NOVER: Incentive Training for Language Models via Verifier-Free Reinforcement Learning

Wei Liu[★], **Siya Qi**[★], **Xinyu Wang**[★], **Chen Qian**[♣], **Yali Du**[★♠], **Yulan He**[★♠✉],

[★]King's College London, [♠]The Alan Turing Institute, [♣]Shanghai Jiao Tong University
{wei.4.liu, yali.du, yulan.he}@kcl.ac.uk

https://huggingface.co/spaces/thinkwee/NOVER

Abstract

Recent advances, such as DeepSeek R1-Zero, highlight the effectiveness of incentive training, a reinforcement learning paradigm that computes rewards solely based on the final answer part of a language model's output, thereby encouraging the generation of intermediate reasoning steps. However, these methods fundamentally rely on external verifiers, which limits their applicability to domains like mathematics and coding, where such verifiers are readily available. Although reward models can serve as verifiers, they require high-quality annotated data and are costly to train. In this work, we propose **NOVER**, NO-VERifier Reinforcement Learning, a general reinforcement learning framework that requires only standard supervised fine-tuning data with no need for an external verifier. NOVER enables incentive training across a wide range of text-to-text tasks and outperforms the model of the same size distilled from large reasoning models such as DeepSeek R1 671B by 7.7%. Moreover, the flexibility of NOVER enables new possibilities for optimizing large language models, such as inverse incentive training.

1 Introduction

Recent progress in Large Language Model (LLM) reasoning has been accelerated by *incentive training* (Guo et al., 2025; Xie et al., 2025; Yu et al., 2025; Zeng et al., 2025; Liu et al., 2025a; Yuan et al., 2025b; Hu et al., 2025), a new Reinforcement Learning (RL) paradigm which optimizes models by computing rewards only on the final answer part in model response, and incentivize models to generate intermediate tokens like reasoning steps spontaneously. Notably, methods such as DeepSeek R1-Zero (Guo et al., 2025) have shown that in domains like mathematics and coding, using a simple rule-based verifier to calculate reward and perform RL-only incentive training can achieve impressive performance, known as Reinforcement Learning with Verifiable Reward (RLVR). It has been demonstrated that such incentivized reasoning ability can generalize to areas like multi-modality (Shen et al., 2025; Zhang et al., 2025; Yang et al., 2025; Feng et al., 2025b) or language agent (Xia and Luo, 2025; Jin et al., 2025; Song et al., 2025; Wang et al., 2025; Feng et al., 2025a).

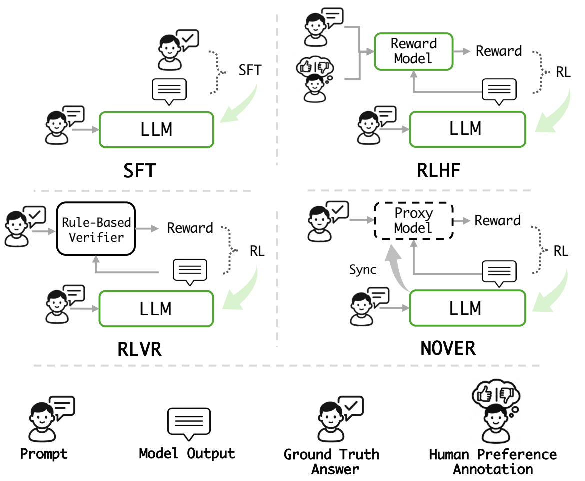

Figure 1: Comparison of NOVER with other post-training paradigms.

However, the success of incentive training with RLVR hinges on external verifiers that can judge the correctness of model outputs. For many highly contextualized and hard-to-grade tasks requiring similar reasoning or cognitive capability, ranging from social behavior analysis to creative writing, such verifiers are infeasible or hard to construct (Weng, 2025). Recent works have explored training general-purpose large verifier models across diverse domains (Ma et al., 2025; Seed et al., 2025; Su et al., 2025). However, building accurate verifier models involves a complex pipeline, which is significantly more costly than traditional reward models for preference alignment (Liu et al., 2025b). Moreover, once trained, these verifier models demand substantial computational resources to be deployed during the subsequent RL training.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 7451–7470
November 4-9, 2025 ©2025 Association for Computational Linguistics

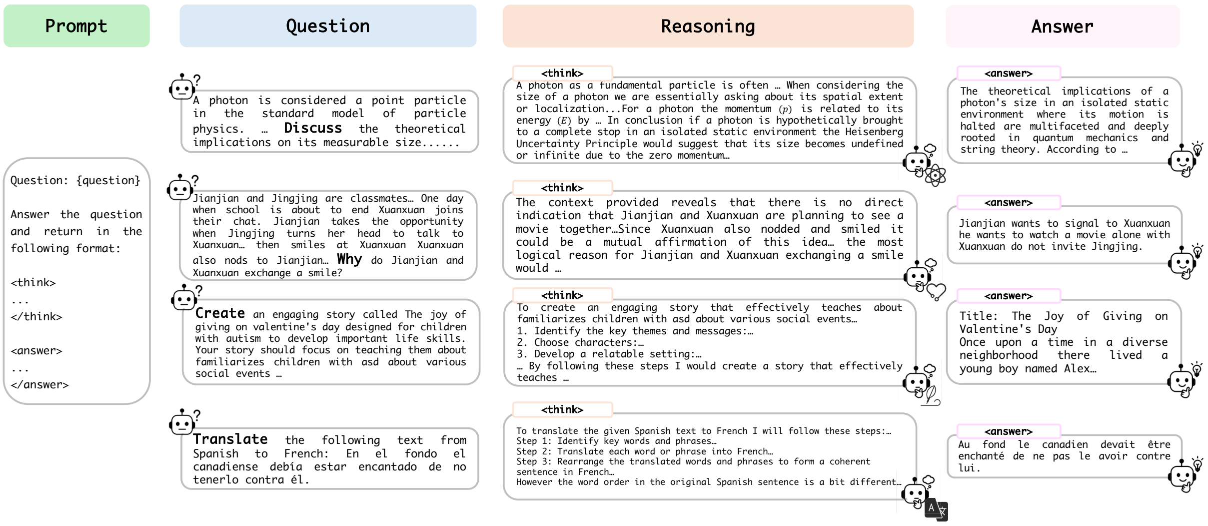

Figure 2: Examples of Qwen2.5-7B-NOVER on a range of text-to-text tasks, demonstrating its ability to handle open-ended questions such as "Discuss", "Why", or "Create". These tasks often require free-form answers that are difficult to verify with clear-cut labels, posing challenges for incentive training.

To address these limitations of RLVR, we propose **NOVER**, NO-VERifier Reinforcement Learning, a novel framework for incentive training without an external verifier. As shown in Figure 1, compared with traditional Supervised Fine-Tuning (SFT), NOVER can perform incentive training similar to RLVR; compared to Reinforcement Learning from Human Feedback (RLHF), NOVER does not need a trained reward model for verification; and compared with RLVR, NOVER extends the incentive training to areas where a reliable verifier is hard to acquire. NOVER requires only standard SFT data and utilizes the model itself to build a reward proxy for lightweight RL training. By calculating perplexity-based reward based on the model's reasoning process, it enables incentive-driven reinforcement learning across a wide range of text-to-text tasks, as shown in Figure 2. Our main contributions are as follows:

1. We introduce **NOVER**, a verifier-free incentive training framework that generalizes RLVR to arbitrary text-to-text tasks with minimal SFT data.

2. Our experiments and analysis demonstrate NOVER's superior performance. It is stable to train compared with model-as-verifier methods and mitigates reward hacking, shapes reasoning patterns, and adapts to diverse tasks.

3. NOVER's flexibility enables imaginative approaches such as inverse incentive training. This paradigm teaches a model how to fish rather than simply giving it a fish, surpassing standard incentive training on tasks that need creativity.

2 Related Work

Language Model Reasoning Early research designed prompting techniques and workflows based on human cognitive priors to enhance the reasoning capabilities of LLMs. Chain-of-Thought (CoT) prompting (Wei et al., 2022) enabled step-by-step reasoning, later extended by Zero-Shot CoT (Kojima et al., 2022) and Self-Consistency (Wang et al., 2023b). More recent methods, such as Tree of Thoughts (Yao et al., 2023a), Least-to-Most (Zhou et al., 2023), Plan-and-Solve (Wang et al., 2023a), Sketch-Navigation (Liu et al., 2024), and Multi-Perspective Self-Reflection (Yan et al., 2024), introduced structured exploration over reasoning trajectories. ReAct (Yao et al., 2023b) combined reasoning with external tool use to enable more interactive problem-solving. Beyond prompting, verification-based approaches were proposed (Cobbe et al., 2021) to solve math reasoning. Neuro-symbolic methods (Pan et al., 2023) fuse LLMs with symbolic solvers. Previous works also utilize Process Reward Models (Lightman et al., 2024) combined with Monte Carlo Tree Search (Kocsis and Szepesvári, 2006) for step-level exploration to replicate OpenAI's o1 (Jaech et al., 2024).

Incentive Training In contrast to the above methods, DeepSeek-R1 (Guo et al., 2025) proposed a simpler paradigm called incentive training. Building on this idea, recent works such as Logic-RL (Xie et al., 2025), DAPO (Yu et al., 2025), SimpleRL (Zeng et al., 2025), OpenReasoner (Hu et al., 2025), Dr.GRPO (Liu et al., 2025a), and VAPO (Yuan et al., 2025b) investigated best practices on aspects such as exploration-exploitation, emergence of "aha" moments, and task difficulty variance. Several recent efforts attempted to extend incentive training beyond math and coding (He et al., 2025; Lu et al., 2025; Gurung and Lapata, 2025; Su et al., 2025; Ma et al., 2025). However, these approaches often rely on domain-specific rules for verification or involve training large verifier models, which can be computationally expensive.

3 Background

Rule-based Reward Given a training prompt p with a template (Guo et al., 2025) asking to generate intermediate tokens t (e.g., reasoning steps) followed by final answer a, incentive training aims to use RL to optimize a model π_θ with outcome reward R_{rule}, which is solely computed by a rule-based verifier $v : \mathcal{A} \to \{0, 1\}$ on a:

$$R_{\text{rule}} = v(a) \tag{1}$$

Such a sparse but accurate outcome reward encourages the model to autonomously generate intermediate tokens t that could lead to a better a. In reasoning tasks, for example, t may be reasoning behaviors like task decomposition or self-reflection. To ensure the verifier can easily parse outputs, DeepSeek-R1 introduces a tag format reward R_{f}, which requires t to be enclosed in `<think>` tags and a in `<answer>` tags:

$$R_{\text{f}} = f_{\text{format}}(t, a) \tag{2}$$

where $f_{\text{format}} : \mathcal{T} \times \mathcal{A} \to \{0, 1\}$ is the regular expression based format check function. Then the combined reward is:

$$R(p, t, a) = w_{\text{rule}} \cdot R_{\text{rule}} + w_{\text{f}} \cdot R_{\text{f}} \tag{3}$$

GRPO After the calculation of reward, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) is then used to optimize π_θ. For each prompt p, the model rolls out a group of completions $\mathcal{C} = \{(t_1, a_1), ..., (t_G, a_G)\}$. The group-normalized advantage is:

$$A_i = \frac{R(p, t_i, a_i) - \mu_{\mathcal{C}}}{\sigma_{\mathcal{C}}} \tag{4}$$

where $\mu_{\mathcal{C}}$ and $\sigma_{\mathcal{C}}$ are the mean and standard deviation of rewards in the group. Then the policy is updated with a clipped objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\{p, t_i, a_i\}} \left(\frac{1}{G} \sum_{i=1}^{G} r_i^{clip} - \beta D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}) \right) \tag{5}$$

$$r_i^{clip} = \min\left(r_i, \text{clip}(r_i, 1-\epsilon, 1+\epsilon)\right) A_i \tag{6}$$

$$r_i = \frac{\pi_\theta(t_i, a_i \mid p)}{\pi_{\theta_{\text{old}}}(t_i, a_i \mid p)} \tag{7}$$

where ϵ is the clip ratio and β is the KL penalty weight. r is the policy ratio (Schulman et al., 2017). This framework enables LLMs to develop explicit, high-quality reasoning processes without dense supervision for intermediate steps.

4 Method

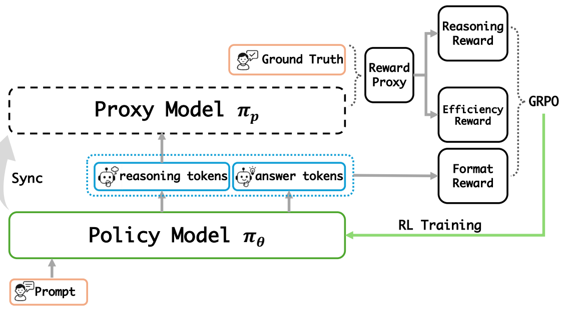

Figure 3: The overall process of NOVER.

It can be seen in Equation 1 that the incentive training described above requires a verifier to compute R_{rule}. **NOVER** eliminates the need for a verifier by introducing reasoning perplexity as a reward proxy, which can be calculated on any SFT data.

4.1 Reasoning Perplexity as Proxy

The core idea of NOVER is that the perplexity of the ground truth answer, conditioned on the model's reasoning trajectory, can serve as a natural proxy for reward. Given a prompt p, the ground truth answer g, and a model response consisting of reasoning tokens t and answer tokens a, we compute the reasoning perplexity P_r using a proxy model π_{p} as follows:

$$P_r(p, t, g) = \exp\left(-\frac{\sum_{i=1}^{|g|} \log \pi_{\text{p}}(g_i \mid p, t, g_{<i})}{|g| \cdot N(|t|)} \right) \tag{8}$$

where $|g|$ denotes the number of tokens in the ground truth. $N(|t|)$ is a simple normalization factor based on the length of the reasoning tokens, which alleviates the length bias of perplexity:

$$N(|t|) = \max\left(1, 1 + \log(|t|)\right) \tag{9}$$

A lower reasoning perplexity P_r indicates a higher probability of generating the correct answer based on the reasoning tokens. As the calculation of P_r is done in a teacher-forcing way without any decoding, computation of P_r requires only $\sim 5\%$ of total training time (see details in Appendix B. Then P_r can be utilized to calculate the reward defined in §4.3.

4.2 Policy-Proxy Synchronization

Unlike recent work that employs an extra frozen model as the proxy (Gurung and Lapata, 2025), we instead use the policy model π_θ itself as the proxy π_p. This choice reflects their shared goal: minimizing the perplexity of correct answers given high-quality reasoning. Moreover, an extra frozen proxy can diverge from the evolving policy, leading to inconsistency (see §6.3). Importantly, using π_θ as π_p does not mean the model acts as both athlete and referee, as π_p itself does not judge but leverages the objective ground truth g for fair evaluation. In practice, we initialize both π_θ and π_p from the same pretrained checkpoint and periodically sync π_θ to π_p every T_sync steps via exponential smoothing, following TR-DPO (Gorbatovski et al., 2025).

$$\pi_\mathrm{p} \leftarrow \alpha \cdot \pi_\mathrm{p} + (1-\alpha) \cdot \pi_\theta \tag{10}$$

where $\alpha \in [0, 1]$ denotes the synchronization rate. Such synchronization enables the proxy model to gradually adapt to improvements in the policy while ensuring a stable calculation of perplexity. Since we employ LoRA (Hu et al., 2022) for efficient training, the π_θ and π_p share the same base model and can be switched seamlessly by replacing the LoRA adapter, which comprises only about 0.4% of the full model parameters.

4.3 Verifier-Free Reward

Given a prompt p and its corresponding group of completions (t_i, a_i), we compute two rewards based on the reasoning perplexity P_r.

Reasoning Reward Reasoning perplexity P_r can serve directly as a reward, as it reflects the quality of the reasoning trajectory. VR-CLI (Gurung and Lapata, 2025) and reasoning advantage SFT (Foster et al., 2024) suggest calculating relative perplexity gain to stabilize training, but we found that GRPO's group normalization already achieves a similar calculation, so there is no need for such a redundant operation. In practice, we observe that relative perplexity improvements still vary widely across different samples, causing high variance in clip ratios and unstable training. To mitigate this, we propose to discretize reasoning perplexity into reasoning rewards R_r. For each completion (p_i, t_i, a_i) with P_r^i, we compute R_r based on its rank among n_valid valid completions. Let $\mathcal{P} = \{P_r^1, P_r^2, ..., P_r^{n_\mathrm{valid}}\}$, sorted as $P_r^{(1)} \leq \cdots \leq P_r^{(n_\mathrm{valid})}$, reasoning rewards are assigned by quantile rank:

$$R_r(p_i, t_i, a_i) = \begin{cases} \frac{n_\mathrm{valid} - \mathrm{rank}(P_r^i) + 1}{n_\mathrm{valid}}, & \text{if } \mathrm{rank}(P_r^i) \leq k \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

In practice, we use either $k = 1$ or $k = n_\mathrm{valid}$. The former assigns a reward of 1.0 only to the best (lowest-perplexity) completion, which is suitable for tasks with objectively correct answers. The latter distributes rewards to all completions based on their normalized rank and is better suited for subjective or open-ended tasks.

Efficiency Reward Recent work (Sui et al., 2025) suggests that longer reasoning is not always better, and effective reasoning length is what matters. We introduce an Efficiency Reward R_e, encouraging the model to generate shorter but better reasoning trajectories. A completion should be rewarded if it achieves lower reasoning perplexity with fewer reasoning tokens compared to other completions. The R_e can be calculated as:

$$R_\mathrm{e}(p_i, t_i, a_i) = \frac{\sum_{j=1, j\neq i}^{n} \mathbf{1}\left[P_r^i < P_r^j \wedge |t_i| < |t_j|\right]}{n_\mathrm{valid} - 1} \tag{12}$$

It is notable that although R_e shares a similar target with R_r, where lower perplexity is better, we cannot use R_e to replace R_r since in the early training stage, most completions could not develop efficient reasoning to acquire the R_e.

These two rewards are then combined with the tag format reward R_f in Equation 2 to form the final reward for GRPO training. The combined reward function is then:

$$R_\mathrm{total} = w_\mathrm{f} R_\mathrm{f} + \mathbb{I}(R_\mathrm{f} = 1) \cdot (w_\mathrm{r} R_\mathrm{r} + w_\mathrm{e} R_\mathrm{e}) \tag{13}$$

where $w_\mathrm{f}, w_\mathrm{r}, w_\mathrm{e}$ are weights that control the relative importance of each reward component. All these rewards are in the range $[0, 1]$. Notably, they are conditionally dependent rather than simply additive: only when the tag format is correct ($R_\mathrm{f} = 1$) are the other two rewards calculated and added; otherwise, they are set to 0. We provide a detailed analysis of how this design effectively prevents the "curse of proxy" in §6.3. NOVER can be combined with various RL algorithms, such as PPO

Method	General Reasoning			Writing	Social Intelligence		Multilingual	Avg.
	NR	GT	WI	SGN	EB	TB	OPUS	
Qwen2.5-3B								
Base Model	21.80 %	43.10 %	18.40 %	18.70 %	32.03 %	46.79 %	16.70 %	28.22 %
+ *CoT*	24.40 %	48.90 %	24.20 %	14.76 %	28.12 %	51.23 %	1.40 %	27.57 %
+ *SFT*	27.00 %	36.20 %	27.30 %	20.08 %	36.72 %	48.66 %	17.30 %	30.47 %
+ *NOVER*	**28.60 %**	**60.30 %**	**28.10 %**	**41.64 %**	**38.28 %**	**57.88 %**	**20.70 %**	**39.36 %**
Qwen2.5-7B								
Base Model	31.80 %	48.50 %	20.70 %	24.21 %	28.91 %	44.22 %	19.30 %	31.09 %
+ *CoT*	31.20 %	57.60 %	29.20 %	33.46 %	38.28 %	50.99 %	1.60 %	34.62 %
+ *SFT*	27.50 %	45.20 %	33.50 %	37.85 %	47.66 %	57.06 %	23.30 %	38.87 %
+ *NOVER*	**38.20 %**	**61.80 %**	**36.60 %**	**50.79 %**	**49.22 %**	**67.79 %**	**26.80 %**	**47.31 %**
Qwen2.5-3B-Instruct	27.10 %	50.00 %	31.50 %	21.25 %	40.62 %	58.69 %	19.90 %	35.58 %
Qwen2.5-7B-Instruct	29.90 %	56.20 %	35.60 %	67.72 %	46.88 %	65.23 %	23.50 %	46.43 %
R1-Distill-Qwen-7B	41.00 %	60.20 %	38.00 %	40.16 %	35.16 %	54.61 %	8.20 %	39.62 %

Table 1: Overall performance. Each cell is shaded based on its relative improvement. Values in **bold** indicate the best-performing variant, excluding other post-trained models. **NR**: Natural Reasoning, **GT**: General Thoughts-430k, **WI**: WebInstruct, **SGN**: SS-GEN, **EB**: EmoBench, **TB**: TomBench, **OP**: OPUS-BOOK-TRANSLATION.

(Schulman et al., 2017), and here we use GRPO for efficient training.

5 Experimental Setup

Dataset To evaluate the effectiveness of the NOVER-incentivized model, we select datasets spanning four broad domains: (1) General Reasoning: This category includes challenging tasks that go beyond standard STEM benchmarks, requiring complex reasoning, including three datasets, **Natural Reasoning** (Yuan et al., 2025a), **General Thought** (General Reasoning, 2025), and **WebInstruct** (TIGER-Lab, 2025), which require models to produce factually grounded answers, often involving multi-step reasoning and justification. (2) Creative Writing: To assess long-form narrative generation and planning ability, we use **SS-GEN** (Feng et al., 2025c), a benchmark focused on coherent, socially themed storytelling. (3) Social Intelligence: We evaluate models on emotionally and socially grounded reasoning using **EmoBench** (Sabour et al., 2024) and **ToMBench** (Chen et al., 2024), which test emotion recognition, social action prediction, and theory-of-mind reasoning. (4) Multilingual Ability: We employ the **OPUS book** corpus (Tiedemann, 2012), which includes translation tasks across 16 languages and 64 source-target language pairs, to evaluate cross-lingual reasoning and generalization. For all datasets and subcategories, we construct training and test sets via uniform sampling. Further details are in Appendix A.

Baselines We evaluate 3B and 7B versions of Qwen 2.5 (Yang et al., 2024) against several baselines. Prior work suggests that instruction-following and basic *CoT* reasoning can emerge during multi-stage pretraining (Wei et al., 2022; Zeng et al., 2025). Thus, we include both vanilla and CoT responses to isolate the effect of NOVER incentive tuning from capabilities already acquired in pretraining. We also consider an SFT baseline, where the same pretrained checkpoints are fine-tuned on each dataset's training split (identical to that used by NOVER). This comparison highlights whether NOVER enhances generalization and abstraction in contrast to SFT (Chu et al., 2025). For reference, we report results from three strong post-trained models: Qwen2.5-3B/7B-Instruct, and R1-Distill-Qwen-7B, a 7B variant distilled from DeepSeek-R1 671B (Guo et al., 2025). These models, trained with massive instruction-following and reasoning data, serve as high-performance baselines.

Training and Evaluation Our training framework is built on Huggingface TRL (von Werra et al., 2020), using LoRA adapters (Hu et al., 2022) for efficient fine-tuning. Full hyperparameters are listed in Appendix C. Training time is mainly determined by rollout efficiency, which depends on the maximum generation length and varies by task (Appendix A). For example, training Qwen2.5-7B for 1,000 steps with a max length of 1,024 tokens on two H100 GPUs takes about 2 hours. For evaluation, we report accuracy on all datasets. For choice questions, we extend the extractor from Team et al. (2025) to identify both the option letter and corresponding text, then match against ground truth. For open-ended QA, we use Gemini-2.0-flash (Anil et al., 2023; Gemini et al., 2024) in an LLM-as-a-Judge setup (see details in Appendix E).

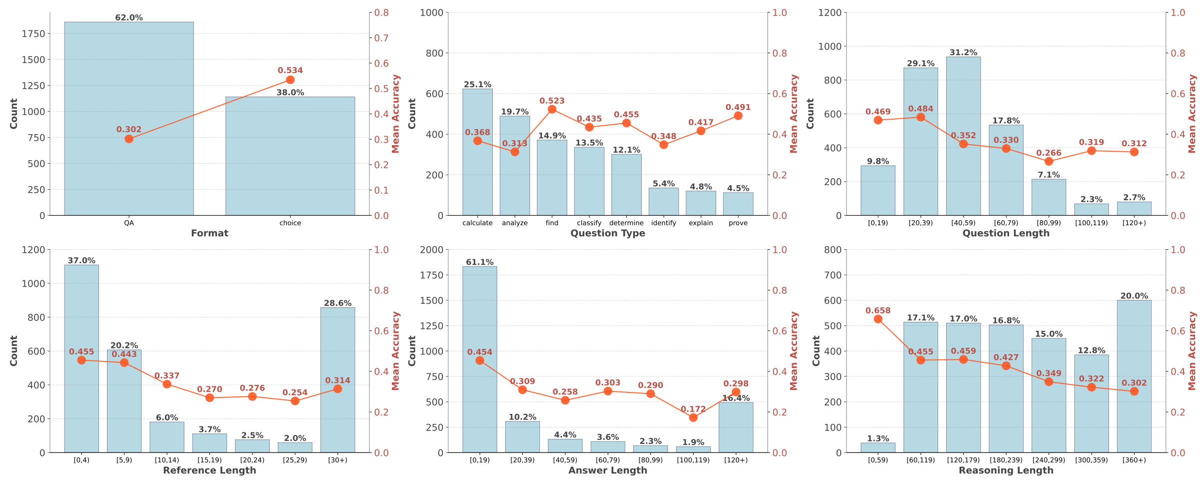

Figure 4: Accuracy of NOVER on three general reasoning tasks by the distribution of different aspects.

6 Results and Analysis

We present a comprehensive and structured experiment organized into three main parts. (1) To examine *when* NOVER performs well, §6.1 analyzes its performance across different task types, and §6.2 investigates how data distribution influence reasoning ability. (2) To understand *why* NOVER is effective, §6.3 examines how NOVER addresses reward hacking for stable learning; §6.4 compares its verifier-free design with Model as Verifier approaches; and §6.5 studies how incentivized reasoning patterns evolve during training. (3) To demonstrates NOVER's *flexibility*, in §6.6 we explored inverse incentive training for creative writing.

6.1 Overall Performance

Table 1 summarizes the performance of NOVER and all baselines. We observe that, for both 3B and 7B model sizes, NOVER consistently outperforms all baselines across all types of tasks. Remarkably, NOVER enables the 3B model to achieve nearly 60% accuracy on General Thoughts, approaching the performance of the much larger R1-Distill-Qwen-7B model distilled from a 671B teacher. In Appendix G, we give a detailed example of how reasoning perplexity discriminates between good and bad responses, thus helping the model to reinforce learning. The improvement is particularly pronounced on datasets that challenge areas where less data is pretrained, such as EmoBench (for social intelligence) and OPUS (for multilingual). In these areas, direct CoT prompting may reduce accuracy. Our manual inspection of CoT outputs reveals that the base model is capable of generating well-structured and fluent CoT. However, these CoT-generated rationales can be easily hallucinated (Huang et al., 2025; Ye et al., 2024; Li et al., 2024). In contrast, NOVER effectively corrects such hallucinated reasoning processes, as illustrated by a detailed example in Appendix F. Notably, SFT sometimes even underperforms the base model, as it encourages the model to directly map questions to answers without an explicit reasoning process. In contrast, the CoT, NOVER, and even the base model can generate intermediate reasoning tokens that aid in answering.

Model	Method	NR	GT	WI
Llama-3.1-8B	*Base Model*	34.20 %	36.70 %	29.90 %
	+ CoT	28.10 %	35.10 %	30.00 %
	+ SFT	23.60 %	23.40 %	**34.50 %**
	+ NOVER	**40.70 %**	**41.50 %**	34.00 %
Mistral-7B	*Base Model*	**33.00 %**	17.80 %	27.00 %
	+ CoT	29.20 %	18.60 %	27.10 %
	+ SFT	22.50 %	20.70 %	27.80 %
	+ NOVER	32.20 %	**21.90 %**	**29.30 %**

Table 2: General Reasoning performance with different instruct model backends.

In addition, as shown in Table 2, we also evaluated other model families such as Mistral (Jiang et al., 2024) and LLaMA (Dubey et al., 2024). Since the base checkpoint of Qwen exhibits strong instruction following ability, it can be directly optimized with reinforcement learning in an R1-Zero fashion. In contrast, the base checkpoints of other models lack such capability, so we employed instruct-tuned checkpoints instead. The results show that NOVER achieves consistent gains across different model backends and checkpoint types (pretrained or instruct-tuned).

6.2 When and Where for Effective Reasoning

Question format We analyze NOVER's performance distribution on the general reasoning area, as shown in Figure 4. For question format, the accuracy on multiple-choice questions remains consistently higher than that on open-ended QA. This is primarily because the presence of candidate options in the question effectively reduces the search space during RL optimization, thereby lowering the task difficulty. We further extract and analyze the key action words from the questions. The model achieves higher accuracy on questions with clear solution directions, such as *find*, *determine*, and *classify*, since the corresponding reasoning chains are more likely to be sampled and reinforced during training. In contrast, for questions with more flexible requirements and less prior constraint on the reasoning process, such as *analyze*, the model's performance is relatively weaker.

Through the lens of length We also analyze the effect of length across four components: question, reference, generated answer, and reasoning process. For the first three, we observe that shorter inputs or outputs generally lead to higher accuracy. Interestingly, for the length of the model-generated reasoning process, accuracy remains relatively stable within the range of 60 to 240 tokens. This suggests that NOVER effectively incentivizes the model to adaptively generate reasoning of appropriate length according to the difficulty of each question, a property that is closely related to the design of the efficiency reward.

Method	3B	7B
Base	12.43%	14.59%
+ *CoT*	14.23%	19.28%
+ *SFT*	26.49%	29.73%
+ *NOVER*	18.74%	23.42%

Table 3: Experiments on FANToM (Kim et al., 2023), a theory of mind task with false premise problems.

False Premise Task Recent work on RLVR suggests that RL may not exceed the capabilities of pretrained models (Yue et al., 2025) in domains with verifiable and structured answers. While NOVER focuses on free-form domains, we conduct a false premise task that similarly reveals the limits of RL. We adopt FANToM (Kim et al., 2023), a theory-of-mind dataset where models answer questions from a specific character's perspective in multi-party dialogues. Some questions rest on false premises (e.g., asking for character A's opinion on a movie when A was absent during the discussion on the movie in the conversation), making them unanswerable. Experiments show that SFT can memorize such refusal patterns from training data, whereas NOVER depends on the pretrained model to generate candidate responses and selectively reinforce the better ones. Lacking exposure to refusal behavior during pretraining, the model struggles to reject false-premise questions, resulting in weaker performance than SFT (see Table 3 and Appendix H). These findings suggest that future work should consider integrating multiple post-training strategies.

6.3 Curse of Proxy

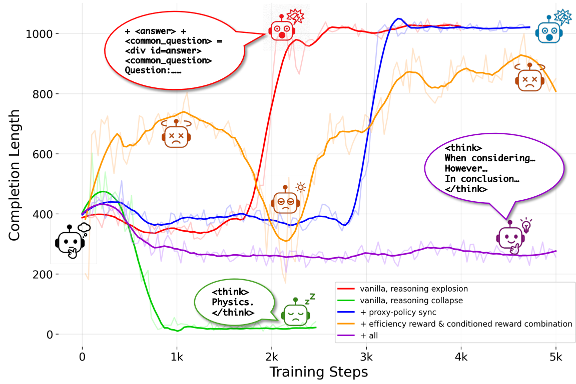

Figure 5: The curse of proxy: inaccurate proxy reward may lead to reward hacking.

NOVER uses reasoning perplexity as a proxy for the ideal reward, a common practice in RL (Ng and Russell, 2000; Christiano et al., 2017). However, this introduces the **curse of proxy**: *imprecise rewards can lead to reward hacking (Amodei et al., 2016), where models exploit flaws in the proxy rather than genuinely improving performance*. To analyze this, we perform ablations on NOVER and track completion lengths during training (Figure 5). (1) Training with only R_f and R_r reveals two failure modes: *reasoning explosion* (red), which is overlong, often garbled outputs, and *reasoning collapse* (green), where models stop thinking and generate minimum reasoning tokens. Both stem from proxy misalignment with the evolving policy. (2) Adding proxy-policy synchronization mitigates this (blue), though eventual explosion still occurs. (3) Introducing the efficiency reward R_e and conditioned reward combination enables partial self-recovery (orange), as invalid completions receive zero reward, encouraging resampling. Yet, recovery is inefficient. (4) Full NOVER integration yields stable training (purple): the model stays "sober," optimizing reasoning length only when it improves outcomes. Synchronization further reduces proxy bias, supporting robust optimization.

6.4 Verifier, Free, or Not

Group	Method	3B	7B
Baselines	*Base*	18.40%	20.70%
	$+CoT$	24.20%	29.20%
	$+SFT$	27.30%	33.50%
Model as Verifier	$+GV$	18.30%	30.00%
	$+LJ$	21.40%	3.80%
	$+LJ_S$	–	21.60%
Verifier-Free	$+NOVER$	28.10%	36.60%

Table 4: Experiments on WebInstruct. We compare LLM-as-a-Judge (LJ) and the officially released verifier model for WebInstruct, the general verifier GV.

To evaluate NOVER on reasoning tasks with difficult verification, we compared it against alternative verifier designs: an LLM-as-a-judge (LJ) and a fine-tuned verifier model. Experiments were conducted on WebInstruct, which includes an official general verifier model (GV) (Ma et al., 2025). For LJ, we used Gemini-2.0-flash with two prompt variants: a lenient "judge" prompt, and a stricter version (LJ_s) (aligned with our evaluation setup). As shown in Table 4, model-based verifiers were highly unstable. With LJ, lenient prompts encouraged reward hacking, where π_θ generated vague but superficially valid outputs to elicit positive rewards (e.g., giving rough ideas instead of precise answers). In contrast, strict prompts yielded sparse rewards and unstable training, thus, the 3B model failed to train. The dedicated verifier also proved unreliable, often misled by the policy model. For example, the policy might only sketch initial steps and prompt the verifier to complete the calculation, causing the verifier to abandon judgment and instead solve the task, then assign an undeserved positive reward.

6.5 Incentivized Reasoning Patterns

We further investigated how reasoning patterns evolved during the training of NOVER. We extracted the reasoning tokens generated by the model at different training steps, including the outputs produced using CoT prompting before training (as 0 step), on a fixed test set. These reasoning traces were then classified using Gemini-2.0-flash into one of several predefined reasoning patterns. Following the design in (Zeng et al., 2025), we defined six types of reasoning patterns (see details in Appendix J). As shown in Figure 6, before training, the model primarily used task decomposition typical of CoT prompting. As training progressed with the influence of the efficiency reward, the model learned to skip redundant reasoning steps when it could directly provide intermediate conclusions, resulting in a notable increase in the proportion of the direct reasoning pattern. At the same time, other reasoning types began to appear and gradually stabilized, reflecting the development of reasoning that is both effective and efficient.

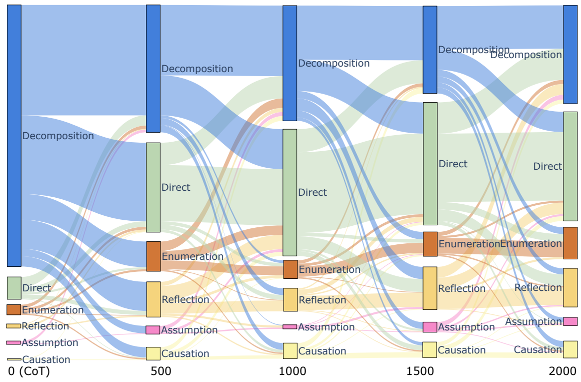

Figure 6: Change of reasoning patterns over steps.

6.6 Inverse Incentive Training

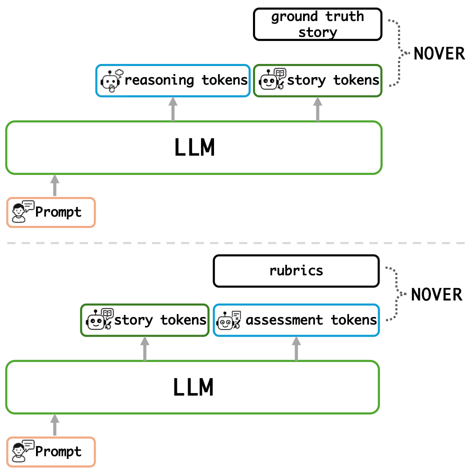

Figure 7: Comparison between standard NOVER training (up) and NOVER_RUBRIC training (down).

We further explore NOVER's flexibility in an interesting setting using the SS-GEN creative writing dataset (Feng et al., 2025c), where both ground truth stories and structural rubrics are provided. Figure 7 illustrates how we construct the inverse incentive training pipeline to enable rubric learning. Traditional incentive training requires the model to first generate thinking tokens, followed by answer tokens, with rewards computed to incentivize the intermediate thinking process. In the SS-GEN dataset's story generation setup, by contrast, we require the model to generate story tokens first, followed by assessment tokens. There is no ground-truth story, but only ground-truth rubrics. In this setting, training with NOVER allows the intermediate tokens being incentivized, namely, the story tokens, to become the desired answer tokens. We adapt NOVER by treating these rubrics as the guidelines for self-assessment, while the story becomes the intermediate process to incentivize, denoted NOVER_RUBRIC. This approach inverts the standard paradigm, implementing a "process as outcome" strategy. What we want is the intermediate tokens (story) instead of the final outputs (assessment). NOVER's design thus effectively steers generation toward rubric satisfaction, without needing ideal story exemplars during training. Empirical results show that Qwen2.5-7B's accuracy improves from **50.79%** (standard NOVER) to **64.37%** after rubric-based training.

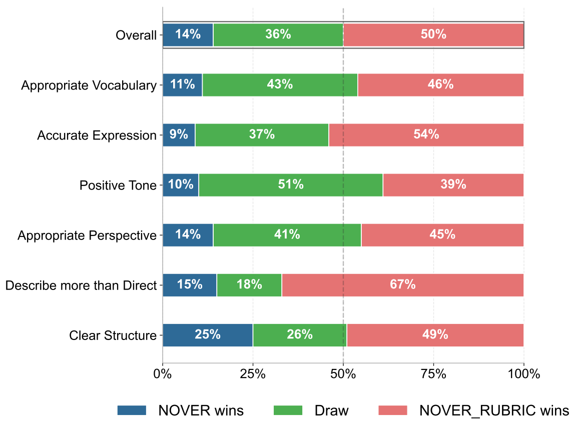

Figure 8: Human judgment on different rubrics for SS-GEN story generation.

To further validate the gains, we conducted human evaluations on each rubric dimension (Figure 8), comparing stories from standard NOVER and the rubric-trained variant. The rubrics are detailed in Appendix I. Results consistently favor the latter across all criteria, highlighting NOVER's flexibility for creative applications beyond classical reasoning tasks. It is worth noting that, due to the lack of explicit procedural rewards, relying solely on outcome-based rubric rewards can lead to occasional hallucinations in generated stories, e.g., repeating "once upon a time" twice. Future approaches will need to combine process and outcome rewards to advance the development of inverse incentive training.

7 Conclusion

In this paper, we present NOVER, a verifier-free framework for incentive training that pushes the boundaries of incentive learning to encompass any text-to-text task. NOVER demonstrates strong performance, and we discuss both the advantages and limitations of such a reinforcement learning paradigm, highlighting promising future directions for achieving robust and effective post-training. The flexibility of NOVER also enables novel learning paradigms such as inverse incentive training.

Limitations

(Zeng et al., 2025; Liu et al., 2025a) show that incentive training requires the base model to possess certain fine-tuned capabilities, such as partial CoT reasoning and instruction-following, to effectively incentivize reasoning. This observation is consistent with our findings on Qwen models. Moreover, general reasoning tasks in free-form format demand stronger base model capabilities than structured-answer tasks like math or coding. Currently, Qwen is a suitable choice in open-source models that meet the above requirements, so we conduct experiments on Qwen instead of other open-source models. Major open-source model teams have also recognized this and strengthened multi-stage pretraining by incorporating large amounts of CoT and instruction data, which gives us confidence that NOVER will be able to incentivize more models in the future.

Acknowledgments

This work was supported in part by the UK Engineering and Physical Sciences Research Council through a Turing AI Fellowship (grant no. EP/V020579/1, EP/V020579/2) and the Prosperity Partnership scheme (grant no. UKRI566). Wei is supported by a PhD studentship provided by King's College London. The authors acknowledge the use of King's Computational Research, Engineering and Technology Environment (CREATE) at King's College London.

References

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman, and Dan Mané. 2016. Concrete Problems in AI Safety. *CoRR*, abs/1606.06565.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, and 33 others. 2023. Gemini: A Family of Highly Capable Multimodal Models. *CoRR*, abs/2312.11805.

Zhuang Chen, Jincenzi Wu, Jinfeng Zhou, Bosi Wen, Guanqun Bi, Gongyao Jiang, Yaru Cao, Mengting Hu, Yunghwei Lai, Zexuan Xiong, and Minlie Huang. 2024. ToMBench: Benchmarking Theory of Mind in Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15959–15983. Association for Computational Linguistics.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep Reinforcement Learning from Human Preferences. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. 2025. SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training. *CoRR*, abs/2501.17161.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training Verifiers to Solve Math Word Problems. *CoRR*, abs/2110.14168.

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

Jiazhan Feng, Shijue Huang, Xingwei Qu, Ge Zhang, Yujia Qin, Baoquan Zhong, Chengquan Jiang, Jinxin Chi, and Wanjun Zhong. 2025a. Retool: Reinforcement learning for strategic tool use in llms. *arXiv preprint arXiv:2504.11536*.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. 2025b. Video-R1: Reinforcing Video Reasoning in MLLMs. *CoRR*, abs/2503.21776.

Yi Feng, Mingyang Song, Jiaqi Wang, Zhuang Chen, Guanqun Bi, Minlie Huang, Liping Jing, and Jian Yu. 2025c. SS-GEN: A Social Story Generation Framework with Large Language Models. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 1300–1308. AAAI Press.

Tom Foster, Eshaan Ahmed, James Cook, Shaked Lifshitz, Tim Rocktäschel, and Jakob N. Foerster. 2024. On Reward Functions For Self-improving Chain-of-thought Reasoning Without Supervised Datasets (Abridged Version). In *Language Gamification - NeurIPS 2024 Workshop*.

Gemini, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

General Reasoning. 2025. GeneralThought-430K: Open Reasoning Dataset. `https://huggingface.co/datasets/GeneralReasoning/GeneralThought-430K`.

Alexey Gorbatovski, Boris Shaposhnikov, Alexey Malakhov, Nikita Surnachev, Yaroslav Aksenov, Ian Maksimov, Nikita Balagansky, and Daniil Gavrilov. 2025. Learn Your Reference Model for Real Good Alignment. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 80 others. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *CoRR*, abs/2501.12948.

Aashish Gurung and Mirella Lapata. 2025. Learning to Reason for Long-form Story Generation. *arXiv preprint arXiv:2503.22828*.

Minggui He, Yilun Liu, Shimin Tao, Yuanchang Luo, Hongyong Zeng, Chang Su, Li Zhang, Hongxia Ma, Daimeng Wei, Weibin Meng, Hao Yang, Boxing Chen, and Osamu Yoshie. 2025. R1-T1: Fully Incentivizing Translation Capability in LLMs via Reasoning Learning. *CoRR*, abs/2502.19735.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. 2025. Open-Reasoner-Zero: An Open Source Approach to Scaling Up Reinforcement Learning on the Base Model. *CoRR*, abs/2503.24290.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.*, 43(2):42:1–42:55.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 80 others. 2024. OpenAI o1 System Card. *CoRR*, abs/2412.16720.

AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, Ddl Casas, F Bressand, G Lengyel, G Lample, L Saulnier, and 1 others. 2024. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. *CoRR*, abs/2503.09516.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Le Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. 2023. FANToM: A Benchmark for Stress-testing Machine Theory of Mind in Interactions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 14397–14413. Association for Computational Linguistics.

Levente Kocsis and Csaba Szepesvári. 2006. Bandit Based Monte-carlo Planning. In *Machine Learning: ECML 2006, 17th European Conference on Machine Learning, Berlin, Germany, September 18-22, 2006, Proceedings*, volume 4212 of *Lecture Notes in Computer Science*, pages 282–293. Springer.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large Language Models are Zero-shot Reasoners. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Bangzheng Li, Ben Zhou, Fei Wang, Xingyu Fu, Dan Roth, and Muhao Chen. 2024. Deceptive Semantic Shortcuts on Reasoning Chains: How Far Can Models Go without Hallucination? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 7675–7688. Association for Computational Linguistics.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's Verify Step by Step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Wei Liu, Chenxi Wang, YiFei Wang, Zihao Xie, Rennai Qiu, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, and Chen Qian. 2024. Autonomous agents for collaborative task under information asymmetry. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025a. Understanding R1-Zero-like Training: A Critical Perspective. *CoRR*, abs/2503.20783.

Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. 2025b. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*.

Yilun Lu, Cheng Zhang, Jiaming Song, Linxi Fan, and William Wang. 2025. ToM-RL: Reinforcement Learning Unlocks Theory of Mind in Small LLMs. *arXiv preprint arXiv:2504.01698*.

Xueguang Ma, Qian Liu, Dongfu Jiang, Ge Zhang, Zejun Ma, and Wenhu Chen. 2025. General-Reasoner: Advancing LLM Reasoning Across All Domains. `https://github.com/TIGER-AI-Lab/General-Reasoner/blob/main/General_Reasoner.pdf`.

Andrew Y Ng and Stuart J Russell. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 663–670.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Yang Wang. 2023. Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 3806–3824. Association for Computational Linguistics.

Siya Qi, Yulan He, and Zheng Yuan. 2024. Can we catch the elephant? the evolvement of hallucination evaluation on natural language generation: A survey. *arXiv e-prints*, pages arXiv–2404.

Sahand Sabour, Siyang Liu, Zheyuan Zhang, June Liu, Jinfeng Zhou, Alvionna Sunaryo, Tatia Lee, Rada Mihalcea, and Minlie Huang. 2024. EmoBench: Evaluating the Emotional Intelligence of Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5986–6004, Bangkok, Thailand. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347.

ByteDance Seed, Yufeng Yuan, Yu Yue, Mingxuan Wang, Xiaochen Zuo, Jiaze Chen, Lin Yan, Wenyuan Xu, Chi Zhang, Xin Liu, and 1 others. 2025. Seed-thinking-v1. 5: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *CoRR*, abs/2402.03300.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, and 1 others. 2025. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-Searcher: Incentivizing the Search Capability in LLMs via Reinforcement Learning. *CoRR*, abs/2503.05592.

Yi Su, Dian Yu, Linfeng Song, Juntao Li, Haitao Mi, Zhaopeng Tu, Min Zhang, and Dong Yu. 2025. Crossing the Reward Bridge: Expanding RL with Verifiable Rewards Across Diverse Domains. *CoRR*, abs/2503.23829.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Ben Hu. 2025. Stop Overthinking: A Survey on Efficient Reasoning for Large Language Models. *CoRR*, abs/2503.16419.

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, Kaixin Deng, Shian Jia, Sichao Jiang, Yiyan Liao, Rui Li, Qinrui Li, Sirun Li, and 77 others. 2025. SuperGPQA: Scaling LLM Evaluation across 285 Graduate Disciplines. *CoRR*, abs/2502.14739.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012*, pages 2214–2218. European Language Resources Association (ELRA).

TIGER-Lab. 2025. WebInstruct-verified Dataset. `https://huggingface.co/datasets/TIGER-Lab/WebInstruct-verified`.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. TRL: Transformer Reinforcement Learning. `https://github.com/huggingface/trl`.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023a. Plan-and-solve Prompting: Improving Zero-shot Chain-of-thought Reasoning by Large Language Models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2609–2634. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, and 1 others. 2025. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Lilian Weng. 2025. Why we think. *lilianweng.github.io*.

Xiaobo Xia and Run Luo. 2025. Gui-r1: A generalist r1-style vision-language action model for gui agents. *arXiv preprint arXiv:2504.10458*.

Tian Xie, Zitian Gao, Qingnan Ren, Haoming Luo, Yuqian Hong, Bryan Dai, Joey Zhou, Kai Qiu, Zhirong Wu, and Chong Luo. 2025. Logic-RL: Unleashing LLM Reasoning with Rule-based Reinforcement Learning. *CoRR*, abs/2502.14768.

Hanqi Yan, Qinglin Zhu, Xinyu Wang, Lin Gui, and Yulan He. 2024. Mirror: Multiple-perspective Self-reflection Method for Knowledge-rich Reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 7086–7103. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 Technical Report. *CoRR*, abs/2412.15115.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, Bo Zhang, and Wei Chen. 2025. R1-Onevision: Advancing Generalized Multimodal Reasoning through Cross-modal Formalization. *CoRR*, abs/2503.10615.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023b. ReAct: Synergizing Reasoning and Acting in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Hongbin Ye, Tong Liu, Aijia Zhang, Wei Hua, and Weiqiang Jia. 2024. Cognitive Mirage: A Review of Hallucinations in Large Language Models. In *Proceedings of the First International OpenKG Workshop: Large Knowledge-Enhanced Models colocacted with The International Joint Conference on Artificial Intelligence (IJCAI 2024), Jeju Island, South Korea, August 3, 2024*, volume 3818 of *CEUR Workshop Proceedings*, pages 14–36. CEUR-WS.org.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, and 16 others. 2025. DAPO: An Open-source LLM Reinforcement Learning System at Scale. *CoRR*, abs/2503.14476.

Weizhe Yuan, Jane Yu, Song Jiang, Karthik Padthe, Yang Li, Dong Wang, Ilia Kulikov, Kyunghyun Cho, Yuandong Tian, Jason E Weston, and 1 others. 2025a. Naturalreasoning: Reasoning in the wild with 2.8 m challenging questions. *arXiv preprint arXiv:2502.13124*.

Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, Xiangpeng Wei, and 1 others. 2025b. VAPO: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. 2025. Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? *arXiv preprint arXiv:2504.13837*.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. SimpleRL-Zoo: Investigating and Taming Zero Reinforcement Learning for Open Base Models in the Wild. *CoRR*, abs/2503.18892.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. 2025. R1-VL: Learning to Reason with Multimodal Large Language Models via Step-wise Group Relative Policy Optimization. *CoRR*, abs/2503.12937.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. 2023. Least-to-most Prompting Enables Complex Reasoning in Large Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

A Dataset Construction Pipeline & Statistics

To minimize data contamination from pretraining, we prioritized the use of the most recent datasets available. We focused on general reasoning datasets that require multi-step inference and span multiple domains, not limited to STEM. In addition, we included text generation datasets that demand reasoning and planning abilities, such as creative writing, social intelligence, and multilingual translation. All datasets were cleaned and filtered to ensure high-quality data. Specifically:

- **Natural Reasoning** We excluded samples lacking a reference answer. We also filtered out samples where the reference answer was merely a number, a single word, or a single sentence, as such cases are often trivial for rule-based verifiers and do not reflect the open-ended reasoning tasks we aim to study.

- **General Thoughts and WebInstruct** These datasets underwent the same preprocessing as *Natural Reasoning*. What's more, these two datasets contain multiple-choice questions, and we converted them into free-form QA format, ensuring that the reference answers included both the correct choice and its content. Due to inconsistent option formatting in the original data (such as 1., A), A, (a).), we designed seven regex-based patterns to clean and standardize the multiple-choice items.

- **EmoBench** We selected two subsets, *emotional understanding* and *emotional application*. For *emotional understanding*, the prompt was adapted to specify a dual-option response format, reflecting the original structure in which each question is paired with two multiple-choice items (with four candidates each), targeting emotion recognition and causality, respectively.

- **FANToM** We chose two subsets aligned with free-form reasoning, which are *full_fact* (asking for an entity) and *full_belief_gen* (asking for opinion in the perspective of somebody).

- **Other Datasets** No preprocessing was applied to the remaining datasets.

For datasets with existing train/test splits, we retained them. For others, we created splits by evenly sampling across subcategories (e.g., academic subjects) to ensure distributional consistency. For very large datasets, we sample 2,000 examples for training, 1,000 for validation, and 1,000 for testing.

Dataset	#Original	#Filtered	#Train	#Validation	#Test
NR	1,150,000	192,178	2,000	1,000	1,000
GT	431,000	78,483	2,000	1,000	1,000
WI	232,000	231,833	2,000	1,000	1,000
SGN	5,087	5,087	4,070	509	508
EB	400	400	272	–	128
TB	2,860	2,860	1,432	571	857
OPUS	1,250,632	1,250,632	2,000	1,000	1,000
FT	1,863	1,863	1,308	–	555

Table 5: Dataset statistics after filtering and splits into training, validation, and test sets.

The detailed statistics are shown in Table 5, where **NR** is Natural Reasoning, **GT** is General Thoughts-430k, **WI** is WebInstruct, **SGN** is SS-GEN, **EB** is EmoBench, **TB** is TomBench, **OP** is OPUS-BOOK-TRANSLATION, **FT** is FANToM.

B Profiling

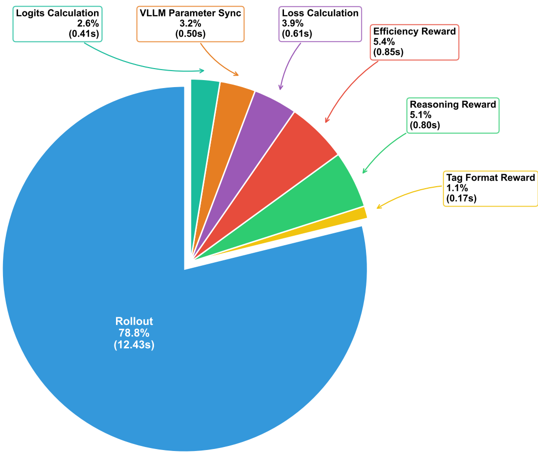

Figure 9: Profiling of the training time in NOVER.

Figure 9 shows the average training time per sample using two H100 GPUs with a completion length of 2,048 tokens. Most of the time is spent on the rollout stage, where vLLM (Kwon et al., 2023) is used to efficiently sample multiple completions from the policy model. In contrast, computing the reasoning and efficiency rewards accounts for only about 5% of the total training time.

C Hyperparameters

All hyperparameters for training NOVER are listed in Table 6. For general reasoning tasks and the SS-GEN creative writing task, we set the maximum completion length to 2048, while for other tasks, it is limited to 512. Training is run for up to 5,000

Notation	Definition	Value
r	LoRA rank	16
α_{LoRA}	LoRA scaling factor	32
p_{dropout}	LoRA dropout ratio	0.1
$dtype$	Training precision	`bfloat16`
G	Group Size	8
B	Batch size per GPU	8
β	KL coefficient	0.1
w_f	Format reward weights	1
w_r	Reasoning reward weights	1
w_e	Efficiency reward weights	1
τ	Rollout temperature	0.6
η_{max}	Maximum learning rate	$1e{-}5$
α	Synchronization coefficient	0.9
T_{sync}	Synchronization interval	100
L_{gen}	Max completion length	[512, 2048]
ϵ	clip range	0.1
ϵ_{high}	Upper clip range	0.2
T_{max}	Max training steps	5000

Table 6: Hyperparameters used for NOVER training.

steps, although we typically apply early stopping when the validation reward, especially the tag format reward, no longer improves. For 3B models, training generally requires more steps than for 7B models.

D System Prompt

Figure 10, Figure 11, and Figure 12 show the system prompt for three kinds of tasks in our experiments. We used minimal instructions in the prompt, relying solely on the `<think>` and `<answer>` tags, without explicitly directing the model to reason. This approach minimizes external intervention and maximizes the model's own incentivized capabilities.

System Prompt for QA

Question: {question}
Answer the question and return in the following format:

<think>
...
</think>
<answer>
...
</answer>

Figure 10: System prompt for most general reasoning tasks.

E LLM-as-Judge Prompt

Figure 13, Figure 14, and Figure 15 show the judgment prompt used in evaluation and model-as-

System Prompt for Choice task

Question: {question}
Options: {options}
Answer the question and return in the following format:

<think>
...
</think>
<answer>
[option letter]: [option text]
</answer>

Figure 11: System prompt for choice-format tasks.

System Prompt for Translation

Translate the following text
from {source_lang_name} to {target_lang_name}:

{source_text}

Answer the question and return in the following format:

<think>
...
</think>
<answer>
...
</answer>

Figure 12: System prompt for translation tasks.

Judge Prompt

Please determine if the following Model answer matches the Ground truth.
Only consider if the response matches the reference, regardless of how detailed or comprehensive it is.
Ground truth: {reference}
Model answer: {response}
Please provide your judgment in the following format:

<think>
[Your detailed reasoning about why the answers match or don't match]
</think>
<conclusion>
[yes/no]
</conclusion>

Return only the formatted response with reasoning and conclusion.

Figure 13: Judge prompt (the strict version) used for evaluation and model-as-judge experiments (§6.4).

Judge Prompt, Lenient Version

Please determine if the following Model answer matches the Ground truth.
Does the model's answer broadly align with the reference answer, even if it's not exactly the same?
Ground truth: {reference}
Model answer: {response}
Please provide your judgment in the following format:

<think>
[Your detailed reasoning about why the answers match or don't match]
</think>
<conclusion>
[yes/no]
</conclusion>

Return only the formatted response with reasoning and conclusion.

Figure 14: Judge prompt (the lenient version) used in model-as-judge experiments (§6.4).

judge experiments in §6.4. The judge prompt for the general verifier is taken from the official Huggingface repository [1]. We also follow the generation configuration on the official repository, where the temperature is set to 0.0 and the maximum tokens to 1,024.

Judge Prompt, General Verifier

User: ### Question: {question}
Ground Truth Answer: {reference_answer}
Student Answer: {answer_content}
For the above question, please verify if the student's answer is equivalent to the ground truth answer.
Do not solve the question by yourself; just check if the student's answer is equivalent to the ground truth answer.
If the student's answer is correct, output "Final Decision: Yes". If the student's answer is incorrect, output "Final Decision: No". Assistant:"

Figure 15: Judge prompt from the official code of general verifier (Ma et al., 2025).

F Reasoning Hallucination Case

Figure 16 presents an example from the OPUS translation dataset where both the base model and NOVER perform reasoning to generate a translation. As shown, the base model is able to produce seemingly correct reasoning steps, sometimes nearly identical to those of NOVER. However, despite generating the right steps, the base model fails to reach the correct conclusion. For instance, its reasoning chain ends with a self-check that asserts the translation is perfect and faithfully conveys the meaning of the original sentence, including the description of the "woman's appearance." Yet, in the earlier steps, the model translates only the subject, verb, and object, omitting any reference to the woman's appearance. This kind of confident but incorrect self-verification, known as one kind of reasoning hallucination (Huang et al., 2025; Qi et al., 2024), can cause the model to overlook essential content, resulting in worse performance than directly producing the translation. In contrast, NOVER follows a similar reasoning path but incorporates critical refinements. For example, it explicitly notes the need to translate not only the core grammatical elements but also any additional descriptive components. As a result, in the initial step of identifying main elements, NOVER successfully detects phrases related to the woman's appearance, such as "perfect figure" and "elegant dress," and accurately translates and integrates them in later steps. Moreover, NOVER provides more rigorous rubrics, such as emphasizing in the rearrangement step that Spanish word order "may differ from English."

G Reasoning Perplexity Example

Figure 17 shows two completions generated by NOVER in response to a medical question from the natural reasoning dataset, each associated with a different level of reasoning perplexity. As illustrated, Completion 1, which has lower reasoning perplexity, is clearly superior to Completion 2 in terms of physiological mechanisms, terminological accuracy, and the discussion on "the role of dietary factors", and overall scientific reasoning. In contrast, Completion 2 is relatively coarse and contains misleading or missing descriptions of relevant terms and mechanisms. Although reasoning perplexity does not directly compare the answer with a reference, it estimates the likelihood of the reference based on the inferred reasoning path, thereby providing a concise and effective measure of answer quality, as evidenced by this example.

H False Premise Example

Figure 18 illustrates an example from the FANToM dataset, shedding light on why NOVER may underperform compared to SFT in the false premise

[1] https://huggingface.co/TIGER-Lab/general-verifier

Case Study in OPUS Translation Task

Task:
Translate the following text from English to Spanish:
With her perfect figure and elegant dress she was, indeed, a strange apparition upon a lonely moorland path.

Reference:
Dada la perfección y elegancia de su vestido, resultaba, desde luego, una extraña aparición en la solitaria senda del páramo.

Base Model CoT:
To translate the given text from English to Spanish, I will follow these steps:
1. Identify the main components of the sentence: subject, verb, and object.
2. Translate each component individually.
3. Rearrange the components to form a grammatically correct sentence in Spanish.
4. Ensure that the translated sentence maintains the original meaning and tone.
Step 1: Identify the main components of the sentence.
Subject: "She"
......
Step 2: Translate each component individually.
Subject: "Ella" (She)
......
Step 3: Rearrange the components to form a grammatically correct sentence in Spanish.
"Ella era, en efecto, una extraña aparición en un sendero solitario de tierra mojada."
Step 4: Ensure that the translated sentence maintains the original meaning and tone.
The translated sentence maintains the original meaning and tone, *as it describes the woman's appearance and the setting*

Base Model Translation:
Ella era, en efecto, una extraña aparición en un sendero solitario de tierra mojada.

NOVER thinking process:
<think>
To translate the given text from English to Spanish, I will follow these steps:
1. Identify the main components of the sentence: subject, verb, object, *and any additional descriptive elements.*
2. Translate each component into Spanish while maintaining the original meaning and context.
3. Rearrange the translated components to fit the Spanish sentence structure, *which may differ from English.*
4. Review the final translation for accuracy and fluency.
Here's the step-by-step translation process:
1. Main components:
- Subject: "she"
......
- *Additional descriptive elements: "perfect figure", "elegant dress", "indeed", "upon a lonely moorland path"*
2. Translations:
- "she" -> "ella"
......
- "perfect figure" -> "figura perfecta"
- "elegant dress" -> "traje elegante"
- "indeed" -> "realmente"
- "upon a lonely moorland path" -> "en una solitaria senda de moorland"
4. Reviewing the final translation:
- The translation maintains the original meaning and context, and the sentence structure is appropriate for Spanish.
I have translated the given text from English to Spanish by identifying the main components, translating each component, rearranging them to fit the Spanish sentence structure, and reviewing the final translation for accuracy and fluency.
</think>

NOVER translation:
<answer>
Ella era, realmente, una extraña aparición en una solitaria senda de moorland con su figura perfecta y traje elegante.
</answer>

Figure 16: A case in the OPUS translation dataset, where the base model generates hallucinated CoT that leads to wrong translation, while NOVER follows similar steps but corrects the wrong parts in the thinking process.

Two Completions of NOVER on Natural Reasoning Dataset

Question:
What is the primary mechanism by which the body eliminates excess cholesterol, and how do dietary factors such as fiber intake and fatty food consumption affect this process? Provide a detailed explanation of the gastrohepatic cycle and the role of bile secretion in cholesterol elimination.

Reference:
The primary mechanism by which the body eliminates excess cholesterol is through the gastrohepatic cycle, where excess cholesterol is secreted by the bile ducts back into the gastrointestinal tract. Dietary factors such as fiber intake can help prevent the digestion of cholesterol, while fatty food consumption can increase cholesterol production.

NOVER completion 1, with reasoning perplexity = 1.406 (the good one)
<think>
The primary mechanism by which the body eliminates excess cholesterol involves a process called the gastrohepatic cycle. This cycle consists of several steps:
1. Cholesterol in the blood is taken up by the small intestine and converted into bile acids, which are then secreted into the intestine.
2. Bile acids are reabsorbed by the small intestine and returned to the liver via the portal vein.
3. In the liver, the bile acids are converted back into cholesterol and stored or used for other purposes.
4. When the body has an excess of cholesterol, the liver produces more bile acids, which are excreted in the feces.
......
In summary, the gastrohepatic cycle is the primary mechanism by which the body eliminates excess cholesterol, and dietary factors such as fiber intake and fatty food consumption can affect this process in various ways.
</think>
<answer>
The primary mechanism by which the body eliminates excess cholesterol involves a process called the gastrohepatic cycle. This cycle consists of several steps: Cholesterol in the blood is taken up by the small intestine and converted into bile acids, which are then secreted into the intestine. Bile acids are reabsorbed by the small intestine and returned to the liver via the portal vein. In the liver, the bile acids are converted back into cholesterol and stored or used for other purposes. When the body has an excess of cholesterol, the liver produces more bile acids, which are excreted in the feces. Dietary factors such as fiber intake and fatty food consumption can affect this process in various ways. Dietary fiber, particularly soluble fiber, can bind to bile acids in the intestine and prevent their reabsorption. This leads to an increased excretion of bile acids in the feces, which in turn increases the amount of cholesterol that is eliminated from the body. Fatty foods, particularly those high in saturated and trans fats, can increase the amount of cholesterol produced by the liver. This can lead to an increase in the amount of cholesterol that needs to be eliminated from the body. Certain fatty acids, such as omega-3 fatty acids, can also have a beneficial effect on cholesterol metabolism. Omega-3 fatty acids can reduce the amount of cholesterol produced by the liver and increase the amount of cholesterol that is eliminated from the body.
</answer>

NOVER completion 2, with reasoning perplexity = 1.516 (the bad one)
<think>
The primary mechanism by which the body eliminates excess cholesterol is through the excretion of cholesterol-rich bile acids in the feces. This process involves the breakdown and conversion of cholesterol into bile acids in the liver, followed by the release of these bile acids into the small intestine during digestion. Bile acids are then excreted from the body in the feces.
Dietary factors such as fiber intake and fatty food consumption can affect this process in several ways. Fiber-rich foods can increase the viscosity of the stool, which slows down the passage of bile acids through the digestive tract. This allows more time for the bile acids to be absorbed back into the bloodstream and reused, reducing the amount of cholesterol that is excreted in the feces.
......
Bile secretion plays a critical role in cholesterol elimination by providing the necessary environment for the breakdown and conversion of cholesterol into bile acids in the liver. Bile acids are then released into the small intestine during digestion, where they interact with fatty acids and triglycerides to form micelles. These micelles help to emulsify the fatty acids and make them more available for absorption by the small intestine.
</think>
<answer>
The primary mechanism by which the body eliminates excess cholesterol is through the excretion of cholesterol-rich bile acids in the feces. Dietary factors such as fiber intake and fatty food consumption can affect this process by increasing or decreasing the amount of cholesterol that is excreted in the feces. The gastrohepatic cycle is the process by which bile is produced in the liver, stored in the gallbladder, and released into the small intestine during digestion. Bile secretion plays a critical role in cholesterol elimination by providing the necessary environment for the breakdown and conversion of cholesterol into bile acids in the liver.
</answer>

Figure 17: We show two completions from NOVER on the Natural Reasoning dataset, showing how reasoning perplexity discriminates good responses from bad ones.

task setting. As shown, the example involves a multi-party dialogue and poses a question requiring higher-order theory of mind reasoning—for instance, inferring how person B would think about a certain topic from person A's perspective. The trap, however, is that B may not have participated in the conversation and thus is unaware of the topic, making the question unanswerable. The corresponding reference typically reflects this, such as "B is unaware of…". SFT can quickly memorize and reproduce such response patterns (e.g., "somebody is unaware of…"), whereas NOVER relies on the base model to first generate a correct answer, which is then reinforced. However, the base model may struggle to recognize the false premise and might not even consider the question potentially unanswerable. Consequently, as shown in the figure, NOVER attempts to answer the question but remains constrained by the flawed assumption that B is aware of the topic, and reasons accordingly. This highlights why NOVER may perform worse than SFT on counterintuitive tasks. Addressing such cases may require a hybrid approach: first using SFT to expose the model to appropriate response patterns and challenge its assumptions, followed by reinforcement learning to strengthen accurate reasoning.

A False Premise Example in FANToM Dataset

Here is a dialogue:

- Eliana: Hello, guys. I find it fascinating how various cultures influence food and cuisine. For instance, in my culture, Brazilian cuisine is a culmination of Europe, Africa, and Native American influences.
- Kade: How so, Eliana? Could you give an example?
- Eliana: Absolutely, Kade! Take feijoada for example, it's a Brazilian dish that's similar to a bean stew, filled with beef and pork. It's traditionally served with rice, collard greens, and farofa, a cassava flour mixture. Feijoada was often made by slaves, who would use whatever meats were available, hence the mixture.
- ......
- Kade: Definitely, Alberto. So, I guess we meet at the next food tasting session, where we have a taste of Brazil, Scotland, Canada, and Mexico!

Question:

What dishes from Scotland and Canada does Eliana think that Alberto believes were discussed by Addison and Kade respectively?

Reference:

Alberto is unaware about which dishes from Scotland and Canada were discussed by Addison and Kade respectively as he was not involved in the conversation when this was discussed. Thus, Eliana does not have any basis to form an opinion on what Alberto believes about the mentioned dishes.

NOVER Response:

<think>
Addison and Kade were the ones who discussed the dishes from Scotland and Canada. From the dialogue, it is evident that Addison mentioned the dish "haggis", which is a combination of sheep's heart, liver, and lungs minced with onion, oatmeal, suet, and spices...... So, Alberto believed that Addison and Kade discussed the dishes from Scotland and Canada, which are haggis and poutine respectively.
</think>
<answer>
Haggis is a dish from Scotland. Kade mentioned the dish "powwow" during the conversation, which is a traditional Mexican festival. So, *Alberto believed that* Addison and Kade discussed the dishes from Scotland and Canada, which are haggis and poutine respectively.
</answer>

Figure 18: A false premise example in FANToM (Kim et al., 2023) dataset, where the question cannot be answered.

I Rubrics in Inverse Incentive Training

To evaluate whether this inverse incentive training approach (treating the process as the desired outcome) successfully aligns the generated story tokens with the rubrics targeted by the final assessment tokens, we developed a human evaluation platform. We sampled 100 examples and asked human annotators to express preferences for each story concerning each rubric. The details of each rubric are provided to human annotators, including:

1. **Clear Structure** A clear structure refers to that there is a straightforward title establishing the intervention goal, an introduction explaining the story theme and setting, a main body depicting the specific situation and appropriate responses, and a conclusion summarizing the core information presented.

2. **Describe more than Direct** The number of descriptive sentences should exceed twice the number of coaching sentences in a Social Story. A descriptive sentence accurately describes the interaction, an event, or an explanation of the rationale that underlies what and

why people think, say, or do, including perspective and affirmative sentences. Coaching sentences gently and safely guide the behavior.

3. **Appropriate Perspective** A): The Social Story should never use the second-person perspective to prevent being overly direct. B): When describing negative behaviors, the Social Story should never employ the first-person perspective to safeguard the dignity and esteem of the audience.

4. **Positive Tone** The Social Story should always describe situations or guide behaviors in a positive manner, especially when depicting typical or expected behaviors in specific situations.

5. **Accurate Expression** The story should use language that is as clear and unambiguous as possible because ASD children typically interpret things literally rather than inferring implicit meanings that require social insight.

6. **Appropriate Vocabulary** The Social Story should choose the most comfortable and accurate vocabulary for the audience. Firstly, use positive verbs while also being mindful of the varying implications of verbs. Avoid using terms that are likely to evoke strong emotional feelings such as "shouldn't", "must", "supposed to" and so on.

J Reasoning Patterns

We categorize reasoning patterns in the NOVER incentivized reasoning process into six main kinds. Here is the definition of each kind:

1. Direct: The direct recall of factual information, definitions, or established concepts without further analysis or transformation. This pattern involves stating information from memory as-is.

2. Decomposition: The systematic organization of a problem into manageable components, establishing clear steps, intermediate goals, or methodical frameworks. This pattern involves creating a structured approach to solving complex problems.

3. Enumeration: The listing of multiple possibilities, options, alternatives, or cases without immediately selecting or committing to any specific one. This pattern involves comprehensively covering various aspects or potential scenarios.

4. Reflection: The process of revisiting, questioning, or reassessing previously stated ideas, assumptions, or conclusions. This pattern involves reflecting on one's own reasoning and making adjustments based on further consideration.

5. Assumption: The introduction of hypothetical conditions or premises that serve as a foundation for further reasoning. This pattern involves making conditional statements to explore potential scenarios or outcomes.

6. Causation: The establishment of cause-effect relationships between events, actions, or conditions. This pattern involves explaining how one factor leads to or influences another.