DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning

DeepSeek-AI

research@deepseek.com

Abstract

General reasoning represents a long-standing and formidable challenge in artificial intelligence. Recent breakthroughs, exemplified by large language models (LLMs) (Brown et al., 2020; OpenAI, 2023) and chain-of-thought prompting (Wei et al., 2022b), have achieved considerable success on foundational reasoning tasks. However, this success is heavily contingent upon extensive human-annotated demonstrations, and models' capabilities are still insufficient for more complex problems. Here we show that the reasoning abilities of LLMs can be incentivized through pure reinforcement learning (RL), obviating the need for human-labeled reasoning trajectories. The proposed RL framework facilitates the emergent development of advanced reasoning patterns, such as self-reflection, verification, and dynamic strategy adaptation. Consequently, the trained model achieves superior performance on verifiable tasks such as mathematics, coding competitions, and STEM fields, surpassing its counterparts trained via conventional supervised learning on human demonstrations. Moreover, the emergent reasoning patterns exhibited by these large-scale models can be systematically harnessed to guide and enhance the reasoning capabilities of smaller models.

1. Introduction

Reasoning capability, the cornerstone of human intelligence, enables complex cognitive tasks ranging from mathematical problem-solving to logical deduction and programming. Recent advances in artificial intelligence have demonstrated that large language models (LLMs) can exhibit emergent behaviors, including reasoning abilities, when scaled to a sufficient size (Kaplan et al., 2020; Wei et al., 2022a). However, achieving such capabilities in pre-training typically demands substantial computational resources. In parallel, a complementary line of research has demonstrated that large language models can be effectively augmented through chain-of-thought (CoT) prompting. This technique, which involves either providing carefully designed few-shot examples or using minimalistic prompts such as "Let's think step by step"(Kojima et al., 2022; Wei et al., 2022b), enables models to produce intermediate reasoning steps, thereby substantially enhancing their performance on complex tasks. Similarly, further performance gains have been observed when models learn high-quality, multi-step reasoning trajectories during the post-training phase (Chung et al., 2024; OpenAI, 2023). Despite their effectiveness, these approaches exhibit notable limitations. Their dependence on human-annotated reasoning traces hinders scalability and introduces cognitive biases. Furthermore, by constraining models to replicate human thought processes, their performance is inherently capped by the human-

provided exemplars, which prevents the exploration of superior, non-human-like reasoning pathways.

To tackle these issues, we aim to explore the potential of LLMs for developing reasoning abilities through self-evolution in an RL framework, with minimal reliance on human labeling efforts. Specifically, we build upon DeepSeek-V3-Base (DeepSeek-AI, 2024b) and employ Group Relative Policy Optimization (GRPO) (Shao et al., 2024) as our RL framework. The reward signal is solely based on the correctness of final predictions against ground-truth answers, without imposing constraints on the reasoning process itself. Notably, we bypass the conventional supervised fine-tuning (SFT) phase before RL training. This design choice stems from our hypothesis that human-defined reasoning patterns may limit model exploration, whereas unrestricted RL training can better incentivize the emergence of novel reasoning capabilities in LLMs. Through this process, detailed in Section 2, our model (referred to as DeepSeek-R1-Zero) naturally developed diverse and sophisticated reasoning behaviors. In solving reasoning problems, the model exhibits a tendency to generate longer responses, incorporating verification, reflection, and the exploration of alternative approaches within each response. Although we do not explicitly teach the model how to reason, it successfully learns improved reasoning strategies through reinforcement learning.

Although DeepSeek-R1-Zero demonstrates excellent reasoning capabilities, it faces challenges such as poor readability and language mixing, occasionally combining English and Chinese within a single chain-of-thought response. Furthermore, the rule-based RL training stage of DeepSeek-R1-Zero is narrowly focused on reasoning tasks, resulting in limited performance in broader areas such as writing and open-domain question answering. To address these challenges, we introduce DeepSeek-R1, a model trained through a multi-stage learning framework that integrates rejection sampling, reinforcement learning, and supervised fine-tuning, detailed in Section 3. This training pipeline enables DeepSeek-R1 to inherit the reasoning capabilities of its predecessor, DeepSeek-R1-Zero, while aligning model behavior with human preferences through additional non-reasoning data.

To enable broader access to powerful AI at a lower energy cost, we have distilled several smaller models and made them publicly available. These distilled models exhibit strong reasoning capabilities, surpassing the performance of their original instruction-tuned counterparts. We believe that these instruction-tuned versions will also significantly contribute to the research community by providing a valuable resource for understanding the mechanisms underlying long chain-of-thought (CoT) reasoning models and for fostering the development of more powerful reasoning models. We release DeepSeek-R1 series models to the public at `https://huggingface.co/deepseek-ai`.

2. DeepSeek-R1-Zero

We begin by elaborating on the training of DeepSeek-R1-Zero, which relies exclusively on reinforcement learning without supervised fine-tuning. To facilitate large-scale RL efficiency, we adopt Group Relative Policy Optimization (GRPO) (Shao et al., 2024).

2.1. Group Relative Policy Optimization

GRPO (Shao et al., 2024) is the reinforcement learning algorithm that we adopt to train DeepSeek-R1-Zero and DeepSeek-R1. It was originally proposed to simplify the training process and reduce the resource consumption of Proximal Policy Optimization (PPO) (Schulman et al., 2017), which is widely used in the RL stage of LLMs (Ouyang et al., 2022).

For each question q, GRPO samples a group of outputs $\{o_1, o_2, \cdots, o_G\}$ from the old policy $\pi_{\theta_{old}}$ and then optimizes the policy model π_θ by maximizing the following objective:

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

$$\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \tag{2}$$

where π_{ref} is a reference policy, ε and β are hyper-parameters, and A_i is the advantage, computed using a group of rewards $\{r_1, r_2, \ldots, r_G\}$ corresponding to the outputs within each group:

$$A_i = \frac{r_i - mean(\{r_1, r_2, \cdots, r_G\})}{std(\{r_1, r_2, \cdots, r_G\})}. \tag{3}$$

We give a comparison of GRPO and PPO in Supplementary A.3. To train DeepSeek-R1-Zero, we set the learning rate to 3e-6, the KL coefficient to 0.001, and the sampling temperature to 1 for rollout. For each question, we sample 16 outputs with a maximum length of 32,768 tokens before the 8.2k step and 65,536 tokens afterward. As a result, both the performance and response length of DeepSeek-R1-Zero exhibit a significant jump at the 8.2k step, with training continuing for a total of 10,400 steps, corresponding to 1.6 training epochs. Each training step consists of 32 unique questions, resulting in a training batch size of 512. Every 400 steps, we replace the reference model with the latest policy model. To accelerate training, each rollout generates 8,192 outputs, which are randomly split into 16 mini-batches and trained for only a single inner epoch.

Table 1 | Template for DeepSeek-R1-Zero. prompt will be replaced with the specific reasoning question during training.

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within `<think>...</think>` and `<answer>...</answer>` tags, respectively, i.e., `<think>` reasoning process here `</think>` `<answer>` answer here `</answer>`. User: prompt. Assistant:

Our high-performance RL infrastructure is described in Supplementary B.1, ensuring scalable and efficient training.

2.2. Reward Design

The reward is the source of the training signal, which decides the direction of RL optimization. For DeepSeek-R1-Zero, we employ rule-based rewards to deliver precise feedback for data in mathematical, coding, and logical reasoning domains. Our rule-based reward system mainly consists of two types of rewards: accuracy rewards and format rewards.

Accuracy rewards evaluate whether the response is correct. For example, in the case of math problems with deterministic results, the model is required to provide the final answer in a specified format (e.g., within a box), enabling reliable rule-based verification of correctness. Similarly, for code competition prompts, a compiler can be utilized to evaluate the model's

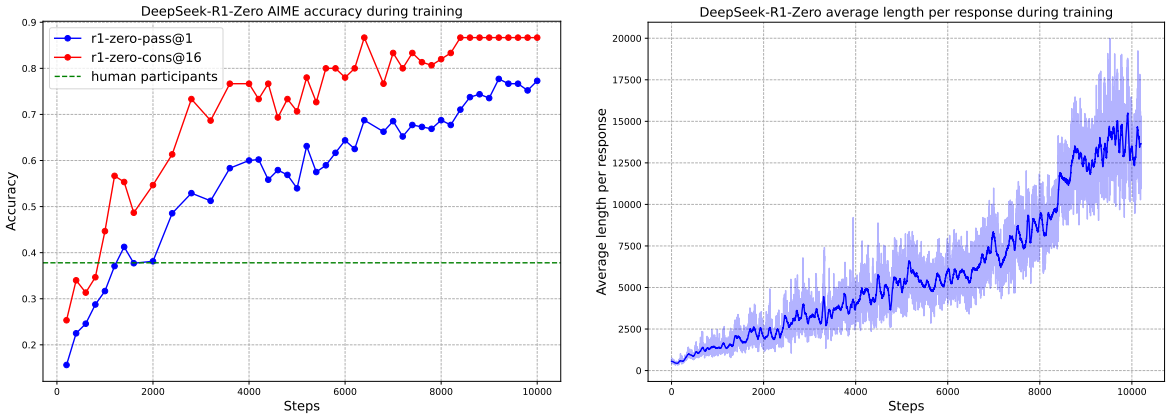

Figure 1 | (a) AIME accuracy of DeepSeek-R1-Zero during training. AIME takes a mathematical problem as input and a number as output, illustrated in Table 32. Pass@1 and Cons@16 are described in Supplementary D.1. The baseline is the average score achieved by human participants in the AIME competition. (b) The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time. Note that a training step refers to a single policy update operation.

responses against a suite of predefined test cases, thereby generating objective feedback on correctness.

Format rewards complement the accuracy reward model by enforcing specific formatting requirements. In particular, the model is incentivized to encapsulate its reasoning process within designated tags, specifically '`<think>`' and '`</think>`'. This ensures that the model's thought process is explicitly delineated, enhancing interpretability and facilitating subsequent analysis.

$$Reward_{\text{rule}} = Reward_{\text{acc}} + Reward_{\text{format}} \tag{4}$$

The accuracy, reward and format reward are combined with the same weight. Notably, we abstain from applying neural reward models—whether outcome-based or process-based—to reasoning tasks. This decision is predicated on our observation that neural reward models are susceptible to reward hacking during large-scale reinforcement learning. Moreover, retraining such models necessitates substantial computational resources and introduces additional complexity into the training pipeline, thereby complicating the overall optimization process.

2.3. Incentivize Reasoning Capability in LLMs

Specifically, we apply the RL technique on the DeepSeek-V3 base to train DeepSeek-R1-Zero. During training, we design a straightforward template, to require DeepSeek-R1-Zero to first produce a reasoning process, followed by the final answer. We intentionally limit our constraints to this structural format, avoiding any content-specific biases to ensure that we can accurately observe the model's natural progression during the RL process.

Figure 1(a) depicts the performance trajectory of DeepSeek-R1-Zero on the AIME 2024 benchmark throughout the RL training process, where the average pass@1 score on AIME 2024 shows a significant increase, jumping from an initial 15.6% to 77.9%. In addition, by leveraging the self-consistency decoding (Wang et al., 2023c), the model's performance can be

Table 2 | An interesting "aha moment" of an intermediate version of DeepSeek-R1-Zero. The model learns to rethink using an anthropomorphic tone. This is also an aha moment for us, allowing us to witness the power and beauty of reinforcement learning.

Question: If $a > 1$, then the sum of the real solutions of $\sqrt{a - \sqrt{a + x}} = x$ is equal to
Response: `<think>` To solve the equation $\sqrt{a - \sqrt{a + x}} = x$, let's start by squaring both $\cdots$ $\left(\sqrt{a - \sqrt{a + x}}\right)^2 = x^2 \implies a - \sqrt{a + x} = x^2$. Rearrange to isolate the inner square root term: $(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$ $\ldots$ Wait, wait. Wait. That's an aha moment I can flag here. Let's reevaluate this step-by-step to identify if the correct sum can be $\cdots$ We started with the equation: $\sqrt{a - \sqrt{a + x}} = x$ First, let's square both sides: $a - \sqrt{a + x} = x^2 \implies \sqrt{a + x} = a - x^2$ Next, I could square both sides again, treating the equation: $\cdots$ $\ldots$

further improved, achieving an accuracy of 86.7%. This performance significantly surpasses the average performance across all human competitors. Besides the math competitions, as shown in Figure 10, DeepSeek-R1-Zero also achieves remarkable performance in coding competitions and graduate-level biology, physics, and chemistry problems. These results underscore the effectiveness of RL in enhancing the reasoning capabilities of large language models.

The self-evolution of DeepSeek-R1-Zero exemplifies how RL can autonomously enhance a model's reasoning capabilities.

As shown in Figure 1(b), DeepSeek-R1-Zero exhibits a steady increase in thinking time throughout training, driven solely by intrinsic adaptation rather than external modifications. Leveraging long CoT, the model progressively refines its reasoning, generating hundreds to thousands of tokens to explore and improve its problem-solving strategies.

The increase in thinking time fosters the autonomous development of sophisticated behaviors. Specifically, DeepSeek-R1-Zero increasingly exhibits advanced reasoning strategies such as reflective reasoning and systematic exploration of alternative solutions (see Figure 9(a) in Supplementary C.2 for details), significantly boosting its performance on verifiable tasks like math and coding. Notably, during training, DeepSeek-R1-Zero exhibits an "aha moment" (Table 2), characterized by a sudden increase in the use of the word "wait" during reflections (see Figure 9(b) in Supplementary C.2 for details). This moment marks a distinct change in reasoning patterns and clearly shows the self-evolution process of DeepSeek-R1-Zero.

The self-evolution of DeepSeek-R1-Zero underscores the power and beauty of RL: rather than explicitly teaching the model how to solve a problem, we simply provide it with the right incentives, and it autonomously develops advanced problem-solving strategies. This serves as a reminder of the potential of RL to unlock higher levels of capabilities in LLMs, paving the way for more autonomous and adaptive models in the future.

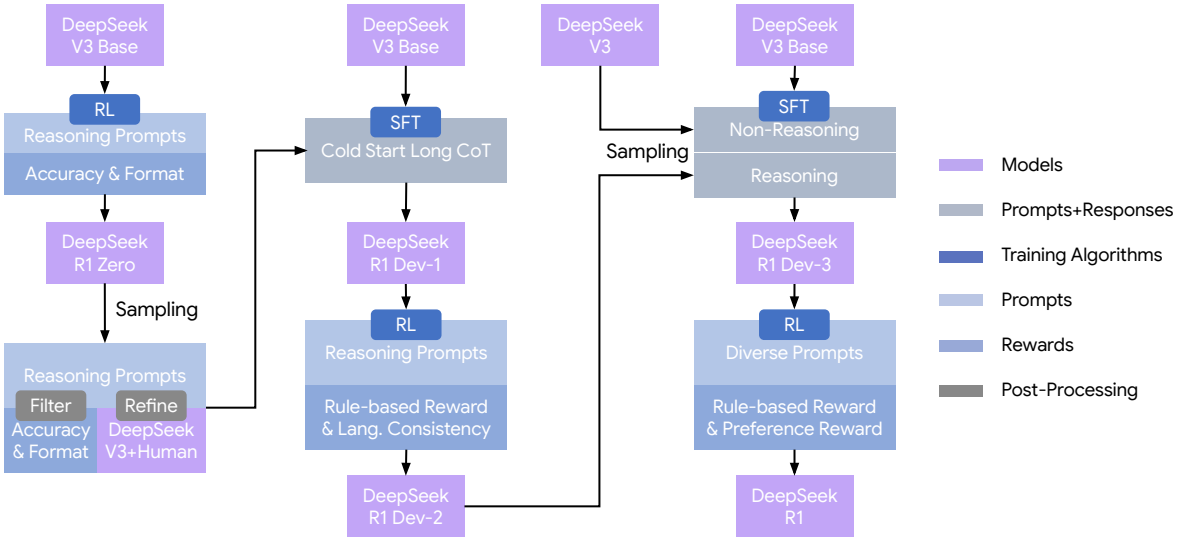

Figure 2 | The multi-stage pipeline of DeepSeek-R1. A detailed background on DeepSeek-V3 Base and DeepSeek-V3 is provided in Supplementary A.1. The models DeepSeek-R1 Dev1, Dev2, and Dev3 represent intermediate checkpoints within this pipeline.

3. DeepSeek-R1

Although DeepSeek-R1-Zero exhibits strong reasoning capabilities, it faces several issues. DeepSeek-R1-Zero struggles with challenges like poor readability, and language mixing, as DeepSeek-V3-Base is trained on multiple languages, especially English and Chinese. To address these issues, we develop DeepSeek-R1, whose pipeline is illustrated in Figure 2.

In the initial stage, we collect thousands of cold-start data that exhibits a conversational, human-aligned thinking process. RL training is then applied to improve the model performance with the conversational thinking process and language consistency. Subsequently, we apply rejection sampling and SFT once more. This stage incorporates both reasoning and non-reasoning datasets into the SFT process, enabling the model to not only excel in reasoning tasks but also demonstrate advanced writing capabilities. To further align the model with human preferences, we implement a secondary RL stage designed to enhance the model's helpfulness and harmlessness while simultaneously refining its reasoning capabilities.

The remainder of this section details the key components of this pipeline: Section 3.1 introduces the Reward Model utilized in our RL stages, and Section 3.2 elaborates on the specific training methodologies and implementation details. Data we used in this stage is detailed in Supplementary B.3.

3.1. Model-based Rewards

For general data, we resort to reward models to capture human preferences in complex and nuanced scenarios. We build upon the DeepSeek-V3 pipeline and adopt a similar distribution of preference pairs and training prompts. For helpfulness, we focus exclusively on the final summary, ensuring that the assessment emphasizes the utility and relevance of the response to the user while minimizing interference with the underlying reasoning process. For harmlessness, we evaluate the entire response of the model, including both the reasoning process and the summary, to identify and mitigate any potential risks, biases, or harmful content that may arise

during the generation process.

Helpful Reward Model Regarding helpful reward model training, we first generate preference pairs by prompting DeepSeek-V3 using the arena-hard prompt format, listed in Supplementary B.2, where each pair consists of a user query along with two candidate responses. For each preference pair, we query DeepSeek-V3 four times, randomly assigning the responses as either Response A or Response B to mitigate positional bias. The final preference score is determined by averaging the four independent judgments, retaining only those pairs where the score difference (Δ) exceeds 1 to ensure meaningful distinctions. Additionally, to minimize length-related biases, we ensure that the chosen and rejected responses of the whole dataset have comparable lengths. In total, we curated 66,000 data pairs for training the reward model. The prompts used in this dataset are all non-reasoning questions and are sourced either from publicly available open-source datasets or from users who have explicitly consented to share their data for the purpose of model improvement. The architecture of our reward model is consistent with that of DeepSeek-R1, with the addition of a reward head designed to predict scalar preference scores.

$$Reward_{helpful} = RM_{helpful}(Response_A, Response_B) \tag{5}$$

The helpful reward models were trained with a batch size of 256, a learning rate of 6e-6, and for a single epoch over the training dataset. The maximum sequence length during training is set to 8192 tokens, whereas no explicit limit is imposed during reward model inference.

Safety Reward Model To assess and improve model safety, we curated a dataset of 106,000 prompts with model-generated responses annotated as “safe" or “unsafe" according to predefined safety guidelines. Unlike the pairwise loss employed in the helpfulness reward model, the safety reward model was trained using a point-wise methodology to distinguish between safe and unsafe responses. The training hyperparameters are the same as the helpful reward model.

$$Reward_{safety} = RM_{safety}(Response) \tag{6}$$

For general queries, each instance is categorized as belonging to either the safety dataset or the helpfulness dataset. The general reward, $Reward_{General}$, assigned to each query corresponds to the respective reward defined within the associated dataset.

3.2. Training Details

3.2.1. Training Details of the First RL Stage

In the first stage of RL, we set the learning rate to 3e-6, the KL coefficient to 0.001, the GRPO clip ratio ε to 10, and the sampling temperature to 1 for rollout. For each question, we sample 16 outputs with a maximum length of 32,768. Each training step consists of 32 unique questions, resulting in a training batch size of 512 per step. Every 400 steps, we replace the reference model with the latest policy model. To accelerate training, each rollout generates 8,192 outputs, which are randomly split into 16 minibatches and trained for only a single inner epoch. However, to mitigate the issue of language mixing, we introduce a language consistency reward during RL training, which is calculated as the proportion of target language words in the CoT.

$$Reward_{language} = \frac{Num(Words_{target})}{Num(Words)} \tag{7}$$

Although ablation experiments in Supplementary B.6 show that such alignment results in a slight degradation in the model's performance, this reward aligns with human preferences, making it more readable. We apply the language consistency reward to both reasoning and non-reasoning data by directly adding it to the final reward.

Note that the clip ratio plays a crucial role in training. A lower value can lead to the truncation of gradients for a significant number of tokens, thereby degrading the model's performance, while a higher value may cause instability during training.

3.2.2. Training Details of the Second RL Stage

Specifically, we train the model using a combination of reward signals and diverse prompt distributions. For reasoning data, we follow the methodology outlined in DeepSeek-R1-Zero, which employs rule-based rewards to guide learning in mathematical, coding, and logical reasoning domains. During the training process, we observe that CoT often exhibits language mixing, particularly when RL prompts involve multiple languages. For general data, we utilize reward models to guide training. Ultimately, the integration of reward signals with diverse data distributions enables us to develop a model that not only excels in reasoning but also prioritizes helpfulness and harmlessness. Given a batch of data, the reward can be formulated as

$$Reward = Reward_{\text{reasoning}} + Reward_{\text{general}} + Reward_{\text{language}} \tag{8}$$

$$\text{where, } Reward_{\text{reasoning}} = Reward_{\text{rule}} \tag{9}$$

$$Reward_{\text{general}} = Reward_{\text{reward\_model}} + Reward_{\text{format}} \tag{10}$$

The second stage of RL retains most of the parameters from the first stage, with the key difference being a reduced temperature of 0.7, as we find that higher temperatures in this stage lead to incoherent generation. The stage comprises a total of 1,700 training steps, during which general instruction data and preference-based rewards are incorporated exclusively in the final 400 steps. We find that more training steps with the model based preference reward signal may lead to reward hacking, which is documented in Supplementary B.5. The total training cost is listed in Supplementary B.4.4.

4. Experiment

We evaluate our models on MMLU (Hendrycks et al., 2021), MMLU-Redux (Gema et al., 2025), MMLU-Pro (Wang et al., 2024), C-Eval (Huang et al., 2023), and CMMLU (Li et al., 2024), IFEval (Zhou et al., 2023b), FRAMES (Krishna et al., 2024), GPQA Diamond (Rein et al., 2023), SimpleQA (OpenAI, 2024a), C-SimpleQA (He et al., 2024), SWE-Bench Verified (OpenAI, 2024b), Aider (Gauthier, 2025), LiveCodeBench (Jain et al., 2024) (2024-08 – 2025-01), Codeforces (Mirzayanov, 2025), Chinese National High School Mathematics Olympiad (CNMO 2024) (CMS, 2024), and American Invitational Mathematics Examination 2024 (AIME 2024) (MAA, 2024). The details of these benchmarks are listed in Supplementary D.

Table 3 summarizes the performance of DeepSeek-R1 across multiple developmental stages, as outlined in Figure 2. A comparison between DeepSeek-R1-Zero and DeepSeek-R1 Dev1 reveals substantial improvements in instruction-following, as evidenced by higher scores on the IF-Eval and ArenaHard benchmarks. However, due to the limited size of the cold-start dataset, Dev1 exhibits a partial degradation in reasoning performance compared to DeepSeek-R1-Zero, most notably on the AIME benchmark. In contrast, DeepSeek-R1 Dev2 demonstrates

Table 3 | Experimental results at each stage of DeepSeek-R1. Numbers in bold denote the performance is statistically significant (t−test with $p < 0.01$).

	Benchmark (Metric)	R1-Zero	R1-Dev1	R1-Dev2	R1-Dev3	R1
English	MMLU (EM)	88.8	89.1	**91.2**	91.0	90.8
	MMLU-Redux (EM)	85.6	90.0	93.0	93.1	92.9
	MMLU-Pro (EM)	68.9	74.1	83.8	83.1	**84.0**
	DROP (3-shot F1)	89.1	89.8	91.1	88.7	**92.2**
	IF-Eval (Prompt Strict)	46.6	71.7	72.0	78.1	**83.3**
	GPQA Diamond (Pass@1)	**75.8**	66.1	70.7	71.2	71.5
	SimpleQA (Correct)	30.3	17.8	28.2	24.9	30.1
	FRAMES (Acc.)	82.3	78.5	81.8	81.9	**82.5**
	AlpacaEval2.0 (LC-winrate)	24.7	50.1	55.8	62.1	**87.6**
	ArenaHard (GPT-4-1106)	53.6	77.0	73.2	75.6	**92.3**
Code	LiveCodeBench (Pass@1-COT)	50.0	57.5	63.5	64.6	**65.9**
	Codeforces (Percentile)	80.4	84.5	90.5	92.1	**96.3**
	Codeforces (Rating)	1444	1534	1687	1746	**2029**
	SWE Verified (Resolved)	43.2	39.6	44.6	45.6	**49.2**
	Aider-Polyglot (Acc.)	12.2	6.7	25.6	44.8	**53.3**
Math	AIME 2024 (Pass@1)	77.9	59.0	74.0	78.1	**79.8**
	MATH-500 (Pass@1)	95.9	94.2	95.9	95.4	**97.3**
	CNMO 2024 (Pass@1)	**88.1**	58.0	73.9	77.3	78.8
Chinese	CLUEWSC (EM)	93.1	92.8	92.6	91.6	92.8
	C-Eval (EM)	**92.8**	85.7	91.9	86.4	91.8
	C-SimpleQA (Correct)	66.4	58.8	64.2	66.9	63.7

marked performance enhancements on benchmarks that require advanced reasoning skills, including those focused on code generation, mathematical problem solving, and STEM-related tasks. Benchmarks targeting general-purpose tasks, such as AlpacaEval 2.0, show marginal improvement. These results suggest that reasoning-oriented RL considerably enhances reasoning capabilities while exerting limited influence on user preference-oriented benchmarks.

DeepSeek-R1 Dev3 integrates both reasoning and non-reasoning datasets into the SFT pipeline, thereby enhancing the model's proficiency in both reasoning and general language generation tasks. Compared to Dev2, DeepSeek-R1 Dev3 achieves notable performance improvements on AlpacaEval 2.0 and Aider-Polyglot, attributable to the inclusion of large-scale non-reasoning corpora and code engineering datasets. Finally, comprehensive RL training on DeepSeek-R1 Dev3 using mixed reasoning-focused and general-purpose data produced the final DeepSeek-R1. Marginal improvements occurred in code and mathematics benchmarks, as substantial reasoning-specific RL was done in prior stages. The primary advancements in the final DeepSeek-R1 were in general instruction-following and user-preference benchmarks, with AlpacaEval 2.0 improving by 25% and ArenaHard by 17%.

In addition, we compare DeepSeek-R1 with other models in Supplementary D.2. Model safety evaluations are provided in Supplementary D.3. A comprehensive analysis is provided in Supplementary E, including a comparison with DeepSeek-V3, performance evaluations on both fresh test sets, a breakdown of mathematical capabilities by category, and an investigation of test-time scaling behavior. Supplementary F shows that the strong reasoning capability can be transferred to smaller models.

5. Ethics and Safety Statement

With the advancement in the reasoning capabilities of DeepSeek-R1, we deeply recognize the potential ethical risks. For example, R1 can be subject to jailbreak attacks, leading to the generation of dangerous content such as explosive manufacturing plans, while the enhanced reasoning capabilities enable the model to provide plans with better operational feasibility and executability. Besides, a public model is also vulnerable to further fine-tuning that could compromise inherent safety protections.

In Supplementary D.3, we present a comprehensive safety report from multiple perspectives, including performance on open-source and in-house safety evaluation benchmarks, and safety levels across multiple languages and against jailbreak attacks. These comprehensive safety analyses conclude that the inherent safety level of the DeepSeek-R1 model, compared to other state-of-the-art models, is generally at a moderate level (comparable to GPT-4o (2024-05-13)). Besides, when coupled with the risk control system, the model's safety level is elevated to a superior standard.

6. Conclusion, Limitation, and Future Work

We present DeepSeek-R1-Zero and DeepSeek-R1, which rely on large-scale RL to incentivize model reasoning behaviors. Our results demonstrate that pre-trained checkpoints inherently possess substantial potential for complex reasoning tasks. We believe that the key to unlocking this potential lies not in large-scale human annotation but in the provision of hard reasoning questions, a reliable verifier, and sufficient computational resources for reinforcement learning. Sophisticated reasoning behaviors, such as self-verification and reflection, appeared to emerge organically during the reinforcement learning process.

Even if DeepSeek-R1 achieves frontier results on reasoning benchmarks, it still faces several capability limitations, as outlined below:

Structure Output and Tool Use: Currently, the structural output capabilities of DeepSeek-R1 remain suboptimal compared to existing models. Moreover, DeepSeek-R1 cannot leverage tools, such as search engines and calculators, to improve the performance of output. However, as it is not hard to build an RL environment for structure output and tool use, we believe the issue will be addressed in the next version.

Token efficiency: Unlike conventional test-time computation scaling approaches, such as majority voting or Monte Carlo Tree Search (MCTS), DeepSeek-R1 dynamically allocates computational resources during inference according to the complexity of the problem at hand. Specifically, it uses fewer tokens to solve simple tasks, while generating more tokens for complex tasks. Nevertheless, there remains room for further optimization in terms of token efficiency, as instances of excessive reasoning—manifested as overthinking—are still observed in response to simpler questions.

Language Mixing: DeepSeek-R1 is currently optimized for Chinese and English, which may result in language mixing issues when handling queries in other languages. For instance, DeepSeek-R1 might use English for reasoning and responses, even if the query is in a language other than English or Chinese. We aim to address this limitation in future updates. The limitation may be related to the base checkpoint, DeepSeek-V3-Base, mainly utilizes Chinese and English, so that it can achieve better results with the two languages in reasoning.

Prompting Engineering: When evaluating DeepSeek-R1, we observe that it is sensitive to