain, and impossible events, and make probability judgments in real-life contexts • Context: Real-life scenarios involving random events (e.g., dice rolling, lottery, baby gender) to enhance relevance and interest while testing key concepts • Bloom’s Taxonomy: Understanding
Question	
Question:	<p>Which of the following describes a <i>random</i> event?</p> <p>A. It will rain tonight B. The sun will rise in the west tomorrow C. There are 12 months in a year D. A coin toss will definitely land heads up</p>
Updated	
Critic:	<p>score: 6.</p> <p>reason: Covers basic concept, but lacks real-life context; omits “impossible event”; low cognitive demand; insufficient instructional depth</p>
Reflection:	<p>thought: Include random, certain, and impossible options with plausible distractors; keep single-choice format and moderate difficulty; use parallel comparisons to guide classification by definition.</p> <p>question: Which of the following describes a <i>random</i> event?</p> <p>A. Rolling a fair die shows a 6. B. The sun will rise in the west tomorrow. C. There are 12 months in a year. D. A tossed coin will definitely land heads up.</p>

Algorithm 1 MCTS with Reflection

Require: Initial question(state) s (root), expand width k , exploration weight w , $max_iterations$, $depth_limit$, reflection module $reflection$, critic module $critic$
Initialize root node with initial state
for $i \leftarrow 1$ **to** $max_iterations$ **do**
 $path \leftarrow []$
 $node \leftarrow root$
 # Selection Phase
 while $node$ has children **do**
 $node \leftarrow \operatorname{argmax}_{child \in node.children} (child.uct)$
 Add $node$ to $path$
 Update $node.visited$
 end while
 # Expansion Phase
 if $node.depth < depth_limit$ **then**
 for $j \leftarrow 1$ **to** k **do**
 Generate new child through reflection:
 $question \leftarrow Reflection$
 $child \leftarrow create_node(question, parent = node)$
 Evaluate $child.reward$ using $Critic$
 Add $child$ to $node.children$
 end for
 end if
 # Simulation Phase
 while not terminal $node$ **do**
 Select child with the highest immediate reward
 Add $node$ to $path$
 Update $node.visited$
 end while
 # Backpropagation Phase
 $cumulative_reward \leftarrow 0$
 for $node$ in $reversed(path)$ **do**
 $cumulative_reward \leftarrow$
 $cumulative_reward + node.reward$
 Update $node.cum_rewards$ with
 $cumulative_reward$
 Calculate new $node.Q$ value
 end for
end for
return the best question from the path

Question Generation Prompt
<p>You are an expert in high school mathematics education. You are analyzing educational objectives to design and create a multiple-choice question. Your goal is to develop a well-structured question that aligns with specific educational objectives while fostering core competencies.</p> <p>Status Determination Rules: concepts: Ensure complete alignment with required content; Maintain logical rigor and clear progression; Consider cognitive levels Competency Development: Integrate core competencies naturally with content; Build connections between concepts Example: {few_shots} Output Format: {"question_design_thought": "detailed explanation of question design approach", "question": "complete multiple-choice question with options"} Required Input: Education_Objectives: {educational objectives}</p>

Table 7: Question generation prompt template.

Critic Evaluation Prompt
<p>You are an expert in high school mathematics education. Your task is to evaluate a mathematical question and its design approach based on given educational objectives. You will assess whether the question meets the educational objectives, provide a strict scoring evaluation, and analyze areas for improvement.</p> <p>Scoring Scale (1-10): Excellent (10): - Complete alignment with Concept and competency requirements - Clear, structured design following stated approach - Deep pedagogical design fostering core competencies - Appropriate cognitive level for students Good (8-9): - Generally meets educational objectives - Minor deviations from target cognitive level Average (6-7): - Meets content requirements but lacks competency development - Cognitive level misalignment with objectives Fair (4-5): - Only partially meets content requirements - Significant cognitive level misalignment Poor (1-3): - Severe deviation from educational objectives - Does not follow design approach - Low quality question Failing (0): - No connection to educational objectives - Completely unrelated to goals and design approach Key evaluation points: Concept Alignment: - Check for complete coverage of required content - Assess logical structure and progression - Evaluate cognitive level appropriateness Competency Development: - Analyze integration of core competencies - Assess effectiveness in building understanding Output Format: {"direction": "detailed analysis of weaknesses, and improvement suggestions", "score": numerical score 1-10} Required Input: Education_Objectives: {educational objectives}; Question: {current question}</p>

Table 8: Critic evaluation prompt template.

Reflection Prompt
<p>You are an expert in high school mathematics education. Your task is to analyze and optimize a math question and its design approach based on given educational objectives, previous feedback, and the question's evolution history. Your goal is to refine both the question design approach and the question itself to better meet educational objectives while maintaining high quality.</p> <p>Key Constraints:</p> <ul style="list-style-type: none"> - All questions must have valid solutions - Modifications should go beyond numerical changes - Changes must align with existing objectives - No new concepts or competencies can be added <p>Analysis Points:</p> <p>Previous Question Analysis: - Review strengths and weaknesses identified - Understand suggested improvements - Study evolution of previous versions</p> <p>Optimization Strategy: - Address identified weaknesses - Maintain existing strengths - Enhance alignment with objectives - Improve question quality</p> <p>Educational Alignment: - Verify concept coverage - Check competency development</p> <p>Output Format:</p> <pre>{"thought": "detailed explanation of optimized design approach", "question": "complete optimized question" }</pre> <p>Required Input: Education_Objectives: {educational objectives} Question: {current question} Direction: {current feedback and suggestions} Trajectory_thoughts: {history of previous versions}</p>

Table 9: Reflection prompt template.

Table 10: Example of the Input and Output for the Education Objectives Evaluation Module

Education Objectives	
Education Objectives:	<ul style="list-style-type: none"> • Concept: Permutation and Combination, Counting Principle • Core Quality: Mathematical Operations, Logical Reasoning • Context: School Course Scheduling (combining combinatorial mathematics with practical school management) • Core Ability: <ul style="list-style-type: none"> – Handle complex permutation and combination problems. – Apply classification counting and step-by-step counting techniques. – Solve practical problems using enumeration and classification addition principles. • Bloom’s Taxonomy: Application
Question Pairs	
Question 1:	A school has 3 courses to assign for the semester, and each course must be taught by one teacher. There are 2 available teachers to choose from for each course. How many possible ways can the teachers be assigned to the courses?
Question 2:	A school needs to arrange courses for the semester. There are 5 courses, and each course needs to be assigned a teacher. Assume each course can be taught by one of 3 teachers, and each teacher can teach at most two courses. How many ways can the courses be assigned to teachers?
Evaluation	
Better Question:	2
Reason:	Question 1 only covers the basic counting principle of assigning teachers to courses, without considering any constraints. It does not address the permutation aspect, which is an important part of the core educational objectives. On the other hand, Question 2 is more complete as it covers both permutation and combination, as well as the counting principle, by introducing a constraint (each teacher can teach at most two courses). This makes Question 2 a better fit for evaluating the student’s ability to apply counting techniques and solve combinatorial problems. Therefore, Question 2 is a more suitable and well-rounded problem for this purpose.

Pass Rate Evaluation Prompt
<p>Consider the following task as a mathematics education assessment assignment. You will act as a senior mathematics education expert, rigorously evaluating a given math problem and its associated educational objectives. You will assess based on the following six key dimensions, checking each item meticulously. If any item does not meet the requirements, the overall assessment result will be marked as a failure (outputting 0):</p> <p>Key Evaluation Dimensions:</p> <ol style="list-style-type: none"> 1. concept Coverage: - Verify complete coverage of all required concepts - Any omission or introduction of unmentioned points is non-compliant 2. Bloom's Taxonomy Level: - Analyze cognitive level alignment with objectives - Must neither exceed nor fall short of target level 3. Student Problem-Solving Skills: - Confirm comprehensive assessment of all targeted skills 4. Mathematical Core Competencies: - Evaluate cultivation of: * Logical reasoning * Mathematical operations * Spatial visualization * Data analysis * Mathematical modeling * Mathematical abstraction 5. Rigor Requirement: - Maintain objectivity and rigor throughout evaluation - Any non-compliance results in direct failure (0) <p>Output Format: { "reason": "Detailed explanation of the reasoning and process behind the evaluation", "pass_rate": 1 or 0 }</p> <p>Required Input: Education_Objectives: {educational objectives} Question: {question}</p>

Table 11: Pass rate evaluation prompt template.

Win Rate Evaluation Prompt

As a senior mathematics education expert, please rigorously evaluate and compare the following question pair. In the evaluation process, analyze each question based on the following dimensions and determine which question better meets the educational objectives.

Evaluation Dimensions:

- 1. Completeness of Concept Coverage:** - Analyze coverage of required concepts - Check for missing or redundant points
- 2. Matching of Cognitive Levels:** - Assess alignment with specified cognitive level - Verify appropriate goal alignment
- 3. Relevance to Ability Development:** - Confirm effective training of specified abilities - Verify alignment with outlined requirements
- 4. Development of Mathematical Literacy:** - Analyze contribution to mathematical literacy development
- 5. Scientific Design of the Structure:** - Evaluate reasonableness of question structure - Assess organization and guidance quality
- 6. Text Clarity and Coherence:** - Assess clarity and conciseness of wording - Evaluate effectiveness of problem-solving communication

Output Format:

```
{ "better_question": 1 or 2, "reason": "Detailed evaluation reasons, explaining why the selected question is better and specifying which dimension(s) show superior performance." }
```

Required Input: Education_Objectives: {educational objectives} Question Pair: {Question pair}

Table 12: Win Rate evaluation prompt template.