VCR: A "Cone of Experience" Driven Synthetic Data Generation Framework for Mathematical Reasoning

Sannyuya Liu[1,2], **Jintian Feng**[1,2], **Xiaoxuan Shen**[1,2], **Shengyingjie Liu**[1,2], **Qian Wan**[1,2*], **Jianwen Sun**[1,2*]

[1]Faculty of Artificial Intelligence in Education, Central China Normal University, Wuhan 430079, China
[2]National Engineering Research Center of Educational Big Data, Central China Normal University, Wuhan 430079, China
{liusy027, shenxiaoxuan, wanq8228, sunjw}@ccnu.edu.cn, {fjt2018, lsyj}@mails.ccnu.edu.cn

Abstract

Large language models (LLMs) have shown excellent performance in natural language processing but struggle with mathematical reasoning. As the training mode gradually solidifies, researchers propose a data-centric concept of artificial intelligence, emphasizing the development of higher-quality data to empower LLMs. Existing studies construct synthetic data for mathematical reasoning by expanding public datasets, thereby performing supervised fine-tuning of LLMs. However, these methods mostly focus on quantity while neglecting quality. The challenging samples fail to receive adequate consideration during data synthesis process, resulting in high construction costs, low-quality density, and serious data homogenization. This paper proposes a multi-agent environment called **V**irtual **C**lass**R**oom (VCR), which leverages various agents driven by LLM to construct high-quality diversified synthetic data. Inspired by the "Cone of Experience" educational theory, VCR introduces three experience levels (direct, iconic, and symbolic) into data synthesis process by analogy with human learning. A user-friendly instruction set and role-playing system are carefully designed, enabling VCR to autonomously plan the scale of synthetic data. This system covers various educational scenarios, including lecture, discussion, problem design and problem-solving. The Adaboost idea embodied in the global iterative process further promotes steady performance improvement. Extensive experiments show that the synthetic data generated by VCR possess higher quality density and generalization capability, which can give LLMs superior mathematical reasoning performance with the same scale.

Introduction

Large Language Models (LLMs) have demonstrated impressive performance in areas such as dialogue and code generation (Minaee et al. 2024; Luo and Yang 2024; OpenAI 2024; Dubey et al. 2024), but they continue to face challenges in mathematical reasoning tasks (Ahn et al. 2024).

The rapid development in the field of LLMs has led to a gradual homogenization of model architectures and training methods. From the perspective of Data-centric AI (Zha et al. 2023), constructing higher-quality synthetic data for supervised fine-tuning (SFT) (Ouyang et al. 2022) of LLMs is expected to further enhance their mathematical reasoning capabilities (Ahn et al. 2024; Yu et al. 2024; Li et al. 2024a,b; Zeng et al. 2024; Liu et al. 2024a).

Research shows that synthetic data, as an effective data augmentation method, can expand existing datasets to achieve data enrichment. For example, MetaMathQA (Yu et al. 2024) and XwinMathQA (Li et al. 2024a) enhanced the queries and responses in the GSM8K and MATH using prompts with strong closed-models such as GPT-3.5 and GPT-4. The quality of synthetic data directly determines the performance of LLMs (Zhang et al. 2024a; Cao et al. 2024; Gunasekar et al. 2023), and its construction process must consider factors such as scale, relevance, accuracy, and diversity (Zeng et al. 2024). Previous works have overly focused on data scale while neglecting quality dimensions, leading to high construction costs, low-quality density, and severe data homogenization issues.

To generate high-quality synthetic data, some works drew an analogy between LLM training and human learning processes, and proposed various methods for generating synthetic data. For instance, LEMA (An et al. 2024) introduced a strategy of learning from mistakes, creating erroneous reasoning data and its correction as training data. MMIQC (Liu et al. 2024a) and SkyworkMathQA (Zeng et al. 2024) rewrote the training set questions into other formats to generate synthetic data, somewhat mimicking the human learning process through diverse practice formats. However, these simple analogy methods cannot fully transfer the multi-level experience in the human learning process, hence they struggle to profoundly guide the data synthesis process.

Human learning is an extremely complex process that includes both observing others and self-practice. The "Cone of Experience" educational theory (Dale 1947) posits that human learning consists of three levels of experience: direct experience (Learning by Doing), iconic experience (Learning through Observation), and symbolic experience (Learning through Abstractions) [1]. Only through the full integration of these three levels of experience can high-quality learning be achieved. Inspired by "Cone of Experience", we design three levels of experiences for LLM training, aiming

*The corresponding authors.

[1]In this paper, we mainly use doing experience, observation experience, and abstraction experience to enhance readability.

to enhance existing datasets and integrate process-oriented information to generate high-quality synthetic data within real educational scenarios. However, due to limitations in ethical concerns and speed of data collection in real educational settings (Yue et al. 2024), we use the strong human-like abilities of LLMs to simulate an educational scenario with LLM-based agents.

In summary, this paper proposes a multi-agent environment called **V**irtual **C**lass**R**oom (VCR), which leverages various agents driven by LLMs to construct high-quality, diversified synthetic data. To embody the three levels of human learning experiences, VCR includes the following stages: **i) Doing Stage**. This stage aims to distinguish the attention of each problem in the training set. Our data augmentation process focuses on challenging problems that LLMs have not been able to solve correctly. **ii) Observation Stage**. In the observation stage, we set multiple roles such as teacher, teaching assistant, student, and classroom noter. Each problem is analyzed in-depth through steps including teacher guidance, student discussion, and summary analysis. **iii) Abstraction Stage**. In this stage, agents summarize problems, extract key elements, and create similar problems for further practice.

Based on these stages, VCR can autonomously analyze sample biases in the target problem domain and adaptively plan the expected scale of synthetic data. A user-friendly set of instructions and a role-playing system are carefully designed to cover teaching activities such as lecturing, discussion, problem design, and problem-solving. The integration of the Adaboost (Ying et al. 2013) ideas into the global iterative process can further promote the steady enhancement of performance. Extensive experiments demonstrate that the synthetic data generated by VCR has higher quality density and generalization performance, providing superior mathematical reasoning capabilities to LLMs at the same scale. Notably, through extensive qualitative analysis, we discover for the first time that indirect data related to thought discussions also contributes to enhancing LLMs' mathematical reasoning abilities, further proving the similarity between LLMs and human learning.

Our contributions can be summarized as follows:

- This paper proposes a multi-agent environment based on the "Cone of Experience" simulation teaching scene, named Virtual Classroom (VCR). VCR achieves the concretization of multi-level experience by analogizing human learning, providing high-quality synthetic data for LLM mathematical reasoning.
- Based on a well-designed instruction set and role-playing system, VCR is entirely driven by LLM-based agents, which not only realize adaptive planning of data scale, but also cover a wide range of educational environments such as lecture, discussion, problem design and problem-solving.
- Extensive experiments demonstrate the effectiveness and advancement of VCR. Under the same data scale, VCR-generated data significantly enhances mathematical reasoning performance more effectively than baselines.
- Qualitative analysis further shows that task-related indirect data is significantly beneficial to LLMs learning, which indicates the feasibility of analogy learning by human experience and can lead to further development for SFT data construction.

Related Work

Mathematical Reasoning with LLMs Although LLMs have demonstrated exceptional capabilities across many tasks (Minaee et al. 2024), significant challenges remain in complex mathematical reasoning tasks (Ahn et al. 2024) such as those in MATH (Hendrycks et al. 2021) and GSM8K (Cobbe et al. 2021). To enhance the mathematical reasoning abilities of LLMs, some work focuses on continual pre-training (CPT) LLMs on large-scale mathematical corpora (Azerbayev et al. 2023; Jiang et al. 2023; Paster et al. 2024), which is believed to supplement LLMs' mathematical knowledge and improve reasoning abilities. Additionally, strategies using reasoning frameworks, such as CoT (Wei et al. 2022; Wu, Jiang, and Shen 2024), guide LLMs to decompose reasoning tasks into sub-steps through prompts. Although these strategies do not involve updating model parameters, they can effectively activate LLMs' reasoning abilities (Wang et al. 2023b). Additionally, recent works in this field have emerged based on large multimodal models, including benchmarks (Zhang et al. 2024b) and construction frameworks (Liu et al. 2024b). In this work, we focus on another strategy, namely SFT-based methods (Yu et al. 2024; Li et al. 2024a), which fine-tune LLMs by constructing query-response pairs related to downstream tasks.

SFT Data Construction The core of SFT-based methods is the collection of high-quality synthetic data. Recent research (Li et al. 2024b; Zeng et al. 2024) indicates that Data Scaling Laws (Kaplan et al. 2020) also apply to mathematical reasoning in LLMs, meaning that a model's mathematical reasoning abilities improve with increasing data size. To achieve performance comparable to GPT-4 in open-source LLMs, large-scale SFT datasets are often constructed by augmenting queries or responses of problems, leveraging powerful closed-source models like GPT-4 or using rejection sampling (Luo et al. 2023; Li et al. 2024a,b; Zeng et al. 2024; Liu et al. 2024a; Shao et al. 2024). However, constructing SFT data is typically time-consuming and costly, so the quality of SFT data is crucial. Higher quality data can mean fewer samples, translating to lower costs. Diversity is one indicator of data quality (Li et al. 2024c; Zhou et al. 2023; Zhang et al. 2024a), and SFT data with higher diversity can bring greater performance improvements to models. In this work, we propose a new SFT data construction approach that goes beyond simple augmentation of queries and responses and achieves higher diversity.

LLM-based Agents for Human Simulation The strong linguistic capabilities of LLMs endow LLM-based agents with excellent anthropomorphic abilities, allowing LLMs to simulate human behaviors with user prompts. LLM-based agents have been applied to simulations in various fields, including psychology (Yang et al. 2024), political science (Moghimifar et al. 2024), social interactions (Gürcan

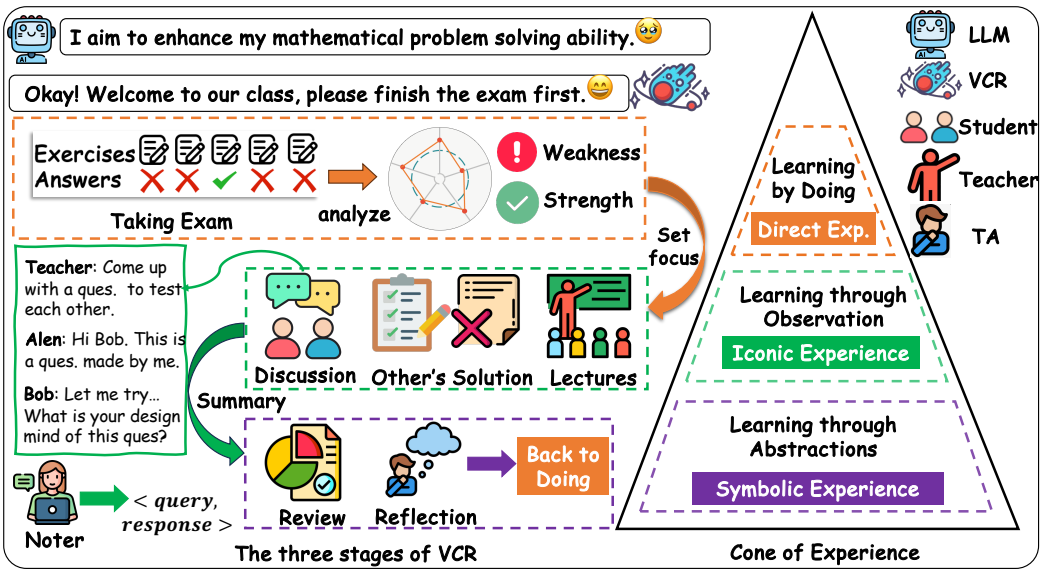

Figure 1: Overview of VCR. Left: The 3-stages generation process of VCR; Right: Cone of Experience proposed by Edgar Dale (Dale 1947). VCR includes 4 roles (Teacher, Student, Teaching Assistant(TA), and Class Noter), which together form three levels of experience through various teaching activities.

2024), and software development (Qian et al. 2024). Agents have also shown impressive anthropomorphic abilities in education. Existing work indicates that agents can act as teachers or students in educational settings, simulating teaching and discussion scenarios, and can formulate teaching plans (Hu et al. 2024). Additionally, LLM-based agents can effectively simulate traditional classroom interaction patterns (Zhang et al. 2024c; Yue et al. 2024). In this work, we use a multi-agent system to simulate the educational process and construct SFT training data by collecting data from these processes.

Methods

Problem Definition

(i) Input/Output of VCR: Given a public dataset $\mathcal{D} = \{(q_i, r_i)\}_{i=1}^{N}$, where N denotes the size of $\mathcal{D}$, q_i and r_i denote the query and response for the i^{th} question, respectively. VCR sequentially performs R expansions on $\mathcal{D}$. The r^{th} expansion results in $\mathcal{D}'_r = \{(q'_i, r'_i)\}_{i=1}^{N'}$, where N' is the user-specified size of the expanded dataset.

(ii) Supervised Fine-Tuning (SFT): For an LLM parameterized by θ, we perform SFT R times, with $\mathcal{D}'_r$ used for the r^{th} SFT. SFT is conducted by maximizing the log-likelihood of the response given the prompt query. The SFT loss is defined as

$$\mathcal{L}(\theta) = -\frac{1}{N'} \sum_{i=1}^{N'} \log \mathbb{P}(r'_i \mid q'_i; \theta) \tag{1}$$

The Proposed Framework (VCR)

Overview This section introduces the workflow of VCR, with an overview illustrated in Figure 1. VCR is a multi-agent environment which involves the following four roles:

Teacher: Responsible for planning the instructional process, providing problems, guiding the students' solution process, and supervising their learning progress; **TA** (Teaching Assistant): Assists in discussions, evaluates, and analyzes the students' responses to problems; **Student**: Obtains knowledge from the Teacher, and engages in learning and practice through participating in discussions and solving problems; **Noter**: Records dialogue data and feedback throughout the learning process into SFT data format.

We set up profiles for each role, and all teaching activities are completed through multi-agent collaboration. The design of VCR is based on the "Cone of Experience" educational theory, incorporating human multi-level learning experiences. It divides the learning process of LLMs into three categories: doing, observation, and abstraction. By engaging with experiences corresponding to these three processes, LLMs enhance their mathematical reasoning capabilities. Note that, just as human learning is a repetitive process, VCR is also iterative rather than linear. In the following, we will describe how to acquire experiences related to each of these three learning processes. Algorithm 1 details the specific workflow.

Stage 1: Learning by Doing Given to-be-trained LLM $\mathcal{M}$ and public dataset(e.g., GSM8K, MATH, etc.) $\mathcal{D} = \{(q_i, r_i)\}_{i=1}^{N}$, and expected synthetic dataset size N'. The process begins with a teacher agent instructing $\mathcal{M}$ to answer each question q_i from $\mathcal{D}$ for t times, allowing the teacher agent to compute each question's error rate k_i. Subsequently, VCR generates αk_i synthetic data for q_i, where

$$\alpha = \frac{N'}{\sum_{i=1}^{N} k_i} \tag{2}$$

The scaling factor α serves as a metric for assessing the

Algorithm 1: VCR for synthetic data construction

1: **Input:** Public training set $\mathcal{D} = \{(q_i, r_i)\}_{i=1}^{N}$, LLM $\mathcal{M}_0$, repeat times R and expected synthetic datasize N'.
2: **Output:** SFT training set $\mathcal{D}'$, final LLM $\mathcal{M}$.
3: $\mathcal{M} \leftarrow \mathcal{M}_0$, $\mathcal{D}' \leftarrow \emptyset$ ▷ Initialization
4: **for** $r \in \{1, 2 \ldots, R\}$ **do**
5: **for** $q_i \in \mathcal{D}$ **do** ▷ Learning by Doing
6: Have $\mathcal{M}$ respond to q_i with t times
7: Calculate the error rate k_i
8: **Teacher** analyze the weakness of $\mathcal{M}$
9: **end for**
10: $\alpha \leftarrow \frac{N'}{\sum_{i=1}^{N} k_i}$ ▷ Scaling factor for data size
11: Construct $\mathcal{D}'_r = \emptyset$ ▷ Training data for r^{th} SFT
12: **for** $q_i \in \mathcal{D}$ **do**
13: $\text{Syn}(q_i) = \emptyset$ ▷ Synthetic data for q_i
14: **while** $(|\text{Syn}(q_i)| < \alpha k_i)$ **do**
15: $\text{Augment}(r_i) = Lecture(\textbf{Teacher}, \textbf{Students}, q_i)$
16: $\text{Expand}(r_i) = Solution(\textbf{Students}, q_i)$
17: $\text{Augment}(q_i) = Discussion(\textbf{Students}, q_i)$
18: $\text{Mind}(q_i) = Discussion(\textbf{Students}, q_i)$
19: ▷ Learning through Observation
20: $\text{Key}(q_i) = Review(\textbf{TA}, q_i)$
21: $\text{Expand}(q_i) = Reflection(\textbf{TA}, q_i)$
22: ▷ Learning through Abstractions
23: **Noter** records all content to $\text{Syn}(q_i)$
24: **end while**
25: **end for**
26: Train $\mathcal{M}$ on $\mathcal{D}'_r$ for producing $\mathcal{M}_r$ ▷ SFT $\mathcal{M}$
27: Update $\mathcal{M} \leftarrow \mathcal{M}_r$, $\mathcal{D}' \leftarrow \mathcal{D}' \cup \{\mathcal{D}'_r\}$
28: **end for**
29: **Return:** SFT training set $\mathcal{D}'$, final model $\mathcal{M}$.

performance and stability of $\mathcal{M}$.

At this stage, VCR identifies challenging samples in $\mathcal{D}$ based on the error rate k_i of $\mathcal{M}$ on q_i, and sets the desired amount of synthetic data for each question. Simultaneously, VCR analyzes $\mathcal{M}$'s weakness from responses, which will serve as seeds for generating questions in the third stage. This stage corresponds to lines 5-9 in Algorithm 1.

Stage 2: Learning through Observation The observation experience corresponding to this stage is central to VCR. Before this stage begins, the teacher agent has already identified the challenging samples in $\mathcal{D}$ for $\mathcal{M}$. The primary task of this stage is to design teaching activities that address these weaknesses, with the activities being carried out by the agent. It is important to note that the LLM $\mathcal{M}$ does not participate in this stage; all learning activities are entirely simulated by the agents. Thus, the data from this stage can be considered as observation experience.

Each question q_i is handled individually, with all teaching activities planned by the teacher. Initially, teacher guides students through answering the questions via **lecture**, generating $\text{Augment}(r_i)$. This provides a more detailed solution process compared to CoT and can be considered as an augment of the answers. Notably, VCR predefines multiple students with different characteristics, meaning their **solutions** will offer diverse perspectives on the same question, which essentially serves as a method for enhancing diversity.

$$\begin{aligned} \text{Augment}(r_i) &= Lecture(\textbf{Teacher}, \textbf{Students}, q_i) \\ \text{Expand}(r_i) &= Solution(\textbf{Students}, q_i) \end{aligned} \tag{3}$$

Subsequently, the teacher plans the topics for **discussion** and asks the students to initiate the discussions. The goal of this stage is to adapt q_i to form $\text{Augment}(q_i)$ and to develop a question design mind for each question.

$$\begin{aligned} \text{Augment}(q_i) &= Discussion(\textbf{Students}, q_i) \\ \text{Mind}(q_i) &= Discussion(\textbf{Students}, q_i) \end{aligned} \tag{4}$$

Stage 3: Learning through Abstractions The purpose of this stage is to have the student agent reflect on the teaching activities of the second stage. VCR requires the TA agent to **review** key information from the questions. Then, based on the weakness of $\mathcal{M}$, new questions are designed and extended by **reflection**.

$$\begin{aligned} \text{Key}(q_i) &= Review(\textbf{TA}, q_i) \\ \text{Expand}(q_i) &= Reflection(\textbf{TA}, q_i) \end{aligned} \tag{5}$$

Finally, the Noter agent organizes all data collected from the second and third stages into the SFT data format, resulting in the synthetic dataset $\mathcal{D}'_r = \{(q'_i, r'_i)\}_{i=1}^{N'}$.

$$\begin{aligned} \mathcal{D}'_r = \textbf{Noter}(\{&\text{Augment}(r_i), \text{Mind}(q_i), \text{Key}(q_i), \\ &\text{Expand}(q_i), \text{Augment}(r_i), \text{Expand}(r_i)\}) \end{aligned} \tag{6}$$

Global Iterative Process Just as human learning is a repetitive process, LLM training also benefits from iterative improvements. In learning doing experiences, VCR adjusts its focus on each question in the training data based on performance feedback, similar to how the Adaboost algorithm iteratively reweights data to improve model accuracy. It is important to note that during this stage, VCR diagnoses the LLM with the core goal of adjusting the weights of the SFT training data, emphasizing a focus on difficult questions. However, this does not mean completely ignoring questions that the LLM already handles well. VCR still allocates a smaller amount of synthetic data to these simpler questions to prevent the LLM from forgetting them.

Specifically, for the LLM $\mathcal{M}$, VCR can repeat the data generation process R times. Before the r^{th} generation, the model undergoes the doing experience learning and allocates focus to training set samples. Then, based on observational and abstract experience learning, the dataset $\mathcal{D}^r_{sft}$ is created. The model $\mathcal{M}_r$ is trained on this dataset, and the final model obtained is $\mathcal{M}_R$. The SFT dataset is:

$$\mathcal{D}' = \bigcup_{1 \le r \le R} \{\mathcal{D}'_r\} \tag{7}$$

This global iterative process allows VCR to enhance the quality of synthetic data, progressively improving the model's performance over time.

Experiments

Datasets

We evaluated the mathematical reasoning abilities of LLMs using 5 benchmarks, including 4 publicly datasets and one modified dataset based on GSM8K, named GSM-Distractor.

- **GSM8K** (Cobbe et al. 2021) is a high-quality dataset of grade school math problems, consisting of 7,473 training samples and 1,319 testing samples.
- **MATH** (Hendrycks et al. 2021) consists of high school math competition problems, containing 7,500 training samples and 5,000 test samples.
- **SVAMP** (Patel, Bhattamishra, and Goyal 2021) includes elementary-level math problems with contextual descriptions. we use its 1,000 test samples for evaluation.
- **ASDiv** (Miao, Liang, and Su 2021) contains 2,305 math problems, covering a broad range of text patterns and question types.
- **GSM-Distractor** modifies the GSM8K test set, incorporating 1 to 3 distractors per question, following the setup of CMATH (Wei et al. 2023). The GSM-Distractor benchmark includes 3 times the original 1,319 problems.

Baselines

- **Evol-Instruct** (Luo et al. 2023) is a reinforcement learning-based method used to construct a dataset of 96K samples, resulting in the WizardMath model.
- **LEMA** (An et al. 2024) created 89K error reasoning paths from LLMs, along with correction data generated by GPT-4.
- **MetaMathQA** (Yu et al. 2024) expanded the training sets of GSM8K and MATH using strategies like FOBAR (Jiang et al. 2024) and Self-Verification (Wang et al. 2023a) with GPT-3.5, resulting in a total of 395K samples.
- **XwinMathQA** (Li et al. 2024a) used a three-step approach with the assistance of GPT-4-Turbo to extend GSM8K and MATH, totally generating 1.44M samples.
- **SkyworkMathQA** (Zeng et al. 2024) combined the strategies of MetamathQA, XwinMathQA, and Evol-Instruct but only extended the MATH dataset, resulting in 2.5M samples, including 0.4M hard questions.

Implementation Details

We use AutoGen (Wu et al. 2023) to simulate the entire VCR process. For SFT training, we follow the settings used in the baselines and employ the LLaMA-2 (Touvron et al. 2023) series models and the Mistral (Jiang et al. 2023) model, including LLaMA-2-7B/13B/70B and Mistral-7B, to facilitate a thorough comparison with baseline methods. Additionally, we include the state-of-the-art open-source model LLaMA-3-8B (Dubey et al. 2024) to evaluate VCR's performance improvements on more advanced models. Given computational resource constraints, we use QLoRA for fine-tuning LLaMA-2-70B and fully fine-tune all other models. The Adam optimizer is used with a weight decay of 3% linear warmup is applied with a warmup ratio of 3%. The learning rates are set as follows: 5e-5 for LLaMA-3-8B, 1e-4 for LLaMA-2-70B, and 2e-5 for other models. All SFTs are conducted for 3 epochs with a batch size of 128. In the QLoRA fine-tuning, LoRA rank and LoRA alpha are set to 96 and 16, respectively, with a dropout rate of 0.05. We employ DeepSpeed with ZeRO-2 stage to accelerate training. Model performance evaluation after SFT is carried out using the vLLM library (Kwon et al. 2023). All experiments are conducted on 8 NVIDIA A800 (80G) GPUs.

Results

Details about VCR-generated Data

In the experiment, we set the SFT dataset size $N' = 60,000$. For evaluating the training set, we require LLM to answer $t = 10$ times for each question. We use the following abbreviations: Aug-MATH/GSM8K refers to augmenting a single dataset, while Aug-Mix refers to augmenting both of them.

Main Results

Data quality of VCR Since baseline methods use one-time data training for 3 epochs, to ensure a fair comparison between VCR-generated data and other methods' data quality, we maintain the same settings and **do not** introduce the global iterative process of VCR in this section, i.e., setting the number of iterations $R = 1$. We will discuss the impact of including global iterative process in the following section.

Table 1 shows that VCR consistently outperforms baselines in enhancing model performance across five different base models. When using a fixed dataset size of 60K, VCR achieves performance improvement of 5.6%-10.3% over MetaMathQA in GSM8K, and 5.6%-7.8% in MATH. For XwinMathQA, VCR shows improvements of 2.3%-4.5% in GSM8K and 2.4%-5.2% in MATH. This advantage can be attributed to the higher quality density of VCR, as it focuses on challenging questions for the model. Notably, when compared with SkyworkMathQA, which only includes Aug-MATH and specifically for MATH, although VCR includes Aug-Mix, its performance on MATH is comparable.

Discussion under more challenge dataset VCR emphasizes identifying and focusing more on challenging samples within the training set. Given that MATH contains more high-difficulty problems compared to GSM8K, we only perform augmentation on the MATH training set, i.e., Aug-MATH, we also keep the dataset size fixed at 60K.

The results in Figure 2 show that, when using only 60K of Aug-MATH data, VCR exhibits the strongest performance compared to baseline methods. The results show that VCR surpasses the current SOTA method SkyworkMathQA by $+1.5\%$ and $+3.0\%$ on GSM8K and MATH using the base LLaMA-2-7B model, respectively, and by $+1.3\%$ and $+3.1\%$ using the base Mistral-7B model.

Furthermore, we analyzed the relationship between model performance on GSM8K and MATH and the scale of SFT data when using only Aug-MATH. Figure 3 shows a performance comparison between VCR and SkyworkMathQA at different data scales. The results indicate that, with smaller data sizes, VCR performs similarly to SkyworkMathQA;

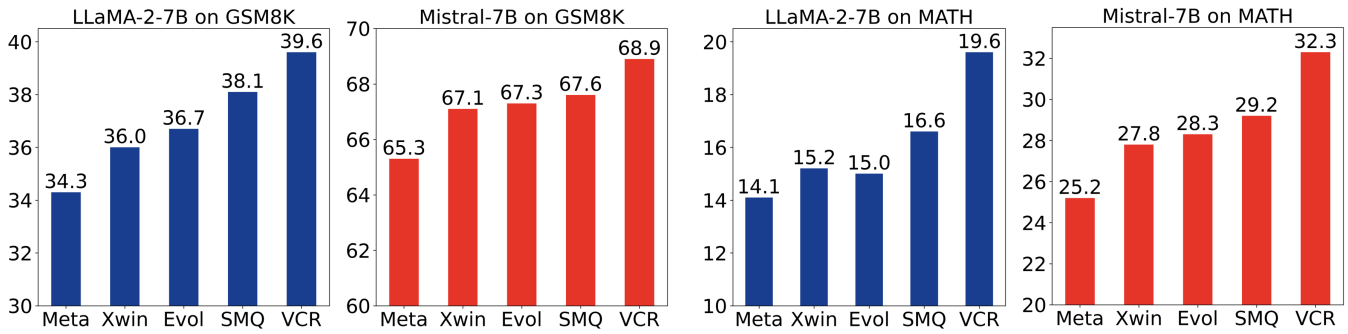

Figure 2: Model performance with various synthetic data generation methods, where SMQ denotes SkyworkMathQA.

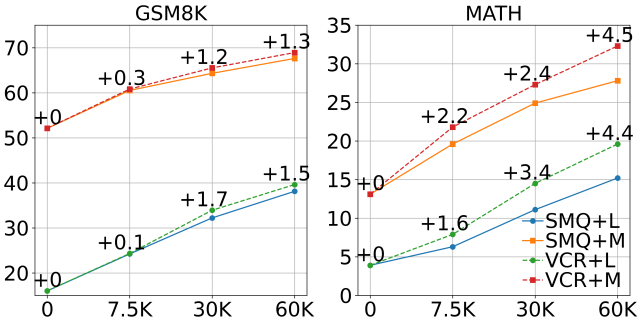

Figure 3: Model performance with varying training data sizes. Here, L and M refer to LLaMA-2-7B and Mistral-7B, respectively, while SMQ denotes SkyworkMathQA.

however, as the data size increases, VCR demonstrates a clear advantage.

These results are consistent with the Scaling Law (Kaplan et al. 2020) and recent studies (Zeng et al. 2024; Yuan et al. 2023), which reveal that increasing data size typically leads to significant improvements in model performance. Consequently, VCR shows greater potential with larger data scales, further validating its effectiveness and superiority.

OOD results We fine-tuned LLaMA-2-7B using 60K Aug-Mix data and evaluated its performance on SVAMP, ASDiv, and GSM-Distractor, which are three out-of-distribution datasets. Table 3 demonstrates the advantages of our method, indicating that the data constructed using the VCR method has higher quality.

Effect of Learning by Doing

In this stage, we assign weights to each question by evaluating the model's performance on the training set. This is based on a personalized approach, where, with a fixed amount of data, we focus on expanding those questions that are more challenging for the model. Table 2 shows the performance of LLaMA-2-7B without incorporating doing experience. In this case, we maintain a total data size of 60K but treat approximately 15K questions from the GSM8K/MATH training sets equally, expanding each question with about 4 synthetic data points using VCR. We observe a performance decline of 1.8% and 1.9% on GSM8K and MATH, respectively.

Although increasing the data scale can mitigate the performance loss due to the absence of the doing experience, it comes at a higher cost. Baselines use an equal treatment strategy, which leads to the generation of a large amount of synthetic data for simpler questions, where the LLM has already achieved high accuracy.

Effect of Learning through Observation

VCR-generated data exhibits greater richness and diversity due to the additional information introduced by its extensive educational activities. Previous research has demonstrated that diversity significantly enhances the mathematical reasoning abilities of LLMs (Zeng et al. 2024; Li et al. 2024a), a viewpoint also validated by the results in Table 2.

We do not rehash the effectiveness of common methods that involve rewriting queries or responses but instead focus on the role of the **problem design mind** introduced in our responses. The results in Table 2 show a performance decline when the problem design mind is removed. This is an intriguing phenomenon, indicating that problem design mind have a substantial impact. To our knowledge, we are the first to present this novel finding. The reason may be that the problem design mind stimulates the model's creativity and stability. Even 7B-level models already possess strong mathematical capabilities but may struggle with stability (Li et al. 2024a). Problem design mind might alleviate stability issues to some extent, contributing to the enhancement of LLMs' reasoning abilities.

Effect of Learning through Abstractions

The results in Table 2 show that removing the third stage leads to a loss in model performance, with a decrease of approximately 0.6% in accuracy for both GSM8K and MATH. To further investigate the role of this stage, we also introduced VCR w/o abstraction in the OOD experiments. Table 3 lists its performance. We found that while there was little change in accuracy on SVAMP and ASDiv, VCR with the complete process exhibited greater robustness. As the number of distractors increased in the GSM-Distractor dataset, VCR's performance degradation was significantly slower compared to VCR w/o abstraction.

Method	DataSize	GSM8K	MATH
LLaMA-2-7B (Touvron et al. 2023)	NA	14.6	2.5
+MuggleMathQA† (Li et al. 2024b)	82K	43.2	14.4
+Evol-Instruct (Luo et al. 2023)	96K	54.9	10.7
+LEMA (An et al. 2024)	89K	54.1	9.4
+MetaMathQA (Yu et al. 2024)	60K	52.3	8.9
+XwinMathQA (Li et al. 2024a)	60K	58.1	12.8
+SkyworkMathQA† (Zeng et al. 2024)	60K	38.1	**15.2**
+VCR	60K	**62.6**	**15.2**
Mistral-7B (Jiang et al. 2023)	NA	52.2	13.1
+MetaMathQA (Yu et al. 2024)	60K	64.5	19.3
+XwinMathQA (Li et al. 2024a)	60K	67.9	22.8
+SkyworkMathQA† (Zeng et al. 2024)	60K	67.6	**27.8**
+VCR	60K	**70.5**	27.1
LLaMA-3-8B (Dubey et al. 2024)	NA	59.9	21.4
+MetaMathQA (Yu et al. 2024)	60K	65.4	24.2
+XwinMathQA (Li et al. 2024a)	60K	68.9	26.8
+VCR	60K	**71.1**	**30.1**
LLaMA-2-13B (Touvron et al. 2023)	NA	28.7	3.9
+Evol-Instruct (Luo et al. 2023)	96K	63.9	14.0
+LEMA (An et al. 2024)	89K	65.7	12.6
+MetaMathQA (Yu et al. 2024)	60K	61.3	11.6
+XwinMathQA (Li et al. 2024a)	60K	64.9	13.9
+VCR	60K	**68.0**	**19.1**
LLaMA-2-70B (Touvron et al. 2023)	NA	56.8	13.5
+Evol-Instruct (Luo et al. 2023)	96K	81.6	22.7
+LEMA (An et al. 2024)	89K	83.5	25.0
+MetaMathQA (Yu et al. 2024)	60K	79.9	22.4
+XwinMathQA (Li et al. 2024a)	60K	82.1	23.2
+VCR	60K	**85.5**	**28.0**

Table 1: The performances of LLMs on GSM8K and MATH with approximately 60K SFT data after training 3 epochs. **Bold** highlights the optimal performance with the data size limit. Results obtained from training primarily on Aug-Mix, with some exceptions indicated by †, which represent results trained on Aug-MATH.

Method	Experience Type			GSM8K	MATH
	Doing	Observation	Abstraction		
SFT	✗	✗	✗	41.6	4.7
VCR	✗	✓	✓	60.8	13.3
	✓	✗	✓	61.5	13.9
	✓	✓	✗	62.0	14.6
	✓	✓	✓	**62.6**	**15.2**

Table 2: Ablation studies with different experience types in VCR. Here we use LLaMA-2-7B as the backbone model and finetune it with 60K VCR-generated Aug-Mix data.

Effect of Global Iterative Process

Previous analysis has demonstrated that the data generated by VCR has higher quality, but it only considered a single generation($R = 1$). Therefore, in this section, we consider the global iterative process with ($R = 3$). For VCR$^{R=1}$, we fine-tuned LLaMA-2-7B with 3 epochs in a single run, We saved the checkpoints after each epoch and evaluated them. For VCR$^{R=3}$, we fine-tune 3 times, each with only 1 epoch. After each fine-tuning, we save the checkpoints and evaluate performance on the test set, while also examining the model's performance on the training set in the Doing Stage. This allows us to adjust the composition of the SFT data for the next epoch, with each epoch fixed at 60K training data.

Method	SVAMP	ASDiv	GSM-Dis.		
			1	2	3
Evol-Instruct	57.3	59.1	46.6	40.1	33.4
LEMA	54.1	65.5	45.4	38.9	32.4
MetaMathQA	58.2	63.6	43.9	37.6	32.4
XwinMathQA	64.1	67.4	48.5	39.8	33.6
VCR w/o Abstraction	65.2	67.1	50.0	42.2	34.8
VCR	**66.5**	**68.9**	**53.2**	**45.8**	**38.9**

Table 3: OOD performance of LLaMA-2-7B after SFT on different Aud-Mix data. Here, GSM-Dis. refers to the GSM-Distractors dataset, with numbers 1, 2, and 3 indicating the number of distractors.

Epoch	VCR$^{R=1}$		VCR$^{R=3}$	
	GSM8K	MATH	GSM8K	MATH
1	60.0	13.6	60.0 (+0)	13.6 (+0)
2	60.9	14.3	62.2 (+1.3)	15.7 (+1.4)
3	62.6	15.2	66.9 (+4.3)	17.8 (+2.6)

Table 4: The performance of LLaMA-2-7B over 3 epochs. VCR$^{R=1}$: training 3 epochs in a single run, data remains consistent across epochs; VCR$^{R=3}$: Add global iterative process, training 3 times with one epoch each time, the SFT data differs in each training.

Results in Table 4 show that VCR$^{R=3}$ achieved gradual performance improvement across the 3 epochs, showing better results compared to directly training 3 epochs with 60K VCR data ($+4.3\%$ in GSM8K, $+2.6\%$ in MATH). This suggests that adjusting the SFT data composition based on model performance on the training set allows the model to focus on more challenging problems, further validating the effectiveness of the Doing stage. It is worth noting that although the data used across the 3 epochs is not completely different, there is a significant overlap, with a total of approximately 71K unique samples after de-duplication.

Conclusion

In this paper, we propose VCR, a virtual learning environment simulated using LLM-powered agents. VCR enhances public mathematical datasets with targeted improvements. Through extensive experiments, we demonstrate that, with the same scale of data, the data generated by VCR can more effectively improve the mathematical reasoning performance of LLMs compared to previous methods. Unlike earlier research that focuses on redesigning prompts to augment queries or answers, VCR is based on the "Cone of Experience" educational theory. It parallels LLM learning with human learning processes, collecting the multi-level experience required for LLM learning through a simulated learning environment, including lecture, discussion, problem design, and problem-solving. We also make a new discovery, indirect data, such as question design mind, which positively impacts model performance. This not only supports the feasibility of analogizing LLM learning to human learning but also provides a new perspective for future research.

Acknowledgments

This work was financially supported by the National Key R&D Program of China (2023YFC3305704), National Natural Science Foundation of China (62437002, 62307015, 62293554), Hubei Provincial Natural Science Foundation of China (2023AFA020, 2023AFB295), China Postdoctoral Science Foundation (2023M741304), and Fundamental Research Funds for the Central Universities (CCNU24AI016).

References

Ahn, J.; Verma, R.; Lou, R.; Liu, D.; Zhang, R.; and Yin, W. 2024. Large Language Models for Mathematical Reasoning: Progresses and Challenges. arXiv:2402.00157.

An, S.; Ma, Z.; Lin, Z.; Zheng, N.; Lou, J.-G.; and Chen, W. 2024. Learning From Mistakes Makes LLM Better Reasoner. arXiv:2310.20689.

Azerbayev, Z.; Schoelkopf, H.; Paster, K.; Santos, M. D.; McAleer, S.; Jiang, A.; Deng, J.; Biderman, S.; and Welleck, S. 2023. Llemma: An Open Language Model For Mathematics. In *The 3rd Workshop on Mathematical Reasoning and AI at NeurIPS'23*.

Cao, Y.; Kang, Y.; Wang, C.; and Sun, L. 2024. Instruction Mining: Instruction Data Selection for Tuning Large Language Models. arXiv:2307.06290.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168.

Dale, E. 1947. Audio-visual materials. *Air Aff.*, 2: 179.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

Gunasekar, S.; Zhang, Y.; Aneja, J.; Mendes, C. C. T.; Giorno, A. D.; Gopi, S.; Javaheripi, M.; Kauffmann, P.; de Rosa, G.; Saarikivi, O.; Salim, A.; Shah, S.; Behl, H. S.; Wang, X.; Bubeck, S.; Eldan, R.; Kalai, A. T.; Lee, Y. T.; and Li, Y. 2023. Textbooks Are All You Need. arXiv:2306.11644.

Gürcan, Ö. 2024. LLM-Augmented Agent-Based Modelling for Social Simulations: Challenges and Opportunities. *HHAI 2024: Hybrid Human AI Systems for the Social Good*, 134–144.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hu, B.; Zheng, L.; Zhu, J.; Ding, L.; Wang, Y.; and Gu, X. 2024. Teaching Plan Generation and Evaluation With GPT-4: Unleashing the Potential of LLM in Instructional Design. *IEEE Transactions on Learning Technologies*, 17: 1471–1485.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; Lavaud, L. R.; Lachaux, M.-A.; Stock, P.; Scao, T. L.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2023. Mistral 7B. arXiv:2310.06825.

Jiang, W.; Shi, H.; Yu, L.; Liu, Z.; Zhang, Y.; Li, Z.; and Kwok, J. 2024. Forward-Backward Reasoning in Large Language Models for Mathematical Verification. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 6647–6661. Bangkok, Thailand: Association for Computational Linguistics.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling Laws for Neural Language Models. arXiv:2001.08361.

Kwon, W.; Li, Z.; Zhuang, S.; Sheng, Y.; Zheng, L.; Yu, C. H.; Gonzalez, J.; Zhang, H.; and Stoica, I. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, 611–626.

Li, C.; Wang, W.; Hu, J.; Wei, Y.; Zheng, N.; Hu, H.; Zhang, Z.; and Peng, H. 2024a. Common 7B Language Models Already Possess Strong Math Capabilities. arXiv:2403.04706.

Li, C.; Yuan, Z.; Yuan, H.; Dong, G.; Lu, K.; Wu, J.; Tan, C.; Wang, X.; and Zhou, C. 2024b. MuggleMath: Assessing the Impact of Query and Response Augmentation on Math Reasoning. arXiv:2310.05506.

Li, M.; Zhang, Y.; Li, Z.; Chen, J.; Chen, L.; Cheng, N.; Wang, J.; Zhou, T.; and Xiao, J. 2024c. From Quantity to Quality: Boosting LLM Performance with Self-Guided Data Selection for Instruction Tuning. arXiv:2308.12032.

Liu, H.; Zhang, Y.; Luo, Y.; and Yao, A. C.-C. 2024a. Augmenting Math Word Problems via Iterative Question Composing. arXiv:2401.09003.

Liu, S.; Feng, J.; Yang, Z.; Luo, Y.; Wan, Q.; Shen, X.; and Sun, J. 2024b. COMET: "cone of experience" enhanced large multimodal model for mathematical problem generation. *Science China Information Sciences*, 67(12): 1–2.

Luo, H.; Sun, Q.; Xu, C.; Zhao, P.; Lou, J.; Tao, C.; Geng, X.; Lin, Q.; Chen, S.; and Zhang, D. 2023. WizardMath: Empowering Mathematical Reasoning for Large Language Models via Reinforced Evol-Instruct. arXiv:2308.09583.

Luo, Y.; and Yang, Y. 2024. Large language model and domain-specific model collaboration for smart education. *Frontiers of Information Technology & Electronic Engineering*, 25(3): 333–341.

Miao, S.-Y.; Liang, C.-C.; and Su, K.-Y. 2021. A Diverse Corpus for Evaluating and Developing English Math Word Problem Solvers. arXiv:2106.15772.

Minaee, S.; Mikolov, T.; Nikzad, N.; Chenaghlu, M.; Socher, R.; Amatriain, X.; and Gao, J. 2024. Large Language Models: A Survey. arXiv:2402.06196.

Moghimifar, F.; Li, Y.-F.; Thomson, R.; and Haffari, G. 2024. Modelling Political Coalition Negotiations Using LLM-based Agents. arXiv:2402.11712.

OpenAI. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray,

A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. arXiv:2203.02155.

Paster, K.; Santos, M. D.; Azerbayev, Z.; and Ba, J. 2024. OpenWebMath: An Open Dataset of High-Quality Mathematical Web Text. In *The Twelfth International Conference on Learning Representations*.

Patel, A.; Bhattamishra, S.; and Goyal, N. 2021. Are NLP Models really able to Solve Simple Math Word Problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2080–2094.

Qian, C.; Liu, W.; Liu, H.; Chen, N.; Dang, Y.; Li, J.; Yang, C.; Chen, W.; Su, Y.; Cong, X.; Xu, J.; Li, D.; Liu, Z.; and Sun, M. 2024. ChatDev: Communicative Agents for Software Development. arXiv:2307.07924.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y. K.; Wu, Y.; and Guo, D. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Wang, H.; Xin, H.; Zheng, C.; Li, L.; Liu, Z.; Cao, Q.; Huang, Y.; Xiong, J.; Shi, H.; Xie, E.; Yin, J.; Li, Z.; Liao, H.; and Liang, X. 2023a. LEGO-Prover: Neural Theorem Proving with Growing Libraries. arXiv:2310.00656.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023b. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 24824–24837.

Wei, T.; Luan, J.; Liu, W.; Dong, S.; and Wang, B. 2023. CMATH: Can Your Language Model Pass Chinese Elementary School Math Test? arXiv:2306.16636.

Wu, Q.; Bansal, G.; Zhang, J.; Wu, Y.; Li, B.; Zhu, E.; Jiang, L.; Zhang, X.; Zhang, S.; Liu, J.; Awadallah, A. H.; White, R. W.; Burger, D.; and Wang, C. 2023. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. arXiv:2308.08155.

Wu, Z.; Jiang, M.; and Shen, C. 2024. Get an A in Math: Progressive Rectification Prompting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 19288–19296. AAAI Press.

Yang, Q.; Wang, Z.; Chen, H.; Wang, S.; Pu, Y.; Gao, X.; Huang, W.; Song, S.; and Huang, G. 2024. PsychoGAT: A Novel Psychological Measurement Paradigm through Interactive Fiction Games with LLM Agents. arXiv:2402.12326.

Ying, C.; Qi-Guang, M.; Jia-Chen, L.; and Lin, G. 2013. Advance and prospects of AdaBoost algorithm. *Acta Automatica Sinica*, 39(6): 745–758.

Yu, L.; Jiang, W.; Shi, H.; Jincheng, Y.; Liu, Z.; Zhang, Y.; Kwok, J.; Li, Z.; Weller, A.; and Liu, W. 2024. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Yuan, Z.; Yuan, H.; Li, C.; Dong, G.; Lu, K.; Tan, C.; Zhou, C.; and Zhou, J. 2023. Scaling Relationship on Learning Mathematical Reasoning with Large Language Models. arXiv:2308.01825.

Yue, M.; Mifdal, W.; Zhang, Y.; Suh, J.; and Yao, Z. 2024. MathVC: An LLM-Simulated Multi-Character Virtual Classroom for Mathematics Education. arXiv:2404.06711.

Zeng, L.; Zhong, L.; Zhao, L.; Wei, T.; Yang, L.; He, J.; Cheng, C.; Hu, R.; Liu, Y.; Yan, S.; Fang, H.; and Zhou, Y. 2024. Skywork-Math: Data Scaling Laws for Mathematical Reasoning in Large Language Models – The Story Goes On. arXiv:2407.08348.

Zha, D.; Bhat, Z. P.; Lai, K.-H.; Yang, F.; Jiang, Z.; Zhong, S.; and Hu, X. 2023. Data-centric Artificial Intelligence: A Survey. arXiv:2303.10158.

Zhang, B.; Liu, Z.; Cherry, C.; and Firat, O. 2024a. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method. In *The Twelfth International Conference on Learning Representations*.

Zhang, R.; Jiang, D.; Zhang, Y.; Lin, H.; Guo, Z.; Qiu, P.; Zhou, A.; Lu, P.; Chang, K.-W.; Gao, P.; and Li, H. 2024b. MathVerse: Does Your Multi-modal LLM Truly See the Diagrams in Visual Math Problems? arXiv:2403.14624.

Zhang, Z.; Zhang-Li, D.; Yu, J.; Gong, L.; Zhou, J.; Liu, Z.; Hou, L.; and Li, J. 2024c. Simulating Classroom Education with LLM-Empowered Agents. arXiv:2406.19226.

Zhou, C.; Liu, P.; Xu, P.; Iyer, S.; Sun, J.; Mao, Y.; Ma, X.; Efrat, A.; Yu, P.; Yu, L.; Zhang, S.; Ghosh, G.; Lewis, M.; Zettlemoyer, L.; and Levy, O. 2023. LIMA: Less Is More for Alignment. arXiv:2305.11206.