DeepSeek-R1-Zero and DeepSeek-R1: A Detailed Technical Report

1. Introduction

DeepSeek-R1-Zero was a pioneering model built on top of the DeepSeek-V3 architecture, primarily to explore whether a model could acquire advanced reasoning skills with only reinforcement learning (RL) and no initial supervised fine-tuning (SFT). By employing a novel training algorithm known as **Group Relative Policy Optimization (GRPO)**, DeepSeek-R1-Zero demonstrated surprisingly strong reasoning capabilities. However, it exhibited certain drawbacks—such as readability issues and language mixing—that eventually led to a more refined approach in DeepSeek-R1.

This report dives into the following:

- How DeepSeek-R1-Zero was built using the DeepSeek-V3 base model.
- Detailed explanation of **Group Relative Policy Optimization (GRPO)**.
- The RL process and reward system that enabled DeepSeek-R1-Zero's reasoning abilities.
- The limitations of DeepSeek-R1-Zero.
- How the training strategy evolved to produce DeepSeek-R1, which combined RL with supervised data to achieve a better balance of reasoning and user-friendliness.

2. DeepSeek-R1-Zero: Built Upon DeepSeek-V3

DeepSeek-V3 served as the foundational large language model for this project. Before the development of R1-Zero, DeepSeek-V3 already had strong language modeling capabilities, particularly in:

- Natural Language Processing (NLP) tasks.
- Some baseline reasoning tasks like reading comprehension and straightforward question-answering.

DeepSeek-R1-Zero started from DeepSeek-V3 but immediately pivoted to a reinforcement learning training regime without performing a dedicated supervised fine-tuning stage on a large curated dataset. The key focus was to see whether RL alone could elicit the emergence of sophisticated "chain-of-thought" reasoning.

3. Group Relative Policy Optimization (GRPO)

3.1 Motivation

Typical policy gradient methods (like PPO) compare new policy outputs against a reference (old) policy using an advantage function. GRPO is a variant designed to:

1. **Leverage a group-based baseline** instead of a single baseline or critic network.
2. **Balance exploration and stability** by sampling multiple outputs for each query from the old policy, then updating the new policy to increase rewards for the best responses while penalizing unwanted behaviors.

3.2 Mathematical Formulation of GRPO

Let:

- q be a query (or input prompt) sampled from some query distribution $P(Q)$.
- $\pi_{old}(o \mid q)$ be the probability of generating output o given q under the old policy.
- $\pi_{\theta}(o \mid q)$ be the probability of generating output o given q under the new policy (parameterized by θ).
- G be the number of outputs sampled (the "group").
- A_i be the advantage associated with the i-th output in the group.

> **GRPO objective:**
>
> $$J_{GRPO}(\theta) = E_{q \sim P(Q),\, o \sim \pi_{old}} \left[\frac{1}{G} \sum_{i=1 \text{ to } G} \min\left(\frac{\pi_{\theta}(o_i \mid q)}{\pi_{old}(o_i \mid q)} * A_i,\ \text{clip}\left(\frac{\pi_{\theta}(o_i \mid q)}{\pi_{old}(o_i \mid q)}, 1-\varepsilon, 1+\varepsilon \right) * A_i \right) - \beta\, D_{KL}\left(\pi_{\theta}(\cdot \mid q) \,||\, \pi_{old}(\cdot \mid q) \right) \right]$$

Breaking this down:

- **Sampling in Groups**: For each query q, GRPO samples G different outputs $\{o_1, o_2, \ldots, o_G\}$ from the **old policy** π_{old}. This ensures the new policy sees a range of possible responses.
- **Ratio $\pi_{\theta}(o_i|q)$ / $\pi_{old}(o_i|q)$**: Measures how much the new policy deviates from the old policy for the specific output o_i.
- **Advantage A_i**: A scalar indicating whether o_i is better or worse than the baseline expectation. High advantage means the response was particularly good (high reward).
- **Clipped Objective** (min(…, …)): Borrowed from PPO, the new policy is prevented from deviating too far from the old policy in a single update (via the *clip* function and parameter ε).

- **KL Penalty** (βD_{KL}): GRPO adds a term $\beta D_{KL}[\pi_\theta \,||\, \pi_{old}]$ to discourage large divergences from the old policy distribution, stabilizing training and preventing catastrophic updates.

Group Baseline vs. Single Critic
Unlike a single value-function-based critic, GRPO uses the group of outputs themselves as a reference for baseline calculations. The advantage is evaluated per group, effectively comparing each sampled output within the context of the group, which can reduce variance in advantage estimates and encourage better overall policy updates.

4. Training DeepSeek-R1-Zero with GRPO

4.1 Self-Evolution with RL

- **No Supervised Fine-Tuning**: DeepSeek-R1-Zero skipped the typical supervised data stage. Instead, it started RL training from the DeepSeek-V3 base.
- **Reward Design**: The model received rewards based on:
  - **Accuracy** (for tasks with a clear correct/incorrect answer).
  - **Format Enforcement**: The model was incentivized to place reasoning in special tags like `<think>` and final answers in `<answer>`.

These two reward signals nudged the model to produce more structured "chain-of-thought" outputs and verify correctness before finalizing an answer.

4.2 Emergence of Complex Reasoning

Remarkably, R1-Zero developed:

- **Self-Verification**: Checking or critiquing its own chain of thought.
- **Reflection**: Revisiting an earlier step of reasoning and correcting mistakes.

This led to *large performance gains* on reasoning-heavy benchmarks. For instance, pass@1 on the AIME 2024 dataset jumped from **15.6%** to **71.0%**.

4.3 Limitations of R1-Zero

- **Readability**: Outputs were not always user-friendly—chain-of-thought text could be rambling, repetitive, or cluttered.
- **Language Mixing**: Sometimes interjected multiple languages in a single response, reducing coherence.

These issues hinted that RL alone wasn't sufficient for polished, human-friendly interaction.

5. Transition to DeepSeek-R1

DeepSeek-R1 aimed to retain the advanced reasoning skills of R1-Zero but address its limitations through a more **hybrid strategy**:

1. **Cold-Start Stage**
    - A small, high-quality dataset was created (some from few-shot prompts and some from refined/human-annotated outputs).
    - This dataset was carefully designed to emphasize **readability** and consistent formatting.
    - The model was fine-tuned on this set before large-scale RL.
2. **Reasoning-Oriented RL**
    - The initial fine-tuned checkpoint went through RL again, focusing on tough domains (math, coding, logic).
    - Rewards were added to promote **language consistency** and avoid the mixing problem.
3. **Rejection Sampling + Supervised Fine-Tuning**
    - **Reasoning Data**: About 600k high-quality samples (chains of thought) were obtained by sampling from the RL model and picking the best outputs.
    - **General Data**: 200k tasks spanning writing, Q&A, etc., to maintain broad capabilities.
    - A supervised fine-tuning pass combined the reasoning data and general data, leading to a more balanced model.
4. **RL for All Scenarios**
    - Another round of RL refined helpfulness, harmlessness, and advanced reasoning performance—ensuring both safety and accuracy.

By merging RL-based reasoning with carefully supervised data, **DeepSeek-R1** overcame the readability and coherence shortcomings while maintaining or even improving upon the reasoning skill.

6. Conclusion

DeepSeek-R1-Zero demonstrated the viability of using reinforcement learning alone (specifically GRPO) to spur advanced reasoning behaviors, though it suffered from inconsistent output formats and occasional language mixing.

DeepSeek-R1 built on R1-Zero's success, introducing curated supervised data and employing multi-stage RL. This resulted in a more *user-friendly*, coherent, and still strongly capable model.

Overall, the progression from R1-Zero to R1 highlights how carefully blending RL with a controlled supervised pipeline can unlock state-of-the-art reasoning while providing reliable, readable outputs.