Scalable Data Synthesis through Human-like Cognitive Imitation and Data Recombination

Zhongyi Ye[1,2], Weitai Zhang[1], Xinyuan Zhou[3], Yongxin Zhu[1,2], Ninghui Rao[4], Enhong Chen[1,2] *

[1]University of Science and Technology of China,
[2]State Key Laboratory of Cognitive Intelligence
[3]iFlytek Research, [4]Xi'an Jiaotong University
{yzy5630, zwt2021, zyx2016}@mail.ustc.edu.cn, xyzhou15@iflytek.com,
rnh661165@stu.xjtu.edu.cn, cheneh@ustc.edu.cn

Abstract

Large language models (LLMs) rely on massive amounts of training data, however, the quantity of empirically observed data is limited. To alleviate this issue, lots of LLMs leverage synthetic data to enhance the quantity of training data. Despite significant advancements in LLMs, the efficiency and scalability characteristics of data synthesis during pre-training phases remain insufficiently explored. In this work, we propose a novel data synthesis framework, Cognitive Combination Synthesis (CCS), designed to achieve highly efficient and scalable data synthesis. Specifically, our methodology mimics human cognitive behaviors by recombining and interconnecting heterogeneous data from diverse sources thereby enhancing advanced reasoning capabilities in LLMs. Extensive experiments demonstrate that: (1) effective data organization is essential, and our mapping-based combination learning approach significantly improves data utilization efficiency; (2) by enhancing data *diversity*, *accuracy*, and *complexity*, our synthetic data scales beyond 100B tokens, revealing CCS's strong scalability. Our findings highlight the impact of data organization methods on LLM learning efficiency and the significant potential of scalable synthetic data to enhance model reasoning capabilities.

1 Introduction

> "*One ounce of practice is worth a thousand pounds of theory*"
>
> – Swami Vivekananda

Large Language Models (LLMs) have achieved remarkable advancements across various domains, including conversation (Achiam et al., 2023; Hurst et al., 2024), mathematics (Guo et al., 2025), coding (Anthropic, 2024), and writing (OpenAI, 2025). The success of LLMs is fundamentally dependent upon large-scale training data, according to scaling laws (Kaplan et al., 2020; Hoffmann et al., 2022), models trained on more data generally tend to exhibit predictable performance improvements. Pre-training data, which are utilized during the pre-training phase of LLMs, serve as a crucial component by providing foundational knowledge and general capabilities that underpin model performance.

Scaling pre-training data has emerged as a critical imperative in the development and advancement of LLMs, however some studies (Villalobos et al., 2022; Muennighoff et al., 2023) indicate that the volume of human-generated data is capped and exhibits slow growth. Consequently, efforts are underway to scaling pre-training data beyond merely scaling raw text, such as multimodal data (Team et al., 2023) and synthetic data (Gunasekar et al., 2023). Synthetic data, serving as a supplement and extension to raw data, is garnering increasing attention (Liu et al., 2024b). The Phi series (Gunasekar et al., 2023; Li et al., 2023; Abdin et al., 2024) and Qwen2.5 series (Yang et al., 2024b,a; Hui et al., 2024) extensively utilized synthetic data during pre-training to enhance their mathematical and coding abilities, leading to remarkable performance. Numerous studies (Zhou et al., 2024; Lu et al., 2024; Toshniwal et al., 2024) utilize synthetic mathematical problems to enhance models' reasoning abilities.

However, existing data synthesis methods primarily focus on supervised fine-tuning (SFT) phase, and their scalability remains unverified, while large-scale synthesis methods specifically designed for pre-training are still under-explored. Furthermore, the lack of transparency in the data synthesis strategies employed by the Phi and Qwen2.5 series impedes the advancement in data synthesis. Finally, the learning efficiency of synthetic data remains understudied; while existing refinement-based methods (Maini et al., 2024; Yue et al., 2024) can improve learning efficiency, their diversity and accuracy are often limited. This raises critical questions:

*Corresponding author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 4722–4736
November 4-9, 2025

Q1: "*What are the key factors governing the scalability of synthetic data generation for large-scale pre-training?*"; **Q2**: "*How can data synthesis be performed efficiently to enhance the learning efficiency of LLMs?*"

To address these limitations, we propose Cognitive Combination Synthesis (CCS), a novel framework that enables scalable and efficient data synthesis through data recombination and forward/backward extension of raw data. To answer **Q1**, we conducted large-scale data collection and comprehensive CCS-based experiments, identifying critical scaling factors in synthetic data generation. To answer **Q2**, we enhanced data learning efficiency through cognitive theory-guided mapping techniques. According to established cognitive models (Illeris, 2018; Wang and Chiew, 2010), skill acquisition from declarative knowledge necessitates deliberate practice and corrective feedback across learning stages. Inspired by those learning theories, our CCS framework accelerates the learning process by establishing knowledge-to-skill mappings. Specifically, our CCS framework includes three stages: (1) Mapping Establishment (Section 3.1): collecting reasoning-density data from multiple sources and constructing knowledge-to-skill mapping data; (2) Solution Refinement (Section 3.2): refining data through LLM to enhance learning efficiency; and (3) Quality Filtering (Section 3.3): applying fine-grained filtering to ensure synthetic data quality.

We validate the efficacy of the proposed approach in mathematical reasoning domains. Extensive experiments show that our CCS framework successfully scales to over 100B tokens, validating the strong scalability of our method. Compared to other data synthesis methods, our mapping-based approach achieves significantly higher learning efficiency, confirming the superiority of our cognitive mapping paradigm. Our findings demonstrate the significant potential of synthetic data in the research and development of LLMs, offering a promising avenue for scaling existing pre-training data to substantially larger scales. Moreover, the methodology within CCS involving the recombination and extension of original data also serves as an effective approach to enhance the utilization efficiency of currently available data. Our contributions are summarized as follows:

1. We introduce CCS, a novel framework that scales beyond 100B tokens, and revealing the key factors governing synthetic data scalability: *diversity*, *accuracy*, and *complexity*.

2. Our mapping-based synthesis approach achieved high learning efficiency, highlighting the importance of data combination strategies.

2 Human Cognitive Process

Learning is a relatively permanent change in behavior or behavioral potential that results from experience (Hilgard and Bower, 1966). This definition encompasses both the acquisition of declarative knowledge ('knowing-that')—facts, concepts, and information that alter our understanding and the development of procedural knowledge ('knowing-how')—skills and habits manifested through performance. According to cognitive psychology, human learning is inherently accompanied by comprehension (Council, 2000), when solving specific problems, individuals must identify which conceptual frameworks or prior knowledge to rely upon, as well as how to effectively apply them to address the given challenges. Fitts and Posner's stage of learning theory (Fitts and Posner, 1967) comprises three distinct phases: the cognitive stage, the associative stage and the autonomous stage. This model describes the progression of a performer from an novice to an expert in acquiring a specific skill.

During the cognitive stage, learners comprehend task requirements and develop an initial conceptual understanding of the skill, primarily relying on declarative knowledge. In the associative stage, the learner is learning how to perform the skill well and how to adapt the skill. At this stage, the focus shifts from declarative knowledge to procedural knowledge — translating what to do into how to do it. This stage mirrors deliberate practice in expertise development, where targeted exercises bridge the gap between theoretical understanding and autonomous skill execution. Finally, in the autonomous stage, the skill becomes highly automatic, requiring minimal conscious effort to perform.

Inspired by human cognitive and learning processes, we propose an efficient data synthesis framework that connects knowledge acquisition with skill learning to enhance learning efficiency. Taking mathematics as an example, we define declarative knowledge in the cognitive stage as fundamental mathematical knowledge (such as arith-

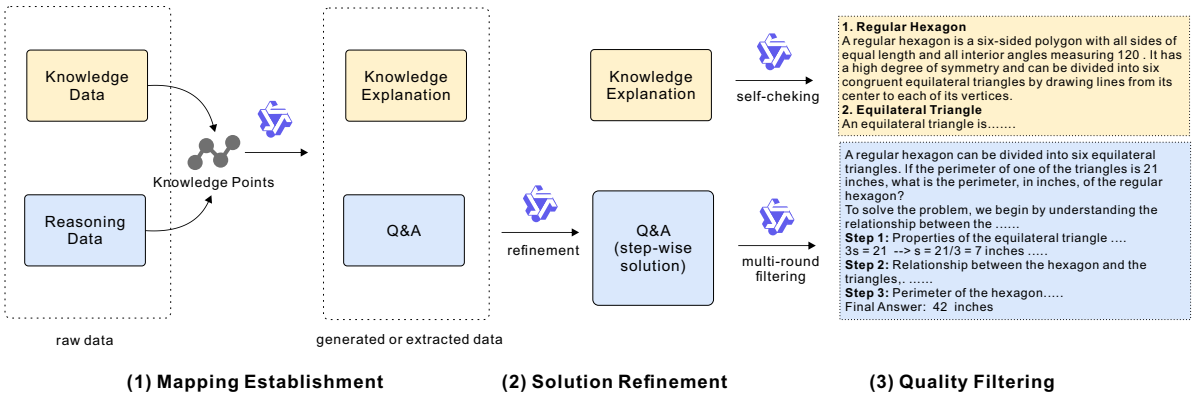


Figure 1: The overall pipeline of our Cognitive Combination Synthesis framework comprises three main stages: Mapping Establishment, Solution Refinement, and Quality Filtering. The Mapping Establishment stage focuses on constructing specific data patterns to achieve knowledge-to-skill mapping. The Solution Refinement stage utilizes a large language model to regenerate more detailed solution steps. Subsequently, the Quality Filtering stage enhances the quality of the synthetic data by validating the knowledge explanations and refined answers.

metic operations, algebraic concepts and basic statistical methods) and define procedural knowledge or skills in the associative stage as mathematical problem-solving ability. Extensive experiments demonstrate the effectiveness of applying human cognitive learning theory to LLMs.

3 The Cognitive Combination Synthesis Framework

In this section, we present details of our Cognitive Combination Synthesis framework for data synthesis. The core principle of our framework lies in linking declarative knowledge and skills to enhance data utilization efficiency. The overall framework comprises three main components: Mapping Establishment, Solution Refinement, and Quality Filtering, as illustrated in Figure 1. In the Mapping Establishment process, acknowledging the heterogeneity of existing data sources (e.g., books, webpages, exercises), we employ multiple strategies to construct the knowledge-to-skill mappings. The Solution Refinement process primarily involves utilizing LLM to rewrite the solution steps of mathematical problems, thereby generating more detailed procedures to facilitate model learning. Finally, the Quality Filtering component performs quality assessments on both the answers and the knowledge explanations to filter out low-quality synthetic data.

3.1 Mapping Establishment

We collect a substantial amount of seed data for synthesis, during the seed construction phase, we systematically collect and extract mathematical knowledge data and problem-solving data, detailed in Appendix A.1. By integrating the aforementioned mathematical knowledge data with problem-solving data to establish knowledge-to-skill mapping, we enhance data utilization efficiency. Given the data heterogeneity, four mapping strategies are utilized, as illustrated in Figure 2

Question-guided Mapping Establishment This method uses math problems as seeds to construct synthetic data. For a given math problem Q, which typically involves multiple knowledge points (KPs), we employ Qwen2.5-7B-Instruct (Yang et al., 2024a) to identify the relevant knowledge points and generate detailed explanations (KEs). These explanations are then followed by the original problem Q (KPs + Q&A), thereby forming synthetic data. We refer to this synthetic data construction method as Question-guided Mapping Establishment (**QME**). We expect this data format to mitigate the model's tendency to memorize problems when lacking sufficient knowledge or mastery of the underlying concepts, thereby promoting a more reasonable knowledge-based reasoning framework during learning. To ensure data diversity, inspired by (Ge et al., 2024) , we employ various explanation styles to enhance data diversity. Detailed prompts are provided in the Appendix A.2.

Knowledge-guided Mapping Establishment In contrast to the QME approach, this method generates math problems from knowledge data to establish knowledge-skill mappings. First, we employ the Qwen2.5-7B-Instruct model to extract

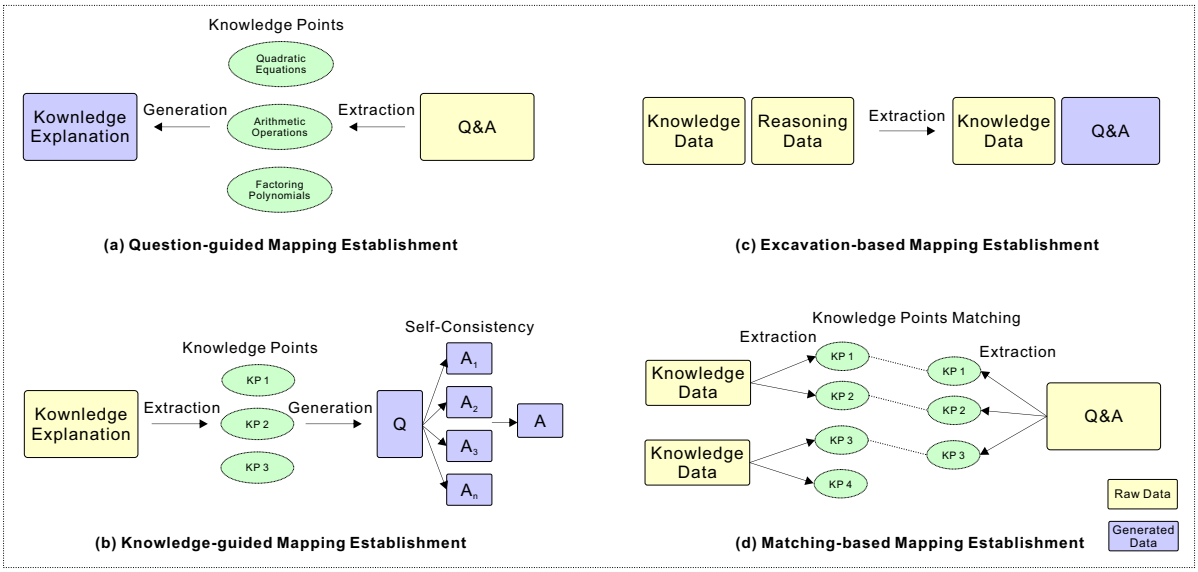


Figure 2: Our mapping establishment process involves four approaches: (a) Question-guided: generating knowledge explanations based on Q&A data; (b) Knowledge-guided: generating problems from knowledge data and utilizes a self-consistency method to derive answers; (c) Excavation-based: mining paired knowledge-reasoning data from existing datasets; (d) Matching-based: bridging existing knowledge data and Q&A data by leveraging shared knowledge points.

knowledge points embedded in the knowledge data. Mathematical problems are then synthesized based on these extracted knowledge points using MathScale's prompt (Tang et al., 2024) with Qwen2.5-72B-Instruct. Subsequently, these synthesized problems are filtered by the above LLM to exclude issues such as incompleteness or contradictions. For the filtered problems, we utilize Qwen2.5-Math-72B-Instruct to generate 32 candidate answers per problem. The final answer for each problem is obtained using a self-consistency approach with majority voting (cons@32). We refer to this synthetic data construction method as Knowledge-guided Mapping Establishment (**KME**). Our method distinguishes itself from other math question synthesis method (Tang et al., 2024; Huang et al., 2025) in two aspects: Firstly, our method established a direct link between the generated problems and their source knowledge explanations. Secondly, we leverage self-consistency to ensure the accuracy of the generated problem answers.

Excavation-based Mapping Establishment We observe that knowledge-skill mappings naturally exist in large-scale unsupervised data, such as textbooks, educational websites and similar resources. We first mine a large corpus of data containing knowledge explanations(KEs) and reasoning processes (Skills). We then extract these reasoning processes and utilize Qwen2.5-72B-Instruct to synthesize corresponding questions and answers (Q&A). The mined KEs and generated Q&A are subsequently combined to form the synthetic dataset (KEs $\rightarrow$ Q&A). We term this approach Excavation-based Mapping Establishment (**EME**).

Matching-based Mapping Establishment Unsupervised data inherently contains a wealth of knowledge and problems, all of which are associated with specific knowledge points. This method establishes connections between existing knowledge data and problem data through knowledge points, thereby creating a mapping from knowledge to skills. We extract corresponding knowledge points from both conceptual knowledge data and problems. These knowledge points then serve as bridges to link the two data types based on shared or similar concepts. We term this approach Matching-based Mapping Establishment (**MME**). Notably, rather than generating new data synthetically, our method recombines existing data to form novel training examples.

3.2 Solution Refinement

Existing raw mathematical problems often have concise and irregular solutions, frequently skipping steps or presenting the final answer before the reasoning. We posit that this data structure is suboptimal for training large models to learn ro-

Model # shots	GSM8K 4-shot	MATH 4-shot	MathQA 4-shot	College 4-shot	Gaokao 4-shot	Olympiad 4-shot	Omni 4-shot	Average
based on Qwen2.5-1.5B								
Qwen2.5-1.5B	68.5	33.0	42.0	22.8	22.1	18.1	13.1	31.4
Qwen2.5-Math-1.5B	76.8	49.0	53.0	37.9	29.9	28.2	20.5	42.2
FineMath-4plus(9.6B)	68.8	34.6	41.5	23.7	23.1	19.1	13.6	32.1
WebInstruct(5B)	69.9	36.2	45.0	24.7	25.2	20.0	14.1	33.6
MegaMath-Synth-Q&A(7B)	71.7	37.4	44.3	25.6	25.5	20.7	16.2	34.5
CCS-QME(5B)	72.9	38.8	46.5	29.1	27.0	21.1	15.6	35.9
CCS-KME(5B)	72.9	37.4	45.4	27.0	25.7	20.7	16.4	35.1
CCS-EME(5B)	73.2	38.4	46.2	28.9	26.0	22.0	17.5	36.0
CCS-MME(5B)	72.5	38.8	46.3	28.7	26.2	21.3	16.8	35.8
CCS(100B)	**81.1**	**51.0**	**56.6**	**41.5**	**36.6**	**30.4**	**22.3**	**45.6**
based on Qwen2.5-7B								
Qwen2.5-7B	85.4	51.0	59.6	33.6	31.7	22.7	15.7	42.8
Qwen2.5-Math-7B	91.6	57.0	63.0	**44.1**	40.0	31.3	21.2	49.8
CCS(100B)	**92.9**	**59.6**	**68.3**	44.0	**40.8**	**31.8**	**23.6**	**51.6**
based on Qwen2.5-72B								
Qwen2.5-72B	91.5	66.8	77.1	48.0	44.9	36.0	20.3	55.0
Qwen2.5-Math-72B	90.8	69.4	76.1	**54.5**	47.5	40.4	27.4	58.0
CCS(100B)	**92.4**	**71.2**	**80.8**	54.4	**50.4**	**41.1**	**28.5**	**59.8**

Table 1: Performance of models of different sizes on math benchmarks. All metrics are reported as percentages (%). We evaluate three model sizes, with the best result highlighted in **bolded**. The number in () represents training sample number. The underlined numbers indicate the best performance among comparative methods. Main results: (1) The CCS synthesis method outperforms alternative approaches, even with less training data; (2) CCS demonstrates strong scaling properties, achieving Qwen2.5-Math level performance at the 100B-token scale.

bust reasoning patterns. To address this issues, we employ Qwen2.5-Math-72B-Instruct to regenerate well-structured solutions for synthetic data, with the goal of maximizing the detail of the reasoning steps. Detailed prompts are provided in the Appendix A.2.

3.3 Quality Filtering

To ensure synthetic data quality, we implement automated quality checks on both knowledge explanations and solution steps generated by models. For knowledge explanations, we employ Qwen2.5-72B-Instruct to detect potential factual errors. For solution steps generated by large language models in both QME, EME and MME approaches, we filter out instances where the model-generated answers deviate from standard reference answers to improve accuracy. In the EME approach, the reference answers are taken directly from the mined data, to further ensure answer reliability, we strictly control data sources by using high-quality materials such as textbooks. All filtered erroneous solutions undergo three regeneration attempts to maximize data utilization efficiency.

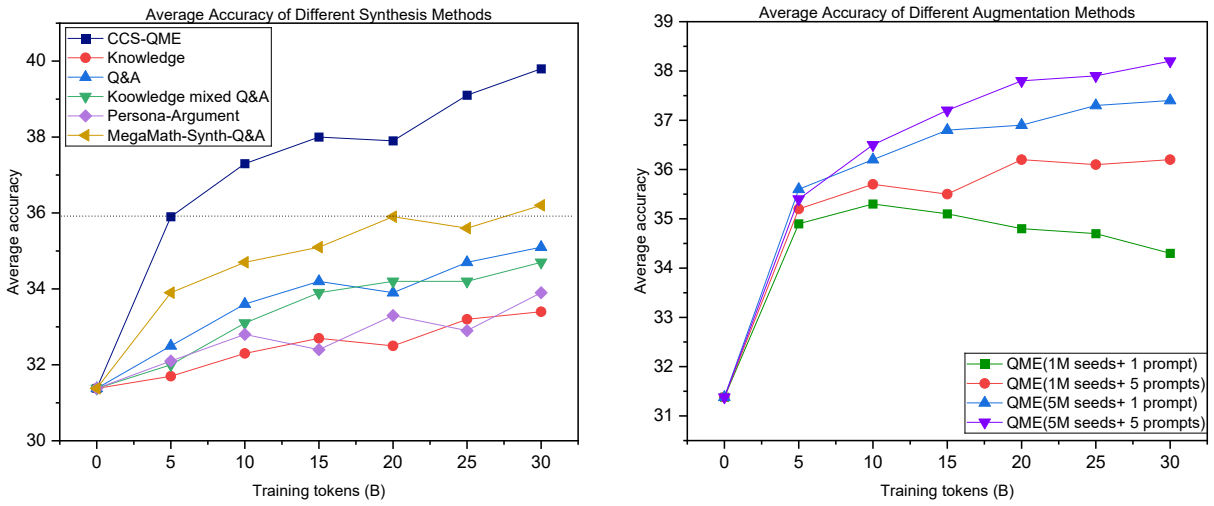
4 Experiments

4.1 Experimental Settings

Datasets. We collected approximately 76M knowledge documents from various online sources and around 21M problems with answers from educational websites, forums, exams, and internal datasets. Ultimately, a total of 100B tokens of data was selected for the experiments, the data synthesis configurations for different mapping establishment methods and their corresponding sampling ratios are provided in the Appendix A.3.

Evaluation. We assess models' mathematical reasoning on synthetic data using diverse benchmarks, including GSM8K (Cobbe et al., 2021), MATH-500(MATH) (Lightman et al., 2023), MathQA (Amini et al., 2019), OlympiadBench-Math(Olympiad) (He et al., 2024), Omni-MATH(Omni) (Gao et al., 2024), Gaokao 2023 En(Gaokao) (Liao et al., 2024), and College-Math(College) (Tang et al., 2024).

Baselines. Large-scale open-source synthetic mathematical data for pre-training remains scarce. We compare CCS synthetic data against FineMath-



(a) efficiency (b) diversity

Figure 3: Average accuracy of different methods trained on Qwen2.5-1.5B. (a) Our CCS method achieves superior learning efficiency over other approaches by establishing knowledge-to-skill mappings. (b) The diversity of original seed data is critical for scaling synthetic data.

4plus (Allal et al., 2025), WebInstruct (Yue et al., 2024) and the recent MegaMath-Synth-Q&A (Zhou et al., 2025). To further evaluate our method's scaling properties, we compare it with the math-specific Qwen2.5-Math series. Training settings are detailed in the Appendix A.4

4.2 Main Results

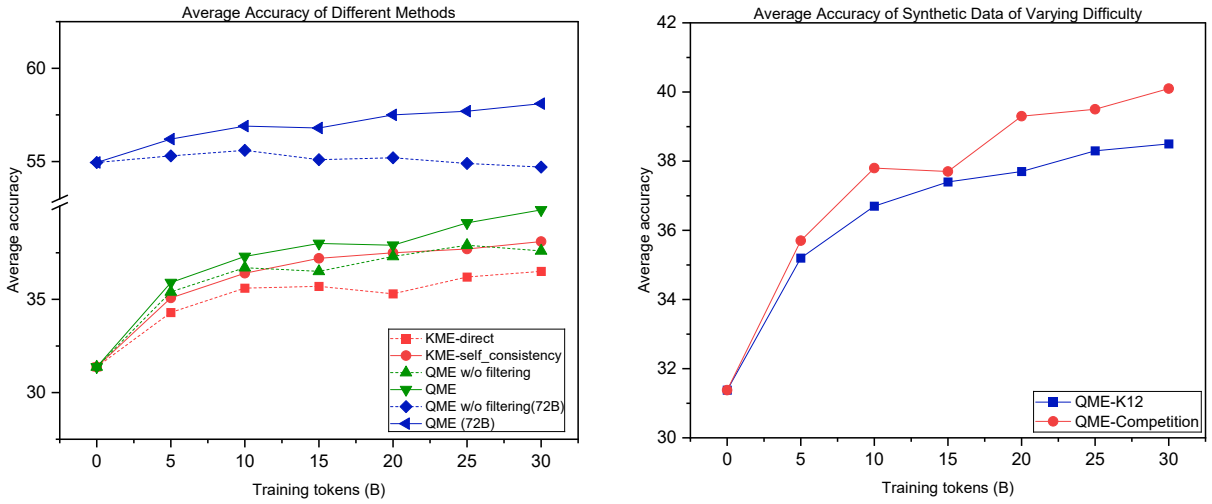
We evaluate our CCS framework on mathmatical benchmarks. First, we validate the effectiveness of the CCS synthesis method. we performed continued pre-training on the Qwen2.5-1.5B base model using different datasets: CCS synthetic data(5B tokens), FineMath-4plus (9.6B tokens), WebInstruct (5B tokens), and MegaMath-Synth-Q&A (7.0B tokens). As presented in Table 1, the CCS data synthesis method yields significantly better results than other methods. Furthermore, to examine scaling properties, we performed continued pre-training on Qwen2.5 base models with 100B tokens synthetic data, as showed in Table 1, which achieving better performance than comparable-sized Qwen2.5-Math models, thereby confirming the scalability of our CCS synthesis method.

Comparison of Different Mapping Methods We compared different mapping establishment approaches while controlling the data scale. As presented in Table 1, strategies using questions as seeds are superior to those using knowledge as seeds. We attribute this primarily to the fact that, for current LLMs, generating reasonable questions and providing correct answers is significantly more challenging than generating correct knowledge.

4.3 Learning Efficiency of Our Method

We compared the effectiveness of mapping-based methods with non-mapping methods. For non-mapping approaches, we evaluated Q&A data and knowledge-based data separately, as well as a 50%-50% hybrid of both to validate our combination method. Figure 3a show that the mapping-based approach significantly outperformed using only Q&A or knowledge data. Furthermore, the CCS method surpassed the hybrid method, indicating that effective data organization, not just the seed data, drives performance improvements.

Additionally, we compared our method with mainstream data synthesis methods: (1) For refinement-based methods, we use the recently open-sourced MegaMath-Synth-Q&A, due to its demonstrated excellent performance. (2) For problem generalization methods, we synthesized questions from the open-source Persona-Hub dataset (Ge et al., 2024) and generated answers via Qwen2.5-Math-72B-Instruct. For datasets smaller than 30B tokens, we train for multiple epochs to reach 30B tokens. As shown in Figure 3a, our CCS method outperformed all baselines, achieving results comparable to MegaMath-Synth-Q&A with only 1/4 of the data, demonstrating superior learning efficiency. We hypothesize this is because this approach, which mimics human learning processes, facilitates reasoning within a more appro-



(a) accuracy (b) complexity

Figure 4: (a)The impact of data accuracy on the scalability of synthetic data. data accuracy is a critical factor for successful scaling. (b) The impact of data complexity on the scalability of synthetic data.

priate framework and establishes a mapping from knowledge to skills, thereby smoothing the learning process and achieving higher learning efficiency.

4.4 The Scalability of Our Method

In contrast to synthetic data used for the SFT phase, scaling synthetic data for pre-training remains a key challenge. Our CCS framework successfully scaled synthetic data generation to the hundreds of billions of tokens. Based on extensive experimentation, we revealed the key factors affecting the scaling properties of synthetic data: *diversity*, *accuracy*, and *complexity*. Furthermore, we evaluated our method's generalization on STEM tasks. The results show that our CCS synthesis method exhibits strong generalization. See Appendix A.5 for details.

Diversity Within the CCS framework, using the QME method as an example, we investigated the influence of original seed data and explanation style prompt diversity on synthetic data scalability. We compared generating knowledge explanations in 1 style versus 5 styles per seed. Figure 3b shows that at equal data scales, using diverse seed questions outperforms multi-style prompts. Moreover, multi-style prompts generation per seed surpasses repeatedly training on each seed without style variation. Our experiments highlight that the diversity of original seed data is critical for scaling synthetic data. In the CCS framework, we enhance the diversity of the final synthetic data through seed diversity, style prompt diversity, and method diversity (employing various mapping strategies).

Accuracy The accuracy of LLM-generated synthetic data is often unreliable. We demonstrate that data accuracy is a critical factor for successful scaling. First, within the KME method, we compared the performance of using the model's direct output as the final answer (KME-direct) against using the result of applying self-consistency to multiple model responses (KME-self_consistency). Figure 4a shows that the self-consistency approach significantly outperforms using the model's direct output. Second, for the QME method, we examined the impact of reference-based consistency filtering. As shown in Figure 4a, while the unfiltered QME method plateaus around 25B tokens, the filtered method demonstrates continuous improvement and is significantly superior, highlighting the critical role of data quality for synthetic data scaling. Finally, we analyzed the sensitivity of different sized models to data quality. Figure 4a shows that on unfiltered QME data, the 7B model continued to improve, whereas the 72B model quickly plateaus or even degrades in performance, suggesting diminishing returns from lower-quality data for stronger base models.

Complexity Using QME as a case study, we simulate data complexity through different question seed sources: QME-K12 (containing K12-level questions) and QME-Competition (containing competition-level questions). Figure 4b shows QME-Competition significantly outperforms QME-K12, demonstrating the importance of synthetic

Data	Average
CCS(10B)	37.9
CCS-QME only	37.3
w/o solution refinement	35.9
w/o quality filtering	36.4

Table 2: The average accuracy across different methods demonstrates that multiple mapping approaches, solution refinement, and quality filtering significantly impact the effectiveness of the synthesized data.

data complexity. For simpler data, LLMs plateau after learning from a certain volume, with additional data providing diminishing returns.

4.5 Ablation Study

In this section, we present detailed ablation studies on our CCS framework. We synthesized 10B tokens of data using the same seeds to evaluate the impact of various factors. For datasets smaller than 10B tokens, we repeated training until reaching 10B tokens to ensure fair comparison.

First, we investigated the effectiveness of employing multiple mapping methods during data synthesis. As shown in Table 2, using four mapping methods outperformed using only the QME method, primarily due to the contribution of method diversity, as mentioned previously. As discussed previously, this is because multiple synthesis methods result in better data diversity, thereby enhancing the final results.

Next, we investigated the impact of solution refinement. Table 2 indicates that skipping this step leads to a 2-point performance drop. These findings validate the effectiveness of our Solution Refinement method in improving the ill-structured problems commonly found in raw Q&A data. This is primarily because raw Q&A data often contains issues such as incomplete reasoning and skipped steps that hinder model learning. Solution refinement generates more structured and detailed step-by-step explanations, facilitating better model training.

Finally, we assessed the effect of quality filtering on synthetic data. Table 2 shows that omitting filtering results in an average 1.5-point performance decline. As previously noted, quality filtering improves accuracy, which significantly influences the effectiveness of the synthesized data.

5 Related Work

Synthetic Data Generation. Synthetic data has emerged as an integral component in the development of LLMs, playing a critical role in both pre-training (Gunasekar et al., 2023; Li et al., 2023; Abdin et al., 2024; Yang et al., 2024b) and post-training phases (Grattafiori et al., 2024; Adler et al., 2024). However, improper use of synthetic data can lead to model collapse (Shumailov et al., 2024) and hallucination issues (Liu et al., 2024b), numerous studies have sought to enhance the usability of synthetic data. (Yu et al., 2023; Li et al., 2024a) increase the volume of synthetic data through problem generalization, (Yue et al., 2024) extracts Q&A from the large-scale websites to ensure data diversity, (Li et al., 2024c; Morishita et al., 2023) ensure the accuracy of synthetic data through code or formal verification. These methods have only been validated on small-scale data, lacking large-scale scalability verification. (Qin et al., 2025) has investigated the scaling laws of synthetic data, however, their approach exhibits performance saturation, revealing its limitations. Although models like the Phi series (Gunasekar et al., 2023; Li et al., 2023; Abdin et al., 2024) and Qwen2.5 series (Yang et al., 2024b,a; Hui et al., 2024) have extensively utilized synthetic data during pre-training, achieving notable results, their underlying methodologies remain opaque. Addressing these limitations, this paper introduces a novel data synthesis framework, demonstrates its scalability properties, and reveals the critical factors underlying successful scaling.

Data Efficiency in LLM Training. Humans possess the remarkable ability to learn and infer from limited data (Illeris, 2018), whereas LLMs typically require extensive samples to acquire comparable knowledge (Allen-Zhu and Li, 2024). Despite the significant advancements achieved by LLMs recently, their learning efficiency remains under-investigated. Existing approaches to improve data efficiency include data augmentation (Ding et al., 2024), curriculum learning (Liu et al., 2024a; Li et al., 2022), and data selection (Li et al., 2024b). (Sachdeva et al., 2024) achieve efficient model training through the automatic assessment of data diversity and coverage. (Maini et al., 2024) demonstrates rephrased data enables significantly more efficient training than original web data. (Ye et al., 2024) shows that carefully synthesized error-correction data yields higher learning efficiency than raw data. In this paper, we propose a data-

efficient synthesis method through strategic data combination, demonstrating superior learning efficiency over alternative approaches.

6 Conclusion

In this work, we propose CCS, a novel data synthesis framework inspired by human cognitive learning processes, and demonstrate its ability to achieve efficient learning through the combination of knowledge-based and skill-based data. This approach achieves superior efficiency over other data synthesis methods, validating the efficacy of this combination learning paradigm. Furthermore, we scale CCS to hundreds of billions of tokens, with extensive experiments revealing three critical factors for scalable data synthesis: *diversity*, *accuracy*, and *complexity*. In future work, we spire to extend CCS to broader reasoning domains and explore additional combination learning data synthesis paradigms to push the capability boundaries of large language models.

Limitations

Although our combination learning approach enhances training efficiency, the underlying learning mechanisms of large language models and more effective data utilization strategies require further investigation. The effectiveness of the CCS framework relies heavily on the quality of the original knowledge-based and skill-based datasets, as biases or inaccuracies in the source data may propagate through the synthesis process. Although quality filtering is employed to enhance synthesized data, there remains significant scope for achieving higher data quality, potentially through the development of improved synthesis models or the integration of formal verification.

Acknowledgements

We would like to thank the anonymous reviewers and meta-reviewer for their constructive feedback on this work. This work was supported by the National Natural Science Foundation of China (No. 62441239)

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, Bryan Catanzaro, Sharon Clay, Jonathan Cohen, and 1 others. 2024. Nemotron-4 340b technical report. *arXiv preprint arXiv:2406.11704*.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, and 1 others. 2025. Smollm2: When smol goes big–data-centric training of a small language model. *arXiv preprint arXiv:2502.02737*.

Zeyuan Allen-Zhu and Yuanzhi Li. 2024. Physics of language models: Part 3.3, knowledge capacity scaling laws. *arXiv preprint arXiv:2404.05405*.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint arXiv:1905.13319*.

AI Anthropic. 2024. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 3(6).

Associate Editors of Mathematical Reviews and zbMATH. 2020. MSC2020-Mathematics Subject Classification System. Available online.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

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. *arXiv preprint arXiv:2110.14168*.

National Research Council. 2000. *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. The National Academies Press, Washington, DC.

Bosheng Ding, Chengwei Qin, Ruochen Zhao, Tianze Luo, Xinze Li, Guizhen Chen, Wenhan Xia, Junjie Hu, Anh Tuan Luu, and Shafiq Joty. 2024. Data augmentation using large language models: Data perspectives, learning paradigms and challenges. *arXiv preprint arXiv:2403.02990*.

Paul Morris Fitts and Michael I. Posner. 1967. *Human Performance*. Brooks/Cole Publishing Company, Belmont, CA.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, and 1 others. 2024. Omni-math: A universal olympiad level mathematic benchmark for large language models. *arXiv preprint arXiv:2410.07985*.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, and 1 others. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ernest R. Hilgard and Gordon H. Bower. 1966. *Theories of Learning*, 3rd edition. Appleton-Century-Crofts, New York.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

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 AAAI Conference on Artificial Intelligence*, pages 24176–24184.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, and 1 others. 2024. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Knud Illeris. 2018. A comprehensive understanding of human learning. In *Contemporary theories of learning*, pages 1–14. Routledge.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. 2024a. Common 7b language models already possess strong math capabilities. *arXiv preprint arXiv:2403.04706*.

Conglong Li, Zhewei Yao, Xiaoxia Wu, Minjia Zhang, Connor Holmes, Cheng Li, and Yuxiong He. 2024b. Deepspeed data efficiency: Improving deep learning model quality and training efficiency via efficient data sampling and routing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 18490–18498.

Conglong Li, Minjia Zhang, and Yuxiong He. 2022. The stability-efficiency dilemma: Investigating sequence length warmup for training gpt models. *Advances in Neural Information Processing Systems*, 35:26736–26750.

Xuefeng Li, Yanheng He, and Pengfei Liu. 2024c. Synthesizing verified mathematical problems. In *The 4th Workshop on Mathematical Reasoning and AI at NeurIPS'24*.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Minpeng Liao, Wei Luo, Chengxi Li, Jing Wu, and Kai Fan. 2024. Mario: Math reasoning with code interpreter output–a reproducible pipeline. *arXiv preprint arXiv:2401.08190*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Ji Liu, Jiaxiang Ren, Ruoming Jin, Zijie Zhang, Yang Zhou, Patrick Valduriez, and Dejing Dou. 2024a. Fisher information-based efficient curriculum federated learning with large language models. *arXiv preprint arXiv:2410.00131*.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and 1 others. 2024b. Best practices and lessons learned on synthetic data. *arXiv preprint arXiv:2404.07503*.

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. *arXiv preprint arXiv:2402.16352*.

Pratyush Maini, Skyler Seto, He Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. 2024. Rephrasing the web: A recipe for compute and data-efficient language modeling. *arXiv preprint arXiv:2401.16380*.

Terufumi Morishita, Gaku Morio, Atsuki Yamaguchi, and Yasuhiro Sogawa. 2023. Learning deductive reasoning from synthetic corpus based on formal logic. In *International Conference on Machine Learning*, pages 25254–25274. PMLR.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. 2023. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376.

OpenAI. 2025. Openai gpt-4.5 system card. Technical report, OpenAI.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, and 1 others. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849.

Zeyu Qin, Qingxiu Dong, Xingxing Zhang, Li Dong, Xiaolong Huang, Ziyi Yang, Mahmoud Khademi, Dongdong Zhang, Hany Hassan Awadalla, Yi R Fung, and 1 others. 2025. Scaling laws of synthetic data for language models. *arXiv preprint arXiv:2503.19551*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Noveen Sachdeva, Benjamin Coleman, Wang-Cheng Kang, Jianmo Ni, Lichan Hong, Ed H Chi, James Caverlee, Julian McAuley, and Derek Zhiyuan Cheng. 2024. How to train data-efficient llms. *arXiv preprint arXiv:2402.09668*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759.

Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. 2024. Mathscale: Scaling instruction tuning for mathematical reasoning. *arXiv preprint arXiv:2403.02884*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

M-A-P Team, Xinrun Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, Kang Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, Kaixing Deng, Shuyue Guo, Shian Jia, Sichao Jiang, Yiyan Liao, Rui Li, Qinrui Li, and 76 others. 2025. Supergpqa: Scaling llm evaluation across 285 graduate disciplines. *Preprint*, arXiv:2502.14739.

Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. 2024. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *Advances in Neural Information Processing Systems*, 37:34737–34774.

Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. 2022. Will we run out of data? limits of llm scaling based on human-generated data. *arXiv preprint arXiv:2211.04325*.

Yingxu Wang and Vincent Chiew. 2010. On the cognitive process of human problem solving. *Cognitive systems research*, 11(1):81–92.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024b. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Tian Ye, Zicheng Xu, Yuanzhi Li, and Zeyuan Allen-Zhu. 2024. Physics of language models: Part 2.2, how to learn from mistakes on grade-school math problems. *arXiv preprint arXiv:2408.16293*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Xiang Yue, Tianyu Zheng, Ge Zhang, and Wenhu Chen. 2024. Mammoth2: Scaling instructions from the web. *Advances in Neural Information Processing Systems*, 37:90629–90660.

Fan Zhou, Zengzhi Wang, Nikhil Ranjan, Zhoujun Cheng, Liping Tang, Guowei He, Zhengzhong Liu, and Eric P. Xing. 2025. Megamath: Pushing the limits of open math corpora. *arXiv preprint arXiv:2504.02807*. Preprint.

Kun Zhou, Beichen Zhang, Zhipeng Chen, Xin Zhao, Jing Sha, Zhichao Sheng, Shijin Wang, Ji-Rong Wen, and 1 others. 2024. Jiuzhang3. 0: Efficiently improving mathematical reasoning by training small data synthesis models. *Advances in Neural Information Processing Systems*, 37:1854–1889.

A Appendix

A.1 Seed Construction

Knowledge-based data. This serve as the primary carriers of mathematical knowledge, encompassing diverse sources such as textbooks, wiki entries, web-pages, and syllabus. To construct a comprehensive mathematical knowledge base, we systematically extract math-relevant content from large-scale unsupervised data through the following methods:

- **Web-pages**: for web-based mathematical content, we use FineMath (Allal et al., 2025), a large scale mathematical educational content filtered from CommonCrawl.
- **Books**: mathematical textbooks and reference materials are retrieved from open-access digital libraries, including Anna's Archive [1] and Project Gutenberg [2]. We apply a mathematical content classifier to filter out non-mathematical books and retain only those with substantial mathematical relevance.
- **Wikis**: We develop a mathematical classifier to automatically identify and extract math-specific concepts and knowledge entities from Wikipedia articles. Following the training methodology proposed in (Shao et al., 2024), we first trained an initial classifier using web data labeled by Qwen2.5-72B-Instruct. We then iteratively refined the model to construct a dataset of 200k samples. The final classifier achieved an accuracy of 87.3% on a human-annotated test set.
- **Educational Supplements**: Additional structured mathematical materials, such as exercise sets, lecture notes, and curriculum-aligned content, are collected from GitHub repositories, open educational datasets (e.g., OpenStax), and institutional websites. A model-based parser is used to standardize the extracted content into a unified format. Specifically, we first applied Qwen2.5-VL-72B-Instruct (Bai et al., 2025) to convert non-textual image content into text. We then discarded articles containing fewer than 50 words. Finally, the data were processed following the FineMath (Allal et al., 2025) pipeline to facilitate subsequent text classification.

This multi-source approach ensures broad coverage of mathematical knowledge while maintaining rigorous filtering to exclude noisy or non-relevant data. For our knowledge data, we will use FineWeb's PII removal method (Penedo et al., 2024) to remove personal information. All collected data adheres to the original data usage agreements.

Problem-Solving data. The problem dataset serves as a critical component of our study, encompassing a comprehensive collection of mathematical problems spanning elementary school to postgraduate levels. This dataset is carefully curated from diverse sources, including examinations, practice exercises and competitions. The dataset comprises tens of millions of problems, providing broad representation across key mathematical domains: Algebra, Functions, Inequalities, Set Theory, Probability and Statistics. Data acquisition involved several methods: collection of open-source datasets (e.g., WebInstruct (Yue et al., 2024)), mining public resources such as CommonCrawl, extraction from textbooks and examination materials, and the use of proprietary data. We employed Qwen2.5-72B-Instruct to filter out questions that are incomplete, ambiguous, or involve proof problems with answers that are difficult to verify, while also extracting final answers for subsequent verification. We perform MinHash-based fuzzy deduplication on these questions against evaluation benchmarks.

A.2 Prompts of Our Method

Knowledge Points Extraction Leveraging an existing mathematic taxonomy - MSC2020 (Mathematics Subject Classification) (Associate Editors

[1] https://annas-archive.org/
[2] https://www.gutenberg.org/

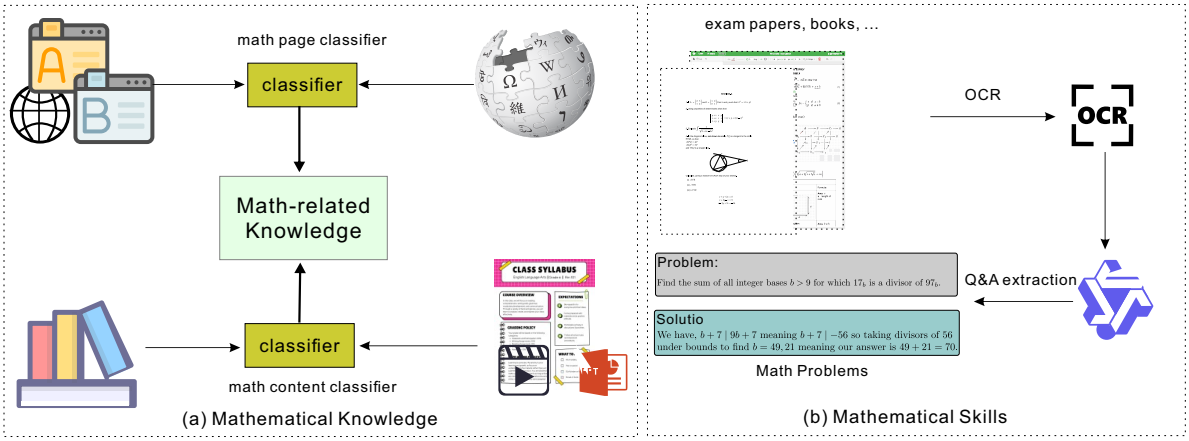


Figure 5: The seed data construction pipeline for data synthesis framework. The whole process contains two categories : (a) mathematical knowledge data, mined from multi-source unsupervised datasets, and (b) mathematical skills data, extracted from examinations, textbooks, and open-source materials.

of Mathematical Reviews and zbMATH, 2020), we identify knowledge points in mathematical texts and problems. Due to MSC2020's multi-level tagging system, we extracted knowledge points hierarchically. The prompt used is shown in Figure 6.

Knowledge Explanation Generation Building upon the extracted knowledge points, given problems, and math-relevant personas selected from Persona Hub (Ge et al., 2024), we utilize a LLM to generate multi-style detailed explanations of the concepts, with the prompt illustrated in Figure 7.

Solution Refinement The generated explanations of knowledge points and the problem are provided as input to a large language model to generate detailed solution steps. The prompt is presented in Figure 8.

A.3 Data composition of the experiment

For different mapping methods, we used different configurations to synthesize data. For QME, 5 distinct styles of knowledge point explanations were generated per problem, yielding 105M entries (see section A.2). for KME, one problem was generated from each of 10M selected knowledge documents, resulting in 10M entries. For EME, problems were generated from 6M knowledge and reasoning documents, resulting in 6M entries. For MME, for each problem, we select one or more pieces of knowledge data based on its identified knowledge points to form knowledge explanations, finally, 5 kind of knowledge explanations were assembled via knowledge point matching, creating approximately 105M total entries. This process ensures comprehensive knowledge coverage, mitigating potential hallucination caused by incomplete knowledge coverage. For our experiments, we sampled 100B tokens from the complete dataset, with the distribution across different methods shown in Table 3. This selection was based on both performance variations observed in preliminary experiments and the available data scale for each method.

Methods	Tokens(B)
QME	40
KME	10
EME	10
MME	40

Table 3: composition of synthetic data in the experiment.

A.4 Training Settings

We employ the Qwen2.5 series for data synthesis. Given the large scale of the training data, unless otherwise specified, we conduct experiments on the Qwen2.5-1.5B base model. We perform continued pre-training on base models. The Adam optimizer is employed with a linear learning rate scheduler, starting at 1e-4 and decaying to a minimum of 3e-5, using a batch size of 12M tokens.

A.5 Generalization of Our CCS method

We evaluate the generalization capability of our CCS framework beyond mathematics using STEM benchmarks, including GPQA (Rein et al., 2024), SuperGPQA (Team et al., 2025), MMLU-STEM (Hendrycks et al., 2021), and OlympiadBench-

As a Mathematics Knowledge Classification Expert, you are tasked with identifying one or more knowledge points from a given mathematical document or problem. Select knowledge points exclusively from the provided list {{KPs}}.

Follow these requirements:
1. For non-mathematical content, return `others`.
2. For a mathematical problem, identify all applicable knowledge points being tested.

Input Text: {{text}}
Output: List all identified knowledge points, separated by `#`.

Figure 6: prompt for knowledge points extraction

As an erudite mathematician, given a mathematical problem and its associated knowledge points, provide an explanation of the foundational knowledge underlying this problem to an eager learner.

The learner's background information is as follows:
{{persona}}

Instruction Requirements for the Final Explanation:
1. When explaining knowledge points, reference the given problem and provide concrete examples where applicable.
2. Tailor the explanation to the learner's knowledge level and background ({{persona}}).
3. Do not merely list concepts—develop each one thoroughly before proceeding to the next. Prioritize depth of understanding and comprehensive exploration over breadth.
4. Maintain rigor (ensure in-depth conceptual coverage) and practical relevance (use specific examples, equations, or proofs where appropriate—e.g., if teaching integration, demonstrate its application through worked solutions).

Given:
- Problem: {{qustions}}
- Knowledge Points: {{KPs}}

Proceed with the detailed explanation.

Figure 7: prompt for knowledge explanation generation

Act as a meticulous mathematician. Given a mathematical problem and its underlying background knowledge, provide the solution steps and the final answer.

Requirements:
1. **Detail:** Include all intermediate steps, calculations, and reasoning. Do not skip any logical transitions. Explain *why* each step is taken.
2. **Rigor:** Ensure mathematical and logical correctness. Use precise terminology and notation. Justify key inferences or assumptions. Include relevant definitions, theorems, formulas, and proofs where necessary to maintain rigor (e.g., using LaTeX for mathematical expressions).
3. **Step-by-Step:** Present the solution as a numbered list of distinct steps. Each step should represent a clear progression towards the final solution.
4. **Completeness:** The solution should be self-contained and fully explain how to arrive at the answer from the problem statement.

Underlying background Knowledge:
{{knowledge explanations}}

Problem:
{{problem}}

Output:

Figure 8: prompt for solution refinement

Model # shots	GPQA 4-shot	SuperGPQA 4-shot	MMLU-STEM 4-shot	OlympiadBench-Physics 4-shot	Average
ability generalization					
Qwen2.5-1.5B	30.3	19.27	50.36	6.12	26.51
Qwen2.5-Math-1.5B	30.3	14.24	46.97	15.96	26.87
CCS-Math(100B)	33.84	17.07	53.03	11.44	28.85
method generalization					
Qwen2.5-1.5B	30.3	19.27	50.36	6.12	26.51
CCS-STEM(5B)	33.84	20.77	57.69	14.63	31.73

Table 4: Average accuracy of our method on STEM benchmarks, demonstrates the strong generalization capability of our CCS data synthesis approach.

Physics(He et al., 2024), with a 4-shot evaluation protocol. OlympiadBench-Physics incorporates a subset of Chinese examination questions, with the remaining benchmark items presented in English. First, we validate our mathematics-based synthetic data on science reasoning tasks. Table 4 shows that our CCS method yields broader improvements in STEM reasoning capabilities compared to the baseline. Second, we demonstrate the generalizability of the CCS method to other domains: using only 5B synthetic data derived from STEM sources, it achieves significant gains on GPQA and OlympiadBench, highlighting the method's applicability beyond mathematics.