CogAtom: From Cognitive Atoms to Olympiad-level Mathematical Reasoning in Large Language Models

Zhuofan Chen[1] **Jiyuan He**[2] **Yichi Zhang**[1] **Xing Hu**[2]
Haoxing Wen[2] **Jun Bai**[3] **Wenge Rong**[1]
[1] School of Computer Science and Engineering, Beihang University, China
[2] Meituan Inc., China
[3] Beijing Institute for General Artificial Intelligence, China
{zhuofanchen, yichizhang, w.rong}@buaa.edu.cn,
{hejiyuan, huxing11, wenhaoxing}@meituan.com, baijun@bigai.ai

Abstract

Mathematical reasoning poses significant challenges for Large Language Models (LLMs) due to its demand for multi-step reasoning and abstract conceptual integration. While recent test-time scaling techniques rely heavily on high-quality, challenging problems, the scarcity of Olympiad-level math problems remains a bottleneck. We introduce CogAtom, a novel cognitive atom-based framework for synthesizing mathematically rigorous and cognitively diverse problems. Unlike prior approaches, CogAtom models problem construction as a process of selecting and recombining fundamental reasoning units, cognitive atoms, extracted from human-authored solutions. A diversity-promoting random walk algorithm enables exploration of the cognitive atom space, while a constraint-based recombination mechanism ensures logical soundness and structural validity. The combinatorial nature of the graph structure provides a near-infinite space of reasoning paths, and the walk algorithm systematically explores this space to achieve large-scale synthesis of high-quality problems; meanwhile, by controlling the number of cognitive atoms, we can precisely adjust problem difficulty, ensuring diversity, scalability, and controllability of the generated problems. Experimental results demonstrate that CogAtom outperforms existing methods in accuracy, reasoning depth, and diversity, generating problems that closely match the difficulty of AIME while exceeding it in structural variation. Our work offers a cognitively grounded pathway toward scalable, high-quality math problem generation.[1]

1 Introduction

Reasoning abilities are core cognitive mechanisms underlying human problem-solving (Hersh, 2015). Among them, mathematical reasoning stands out for its unique cognitive complexity, requiring abstract concept comprehension, logical inference across domains, and multi-step solution strategies (Hendrycks et al., 2021; Wang et al., 2024b; Yue et al., 2024a; Wei et al., 2024). As a result, mathematical reasoning has become a key benchmark for evaluating the progress of Large Language Models (LLMs) toward Artificial General Intelligence (AGI) (Zhong et al., 2024).

With the paradigm of LLMs shifting from training-time compute scaling toward test-time compute scaling (Snell et al., 2024; DeepSeek-AI et al., 2025; Yang et al., 2025a), it heavily depends on the high-quality, multi-step mathematical problems as these high-quality problems are essential for effectively exposing model deficiencies in multi-step reasoning and validating the performance boundaries of test-time optimization strategies (Lu et al., 2024; Xu et al., 2025). This shift has generated a substantial demand for data that is simultaneously: (1) large-scale and scalable, to support compute-intensive optimization strategies; (2) highly challenging, to drive models beyond their current capabilities; and (3) cognitively diverse, to mitigate overfitting to familiar solution patterns. However, the existing collection of human-authored, Olympiad-level problems is static, limited in size, and does not fulfill these requirements. This pronounced scarcity of suitable data has become a primary obstacle to the rigorous development and evaluation of advanced reasoning systems (Hendrycks et al., 2021; Yue et al., 2024a; NuminaMath, 2024).

The most straightforward solution to this data scarcity problem is to synthesize challenging problems automatically. Existing synthesis methods can be categorized as prompt engineering-based methods (Toshniwal et al., 2025; Yu et al., 2024; Liu et al., 2025), corpus mining methods (Zhao et al., 2024b; Yue et al., 2024b), evolutionary and transfer-based methods (Yu et al., 2024; Luo et al.,

[1] Our code is publicly available at https://github.com/Icarus-1111/CogAtom.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 24108–24125
November 4-9, 2025

2025) and knowledge-driven synthesis methods (Tang et al., 2024; Huang et al., 2025; Zhao et al., 2025). However, these methods fail to model fundamental units of thought—cognitive atoms—and their combinatorial principles from a cognitive science perspective. In contrast, when mathematics experts design Olympiad problems, they carefully craft cognitive associations between concepts and construct rigorous logical frameworks that demand multi-level reasoning. Current methods are unable to effectively emulate this process, leading to substantial gaps between generated problems and human-authored Olympiad questions in terms of accuracy, logical coherence, and cognitive diversity.

To overcome these structural limitations, we introduce CogAtom, a framework that implements a paradigm shift from linear generation to structured synthesis, centered on the Cognitive Association Graph. This framework operates through a systematic, three-stage generation process. First, reasoning atoms are extracted from a curated seed set and assembled into the global graph; its vast combinatorial nature provides a near-infinite space of potential reasoning paths, directly enabling scalable generation. Next, a diversity-promoting random walk algorithm explores this structured space to sample long and intricate reasoning paths, the topological complexity of which forms the basis for problems of high difficulty. Finally, these paths are transformed by a constraint-based recombination mechanism, driven by three Cognitive Transfer Operators, which ensures the final combination is logically coherent while exhibiting high conceptual diversity.Our contributions are summarized as follows:

(1) We innovatively incorporate the established concept of reasoning atoms into a graph-based mathematical problem synthesis framework, enabling systematic extraction and representation of cognitive connections between fundamental mathematical concepts.

(2) We introduce three cognitive transition operators—Path Extension, Bridge Replacement, and Counterfactual Perturbation—that collectively ensure reasoning depth, logical coherence, and conceptual diversity in synthesized problems.

(3) Through extensive experimentation, we demonstrate that our CogAtom framework generates high-quality training data that significantly enhances foundation models' mathematical reasoning capabilities, with particularly pronounced improvements on advanced multi-step reasoning tasks.

2 Related Work

2.1 Math Reasoning with LLMs

While LLMs have demonstrated remarkable capabilities, their mathematical reasoning remains fragile, vulnerable to common failure modes such as distraction by irrelevant context (Yang et al., 2025b) and an inability to identify ill-defined problems with missing or contradictory conditions (Tian et al., 2024). To address these limitations, researchers have explored several approaches. Some work focuses on curating specialized datasets for math reasoning, aiming to offer more effective benchmarks and training resources for evaluating and enhancing model capabilities (Hendrycks et al., 2021; Wang et al., 2024b; Yue et al., 2024a). Another work leverages prompting strategies, such as chain-of-thought, to elicit more structured and accurate reasoning (Wei et al., 2022; Zhang et al., 2025; Fu et al., 2023; Chen et al., 2023; Kim et al., 2025). Beyond prompting, fine-tuning (Wang et al., 2023; Wen et al., 2024; Ye et al., 2025), in-context learning (Zhao et al., 2024b), reinforcement learning (Yu et al., 2023; Wang et al., 2024a; Trung et al., 2024), test-time scaling (DeepSeek-AI et al., 2025; Guan et al., 2025) and other strategies are also utilized to improve mathematical generalization and symbolic reasoning (Zhao et al., 2024c; Ma et al., 2025; Chen et al., 2025; Fu et al., 2025).

2.2 Data Synthesis For Math Reasoning

Early efforts in mathematical reasoning research often relied on manually constructed datasets (Hendrycks et al., 2021; Cobbe et al., 2021; NuminaMath, 2024). However, their limited scale and diversity have constrained the potential of LLMs in math reasoning tasks. Recent work has explored the use of LLMs themselves to generate synthetic data. Some work leverages LLMs to generate diverse problems through self-instruct, chain-of-thought prompting (Luo et al., 2025; Toshniwal et al., 2025; Yu et al., 2024; Liu et al., 2025). Subsequent work introduces rejection sampling to alleviate the problem of low-quality data from direct prompting (Neal, 2003; Li et al., 2025b). To move beyond simple prompting, recent work has focused on knowledge-driven synthesis pipelines. These approaches often extract structured elements, such as logically consistent templates (Huang et al., 2023) or key knowledge points (Tang et al., 2024; Huang

et al., 2025), from seed data to guide generation.

Although recent methods, especially knowledge-driven synthesis pipelines, have improved mathematical problem synthesis, they still suffer from shallow conceptual hierarchies and limited diversity. Furthermore, these approaches often apply a uniform generation budget, potentially overlooking the varied learning utility of problems at different difficulty levels (Xiong et al., 2025). Our CogAtom framework addresses these challenges by introducing cognitive atoms and unique walk and recombination mechanism.

3 Methodology

As shown in Figure 1, our approach presents a comprehensive framework for mathematical problem synthesis, encompassing three crucial stages: (1) the extraction of reasoning atoms, (2) the construction of a cognitive association graph, and (3) the synthesis of challenging mathematical problems.

3.1 Reasoning Atom Extraction

The foundation of our framework is a rich and diverse set of reasoning atoms. The quality of these atoms is fundamentally constrained by the seed problems from which they are extracted. Guided by the principle that high-quality outputs necessitate high-quality inputs, we begin not with random data, but with a meticulously curated set of seed problems. To this end, mirroring efforts in computational education to automatically assess and filter high-quality simulated agents (Li et al., 2025a), we introduce a systematic and reproducible procedure: the Automated Quality and Complexity Assessment Protocol.

This protocol leverages GPT-4o as an expert judge to evaluate candidate problems. First, we established a 5-point rubric to score problems based on the depth and complexity of reasoning required, inspired by established pedagogical principles and the design of large-scale educational datasets (Penedo et al., 2024). The full rubric is detailed in Appendix H. We then prompted GPT-4o to score each problem according to this rubric. To ensure robustness and mitigate potential biases, each problem was scored three times, and the average score was used. Finally, we applied a stringent filter, retaining only problems with an average score of 3.0 or higher. The rationale for this rigorous curation is twofold. First, by selecting problems that demand at least moderate multi-step or conceptual reasoning, we ensure that our initial pool of cognitive atoms is sufficiently rich and sophisticated to support the generation of novel, non-trivial problems. Second, this "quality-over-quantity" approach is supported by findings that an LLM's reasoning is more effectively unlocked by smaller, high-quality datasets (Ye et al., 2025), and that problems near the boundary of a model's competence offer the most learning utility (Xiong et al., 2025). Our protocol thus serves as a principled method for curating a high-quality set of problem-solving demonstrations.From this curated seed set of 9,403 problems, we then extract their constituent reasoning atoms. Inspired by AoT (Teng et al., 2025), we prompt GPT-4o to solve each problem while reversely extracting the required atoms. To consolidate semantically redundant elements, we generate vector embeddings for the extracted atoms and cluster them based on cosine similarity, yielding a final, refined set of $|\mathcal{A}| = 44{,}117$ unique reasoning atoms, where $\mathcal{A}$ denotes the set of all reasoning atoms.

3.2 Graph-Based Reasoning Chain Generation

Our methodology transforms the curated set of reasoning atoms into novel and coherent problem structures through a sequential pipeline: (1) construction and refinement of a global knowledge graph to map conceptual associations, and (2) a sample-and-refine procedure on this graph to generate logical reasoning chains.

1. Global Graph Construction and Refinement. We begin by constructing a global, undirected Cognitive Association Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega)$, built on the principle of co-occurrence. This principle serves as a strong, data-driven proxy for the underlying cognitive associations in expert problem-solving. Each node $v \in \mathcal{V}$ is a unique reasoning atom. To mitigate the bias from high-frequency pairs, a common challenge in corpus-based network analysis, edges are weighted using a logarithmic transformation of their co-occurrence count: $\omega_{ij} = \log(1 + n_{ij})$. To enhance the graph's utility for generating non-trivial problems, we then prune "supernodes"—nodes corresponding to overly generic concepts (e.g., "equation substitution"). These are identified statistically as nodes whose degree exceeds a threshold of $\mu + 2\sigma$, where μ and σ are the mean and standard deviation of node degrees in G. This process yields a pruned

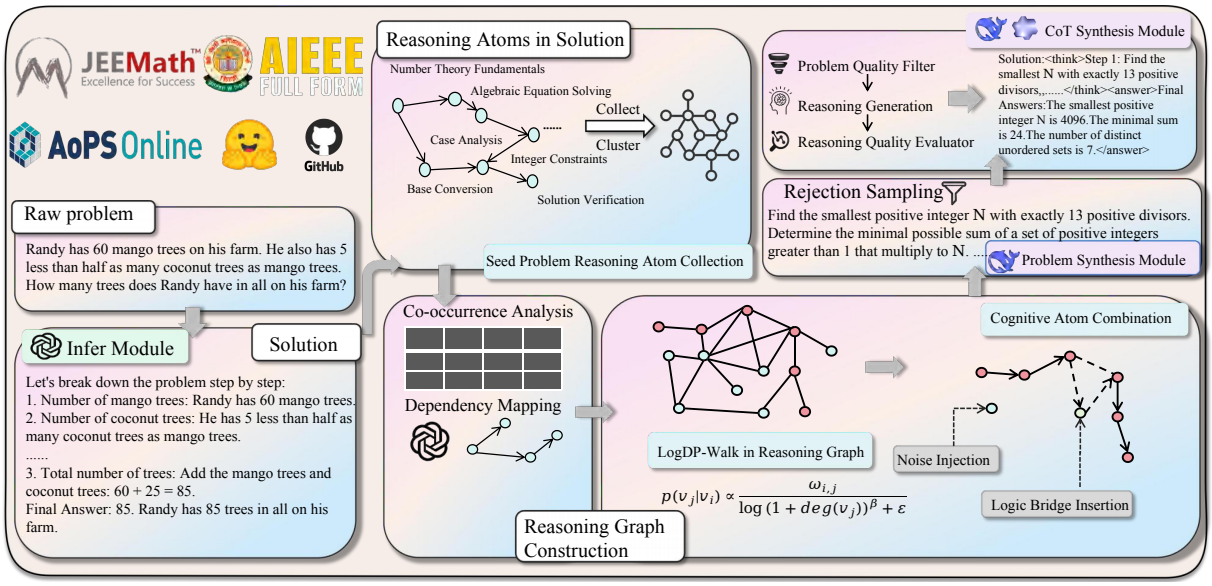

Figure 1: The overall pipeline of CogAtom, which consists of three main stages. **(1) Knowledge Base Construction:** Reasoning atoms are extracted from a curated seed set to build a global Cognitive Association Graph based on co-occurrence. **(2) Reasoning Chain Generation:** A sample-and-refine process generates the final reasoning combination: diverse paths are first sampled from the global graph, then logically refined using local dependency information and Cognitive Transfer Operators. **(3) Problem Synthesis:** The refined combination serves as a logical blueprint to prompt a powerful LLM for the final synthesis of a new problem and its solution.

graph, G', which serves as the foundational structure for exploration.

2. Reasoning Chain Generation. The generation of the final reasoning atom combinations is a two-stage process that moves from broad exploration to fine-grained logical refinement.

First, to generate diverse conceptual skeletons, we perform a biased random walk on the pruned graph G', termed Diversity-Promoting Degree-Regularized Path Expansion (DPDRPE). At each step, the algorithm probabilistically selects the next node v_{next} from the neighbors of the current node v_{curr}. The selection is governed by an association score that penalizes high-degree nodes, thus favoring less common and potentially more novel conceptual connections, defined as:

$$\text{score}(v_{\text{next}}) = \frac{\omega_{v_{\text{curr}},v_{\text{next}}}}{(\deg(v_{\text{next}}) + \epsilon)^{\alpha}} \quad (1)$$

where ω is the co-occurrence weight, $\deg(\cdot)$ is the node degree, and α is a hyperparameter controlling the penalty strength. This yields a set of "reasoning paths" that are diverse but may lack strict logical coherence.

Second, to instill logical rigor, each sampled path undergoes an iterative refinement process formalized in Algorithm 1. The target combination size, K, is set to 10. This choice is informed by an analysis of human-authored Olympiad-level problems, which our analysis shows typically involve the synthesis of 8–12 core concepts. Our selection of $K = 10$ thus aims to emulate a comparable level of cognitive complexity. For each path, we dynamically construct a local, directed Dependency Graph $G_{D,path} = (V_{path}, E_D)$, where V_{path} contains the atoms in the sampled path P. Here, s_{ij} represents the logical dependency strength between atoms v_i and v_j on a 5-point scale (detailed in Appendix B). We retain only edges with $s_{ij} \geq 3$ (on a 5-point scale). This threshold acts as a crucial denoising step, filtering out weak or irrelevant connections while preserving meaningful, non-obvious relationships (scores 3 and above indicate moderate to strong logical dependencies) that are vital for creative synthesis. On this local graph, the conditional probability of a dependency is defined as $P(v_j|v_i) = s_{ij} / \sum_{v_k \in \text{succ}(v_i)} s_{ik}$, where $\text{succ}(v_i)$ is the set of successor nodes of v_i in $G_{D,path}$.

The refinement is driven by three Cognitive Transfer Operators, which are applied iteratively as outlined in Algorithm 1. Each operator is designed to optimize a specific property of the reasoning combination:

Bridge Replacement enhances logical coherence by inserting an intermediary node v_k to connect a weakly linked pair (v_i, v_j). The optimal bridge

Algorithm 1: Reasoning Combination Refinement

Data: A sampled reasoning path P, Target size K
Result: A refined reasoning combination C
$C \leftarrow \text{BackboneConstruction}(P)$;
// Select key nodes from P
Construct local dependency graph $G_{D,path}$ for nodes in C;
while $|C| < K$ **do**
  $C \leftarrow \text{BridgeReplacement}(C, G_{D,path})$;
  $C \leftarrow$ CounterfactualPerturbation$(C, P, G_{D,path})$;
  $C \leftarrow \text{PathExtension}(C, G_{D,path})$;
end
return C

node is selected by maximizing the compound dependency strength, formalized as:

$$v_k^* = \arg \max_{v_k \in V \setminus C} P(v_k|v_i) \cdot P(v_j|v_k) \quad (2)$$

Counterfactual Perturbation promotes cognitive diversity by introducing an atom v^* from the original path P that is minimally associated with the current combination C. This encourages the exploration of novel conceptual links and is guided by:

$$v^* = \arg \min_{v \in P \setminus C} \max_{v_j \in C} P(v_j|v) \quad (3)$$

Path Extension ensures the completeness and logical flow of the reasoning chain by appending a strongly dependent successor node v_{next} to a node $v_i \in C$, governed by the condition:

$$P(v_{next}|v_i) \geq \theta \quad (4)$$

where θ is a predefined dependency threshold. Through this iterative process, diverse conceptual skeletons are transformed into combinations that are both logically sound and cognitively novel.Ultimately, we posit that the quality of a synthesized problem is directly determined by the logical coherence and conceptual novelty of its underlying reasoning chain—properties our sample-and-refine process is explicitly designed to optimize.

3.3 Synthesis of Challenging Mathematical Problems

Given a combination of reasoning atoms, we design an efficient pipeline for problem generation and quality control. Tailored prompts are crafted to guide large language models in synthesizing mathematical problems that are both challenging and diverse.To ensure the quality of the generated problems, we employ a rigorous multi-dimensional evaluation process that filters out questions lacking logical consistency, sufficient solvability, appropriate difficulty, or adequate concept coverage. For problems that pass this screening, we further utilize a strong teacher model to generate detailed step-by-step reasoning solutions. Each reasoning chain is then subjected to comprehensive quality assessment, focusing on conceptual integration, reasoning depth and rigor, key insight demonstration, error path exploration, and training applicability. Only those problems and reasoning chains that meet all quality criteria are retained in the final dataset.Through this dual-stage quality assurance process, we construct a dataset $\mathcal{D} = \{(q_i, s_i, a_i)\}$, where q_i denotes the problem statement, s_i is the step-by-step solution, and a_i is the final answer. This dataset provides a solid foundation for training and evaluating advanced mathematical reasoning models.

4 Experiments

4.1 Datasets

Seed Dataset Construction. To establish a high-quality foundation for our synthetic data generation process, we constructed a comprehensive seed dataset comprising 9,403 mathematical problems carefully selected from multiple established datasets. These source datasets include: GSM8K (Cobbe et al., 2021) (MIT License), MATH (Hendrycks et al., 2021) (MIT license), TAL-SCQ5K-EN (Math-Eval, 2023) (MIT License), JEEE and AIEEE[2], ranging from elementary and middle school difficulty to Olympic-level difficulty. To ensure quality and appropriate difficulty distribution, we employed GPT-4o (Zhao et al., 2024a) to filter and categorize the problems based on their complexity and reasoning requirements, resulting in a diverse and balanced seed collection suitable for our synthetic data generation pipeline. The detailed composition of the seed dataset is summarized in Table 1.

4.2 Baseline Methods

Our evaluation employs both short-CoT (concise intermediate reasoning steps) and long-CoT (ex-

[2]https://jeemath.in/

Source Dataset	Number of Samples
GSM8K (train)	1351
TAL-SCQ5K-EN (train)	526
MATH (train)	3994
AIEEE/JEEE	3472
Total	**9343**

Table 1: Composition of the seed dataset used for synthetic data generation.

tended reasoning with self-reflection and alternative paths) approaches. We compare against five state-of-the-art mathematical problem generation methods:

For comprehensive assessment, we compare our method against several prominent approaches in mathematical problem generation: **Evol-Instruct** (Luo et al., 2025) implements an iterative refinement mechanism using LLMs to progressively enhance instructional data complexity; **KPDDS** (Huang et al., 2025) extracts key points from authentic sources to generate mathematically coherent question-answer pairs; **OpenMath** (Toshniwal et al., 2025) synthesizes solutions for established benchmarks through open-source language models; **NuminaMath** (NuminaMath, 2024) provides competition-level mathematical problems with detailed reasoning traces; and **MathScale** (Tang et al., 2024) constructs concept graphs from seed questions to guide diverse problem generation.

For methods without publicly released problem sets (specifically Evol-Instruct and KPDDS), we followed their documented methodologies using Qwen2.5-Math-72B-Instruct (Yang et al., 2024) to generate comparable problem collections. For NuminaMath, OpenMathInstruct, and MathScale, we utilized their published problem sets directly.

4.3 Implementation Details

We employ GPT-4o to generate step-by-step solutions, from which atomic reasoning steps were extracted. Each reasoning atom is encoded as a dense vector using the BGE-M3 model (Chen et al., 2024), followed by L2 normalization. We apply MiniBatch KMeans clustering to these embeddings and further merged highly similar clusters using cosine similarity filtering. We then construct a cognitive association graph comprising 44,177 reasoning atom nodes and 149,576 edges. To generate diverse reasoning paths, we perform iterative degree-penalized random walks of order $n = 5$ starting from each node in the concept graph, thereby constructing cognitive reasoning paths. Along each path, we apply three types of cognitive leap operations to obtain combinations of reasoning atoms, with each combination containing 10 nodes. For problem synthesis, we use Qwen2.5-72B-Instruct to generate CogAtom-short problems and DeepSeek-R1-Distill-Qwen-32B (DeepSeek-AI et al., 2025) for CogAtom-long problems. For quality control, we employ Qwen2.5-72B-Instruct as an LLM-based judge. We further fine-tune Qwen2.5-Math-7B (Yang et al., 2024) and Qwen2.5-14B-Origin using the Adam optimizer with initial learning rates of 2×10^{-5} and 1×10^{-5}, respectively. To further investigate the reasoning ability of long-CoT, we fine-tuned DeepSeek-R1-Distill-Qwen-7B and evaluate on three high-difficulty datasets (MATH, AIME 2024 and AIME 2025). All training was conducted with BF16 mixed precision and Flash Attention for two epochs, and greedy decoding is used during evaluation. All experiments were conducted on a cluster of 8 machines, each equipped with NVIDIA A100 GPUs.

4.4 Main Result

Tables 2 and 3 present comprehensive evaluation results comparing our CogAtom framework against state-of-the-art mathematical problem generation methods across five benchmarks of increasing difficulty. Our analysis reveals several significant findings:

(1) When used for fine-tuning identical base models, CogAtom-generated data consistently yields superior performance across all benchmarks. For Qwen2.5-Math-7B, fine-tuning with CogAtom-long data achieves 91.2% accuracy on GSM8K and 83.0% on MATH500, outperforming the next best data sources (KPDDS at 89.5% and Evol-Instruct at 73.8%, respectively). Similar advantages are observed with Qwen2.5-14B-Origin, demonstrating the cross-architectural transferability of our approach.

(2) The performance advantage from CogAtom-generated training data becomes increasingly pronounced as problem difficulty increases. With Qwen2.5-Math-7B, the improvement margin grows from 1.7 percentage points on GSM8K to 9.2 percentage points on MATH500 (compared to the strongest baselines). For Olympic-level problems, Qwen2.5-Math-7B fine-tuned on CogAtom-long data correctly solves $5/30$ AIME2024 problems, compared to $4/30$ for the best baseline method.

Methods	gsm8k	math500	AMC	AIME2024	AIME2025
Models based on Qwen2.5-Math-7B					
KPDDS	89.5%	73.6%	50.0%	4/30	2/30
Evol-Instruct	87.9%	73.8%	45.0%	2/30	1/30
NuminaMath	84.4%	70.6%	47.5%	2/30	1/30
MathScale	80.8%	71.2%	45.0%	1/30	2/30
OpenMath	87.2%	69.0%	47.5%	1/30	2/30
CogAtom-short	90.4%	75.0%	**52.5%**	3/30	2/30
CogAtom-long	**91.2%**	**83.0%**	47.5%	**5/30**	**3/30**
Models based on Qwen2.5-14B-Origin					
KPDDS	88.8%	58.8%	29.1%	**3/30**	1/30
Evol-Instruct	87.8%	63.0%	32.5%	2/30	1/30
NuminaMath	79.0%	59.0%	37.5%	1/30	1/30
MathScale	80.6%	59.4%	22.5%	1/30	0/30
OpenMath	87.8%	64.2%	40.0%	2/30	1/30
CogAtom-short	89.2%	68.8%	**45.0%**	**3/30**	**2/30**
CogAtom-long	**91.7%**	**76.4%**	32.5%	**3/30**	**2/30**

Table 2: Evaluation results on five mathematical benchmarks for model Qwen2.5-Math-7B and Qwen2.5-14B-Origin, both fine-tuning with 100K synthetic problems. Within each section, the best results are highlighted in bold font, and the second best results are underlined. The number of correct answers (out of 30) is reported for both AIME2024 and AIME2025.

This pattern highlights our framework's effectiveness in generating training data that encodes complex reasoning patterns required for advanced mathematical problem-solving.

(3) Training with CogAtom-short data yields models that excel on structured problems with clear solution paths (e.g., 52.5% on AMC using Qwen2.5-Math-7B), while CogAtom-long data produces models that perform better on problems requiring multi-step reasoning (e.g., 83.0% vs. 75.0% on MATH500). This differentiation reflects how our cognitive leap operators create training examples that develop distinct reasoning capabilities based on the complexity of target tasks.

(4) As shown in Table 3, when fine-tuning DeepSeek-R1-Distill-Qwen-7B, CogAtom-long data enables achieving 90.8% accuracy on MATH500, surpassing the next best method (Evol-Instruct at 79.6%) by 11.2 percentage points. Most remarkably, it facilitates solving $10/30$ AIME2024 problems and $9/30$ AIME2025 problems. These substantial improvements demonstrate that training examples generated by CogAtom capture complex conceptual dependencies and multi-step reasoning paths, thereby unlocking the full potential of advanced reasoning models.

Methods	math500	AIME2024	AIME2025
KPDDS	76.6%	6/30	1/30
Evol-Instruct	79.6%	6/30	2/30
NuminaMath	72.2%	4/30	2/30
MathScale	75.3%	1/30	0/30
OpenMath	68.0%	2/30	2/30
CogAtom-long	**90.8%**	**10/30**	**9/30**

Table 3: Evaluation results on three mathematical benchmarks with high-difficulty for model Qwen2.5-7B-DeepSeek-R1-Distill-Qwen-7B.

4.5 Analysis of Data Scale

To evaluate the scalability of our data synthesis engine, we fine-tuned Qwen2.5-7B-Math on CogAtom-generated datasets of increasing sizes, from 300k up to 1.6 million problems. The results, presented in Figure 2 and detailed in Table 4, reveal a strong, positive correlation between data scale and performance on core mathematical reasoning benchmarks.

The scaling trend is particularly pronounced on the most challenging benchmarks. On the competition-level AIME dataset, for instance, accuracy exhibits a clear monotonic ascent with data volume: it rises from 27.0% at 300k samples, to 29.0% (+500k), 31.0% (+1M), and culminates at 33.0% with the full 1.6M dataset. This scaling behavior, characterized by diminishing returns, is consistent with established logarithmic scaling laws in large language models. Furthermore, the data reveals a differentiated impact: the absolute performance gain on the complex reasoning required for AIME (+6.0% from 300k to 1.6M) is substantially larger than on the more algorithmic GSM8K benchmark (+1.5% over the same interval). While

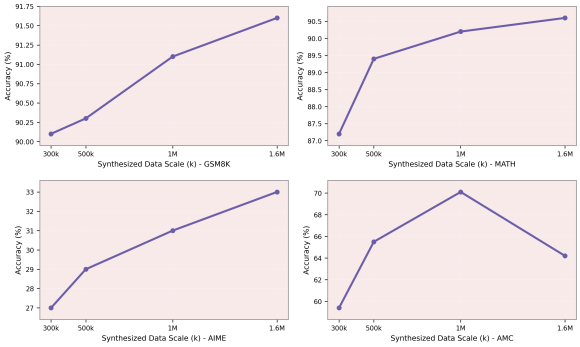

Figure 2: Performance on mathematics benchmarks with increasing scale of synthesized data from CogAtom-long. We report accuracy for GSM8K, MATH, AIME, and AMC.

Data Scale	GSM8K	MATH	AIME	AMC
+300k	90.1%	87.2%	27.0%	59.4%
+500k	90.3%	89.4%	29.0%	65.5%
+1M	91.1%	90.2%	31.0%	70.1%
+1.6M	91.6%	90.6%	33.0%	64.2%

Table 4: Detailed performance on mathematics benchmarks with increasing scale of synthesized data from CogAtom-long.

performance on AMC shows a different trajectory, peaking at 1M samples, the overall results strongly validate that the CogAtom engine is a scalable and effective method for enhancing advanced mathematical reasoning.

4.6 Cross-Domain Generalization to Physics

To assess the domain-agnostic nature of the CogAtom paradigm, we applied it to physics—a domain that, like mathematics, is characterized by complex principles and multi-step reasoning.We synthesized two large-scale datasets of 300k and 600k physics problems, respectively, and used them to fine-tune the Qwen2.5-7B base model.The empirical results, presented in Table 5, provide strong support for this hypothesis. Fine-tuning on the generated data yields consistent and substantial performance gains across a suite of standard physics benchmarks in both Chinese (C-MMLU) and English (MMLU). The performance improvements scale positively with data volume; for instance, on C-MMLU High School Physics, the accuracy gain grows from +1.3% with 300k samples to a remarkable **+9.3%** with 600k samples. This monotonic trend underscores the effectiveness of our synthesized data in imparting robust physical reasoning skills. A qualitative case study of a generated physics problem, which integrates concepts from thermodynamics and black-body radiation, is detailed in Appendix C.

4.7 Ablation Study on Synthesis Components

Table 6 reports the results of our ablation study on the key components of the framework. The degree-penalty mechanism is crucial for promoting conceptual diversity by mitigating the selection bias toward high-degree nodes; its removal causes a dramatic drop in CogAtom-long's performance on AIME2024 from $4/30$ to $1/30$, especially for complex problems. Cognitive leap operators are particularly beneficial for long-form reasoning: their ablation leads to a 2.6 percentage point decrease in MATH500 accuracy and halves the AIME2024 score for CogAtom-long. In contrast, these operators have minimal effect on CogAtom-short, indicating their primary role in enhancing complex reasoning chains. Quality-based rejection sampling also plays a significant role in challenging benchmarks; without it, CogAtom-short's AIME2024 result declines from $3/30$ to $1/30$. Collectively, these findings confirm that each component makes a meaningful contribution, with their importance varying according to the complexity of mathematical reasoning tasks.

4.8 Analysis of Problem Difficulty

We assess problem difficulty using two complementary metrics: answer consistency and inference tokens. A problem is marked as consistent if both models generate identical answers. We also record the total number of tokens consumed during their reasoning to assess difficulty. We additionally extend our analysis to the challenging Olympiad-level AIME2024 dataset. The results are presented in Table 7. Compared with other baseline synthetic methods, our CogAtom framework achieving the lowest answer consistency (67.47%) and most tokens (3022), indicating synthesis of the most challenging problem. Although AIME2024 benchmark still yields an even lower consistency of 50.00% and more tokens of 5260, CogAtom narrows this gap more than any other methods, demonstrating its effectiveness at generating higher-difficulty, more realistic mathematical challenges.

4.9 Analysis of Problem Diversity

We further analyze the diversity of synthesized problems. We introduce the Problem Type Diver-

Table 5: Performance on physics benchmarks as the scale of synthetic data increases. We report accuracy scores. The best results are in bold.

Synthetic Data Scale	C-MMLU Concept. Physics	C-MMLU High School Physics	MMLU Concept. Physics	MMLU High School Physics	MMLU Univ. Physics
Baseline (0 samples)	0.7687	0.6818	0.7050	0.5762	0.5000
+300k (Our Method)	0.7823	0.6909	0.7450	0.5960	**0.5882**
+600k (Our Method)	**0.7959**	**0.7455**	0.7200	**0.6093**	0.5294

Methods	Ablation	gsm8k	math500	AIME2024	AIME2025
CogAtom-short	Full model	**88.5%**	**70.6%**	**3/30**	**2/30**
	w/o degree-penalty	87.1%	69.6%	2/30	1/30
	w/o cognitive	87.6%	70.2%	**3/30**	**2/30**
	w/o reject-sampling	86.9%	70.6%	1/30	**2/30**
CogAtom-long	Full model	**91.0%**	**70.6%**	**4/30**	**3/30**
	w/o degree-penalty	90.3%	69.2%	1/30	2/30
	w/o cognitive	90.6%	68.0%	2/30	2/30
	w/o reject-sampling	89.5%	69.0%	3/30	2/30

Table 6: Ablation study results of different synthesis components for model Qwen2.5-Math-7B finetuning with 10K synthetic problems.

Methods	Answer Consistency	Tokens
KPDDS	75.7%	2328
Evol-Instruct	70.0%	2282
NuminaMath	86.3%	1954
MathScale	68.6%	2103
OpenMath	80.5%	2532
AIME2024	**50.0%**	**5260**
CogAtom	67.5%	3022

Table 7: Analysis of problem difficulty with model Qwen2.5-72B and DeepSeek-R1-distill-Qwen-32B.

Methods	PTD
KPDDS	1.7903
Evol-Instruct	1.7931
NuminaMath	1.7936
MathScale	1.7836
OpenMath	1.7190
AIME2024	1.7896
CogAtom	**1.7961**

Table 8: Analysis of problem diversity

sity (PTD) metric to rigorously quantify semantic diversity in mathematical problem datasets. PTD jointly captures the breadth of problem type coverage and the uniformity of their distribution:

$$\text{PTD} = \frac{N_c}{\sqrt{N}} \left(1 - \frac{\sigma_c}{\mu_c \sqrt{N_c}}\right) \tag{5}$$

Here, N_c is the semantic cluster count, N is the total sample size, μ_c is the mean cluster size, and σ_c is the cluster size standard deviation. Empirically, CogAtom-Long achieves the highest PTD score (1.7961), reflecting comprehensive and balanced problem coverage. In contrast, baseline datasets like KPDDS (1.7190) show lower PTD, indicating more homogeneous content. These results highlight our approach's advantage in fostering semantic diversity, critical for enhancing model generalization to novel mathematical reasoning tasks.

5 Conclusion

In this paper, we introduced CogAtom, a novel framework for mathematical problem synthesis that integrates reasoning atoms and cognitive association graphs to generate high-quality training data. Our extensive experiments demonstrated that models fine-tuned on CogAtom-generated problems achieve substantial improvements in mathematical reasoning capabilities, particularly on advanced multi-step reasoning tasks, outperforming existing methods by significant margins on challenging benchmarks including MATH500 and AIME. The effectiveness of our approach highlights the importance of cognitive science principles in designing synthetic training data for enhancing reasoning abilities in foundation models.

Limitations

Despite the promising results of our work, a primary limitation is that the CogAtom framework currently operates exclusively in the textual modality. This constraint limits its applicability to mathematical domains that are inherently visual, such as geometry and graph theory, where diagrams and

figures are often integral to the problem statement. A significant direction for future research is to extend CogAtom into a multimodal framework. This would involve developing methods to represent visual components as a new type of cognitive atom and learning the cross-modal associations between textual and visual elements. Such an extension would enable the synthesis of a far richer and more comprehensive class of problems, better reflecting the multimodal nature of human mathematical reasoning.

Ethics Statement

All experiments in this study are conducted on publicly available and commonly used datasets. We acknowledge the risks associated with automated content generation by large language models. Our research on structured data synthesis contributes to the development of more controllable and reliable methodologies, aiming to foster beneficial applications of AI in specialized reasoning domains.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62477001.

References

Jiaao Chen, Xiaoman Pan, Dian Yu, Kaiqiang Song, Xiaoyang Wang, Dong Yu, and Jianshu Chen. 2023. Skills-in-context prompting: Unlocking compositionality in large language models. *CoRR*, abs/2308.00304.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *CoRR*, abs/2402.03216.

Zui Chen, Tianqiao Liu, Tongqing, Mi Tian, Weiqi Luo, and Zitao Liu. 2025. Advancing mathematical reasoning in language models: The impact of problem-solving data, data synthesis methods, and training stages. In *Proceedings of the 13th International Conference on Learning Representations*.

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

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, and S. S. Li. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *CoRR*, abs/2501.12948.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2023. Complexity-based prompting for multi-step reasoning. In *Proceedings of the 11th International Conference on Learning Representations*.

Yuqian Fu, Yuanheng Zhu, Jiajun Chai, Guojun Yin, Wei Lin, Qichao Zhang, and Dongbin Zhao. 2025. RLAE: Reinforcement learning-assisted ensemble for LLMs. *CoRR*, abs/2506.00439.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. 2025. rStar-Math: Small llms can master math reasoning with self-evolved deep thinking. *CoRR*, abs/2501.04519.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

Reuben Hersh. 2015. *How Humans Learn to Think Mathematically. Exploring the Three Worlds of Mathematics*. *The American Mathematical Monthly*, 122(3):292–296.

Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen. 2025. Key-point-driven data synthesis with its enhancement on mathematical reasoning. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, pages 24176–24184.

Zeyu Huang, Xiaofeng Zhang, Jun Bai, Wenge Rong, Yuanxin Ouyang, and Zhang Xiong. 2023. Solving math word problems following logically consistent

template. In *Proceedings of the 2023 International Joint Conference on Neural Networks*, pages 1–8.

Juno Kim, Denny Wu, Jason Lee, and Taiji Suzuki. 2025. Metastable dynamics of chain-of-thought reasoning: Provable benefits of search, RL and distillation. *CoRR*, abs/2502.01694.

Haoxuan Li, Jifan Yu, Xin Cong, Yang Dang, Yisi Zhan, Huiqin Liu, and Zhiyuan Liu. 2025a. Exploring LLM-based student simulation for metacognitive cultivation. *CoRR*, abs/2502.11678.

Peiji Li, Kai Lv, Yunfan Shao, Yichuan Ma, Linyang Li, Xiaoqing Zheng, Xipeng Qiu, and Qipeng Guo. 2025b. FastMCTS: A simple sampling strategy for data synthesis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 24405–24422.

Haoxiong Liu, Yifan Zhang, Yifan Luo, and Andrew C. Yao. 2025. Augmenting math word problems via iterative question composing. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, pages 24605–24613.

Zimu Lu, Aojun Zhou, Houxing Ren, Ke Wang, Weikang Shi, Junting Pan, Mingjie Zhan, and Hongsheng Li. 2024. MathGenie: Generating synthetic data with question back-translation for enhancing mathematical reasoning of llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 2732–2747.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, and Dongmei Zhang. 2025. WizardMath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *Proceedings of the 13th International Conference on Learning Representations*.

Yiran Ma, Zui Chen, Tianqiao Liu, Mi Tian, Zhuo Liu, Zitao Liu, and Weiqi Luo. 2025. What are step-level reward models rewarding? counterintuitive findings from mcts-boosted mathematical reasoning. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence*, pages 24812–24820.

Math-Eval. 2023. TAL-SCQ5K. https://github.com/math-eval/TAL-SCQ5K.

Radford M Neal. 2003. Slice sampling. *The Annals of Statistics*, 31(3):705–767.

NuminaMath. 2024. Numinamath. https://huggingface.co/collections/AI-MO.

Guilherme Penedo, Hynek Kydlícek, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin A. Raffel, Leandro von Werra, and Thomas Wolf. 2024. The FineWeb datasets: Decanting the web for the finest text data at scale. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems*, pages 30811–30849.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314.

Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. 2024. MathScale: Scaling instruction tuning for mathematical reasoning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 47885–47900.

Fengwei Teng, Zhaoyang Yu, Quan Shi, Jiayi Zhang, Chenglin Wu, and Yuyu Luo. 2025. Atom of thoughts for markov LLM test-time scaling. *CoRR*, abs/2502.12018.

Shi-Yu Tian, Zhi Zhou, Kun-Yang Yu, Ming Yang, Lin-Han Jia, Lan-Zhe Guo, and Yu-Feng Li. 2024. VC search: Bridging the gap between well-defined and ill-defined problems in mathematical reasoning. *CoRR*, abs/2406.05055.

Shubham Toshniwal, Wei Du, Ivan Moshkov, Branislav Kisacanin, Alexan Ayrapetyan, and Igor Gitman. 2025. OpenMathInstruct-2: Accelerating AI for math with massive open-source instruction data. In *Proceedings of the 13th International Conference on Learning Representations*.

Luong Quoc Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. 2024. ReFT: Reasoning with reinforced fine-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 7601–7614.

Peiyi Wang, Lei Li, Liang Chen, Feifan Song, Binghuai Lin, Yunbo Cao, Tianyu Liu, and Zhifang Sui. 2023. Making large language models better reasoners with alignment. *CoRR*, abs/2309.02144.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024a. Math-Shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 9426–9439.

Zengzhi Wang, Xuefeng Li, Rui Xia, and Pengfei Liu. 2024b. MathPile: A billion-token-scale pretraining corpus for math. In *Proceedings of the 38th Annual Conference on Neural Information Processing System*, pages 25426–25468.

Chenrui Wei, Mengzhou Sun, and Wei Wang. 2024. Proving Olympiad algebraic inequalities without human demonstrations. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems*, pages 82811–82822.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th Annual Conference on Neural Information Processing System*, pages 24824–24837.

Jiaxin Wen, Jian Guan, Hongning Wang, Wei Wu, and Minlie Huang. 2024. CodePlan: Unlocking reasoning potential in large langauge models by scaling code-form planning. *CoRR*, abs/2409.12452.

Feng Xiong, Hongling Xu, Yifei Wang, Runxi Cheng, Yong Wang, and Xiangxiang Chu. 2025. HS-STAR: hierarchical sampling for self-taught reasoners via difficulty estimation and budget reallocation. *CoRR*, abs/2505.19866.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2025. Magpie: Alignment data synthesis from scratch by prompting aligned LLMs with nothing. In *Proceedings of the 13th International Conference on Learning Representations*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025a. Qwen3 technical report. *CoRR*, abs/2505.09388.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024. Qwen2.5-Math technical report: Toward mathematical expert model via self-improvement. *CoRR*, abs/2409.12122.

Minglai Yang, Ethan Huang, Liang Zhang, Mihai Surdeanu, William Wang, and Liangming Pan. 2025b. How is LLM reasoning distracted by irrelevant context? An analysis using a controlled benchmark. *CoRR*, abs/2505.18761.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. LIMO: Less is more for reasoning. *CoRR*, abs/2502.03387.

Fei Yu, Anningzhe Gao, and Benyou Wang. 2023. Outcome-supervised verifiers for planning in mathematical reasoning. *CoRR*, abs/2311.09724.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. MetaMath: Bootstrap your own mathematical questions for large language models. In *Proceedings of the 12th International Conference on Learning Representations*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2024a. MAmmoTH: Building math generalist models through hybrid instruction tuning. In *Proceedings of the 12th International Conference on Learning Representations*.

Xiang Yue, Tianyu Zheng, Ge Zhang, and Wenhu Chen. 2024b. MAmmoTH2: Scaling instructions from the web. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems*, pages 90629–90660.

Yifan Zhang, Jingqin Yang, Yang Yuan, and Andrew C. Yao. 2025. Cumulative reasoning with large language models. *Transactions on Machine Learning Research*, 2025:1–40.

Justin Zhao, Timothy Wang, Wael Abid, Geoffrey Angus, Arnav Garg, Jeffery Kinnison, Alex Sherstinsky, Piero Molino, Travis Addair, and Devvret Rishi. 2024a. LoRA land: 310 fine-tuned LLMs that rival GPT-4, A technical report. *CoRR*, abs/2405.00732.

Xueliang Zhao, Wenda Li, and Lingpeng Kong. 2024b. Subgoal-based demonstration learning for formal theorem proving. In *Proceedings of the 41st International Conference on Machine Learning*, pages 60832–60865.

Xueliang Zhao, Wei Wu, Jian Guan, and Lingpeng Kong. 2025. PromptCoT: Synthesizing olympiad-level problems for mathematical reasoning in large language models. In *Findings of the Association for Computational Linguistics, ACL*, pages 18167–18188.

Yu Zhao, Huifeng Yin, Bo Zeng, Hao Wang, Tianqi Shi, Chenyang Lyu, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2024c. Marco-o1: Towards open reasoning models for open-ended solutions. *CoRR*, abs/2411.14405.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2024. AGIEval: A human-centric benchmark for evaluating foundation models. In *Findings of the Association for Computational Linguistics: NAACL*, pages 2299–2314.

A Seed Problem Curation Strategy

To ensure our seed problem set possesses sufficient complexity and quality to foster advanced reasoning, we developed and implemented a systematic curation strategy. Our approach is grounded in the principle that the quality of synthesized data is fundamentally dependent on the quality of the initial seeds. This strategy automates the curation process in a principled and reproducible manner.

The core of this strategy is an LLM-as-an-expert-judge assessment. For each candidate problem, we prompted GPT-4o to score it three independent

times to ensure robustness and mitigate potential scoring anomalies. The final score was the average of these three ratings. Subsequently, we applied a stringent quality filter, retaining only problems with an average score of 3.0 or higher. The evaluation rubric used in this process, which provides a structured hierarchy for assessment from foundational recall to abstract reasoning, is presented in Figure 3.

B Prompt for Cognitive Atom Extraction

Figure 4 instructs the language model to analyze mathematical problems through a step-by-step reasoning process, then identify and extract the fundamental cognitive atoms—atomic knowledge entities at appropriate granularity—required to master the problem, enabling systematic representation of the core mathematical concepts and principles underlying complex reasoning tasks.

Figure 5 presents the prompt used to quantify logical dependencies between cognitive atoms. This prompt guides the language model to evaluate the strength of prerequisite relationships on a 5-point scale, enabling the construction of dependency graphs for reasoning chain refinement.

Figure 6 shows the prompt generating problems after we obtain the combinations of cognitive atoms. We apply strict filtering for both synthetic problems and generated answers, as shown in Figure 7 and 8.

C Case Study in Physics Generalization

Figure 9 presents a representative problem synthesized by CogAtom in the physics domain. This problem exemplifies a high-quality synthesis, as its solution requires the integration of four distinct cognitive atoms: Energy Absorption, Black-Body Emission, Thermodynamic Equilibrium, and Heat Capacity. The core reasoning challenge lies in reconciling the different geometric dependencies of energy absorption (proportional to the Earth's cross-sectional area) and thermal emission (proportional to the total surface area). A solver must correctly navigate this interplay and then deduce how a systemic property, such as heat capacity, influences the final equilibrium state. This demand for synthesizing a coherent model from competing principles is a hallmark of the cognitively complex problems our framework is designed to generate, built to stress-test and cultivate advanced reasoning capabilities.

D Case Study: Mathematical Problem Synthesis

To provide a transparent, step-by-step illustration of our core Reasoning Chain Generation stage, this appendix presents a detailed case study. The process, visualized in Figure 10, deconstructs how our Cognitive Transfer Operators systematically transform a simple conceptual path into a sophisticated, multi-domain mathematical problem, showcasing the framework's capacity for controlled and creative synthesis.

5-Point "Level-Based" Rubric for Mathematical Problem Complexity

Problems are evaluated against the following criteria sequentially. A problem is awarded the score corresponding to the highest level for which it fully satisfies all criteria.

Score 1: Foundational Recall. The problem requires the recall and direct application of a single, elementary definition, theorem, or routine computational procedure. The solution comprises a single, non-decomposable cognitive step.

Score 2: Algorithmic Procedure. The problem is solved by executing a standard, multi-step algorithm or a well-defined procedural sequence. The solution path is linear and requires no significant strategic deviation or planning.

Score 3: Conceptual Synthesis. The problem necessitates the synthesis of two or more distinct conceptual domains (e.g., algebra and geometry). The solver must independently formulate a solution plan to connect these disparate concepts.

Score 4: Non-Routine Insight. In addition to conceptual synthesis, the solution requires a non-obvious insight, an ingenious construction, or a strategic shortcut. A solution via standard algorithms is either intractable or substantially more complex than one employing the required insight.

Score 5: Abstract or Creative Reasoning. In addition to all lower-level requirements, the problem is characterized by a high degree of abstraction, demanding creative reframing of the problem space or the application of advanced and specialized mathematical formalism.

Figure 3: The 5-level, level-based rubric used in our seed problem curation strategy. Each level defines a specific set of cumulative criteria for assessing the reasoning depth of a mathematical problem, from Foundational Recall (Score 1) to Abstract or Creative Reasoning (Score 5).

Prompt for Cognitive Atom Extraction

Problem: {problem}
1.Think step-by-step through the process of solving this problem.
2.What knowledge points need to be mastered to correctly solve this problem? The granularity of the knowledge points should be individual knowledge entities, equivalent to atoms in a knowledge graph. Provide the name and explanation of each knowledge point.

Respond to the knowledge points in JSON standard format, ensuring that the output can be loaded using json.loads. Do not include any unrelated information or content that is not in JSON format: Keys represent the serial number of the knowledge point, and values are the concatenation of each knowledge point and its corresponding explanation, separated by ' - ':
{"1": "Knowledge Point - Explanation", "2": "Knowledge Point - Explanation", "3": "Knowledge Point - Explanation", ...}
output:

Figure 4: Prompt for extracting cognitive atom from problems

Prompt for Dependency Extraction

You are a senior mathematics education expert specializing in curriculum design and knowledge structure analysis.

Given the following ordered list of knowledge points, please analyze all possible knowledge point pairs (A, B), where A appears before B. Only output the pairs where A is a prerequisite (i.e., has a dependency relationship) for B, with a dependency strength score of 3 (moderate), 4 (strong), or 5 (essential). The scoring criteria are as follows:
• 3 (Moderate): A is helpful for learning B; mastering A makes B easier to understand, though B can still be learned in other ways.
• 4 (Strong): A is very important for B; lack of understanding of A significantly hinders learning B.
• 5 (Essential): A is a strict prerequisite for B; without A, it is not possible to properly understand or learn B.
For each such dependency pair, provide the score (3, 4, or 5) and a concise academic justification.

Input knowledge point list: {kp_list_str}
Output strictly in the following JSON format (only output JSON, minimize line breaks):
[{"from": "A", "to": "B", "score": 3, "reason": "Concise academic justification"}, {"from": "A", "to": "C", "score": 4, "reason": "Concise academic justification"}]

Do not include pairs with scores of 1 or 2. Output only the JSON array in a single line without unnecessary line breaks.

Figure 5: Prompt for extracting dependency

Prompt for Problem Generation

What advanced mathematics or mathematics competition problems are related to {knowledge_points}? Please follow the steps below to design a new comprehensive advanced mathematics or mathematics competition problem. You do NOT need to provide the answer or solution process.
Steps:
Initial Problem Design: First select the main knowledge points that are most suitable for combining into a problem (try to minimize discarding knowledge points at this stage, but you may omit a few that are clearly unsuitable for integration). Design an initial version of the problem that integrates these main knowledge points.
Problem Enhancement: Next, to increase the diversity and difficulty of the problem, please actively consider which advanced knowledge points (not limited to the original set) can be reasonably and naturally integrated into the problem, based on the structure and objectives of the initial problem. Select no more than 3 higher-level knowledge points that are most challenging and easiest to integrate.
Final Problem Design: Based on the initial problem, design a new, more difficult and diverse advanced mathematics problem by integrating these additional knowledge points.
Required Specifications:
The initial version of the problem (with the main knowledge points used for the first combination)
The knowledge points discarded (relative to the original set provided)
The knowledge points added (including those beyond the original set, if any) - strictly limit to a maximum of 3 additional knowledge points
The design rationale for the more difficult, extended version of the problem, explaining why you chose these additional knowledge points and how they improve the problem's difficulty, comprehensiveness, and educational value
The final, extended version of the problem
Output Format Requirement: Use standard JSON format, strictly following the structure below, and ensure the result can be loaded via json.loads. Do not include any irrelevant information or non-JSON content:
{"initial_problem": "xxx", "discarded_knowledge_points": "xxx", "added_knowledge_points": "xxx", "extended_design_idea": "xxx", "extended_problem": "xxx"}

Figure 6: Prompt for problem generation

Prompt for Quality Filtering

Evaluation Criteria
Please evaluate each criterion as follows: [Perfect | Acceptable | Bad]
Logical Consistency: Check whether the problem and its design rationale contain logical gaps or imprecisions, whether they conform to rigorous logical reasoning standards, and whether they are clear without ambiguity.
Solvability: Confirm whether the problem has one and only one valid solution, check if all key information needed for solving has been provided, and ensure the problem can be reasonably solved.
Compliance with Difficulty Standards: Determine whether the knowledge depth and technique complexity required to solve the problem meet the difficulty standards of higher mathematics or mathematical competitions.
Concept Coverage: Evaluate whether the problem adequately reflects and applies the given basic concepts. Check for any omissions, weakening, or improper use of relevant concepts.
Response Format
For each criterion, please provide:
Rating: [Perfect | Acceptable | Bad]
Explanation: Clear justification for the rating
Final Determination
After completing all standard evaluations, please summarize at the end of your response:"Final Judgement: [verdict]"Where verdict must be one of the following:
- 'perfect' (if LOGICAL CONSISTENCY and SOLVABILITY are both Perfect, and at least one other criterion is Perfect, with no Bad ratings)
- 'acceptable' (if there are no Bad ratings, but the conditions for 'perfect' are not met)
- 'bad' (if any criterion is rated as Bad)

Note: "Final Judgement: [verdict]" must appear as the last line of your response.
Given Materials
Basic Concepts: {concept_text}
Target Difficulty Level: Advanced Mathematics or Competition Level
Problem and Design Rationale: (The rationale section describes the author's thinking process and reasoning when designing the problem): {rationale_and_problem

Figure 7: Prompt for quality filtering

Prompt for CoT Answer Quality Filtering

Evaluation Criteria
Please evaluate each criterion as follows: [Perfect | Acceptable | Bad]
1. Problem Value:Concept Integration & Complexity: Assess the degree to which the problem integrates multiple mathematical concepts and its reasoning difficulty, whether it sufficiently challenges model capabilities.Problem Uniqueness: Evaluate the innovativeness of the problem, whether it goes beyond common problem patterns, and can complement existing training data gaps.
2. Thinking Process Quality:Reasoning Depth & Rigor: Assess the logical coherence of the reasoning chain, whether it includes sufficient mathematical proofs and rigorous derivations.Key Insight Demonstration: Evaluate whether the thinking process clearly demonstrates breakthrough points and innovative approaches to solving the problem.Error Path Exploration: Assess whether it includes identification and correction of potential error pathways, helping the model learn to avoid common pitfalls.
3. Training Applicability:Thought Traceability: Evaluate the transparency and explainability of the reasoning process, whether it facilitates model learning of reasoning paths.Answer Accuracy & Completeness: Assess whether the answer is correct and includes comprehensive explanations, suitable as a high-quality learning target.Reasoning Method Generalizability: Evaluate whether the thinking method can be applied to similar problems, and whether it universally enhances model reasoning capabilities
Response Format
For each criterion, please provide:
Rating: [Perfect | Acceptable | Bad]
Explanation: Clear justification for the rating
Final Determination:After completing all standard evaluations, please summarize at the end of your response:
"Final Assessment: [conclusion]"
Where conclusion must be one of the following:
- 'Perfect' (if at least 5 criteria are rated Perfect, the rest are Acceptable, and at least 2 Perfect ratings in Thinking Process Quality)
- 'Acceptable' (if there are no Bad ratings, but conditions for Perfect are not met)
- 'Bad' (if any criterion is rated as Bad)

Note: "Final Assessment: [conclusion]" must appear as the last line of your response.
Given Materials
Problem: {extended_problem}Thinking Process: {thinking_process}Answer: {answer}

Figure 8: Prompt for CoT answer quality filtering

Generated Problem
Question:
The Earth absorbs solar radiation with a power given by $P = S \times \pi R_{\text{earth}}^2$ where S is the solar constant (~1361 W/m²). Assume the Earth behaves as a black body (albedo = 0). If the Earth's atmosphere is modeled as an ideal gas in thermal equilibrium with the surface, and the heat capacity of the atmosphere is proportional to its mass, which statement is correct?
A. A more massive atmosphere would lead to a higher equilibrium temperature.
B. The cross-sectional area of the Earth determines the total emitted power.
C. The ideal gas assumption implies no heat exchange occurs between the surface and atmosphere.
D. The absorbed solar power depends on the mass ratio between the Earth and the atmosphere.
Model's Solution and Reasoning
The absorbed solar power (P) depends on the cross-sectional area (πR_{earth}^2) and the solar constant, not the mass ratio (eliminating D). The ideal gas assumption allows for heat exchange to reach thermal equilibrium (eliminating C). The Earth's emitted power, according to the Stefan-Boltzmann law, depends on its total surface area ($4\pi R_{\text{earth}}^2$), not its cross-sectional area (eliminating B). A more massive atmosphere possesses a higher heat capacity. To reach thermal equilibrium under a constant energy input, a higher heat capacity requires a higher equilibrium temperature to radiate energy at the same rate it is absorbed (A is correct).
Final Answer: A
Analysis of Integrated Cognitive Atoms
This problem effectively assesses a student's ability to synthesize several distinct cognitive atoms from physics:
• Energy Absorption from Radiation: Requires understanding that the power absorbed by a spherical body from a parallel radiation source is proportional to its cross-sectional area (πR^2), not its surface area.
• Black-Body Emission: Requires knowing that the power emitted by a body is determined by its total surface area ($4\pi R^2$) and temperature, as described by the Stefan-Boltzmann law.
• Thermodynamic Equilibrium: Involves the core concept that equilibrium is reached when energy absorbed equals energy emitted. The properties of the system (like heat capacity) determine the final state (temperature) at which this balance occurs.
• Heat Capacity: Requires the knowledge that heat capacity is an extensive property proportional to mass, and a higher heat capacity implies more energy is stored for a given temperature rise

Figure 9: A case study illustrating the cross-domain generalization of CogAtom. The synthesized physics problem requires integrating distinct cognitive atoms from thermodynamics, black-body radiation, and mechanics.

Case Study: Generating a Multi-domain Problem

This case study illustrates the step-by-step process of generating a problem that integrates Number Theory, Combinatorics, and Optimization.

Step 1: Initial Path Sampling
The process begins by sampling an initial "seed path" of related cognitive atoms, representing a standard optimization problem:
Initial Path:["Budget Constraint", "Unit Cost Calculation", "Total Quantity Calculation", "Linear Consumption Model", "Maximization"]

Step 2: Path Enhancement via Cognitive Transfer Operators
The framework then applies three cognitive operators to inject complexity and novelty into the initial path.
A. Path Extension (Deepening a Concept) The system replaces the simple "Unit Cost Calculation" with a more complex prerequisite, "Combinatorial Counting," forcing the unit value to be derived from a counting sub-problem.
B. Bridge Replacement (Connecting Disparate Ideas) To enrich the counting task, the system introduces "Prime Factorization" as a bridge, elegantly connecting the domains of Combinatorics and Number Theory.
C. Counterfactual Perturbation (Introducing a Novel Constraint) To make the final optimization non-trivial, the system perturbs the "Maximization" goal with a distant concept, "Modular Arithmetic," transforming a standard optimization into a constrained one.
This iterative refinement results in a final, more sophisticated combination of cognitive atoms:

Final Combination:["Prime Factorization", "Combinatorial Counting", "Budget Constraint", "Total Quantity Calculation", "Linear Consumption Model", "Modular Arithmetic", "Solving Linear Congruence", "Maximization"]

Step 3: Problem Synthesis
Finally, the enriched atom combination is provided to GPT-4o to synthesize a coherent problem narrative.

Generated Problem

Problem Statement:
Patty has $45 to spend on cookie packages. Each package costs $5 and contains C cookies, where C is the number of ordered triples of integers (l, w, h) such that $l \leqslant w \leqslant h$ and $l \times w \times h = 360$. Her siblings do 10 chores per week, and she pays them 4 cookies per chore. After buying the cookies, she finds that her total number of cookies must be 2 more than a multiple of 7 to unlock a bonus. What is the maximum number of full weeks she can pay for the chores under this modular constraint?

Problem Analysis
This generated problem successfully integrates three mathematical domains:
1. Number Theory: Prime factorization ($360 = 2^3 \times 3^2 \times 5$) and modular arithmetic constraints
2. Combinatorics: Counting ordered triples with constraints
3. Optimization: Maximizing the number of weeks under constraints
The problem requires solvers to first perform combinatorial counting, then apply modular arithmetic constraints for optimization, demonstrating the cognitive atom framework's ability to generate complex, multi-domain mathematical problems.

Figure 10: A visualization of the Reasoning Chain Generation process for a multi-domain math problem. **(1) Path Sampling:** An initial, simple "seed path" focused on optimization is sampled from the global graph. **(2) Iterative Refinement:** The three Cognitive Transfer Operators—Path Extension, Bridge Replacement, and Counterfactual Perturbation—iteratively refine the path, injecting complexity by linking disparate domains like Number Theory and Combinatorics.