Distilling Robustness: Mitigating Persona Sensitivity in Language Models via RLVR Teacher-Student Training

Jihwan Oh*, **Murad Aghazada***, **Se-Young Yun**
KAIST AI
{ericoh929, aghazadamurad, yunseyoung}@kaist.ac.kr

Abstract

While persona prompting can boost Large Language Model (LLM) performance, prior work shows that finding the optimal persona is notoriously difficult, with its effects often being unpredictable and even detrimental. Current research has attempted to mitigate this volatility with prompt-level interventions, such as using costly inference-time ensembles to select the best output from a sensitive model. We argue for a paradigm shift: instead of searching for the best prompt for a volatile model, we should build an inherently robust model that is insensitive to persona variations. We hypothesize that persona sensitivity is not a random phenomenon but a systematic outcome of a model's alignment process. Our research reveals that models aligned with Reinforcement Learning with Verifiable Rewards (RLVR) are highly robust, unlike their sensitive, preference-optimized counterparts. Based on this insight, we propose leveraging knowledge distillation as a model-centric technique to transplant this robustness from an RLVR teacher to a student model. Experiments on mathematical and general reasoning benchmarks show that the distilled model inherits the teacher's robustness, drastically reducing performance gaps across personas. Specifically, for the Qwen3 family, the average persona stability score increased by +0.23 on MATH500, +0.32 on AIME2024, and +0.18 on MMLU-Pro, while for the Llama-8B models, the improvement reached +0.24, +0.72, and +0.05, respectively.

Introduction

Persona prompting, which involves assigning a role such as "*You are a mathematical expert*", is a widely adopted technique for interacting with Large Language Models (LLMs) (Pataranutaporn et al. 2021; Bai et al. 2022; Liu et al. 2023; Shanahan, McDonell, and Reynolds 2023; Kong et al. 2023; Luo et al. 2024; Luz de Araujo and Roth 2025; Sandwar et al. 2025). Recent studies have converged on a key insight: while an optimal persona for a given task can significantly boost performance, this ideal persona is notoriously difficult to identify. Zheng et al. (2024) demonstrate that automatically finding the best persona is challenging, with selection strategies often performing no better than random. Similarly, Kim, Yang, and Jung (2024) and Luz de Araujo and Roth (2025) frame the persona as a *double-edged sword*; the potential for performance gain is high, but so is the risk of degradation from a poorly chosen prompt.

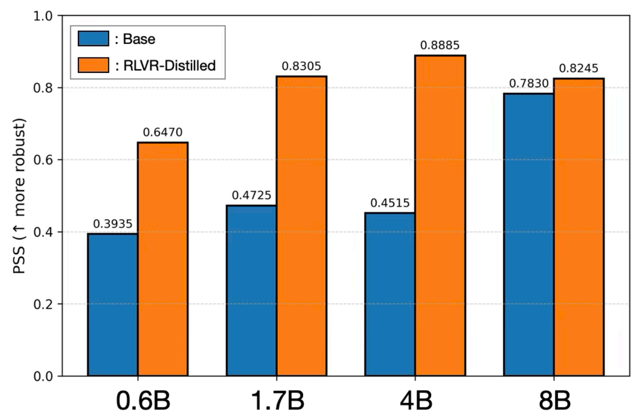

Figure 1: RLVR-Distilled Qwen3 models shows robust performance across 4 various personas than Base models. We average the PSS value from MATH500 and AIME2024 datasets.

Faced with this challenge, current research has focused on prompt-level interventions. While Zheng et al. (2024) primarily diagnose the problem's difficulty without proposing a concrete solution, Kim, Yang, and Jung (2024) offer an ensemble method that generates outputs from both persona-based and neutral prompts, then uses an LLM evaluator to select the better one. However, this approach is fundamentally a reactive, inference-time solution that remains focused on finding the best output from a volatile model, which incurs significant computational overhead from multiple LLM calls.

We argue for a paradigm shift. Instead of asking how to find the optimal persona for a sensitive model, we ask: why not build an inherently robust model that is insensitive to persona variations in the first place? This model-centric approach is far more attractive from a user's perspective, as it promises reliable performance without the need for meticulous prompt engineering or the uncertainty of trial-and-error. We challenge the notion that persona sensitivity is a random or unavoidable phenomenon. We hypothesize and provide empirical evidence that a model's robustness is highly related to its post-alignment process.

Our investigation reveals a clear behavioral divergence: models aligned with Reinforcement Learning with Verifiable

*These authors contributed equally.

Category	Persona Name	Prompt Text
Domain Expert	`math expert`	*You are a mathematical expert with deep knowledge of various mathematical concepts. Solve problems with precision and clarity. Think step by step.*
Out-of-Domain Expert	`carpenter`	*You are a master carpenter with deep knowledge of various woodworking concepts. Solve problems with precision and clarity. Think step by step.*
Baseline (No Persona)	`zero-shot (Z/S) CoT`	*Solve problems with precision and clarity. Think step by step.*
Role-Play Stress Test	`kindergartener`	*You are a kindergartener who just learned their numbers. Solve problems with precision and clarity. Think step by step.*
Instruction Ablation	`kind. w/o Z/S CoT`	*You are a kindergartener who just learned their numbers. Solve problems.*

Table 1: Experimental prompts and conceptual categorization for math tasks.

Rewards (RLVR) (Lample and Charton 2019; Cobbe et al. 2021; Le et al. 2022; Lewkowycz et al. 2022; Lightman et al. 2023), such as the Qwen3 (Yang et al. 2025) and Deepseek R1 (Guo et al. 2025) series, exhibit remarkable robustness to persona variations. In stark contrast, models aligned with Preference Optimization (PO) methods (Lample and Charton 2019; Ouyang et al. 2022; Rafailov et al. 2023; Zhao et al. 2023), like the Llama3 (Grattafiori et al. 2024) and Gemma3 (Team et al. 2025) series, show extreme sensitivity. Based on this insight, we move beyond prompt-level fixes and propose a model-centric solution that leverages knowledge distillation to transplant the intrinsic robustness of an RLVR teacher into a student model, allowing widely used small models to inherit RLVR-style stability without the prohibitive computational cost and training instability of applying RLVR to each model directly. We term this *Robustness Distillation*, and it aims to solve the root cause of the problem by creating a model that is stable by design. As shown in Figure 1, distilled model from RLVR teacher show robust performance across various personas.

To validate our hypothesis, we systematically compare three model archetypes on complex mathematical reasoning benchmarks: (1) RLVR-native models, (2) non-RLVR models, and (3) RLVR-teacher-student distilled models. Our contributions are threefold:

- We systematically analyze the impact of persona prompting across different model scales, families, and task difficulties. Our findings reveal novel insights distinct from previous studies (Kim, Yang, and Jung 2024; Zheng et al. 2024; Luz de Araujo and Roth 2025).
- We provide empirical and theoretical evidence that the post-alignment method (RLVR vs. PO) is a primary determinant of a model's sensitivity to persona prompts.
- We validate our discovery that simple teacher-student distillation is sufficient to transplant an RLVR teacher's robustness to a student model, offering a practical and principled solution to the challenge of prompt sensitivity.

The Conditional Effectiveness of Persona Prompting

In this section, we detail our experimental design for analyzing the interplay between model scale and persona prompts with task difficulty, followed by a presentation of our key findings.

Experimental Setup

Datasets and Models. To comprehensively evaluate the mathematical and logical reasoning capabilities of the models, we utilized two representative benchmark datasets: testsets of MATH500 (Hendrycks et al. 2021) and AIME2024 (Mathematical Association of America. 2024). To isolate the effect of scale while controlling for architectural differences, our experiments were conducted on three primary model families: Qwen3 (0.6B, 1.7B, 4B, 8B, 32B) (Yang et al. 2025), Llama3.1&3.2 (1B, 8B, 70B) (Grattafiori et al. 2024), and Gemma3 (1B, 4B, 12B, 27B) (Team et al. 2025). Here, we choose all LLMs which have undergone their post-training procedure. Especially, Qwen3-32B is post-trained including RLVR and other Qwen3 models are distilled from Qwen3-32B/235B-A22B without any post-training as revealed in Yang et al. (2025). We perform five independent runs across all datasets and models to compute each metric, and report the averaged results.

Prompt Design. To test our core hypotheses, we designed five distinct prompts, conceptually categorized as shown in Table 1. Our design aims to systematically disentangle the effect of the persona frame from the core procedural directive,"Think step by step", a technique we refer to as zero-shot Chain-of-Thought (Z/S CoT) (Kojima et al. 2022). Each prompt is built upon the base instruction,"Solve problems with precision and clarity," to ensure a consistent task definition. The specific role of each prompt category is as follows:
math expert represents the conventional and most intuitive use of persona prompting, where the assigned role is directly aligned with the task domain.
carpenter tests a model's ability to generalize abstract principles. The role is thematically unrelated to mathematics, but still implies a methodical and precise problem-solving process.
zero-shot (Z/S) CoT prompt contains only the procedural instructions. It serves as our primary control group to measure the performance of a persona-less, instruction-driven approach.
kindergartener introduces a persona intentionally mismatched with task complexity. This measures the model's

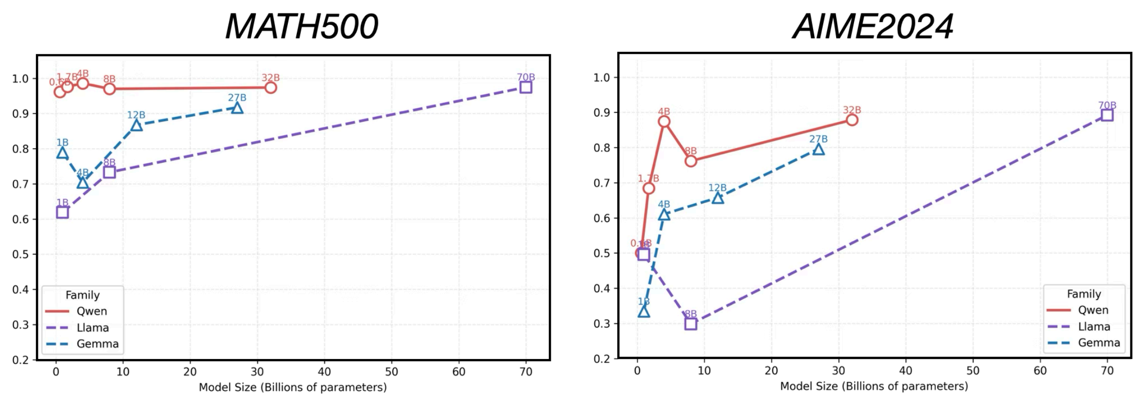

Figure 2: PSS comparison across model families, model scale, and task difficulty.

Model	MATH500 Dataset				AIME2024 Dataset			
	`math expert`	`kindergartener`	`carpenter`	PSS ↑	`math expert`	`kindergartener`	`carpenter`	PSS ↑
Qwen-0.6B	**47.32 ± 1.66**	45.48 ± 0.87	46.20 ± 1.61	0.9611	1.98 ± 1.81	1.98 ± 1.81	**3.96 ± 2.76**	0.5000
Qwen-1.7B	**69.76 ± 1.04**	68.08 ± 1.20	69.08 ± 2.27	0.9759	**12.62 ± 4.34**	11.30 ± 2.99	8.64 ± 1.86	0.6846
Qwen-4B	**79.28 ± 0.66**	78.16 ± 1.08	78.52 ± 0.72	0.9859	**21.33 ± 3.80**	18.66 ± 5.58	19.33 ± 4.94	0.8748
Qwen-8B	80.28 ± 1.27	78.24 ± 0.55	**80.66 ± 1.05**	0.9700	**28.00 ± 6.91**	21.33 ± 2.98	27.33 ± 3.65	0.7618
Qwen-32B	82.08 ± 1.38	**83.84 ± 1.65**	81.64 ± 0.99	0.9738	23.99 ± 8.94	**27.30 ± 2.81**	24.66 ± 2.98	0.8788
PSS Avg.	-	-	-	0.9733	-	-	-	0.7400
Llama-3.2-1B-Ins	**14.80 ± 2.63**	9.16 ± 1.28	13.40 ± 2.09	0.6189	0.66 ± 1.49	**1.33 ± 1.82**	0.66 ± 1.49	0.4962
Llama-3.1-8B-Ins	**43.88 ± 1.00**	32.16 ± 1.99	42.96 ± 1.85	0.7329	**6.66 ± 2.79**	1.99 ± 2.98	3.33 ± 3.33	0.2988
Llama-3.3-70B-Ins	71.76 ± 1.15	70.76 ± 0.91	**72.60 ± 0.68**	0.9747	**24.66 ± 3.80**	22.00 ± 1.82	23.33 ± 6.23	0.8921
PSS Avg.	-	-	-	0.7755	-	-	-	0.5624
Gemma-3-1B-IT	26.42 ± 1.67	25.00 ± 1.15	**31.64 ± 0.55**	0.7901	**2.00 ± 2.98**	0.67 ± 1.49	1.33 ± 1.82	0.3350
Gemma-3-4B-IT	**70.40 ± 0.93**	49.56 ± 1.32	68.08 ± 0.92	0.7040	7.33 ± 4.94	8.00 ± 5.06	**12.00 ± 3.80**	0.6108
Gemma-3-12B-IT	**80.92 ± 0.40**	70.20 ± 0.68	78.80 ± 1.08	0.8675	**25.33 ± 5.05**	16.66 ± 3.34	22.00 ± 6.91	0.6577
Gemma-3-27B-IT	**86.00 ± 0.84**	78.88 ± 1.00	84.16 ± 0.79	0.9172	26.66 ± 5.27	26.00 ± 7.60	**32.66 ± 4.95**	0.7961
PSS Avg.	-	-	-	0.8197	-	-	-	0.5999

Table 2: Performance comparison for personas across models (Qwen3, Llama3, Gemma3) and datasets (MATH500, AIME2024). **Bold** indicates the highest performance within each model-dataset group, while underlined values indicate the lowest. For Llama-3 models, 'Ins' stands for 'Instruct'.

ability to prioritize the procedural "Think step by step" directive over a distracting or non-sensical role.

kind. w/o Z/S CoT is identical to the stress test (`kindergartener`) but crucially removes the "Think step by step" directive. This ablation allows us to directly measure the performance impact of this core cognitive scaffolding, isolating its effect from that of the persona.

Metric. We introduce Persona Stability Score (PSS) which measures how the performance varies according to the persona prompt. We define PSS as follows:

$$\mathrm{PSS}(m,d) = \frac{\min_{p\in\mathcal{P}} a_{m,p,d}}{\max_{p\in\mathcal{P}} a_{m,p,d}} \in [0,1] \tag{1}$$

where m denotes a model, d a dataset, $\mathcal{P}$ the set of personas under comparison, and $a_{m,p,d} \in [0,1]$ the performance (e.g., accuracy) of model m with persona p on dataset d. By construction, larger values (closer to 1) indicate *persona-stable* behavior—performance is similar across personas—whereas smaller values (closer to 0) indicate *persona-sensitive* behavior with large disparities between the best and worst personas.

Comparing Personas Across Models and Datasets

Our analysis of the results presented in Table 2 and Figure 2 reveals a distinct trend: (1) As model size increases, the performance gap between different personas narrows. (2) As task difficulty gets harder, the performance gap between different personas become wider. (3) Model family matters for persona gaps.

Model scale. From Figure 2, we observe a consistent trend that the PSS value increases as the model size grows. For instance, within the Llama3 family on the MATH500 dataset, the PSS rises from 0.6189 at 1B to 0.9747 at 70B, while on AIME2024 it increases from 0.4962 to 0.8921. A similar

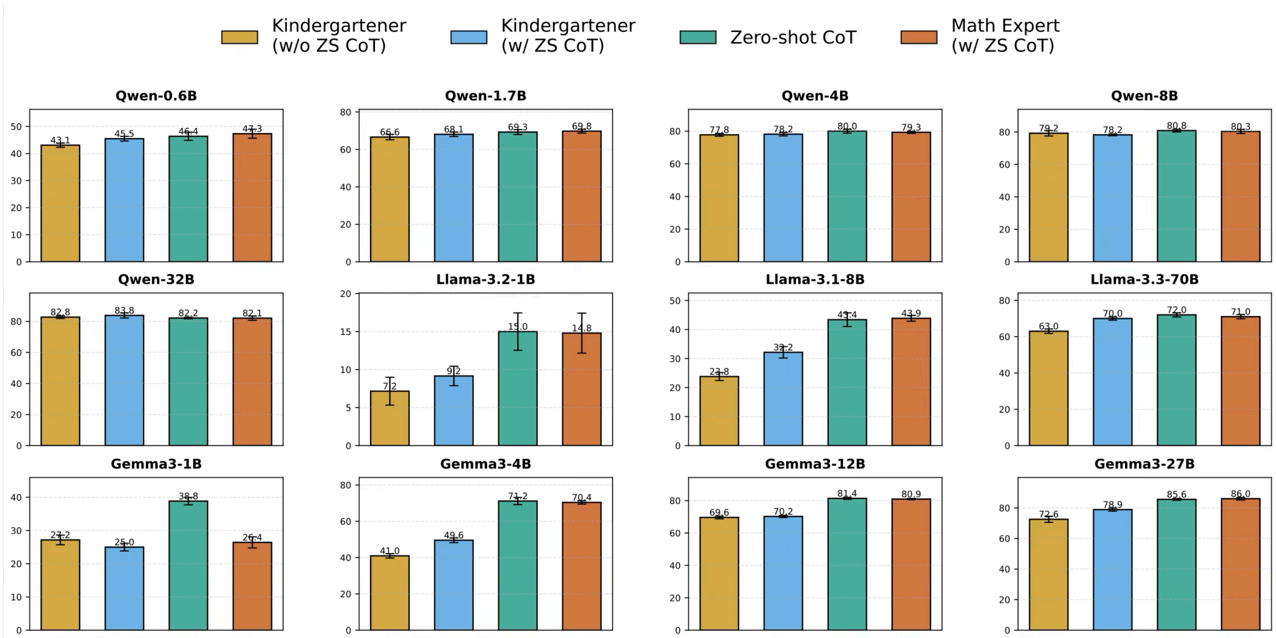

Figure 3: Performance evolution per persona according to the model size on MATH500.

upward pattern is seen in the Gemma3 family, where the PSS improves from 0.7901 at 1B to 0.9172 at 27B on MATH500, and from 0.3350 to 0.7961 on AIME2024. Nevertheless, even for Qwen3, while MATH500 shows little variation due to already high stability, AIME2024 still demonstrates an upward trend in PSS as the model size increases. Excluding this case, all other model families across both datasets clearly demonstrate that increasing model scale reduces the performance disparity across personas.

Task difficulty. When comparing the average PSS values across datasets, we find that the more challenging AIME2024 dataset consistently yields lower PSS values than MATH500. Specifically, the average PSS on MATH500 is 0.9733 for Qwen3, 0.7755 for Llama3, and 0.8197 for Gemma3, whereas on AIME2024 these values drop to 0.7400, 0.5624, and 0.5999, respectively. Taking the overall average across model families further confirms this gap: 0.8562 on MATH500 vs. 0.6341 on AIME2024. This cross-dataset comparison highlights a clear pattern: as the task becomes more difficult, the PSS decreases, meaning that performance becomes more sensitive to the choice of persona. In other words, harder reasoning tasks amplify the performance disparity across personas, indicating that model robustness to stylistic variations is less stable under increased task complexity.

Model family. Across model families, we observe clear differences in PSS. Qwen3 consistently achieves the highest PSS values by a large margin, while Llama3 and Gemma3 show substantially lower and relatively comparable scores. This discrepancy can be directly attributed to the alignment method: unlike Llama3 and Gemma3, Qwen3 is aligned with RLVR (or distilled from RLVR teacher), which inherently encourages robustness to stylistic variations such as persona prompts. Between the two non-RLVR families, Gemma3 exhibits slightly higher PSS than Llama3, which we hypothesize is due to its post-training design. Specifically, Gemma3 incorporates a weighted reward signal that mixes correctness-based feedback into the alignment process (Ramé et al. 2024), whereas Llama3 relies solely on Direct Preference Optimization (Rafailov et al. 2023). This additional signal likely provides Gemma3 with greater stability against persona variation, explaining its modest advantage in PSS over Llama3.

Impact of Inappropriate Personas and the Essential Role of CoT

This section analyzes the results of our instruction ablation study, revealing two key insights: first, that an inappropriate persona actively degrades reasoning performance, and second, that the `Z/S CoT` instruction serves as an essential mechanism for performance preservation, even under adverse conditions.

The Limited Efficacy of Expert Personas. As shown in the Table 2 and Figure 3, while the expert persona `math expert` performs better than out-of-domain personas like `kindergartener` or `carpenter`, it was surprisingly found that its performance is often similar to, or even worse than, the `Z/S CoT` baseline, which is not assigned any persona. This result is consistent with the findings in Zheng et al. (2024); Kim, Yang, and Jung (2024), but it differs from the outcomes discussed in Luz de Araujo and Roth (2025).

CoT as an Essential Performance Safety Net. The critical importance of the CoT instruction is most starkly revealed when it is removed from the already handicapped `kindergartener` persona. Seeing the Figure 3, the comparison between `kindergartener` (persona + CoT) and `kind. w/o Z/S-CoT` (persona only, CoT removed) shows a catastrophic collapse in performance. Without Qwen3 models, in six out of seven models, removing the *"with precision and clarity. Think step by step."* directive resulted in significantly lower scores. The effect was particularly drastic for the Llama models; for instance, Llama-3.1-8B-Instruct dropped from 32.16% to 23.80%, and Llama-3.3-70B-Instruct fell from 70.76% to 63.60%. This finding demonstrates that CoT is not merely a performance enhancement tool but also a crucial performance preservation mechanism. When a model is given a confusing or inappropriate persona, the CoT instruction acts as a "safety net" that constrains the model's reasoning process and prevents a complete failure.

Distilling Robustness: RLVR-trained Model as a Teacher

A central claim of our work is that the *alignment method* largely dictates a model's sensitivity to stylistic (persona) prompts. In particular, we argue that models trained with *Reinforcement Learning with Verifiable Rewards* (RLVR) naturally develop a reasoning policy that is robust to perturbations in persona prompts. In this section, we first formalize RLVR as a policy optimization scheme with an objective verifier, and then provide a simple mechanistic explanation of why this training procedure encourages persona robustness.

Policy Optimization with an Objective Verifier

RLVR is a variant of reinforcement learning tailored for domains (e.g., mathematics) where the quality of a response can be automatically verified. Unlike traditional RLHF, which uses a learned reward model trained on subjective human preferences, RLVR utilizes a deterministic, rule-based verifier to provide the reward signal.

Let $\mathcal{D}$ be a dataset of problem instances x (and, when applicable, persona prompts p), and let π_θ denote the language model policy that generates a response $y = (y_{1:T})$. The key component is the verifier function $V(y)$, defined as:

$$V(y) = \begin{cases} +1 & \text{if the final answer extracted from } y \\ & \text{is correct,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

This verifier provides a sparse, *objective* reward based solely on correctness. The goal of RLVR is to optimize the policy π_θ to maximize the expected reward from this verifier, while regularizing the policy so that it does not deviate too far from a trusted reference policy π_{ref}. This is achieved using a Kullback–Leibler (KL) divergence penalty by maximizing:

$$\mathbb{E}_{\substack{x,p\sim\mathcal{D} \\ y\sim\pi_\theta(\cdot|x,p)}} \Big[V(y) - \beta \cdot D_{\text{KL}}\big(\pi_\theta(y \mid x,p) \,\big\|\, \pi_{\text{ref}}(y \mid x,p)\big) \Big], \tag{3}$$

where $\beta > 0$ controls the strength of KL regularization. We hypothesize that the specific structure of (3), together with the verifier's indifference to style, is responsible for the prompt robustness observed in RLVR-trained models. After the RLVR teacher learns such a style-invariant reasoning policy, we then transfer this robustness into smaller student models via distillation.

Why RLVR Induces Persona Robustness

We provide a brief sketch of why RLVR tends to produce policies that are insensitive to persona prompts, with full derivations deferred to Appendix C.

We decompose a model response into *reasoning tokens* z and *stylistic tokens* s (e.g., tone, persona-specific phrases), and assume a factorized policy

$$y = (z, s), \qquad \pi_\theta(y \mid x,p) = \pi_\theta(z \mid x,p)\, \pi_\theta(s \mid z,x,p), \tag{4}$$

where x is the problem and p is the persona prompt. Intuitively, z encodes the problem-solving trajectory and final answer, while s only controls how this content is wrapped in a particular style. The verifier reward is *style-invariant*: it depends only on whether the reasoning is correct,

$$V(y) = V(z) = \begin{cases} +1 & \text{if } z \text{ is correct,} \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

and is indifferent to the persona tokens s and prompt p. Under this decomposition, the RLVR objective in (3) can be written as:

$$J(\theta) = \mathbb{E}_{x,p,z,s} \Bigg[V(z) - \beta \log \frac{\pi_\theta(z \mid x,p)}{\pi_{\text{ref}}(z \mid x,p)} - \beta \log \frac{\pi_\theta(s \mid z,x,p)}{\pi_{\text{ref}}(s \mid z,x,p)} \Bigg]. \tag{6}$$

Crucially, $V(z)$ does not depend on s, so the style policy $\pi_\theta(s \mid z,x,p)$ is optimized only through the KL penalty. As we show in Appendix X, the KL term is minimized when the style matches the reference, $\pi_\theta^\star(s \mid z,x,p) = \pi_{\text{ref}}(s \mid z,x,p)$, which removes any incentive to move the style away from the reference persona. Plugging this optimal style policy back into (7) yields an *effective* objective that depends only on the reasoning policy:

$$\max_{\pi_\theta}\ \mathbb{E}_{x,p,z} \Bigg[V(z) - \beta \log \frac{\pi_\theta(z \mid x,p)}{\pi_{\text{ref}}(z \mid x,p)} \Bigg], \tag{7}$$

so RLVR pushes the model to change *what it thinks* (the reasoning trajectory z) while keeping *how it speaks* (the style s) anchored to the reference. This explains why RLVR-trained models tend to be robust to persona variations in our experiments.

Distillation Objective: Transferring Robust Policies

After the teacher models are trained with RLVR such as GRPO to obtain robust reasoning policies, smaller student models inherit these behaviors through knowledge distillation. Depending on the family of models, the distillation process follows two distinct yet complementary paradigms: (1) off-policy policy distillation (supervised fine-tuning), and (2) on-policy distillation (distribution matching).

Model Size	Model Version	math expert	kindergartener	carpenter	Perf. Gap (Δ) ↓	PSS ↑
		MATH500				
Qwen3-0.6B	Base	34.84 ± 4.75	28.44 ± 3.43	33.88 ± 3.06	6.40	0.8163
	Distilled	**47.32** ± **1.66**	**45.48** ± **0.87**	**46.20** ± **1.61**	**1.84**	**0.9611**
Qwen3-1.7B	Base	29.56 ± 11.83	18.08 ± 14.23	23.12 ± 5.81	11.48	0.6116
	Distilled	**69.76** ± **1.04**	**68.08** ± **1.20**	**69.08** ± **2.27**	**1.68**	**0.9759**
Qwen3-4B	Base	59.08 ± 7.62	41.84 ± 13.06	63.92 ± 7.40	22.08	0.6546
	Distilled	**79.28** ± **0.66**	**78.16** ± **1.08**	**78.52** ± **0.72**	**1.12**	**0.9859**
Qwen3-8B	Base	67.40 ± 5.43	60.88 ± 2.84	65.92 ± 6.36	6.52	0.9033
	Distilled	**80.28** ± **1.27**	**78.24** ± **0.55**	**80.66** ± **1.05**	**2.42**	**0.9700**
Llama-3.1-8B	Instruct	43.88 ± 1.00	32.16 ± 1.99	42.96 ± 1.85	11.72	0.733
	Deepseek-R1-Distlled	**62.48** ± **2.15**	**61.32** ± **1.35**	**61.04** ± **2.25**	**1.44**	**0.977**
		AIME2024				
Qwen3-0.6B	Base	0.66 ± 1.49	0.00 ± 0.00	0.66 ± 1.49	0.66	0.0000
	Distilled	**1.98** ± **1.81**	**1.98** ± **1.81**	**3.96** ± **2.76**	**1.98**	**0.5000**
Qwen3-1.7B	Base	2.00 ± 2.98	2.00 ± 2.98	0.67 ± 1.49	1.33	0.3333
	Distilled	**12.62** ± **4.34**	**11.30** ± **2.91**	**8.64** ± **1.86**	**3.98**	**0.6846**
Qwen3-4B	Base	14.00 ± 7.23	7.33 ± 5.96	12.00 ± 5.58	6.67	0.5236
	Distilled	**21.33** ± **3.80**	**18.66** ± **5.58**	**19.33** ± **4.94**	**2.67**	**0.8748**
Qwen3-8B	Base	12.67 ± 7.23	8.66 ± 5.05	8.66 ± 3.80	4.00	0.6839
	Distilled	**28.00** ± **6.91**	**21.33** ± **2.98**	**27.33** ± **3.65**	**6.67**	**0.7618**
Llama-3.1-8B	Instruct	3.996 ± 2.79	1.998 ± 2.98	3.330 ± 3.33	2.00	0.500
	Deepseek-R1-Distilled	**14.664** ± **6.63**	**11.998** ± **7.44**	**11.332** ± **9.81**	**3.33**	**0.773**

Table 3: Performance comparison (accuracy % ± std. dev.) of personas on base vs. distilled Qwen3 and Llama3 models for MATH500 (top) and AIME2024 (bottom).

(1) Off-policy distillation: Supervised fine-tuning. In the DeepSeek-R1 framework (Guo et al. 2025), the reasoning capability acquired by the large GRPO-trained teacher is transferred to smaller dense models (e.g., Deepseek-R1-Distill-Llama-8B) purely via supervised fine-tuning. Let $\mathcal{D}_T = \{(x, y_T)\}$ denote the dataset of reasoning trajectories generated and verified by the teacher policy π_T. The student π_S minimizes the cross-entropy between its output and the teacher's verified trajectory (Agarwal et al. 2024):

$$\mathcal{L}_{\text{OffPD}} = -\mathbb{E}_{(x,y_T)\sim\mathcal{D}_T}\left[\sum_{t=1}^{|y_T|} \log \pi_S(y_{T,t} \mid x, y_{T,<t})\right]. \tag{8}$$

This objective treats the teacher's reasoning traces as ground-truth labels, allowing the student to imitate the RLVR-trained reasoning distribution without any reinforcement updates.

(2) On-policy distillation: Distribution matching. Qwen3 (Yang et al. 2025) employs a two-stage pipeline: first performing supervised fine-tuning from teacher generated data, followed by on-policy distillation where the student generates response given prompts (prefix), and the teacher generates tokens given the prefix from students. Then the students is fine-tuned by aligning its token distribution to the teacher's one. Formally, the objective minimizes the KL divergence between teacher and student output distributions (Agarwal et al. 2024):

$$\mathcal{L}_{\text{OnPD}} = \mathbb{E}_{\substack{x\sim\mathcal{D} \\ y_{<t}\sim\pi_S}}\Big[D_{\text{KL}}\big(\pi_T(\cdot|x, y_{<t}) \parallel \pi_S(\cdot|x, y_{<t})\big)\Big]. \tag{9}$$

This formulation enables the student to align with the teacher's robust policy under on-policy supervision, yielding strong-to-weak transfer where small Qwen3 models (e.g., 0.6B–14B) inherit the reasoning stability of large RL-trained teachers (e.g., Qwen3-32B). We compare off-policy distilled model (Deepseek-R1-Distill-Llama-8B) and off-policy distilled model with on-policy distillation (Qwen3-distilled family).

Results of RLVR Distilled vs Non-distilled Models

We report the PSS metric performance between RLVR distilled and non-distilled model utilizing Qwen and Llama model families. We find RLVR-distilled models robust to persona prompts, while non-distilled baselines drift with persona; Appendix B also documents key features of the Qwen3/Llama3 families.

Qwen3 Results. The results of our comparative analysis between the Base and Distilled Qwen3 models, as presented in Table 3, reveal a significant finding. The *Robustness Distillation* process reduces the model's sensitivity to persona variations. The Base models exhibit high sensitivity; the Qwen-4B-Base model, for example in MATH500, shows a massive

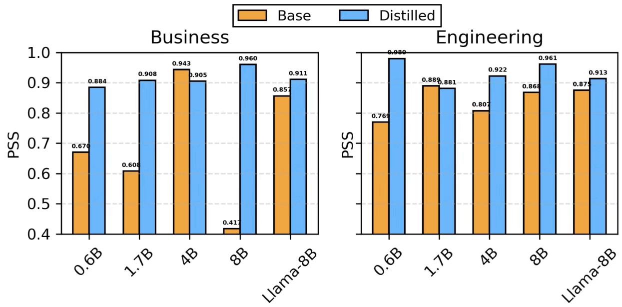

Figure 4: PSS value comparison on base vs. distilled Qwen3 and Llama3 models for MMLU-Pro. Similarly to math benchmarks, we utilized three different personas.

22.08 % point performance gap and 0.6546 PSS value between its best `carpenter` and worst `kindergartener` performing personas. In stark contrast, the Qwen-4B-Distilled model is highly robust, with the same gap shrinking to a mere 1.12 % points and improved PSS value to 0.9859. This pattern—a large sensitivity gap in the Base and Distilled models—is consistent across all scales and datasets, demonstrating that the *Robustness Distillation* from RLVR process is a simple yet key driver of prompt robustness.

Llama3 Result. The distillation process drastically reduces the model's sensitivity to persona variations. The standard Llama-3.1-8B-Instruct model is highly sensitive, exhibiting a massive 11.72 percentage point performance gap and 0.733 PSS value between its best-performing persona (`math expert` at 43.88%) and its worst (`kindergartener` at 32.16%). However, the distilled model demonstrates remarkable robustness, with the performance gap shrinking to a mere 1.44 percentage points and improved PSS value to 0.977. And same tendency is also observed in AIME2024 dataset. This represents a significant reduction in sensitivity, confirming that the teacher's robust policy was successfully transplanted.

Generalization Beyond Math Tasks

RLVR training is primarily applied to the math tasks which can allow verified rewards. This raises the question of whether the robustness to persona prompt can be generalized beyond the math tasks. To address this question, we conduct additional experiments on MMLU-Pro with two categories: Business and Engineering. For each category, we randomly sample 100 problems and run each model 5 times with three personas ("expert", "off-domain professional", "child-like"). Across four Qwen scales (0.6B, 1.7B, 4B, 8B), *Robustness Distillation* consistently increases mean accuracy over personas on both Business and Engineering, and raises PSS from roughly for Base models to for Distilled models as shown in Figure 4. Llama models show the same trend. We fully disclose entire results and used prompts for MMLU-Pro Business and Engineering categories in Appendix A (Table 5 and Table 4).

Related works

Prompt Engineering for Complex Reasoning

The advent of large-scale pre-trained language models has established prompting as the primary paradigm for interacting with these models (Qiu et al. 2020; Mann et al. 2020; Gao, Fisch, and Chen 2020; Lester, Al-Rfou, and Constant 2021). While early work focused on simple template design for few-shot learning, the need for more complex reasoning led to a seminal breakthrough with Chain-of-Thought (CoT) prompting. CoT demonstrated that guiding a model to generate explicit, step-by-step explanations significantly improves its reasoning abilities (Wei et al. 2022; Kojima et al. 2022). This core principle was further refined by subsequent work (Kojima et al. 2022; Wei et al. 2022; Zhou et al. 2022; Yao et al. 2023; Wang et al. 2023; Besta et al. 2024), such as using simple, task-agnostic instructions like "Let's think step by step" or enhancing robustness by sampling multiple reasoning paths (Wang et al. 2022). These studies collectively establish that the instructional directives guiding a model's process are a key determinant of performance. However, this body of literature primarily focuses on the efficacy of specific instructional phrases in isolation. It does not systematically investigate how these instructions interact with other common components of a prompt, most notably the assignment of a persona.

Persona Prompting

Parallel to the development of reasoning-focused techniques, *persona prompting* has become a widely disseminated "best practice" in the practitioner community. This strategy involves assigning the LLM a specific role or identity, such as an expert, a character, or even an object, with the goal of steering its knowledge, tone, and response style (Gu et al. 2023; Shao et al. 2023; Bubeck et al. 2023). The technique is frequently recommended in guides for controlling model behavior and is implicitly used in various applications, from building specialized chatbots to enhancing the safety and alignment of models by instructing them to be helpful and harmless (Bai et al. 2022). The underlying intuition is that a persona helps to contextualize the task, thereby activating a more relevant and effective subspace of the model's vast parametric knowledge. Despite its widespread adoption, persona prompting remains a under-studied phenomenon in a rigorous academic context (Battle and Gollapudi 2024; Shanahan, McDonell, and Reynolds 2023). Most empirical studies that employ personas often do so in conjunction with other detailed instructions.

Conclusion

By transferring verifier-aligned reasoning from a robust teacher into smaller student models, we show that persona-induced performance variance can be substantially reduced without harming overall task accuracy. Our results indicate that robustness to stylistic variation can be learned in a model-centric way—offering a scalable alternative to prompt engineering and pointing toward more stable, human-aligned generation.

Acknowledgments

This work was supported by Center for Applied Research in Artificial Intelligence (CARAI) grant funded by Defense Acquisition Program Administration (DAPA) and Agency for Defense Development (ADD) (UD230017TD). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2019-II190075, Artificial Intelligence Graduate School Support Program(KAIST)).

References

Agarwal, R.; Vieillard, N.; Zhou, Y.; Stanczyk, P.; Garea, S. R.; Geist, M.; and Bachem, O. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *The twelfth international conference on learning representations*.

Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Battle, R.; and Gollapudi, T. 2024. The unreasonable effectiveness of eccentric automatic prompts. *arXiv preprint arXiv:2402.10949*.

Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 17682–17690.

Bubeck, S.; Chandrasekaran, V.; Eldan, R.; Gehrke, J.; Horvitz, E.; Kamar, E.; Lee, P.; Lee, Y. T.; Li, Y.; Lundberg, S.; et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Gao, T.; Fisch, A.; and Chen, D. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Gu, J.; Han, Z.; Chen, S.; Beirami, A.; He, B.; Zhang, G.; Liao, R.; Qin, Y.; Tresp, V.; and Torr, P. 2023. A systematic survey of prompt engineering on vision-language foundation models. *arXiv preprint arXiv:2307.12980*.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Kim, J.; Yang, N.; and Jung, K. 2024. Persona is a double-edged sword: Mitigating the negative impact of role-playing prompts in zero-shot reasoning tasks. *arXiv preprint arXiv:2408.08631*.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Kong, A.; Zhao, S.; Chen, H.; Li, Q.; Qin, Y.; Sun, R.; Zhou, X.; Wang, E.; and Dong, X. 2023. Better zero-shot reasoning with role-play prompting. *arXiv preprint arXiv:2308.07702*.

Lample, G.; and Charton, F. 2019. Deep learning for symbolic mathematics. *arXiv preprint arXiv:1912.01412*.

Le, H.; Wang, Y.; Gotmare, A. D.; Savarese, S.; and Hoi, S. C. H. 2022. Coderl: Mastering code generation through pretrained models and deep reinforcement learning. *Advances in Neural Information Processing Systems*, 35: 21314–21328.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Lewkowycz, A.; Andreassen, A.; Dohan, D.; Dyer, E.; Michalewski, H.; Ramasesh, V.; Slone, A.; Anil, C.; Schlag, I.; Gutman-Solo, T.; et al. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35: 3843–3857.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9): 1–35.

Luo, J.; Luo, R.; Chen, L.; Zhu, L.; Ao, C.; Li, J.; Chen, Y.; Cheng, X.; Yang, W.; Su, J.; et al. 2024. PersonaMath: Enhancing Math Reasoning through Persona-Driven Data Augmentation. *CoRR*.

Luz de Araujo, P. H.; and Roth, B. 2025. Helpful assistant or fruitful facilitator? Investigating how personas affect language model behavior. *PloS one*, 20(6): e0325664.

Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1(3): 3.

Mathematical Association of America. 2024. AIME 2024 Problems.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.

Pataranutaporn, P.; Danry, V.; Leong, J.; Punpongsanon, P.; Novy, D.; Maes, P.; and Sra, M. 2021. AI-generated characters for supporting personalized learning and well-being. *Nature Machine Intelligence*, 3(12): 1013–1022.

Qiu, X.; Sun, T.; Xu, Y.; Shao, Y.; Dai, N.; and Huang, X. 2020. Pre-trained models for natural language processing: A survey. *Science China technological sciences*, 63(10): 1872–1897.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36: 53728–53741.

Ramé, A.; Vieillard, N.; Hussenot, L.; Dadashi, R.; Cideron, G.; Bachem, O.; and Ferret, J. 2024. Warm: On the benefits of weight averaged reward models. *arXiv preprint arXiv:2401.12187*.

Sandwar, V.; Jain, B.; Thangaraj, R.; Garg, I.; Lam, M.; and Zhu, K. 2025. Town Hall Debate Prompting: Enhancing Logical Reasoning in LLMs through Multi-Persona Interaction. *arXiv preprint arXiv:2502.15725*.

Shanahan, M.; McDonell, K.; and Reynolds, L. 2023. Role play with large language models. *Nature*, 623(7987): 493–498.

Shao, Y.; Li, L.; Dai, J.; and Qiu, X. 2023. Character-llm: A trainable agent for role-playing. *arXiv preprint arXiv:2310.10158*.

Team, G.; Kamath, A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ramé, A.; Rivière, M.; et al. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Wang, L.; Xu, W.; Lan, Y.; Hu, Z.; Lan, Y.; Lee, R. K.-W.; and Lim, E.-P. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; and Zhou, D. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T. L.; Cao, Y.; and Narasimhan, K. 2023. Tree of thoughts: Deliberate problem solving with large language models, 2023. *URL https://arxiv. org/abs/2305.10601*, 3: 1.

Zhao, Y.; Joshi, R.; Liu, T.; Khalman, M.; Saleh, M.; and Liu, P. J. 2023. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*.

Zheng, M.; Pei, J.; Logeswaran, L.; Lee, M.; and Jurgens, D. 2024. When" a helpful assistant" is not really helpful: Personas in system prompts do not improve performances of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 15126–15154.

Zhou, D.; Schärli, N.; Hou, L.; Wei, J.; Scales, N.; Wang, X.; Schuurmans, D.; Cui, C.; Bousquet, O.; Le, Q.; et al. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

A: Ablation studies

We show further evidence that RLVR models are more robust to the various persona prompts. Comparative analysis of persona stability on MATH500, where Qwen3 demonstrates strong robustness while Llama3 and Gemma3 exhibit notable sensitivity across personas as shown in Figure 5. In addition, we report the full experiments results of Qwen and Llama models on MMLU-Pro (Business and Engineering) in Table 5 together with prompt designs in Table 4. Also, we add qualitative outcome examples showing how the output tokens differ between the RLVR-distilled and non-distilled (preference-optimized) Llama3-8B models in Tables 8 and 7, given a math problem from the MATH500 dataset, as shown in Table 6.

B: Qwen and Llama models' post training

We briefly describe how the models used in this paper, were post-trained by each company.

Qwen3 Family

To compare between distilled one and non-distilled one, we utilize the open-sourced Qwen3 models.

Qwen3-Base. The Qwen3-Base models represent the foundational models after the completion of a three-stage pre-training process on 36 trillion tokens, which includes *general, reasoning*, and *long context* stages. Crucially, these models have not undergone any of the subsequent post-training alignment procedures, such as the multi-stage reinforcement learning pipeline detailed in the Qwen3 technical report (Yang et al. 2025). Therefore, the Base models have not been exposed to the "Reasoning RL" (RLVR) or "General RL" stages. In the context of our study, these models serve as a critical baseline to evaluate the inherent persona sensitivity of the pre-trained architecture before the introduction of the robustness-inducing policy learned during the RLVR-centric alignment.

Qwen3-Distilled. The publicly released lightweight Qwen3 models (from 0.6B to 8B) are the result of a specialized alignment process called Strong-to-Weak Distillation. As illustrated in the Qwen3 technical report, these smaller models are not trained directly with the complex four-stage RL pipeline. Instead, they are distilled from the flagship teacher models (e.g., Qwen3-32B), which have undergone the RLVR stages. Yang et al. (2025) states this approach is significantly more efficient and effective for smaller models than direct reinforcement learning. Consequently, these Qwen3-Distilled models are the ideal subjects for our analysis, as they represent student models whose final policy has been shaped by the knowledge and robust characteristics of an RLVR-trained teacher.

Llama Family

Llama-3.1-8B-Instruct. The Llama-3.1-8B-Instruct model serves as a representative example of a state-of-the-art model aligned primarily with Preference Optimization (PO) methods. According to its technical report (Grattafiori et al. 2024), the post-training process for the Llama3 family involves several rounds of Supervised Finetuning (SFT) and Direct Preference Optimization (DPO) (Grattafiori et al. 2024). The authors explicitly state that they adopted this relatively simple procedure over more complex reinforcement learning algorithms to maximize training stability and scalability. This PO-aligned model, therefore, provides a clear baseline for evaluating the behavior of a model whose policy has been shaped by directly learning from human preference pairs, making it an ideal subject to test for persona sensitivity.

Llama-3.1-8B-Distilled. To test our central hypothesis, we tested a distilled version of the Llama-3.1-8B model (Guo et al. 2025). This model starts from a Llama-3.1-8B pre-trained checkpoint but undergoes a different alignment process. Instead of the native PO-based instruction fine-tuning like Llama-3.1-8B-Instruct, this model is fine-tuned via knowledge distillation from a powerful, RLVR-trained teacher model Deepseek-R1 (Guo et al. 2025). This student model is trained on the reasoning traces and outputs of the teacher, a process designed to transfer not just the teacher's knowledge but also its robust, objective-driven reasoning policy. This model allows us to create a controlled comparison, isolating the effect of the alignment signal (RLVR-distilled vs. native PO) on the same underlying base architecture.

C: Theoretical Analysis

We now provide a simple theoretical argument for why RLVR tends to produce policies that are insensitive to persona perturbations.

Setup: decomposing reasoning and style. Let x be a problem instance, p a persona prompt, and $y = (y_{1:T})$ a response sampled from the policy $\pi_\theta(y \mid x, p)$. We decompose the response into *reasoning tokens* z and *stylistic tokens* s (e.g., tone, persona-specific phrases), so that

$$y = (z, s), \qquad \pi_\theta(y \mid x, p) = \pi_\theta(z \mid x, p)\, \pi_\theta(s \mid z, x, p). \tag{10}$$

Intuitively, z contains the problem-solving trajectory and final answer, while s encodes how this content is wrapped in a particular persona's style.

Assumption 1 (Weak separability of policy). We assume that RLVR trains the policy towards a state of *weak separability*, where the core reasoning policy is largely disentangled from stylistic variation:

$$\frac{\partial V(z)}{\partial s} \approx 0, \tag{11}$$

i.e., the verifier reward $V(z)$ depends only on the correctness of the reasoning (e.g., the final answer), and is completely indifferent to the style s or the persona prompt p:

$$V(y) = V(z) = \begin{cases} +1 & \text{if } z \text{ is correct,} \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

Full RLVR objective with reasoning–style decomposition. Under the decomposition $y = (z, s)$, the RLVR objective in

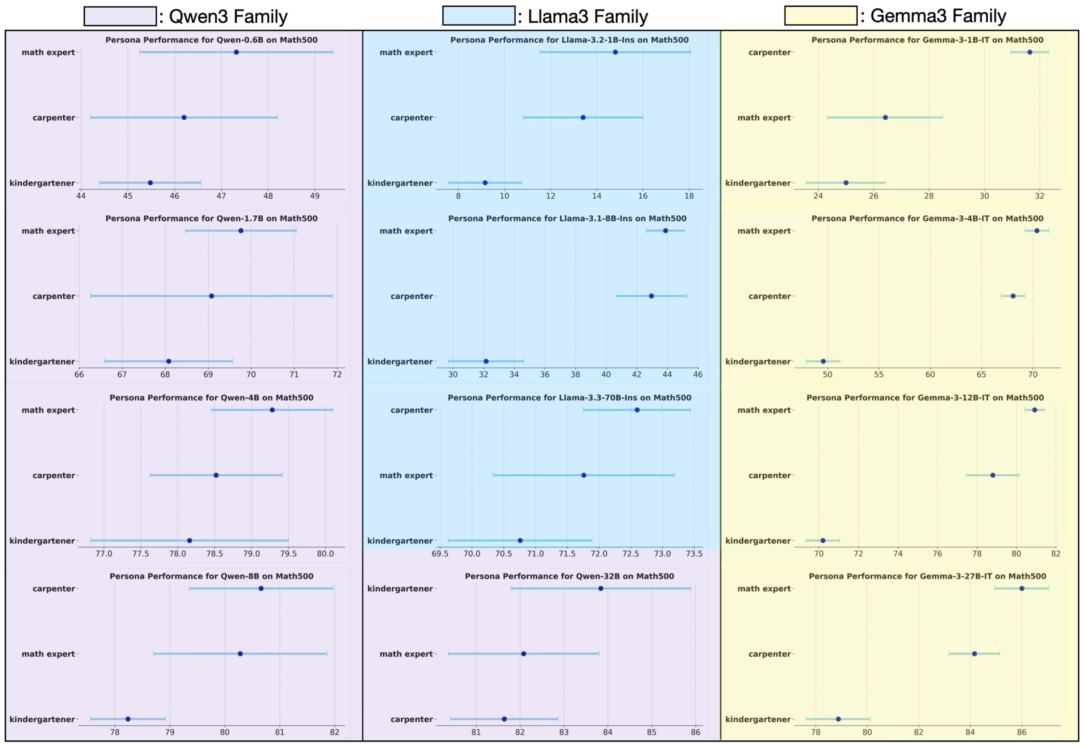

Figure 5: Comparative analysis of persona performance and stability on the Math500 dataset. The plots show mean accuracies and 95% confidence intervals (t-distribution, n=5 trials) across the Qwen, Llama, and Gemma families. The figure highlights a stark difference in sensitivity: the Qwen family (left) exhibits high robustness with largely overlapping intervals, while the Llama (middle) and Gemma (right) families show significant performance gaps between personas.

(3) can be written as:

$$J(\theta) = \mathbb{E}_{x,p,z,s}\Bigg[V(z) - \beta \log \frac{\pi_\theta(z \mid x,p)}{\pi_{\text{ref}}(z \mid x,p)} - \beta \log \frac{\pi_\theta(s \mid z,x,p)}{\pi_{\text{ref}}(s \mid z,x,p)}\Bigg]. \tag{13}$$

Here, $V(z)$ interacts only with the reasoning component z, while the two KL terms regularize the reasoning policy $\pi_\theta(z \mid x,p)$ and the style policy $\pi_\theta(s \mid z,x,p)$ toward their respective reference policies.

Optimal style policy is anchored to the reference. Critically, $V(z)$ does not depend on s, i.e., $\partial V(z)/\partial s \approx 0$. Therefore, when optimizing $J(\theta)$ with respect to the style policy $\pi_\theta(s \mid z,x,p)$, the only relevant term is the KL penalty:

$$\max_{\pi_\theta(s|z,x,p)} \mathbb{E}_{x,p,z,s}\left[-\beta \log \frac{\pi_\theta(s \mid z,x,p)}{\pi_{\text{ref}}(s \mid z,x,p)}\right]. \tag{14}$$

This term is maximized when the KL divergence is minimized ($D_{\text{KL}} \geq 0$), whose minimum value 0 is achieved if and only if the style policy matches the reference:

$$\pi_\theta^\star(s \mid z,x,p) = \pi_{\text{ref}}(s \mid z,x,p). \tag{15}$$

In other words, RLVR has no incentive to move the style away from the reference model; it only encourages the policy to adopt the reference's stylistic behavior, regardless of persona variations.

Reduced effective objective for reasoning. Substituting the optimal style policy $\pi_\theta^\star(s \mid z,x,p)$ back into (6), the style-dependent KL term vanishes. The resulting *effective* objective depends only on the reasoning policy:

$$\max_{\pi_\theta} \mathbb{E}_{x,p,z}\left[V(z) - \beta \log \frac{\pi_\theta(z \mid x,p)}{\pi_{\text{ref}}(z \mid x,p)}\right]. \tag{16}$$

Thus, under weak separability, RLVR training is driven entirely by the verifiable reward over z (correctness of reasoning) and a KL regularizer that keeps the reasoning distribution close to the reference.

Consequence: robustness to persona prompts. Persona prompts p primarily influence the style component s and, to a lesser extent, the reasoning trajectory z. However, because: (i) the reward $V(z)$ is insensitive to s and p, and (ii) the optimal style policy is anchored to the reference model, RLVR allocates its learning capacity to adjusting $\pi_\theta(z \mid x,p)$ in a way that improves correctness uniformly across personas, while keeping stylistic behavior close to π_{ref}. As a result, the reasoning policy becomes largely *persona-invariant*: changing

MMLU-Pro Category	Persona Category	Persona Name	Prompt Text
Business	Domain Expert	`business expert`	*You are a business expert with deep knowledge of various business concepts. Solve problems with precision and clarity. Think step by step.*
	Out-of-Domain Expert	`master chef`	*You are a master chef with deep knowledge of various culinary concepts. Solve problems with precision and clarity. Think step by step.*
	Role-Play Stress Test	`preschooler`	*You are a preschooler who just learned what money is. Solve problems with precision and clarity. Think step by step.*
Engineering	Domain Expert	`engineering expert`	*You are an engineering expert with deep knowledge of various engineering concepts. Solve problems with precision and clarity. Think step by step.*
	Out-of-Domain Expert	`master musician`	*You are a master musician with deep knowledge of various musical concepts. Solve problems with precision and clarity. Think step by step.*
	Role-Play Stress Test	`kindergartener`	*You are a kindergartener who just learned about shapes. Solve problems with precision and clarity. Think step by step.*

Table 4: Experimental prompts and conceptual categorization for business and engineering tasks.

p may slightly alter stylistic tokens s, but has limited effect on the core reasoning trajectory and final answer z. This mechanistic view explains why RLVR-trained models empirically exhibit strong robustness to persona prompt variations, making them ideal teachers for our robustness distillation framework.

D: About Computational Experiments

Model size and budget

We utilized two A6000 GPUs or four A5000 GPUs to serve 3 kinds of models (Llama3, Qwen3, Gemma3) up to 70B size of models. We report the exact size of models used for our paper in Section .

Experimental Setup and Hyperparameters

We discussed about the experimental setup in Section . And we didn't conduct the hyperparameters search because hyperparameters are not in our interest, in which we just find the tendency across the various personas.

E: Scientific Artifact Usage

License for Artifacts

All models and datasets used in this work are publicly available under their respective open licenses. Specifically, we used publicly released language models (e.g., Llama-3.1-8B-Instruct, Qwen3, Gemma3, etc.) and benchmark datasets (e.g., MATH500, AIME2024), each distributed under their original terms of use. We did not create or distribute any new proprietary datasets or models.

Artifact Use Consistent with Intended Use

All models and datasets used in this work were employed strictly within their intended research and evaluation purposes. We adhered to the usage terms of each artifact — including publicly released LLMs (e.g., Llama3, Qwen3, Gemma3) and open-source datasets (e.g., MATH500, AIME24). No artifacts were used in ways inconsistent with their original access conditions, and all data derivatives were created and analyzed solely for research purposes.

F: Parameters for Packages

We used the Hugging Face Transformers library for model loading and inference. All models were accessed via their official repositories (e.g., Qwen/Qwen3-8B, meta-llama/Llama-3.1-8B-Instruct) without additional fine-tuning.

G: AI Usage

We leveraged AI tools (OpenAI's ChatGPT and Google's Gemini) to assist with code generation and elaborating the text. However, all core concepts, methodological designs, and the vast majority of the manuscript's writing were developed and authored directly by the authors.

H: Limitations

Our approach currently assumes access to at least one RLVR-trained "teacher" model, which may not always be available in every domain or deployment setting. Training such verifier-aligned teachers can itself be computationally demanding, even though once obtained they can be reused to distill robustness into many smaller students. In addition, our empirical study focuses on open-source models (e.g., Qwen and LLaMA) and publicly available reasoning benchmarks; while this choice improves reproducibility, it leaves open how well robustness distillation transfers to proprietary frontier models and broader application domains. We view these constraints as limitations of the current evaluation scope rather than of the framework itself, and leave exploring alternative teacher sources and closed-weight models to future work.

Model Size	Model Version	`domain expert`	`role-play stress test`	`out-of-domain expert`	**Perf. Gap** (Δ) $\downarrow$	**PSS** $\uparrow$
		Business				
Qwen3-0.6B	Base	20.0 ± 6.44	16.6 ± 5.13	13.4 ± 2.97	6.6	0.6700
	Distilled	32.8 ± 2.59	30.2 ± 2.77	29.0 ± 3.32	3.8	**0.8841**
Qwen3-1.7B	Base	29.6 ± 14.8	25.4 ± 10.7	18.0 ± 12.7	11.6	0.6081
	Distilled	51.8 ± 4.15	52.2 ± 1.79	47.4 ± 3.85	4.8	**0.9080**
Qwen3-4B	Base	52.4 ± 9.71	52.6 ± 7.16	49.6 ± 13.0	3	**0.9430**
	Distilled	71.6 ± 1.82	71.0 ± 2.24	64.8 ± 2.95	6.8	0.9050
Qwen3-8B	Base	52.0 ± 4.06	52.2 ± 6.53	43.4 ± 6.66	8.8	0.4173
	Distilled	74.2 ± 3.03	75.6 ± 1.67	72.6 ± 4.10	3	**0.9603**
Llama3.1-8B	Instruct	46.0 ± 2.12	43.2 ± 4.21	39.4 ± 2.41	6.6	0.8565
	Deepseek-R1-Distilled	42.8 ± 7.56	40.0 ± 8.09	39.0 ± 7.03	3.8	**0.9112**
		Engineering				
Qwen3-0.6B	Base	13.0 ± 1.22	12.4 ± 1.14	10.0 ± 3.08	3	0.7692
	Distilled	29.6 ± 2.79	29.0 ± 3.89	29.0 ± 4.47	0.6	**0.9797**
Qwen3-1.7B	Base	18.0 ± 5.66	17.0 ± 3.74	16.0 ± 5.29	2	**0.8889**
	Distilled	34.0 ± 3.46	38.6 ± 5.50	36.2 ± 2.95	4.6	0.8808
Qwen3-4B	Base	38.4 ± 6.58	32.4 ± 11.93	31.0 ± 6.52	7.4	0.8073
	Distilled	51.4 ± 1.95	48.8 ± 1.09	47.4 ± 4.77	4	**0.9222**
Qwen3-8B	Base	38.4 ± 2.07	39.4 ± 1.52	34.2 ± 4.82	5.2	0.8680
	Distilled	56.6 ± 2.70	54.4 ± 1.52	54.8 ± 3.70	2.2	**0.9611**
Llama3.1-8B	Instruct	27.4 ± 1.14	28.8 ± 5.22	25.2 ± 6.46	3.6	0.8750
	Deepseek-R1-Distilled	26.0 ± 1.87	27.6 ± 3.36	25.2 ± 2.39	2.4	**0.9130**

Table 5: Performance comparison (accuracy % $\pm$ std. dev.) of personas on base vs. distilled Qwen3 and Llama3 models for Business (top) and Engineering (bottom).

Problem	**Text**
`Given Problem`	**Define** $$p = \sum_{k=1}^{\infty} \frac{1}{k^2} \quad \text{and} \quad q = \sum_{k=1}^{\infty} \frac{1}{k^3}.$$ **Find** a way to write $$\sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \frac{1}{(j+k)^3}$$ in terms of p and q.

Table 6: Example Problem.

Response	Text
`child-like response`	I don't know what that means. My teacher told me to use numbers 1–10 to help us learn math. Let me try. We need to find a way to write a big sum in terms of p and q. First, we need to find p and q. To find p, we need to add $\frac{1}{1^2}$, $\frac{1}{2^2}$, $\frac{1}{3^2}$, and so on. $$p \;=\; \frac{1}{1^2}+\frac{1}{2^2}+\frac{1}{3^2}+\frac{1}{4^2}+\frac{1}{5^2}+\frac{1}{6^2}+\frac{1}{7^2}+\frac{1}{8^2}+\frac{1}{9^2}+\frac{1}{10^2}+\cdots$$ To find q, we need to add $\frac{1}{1^3}$, $\frac{1}{2^3}$, $\frac{1}{3^3}$, and so on. $$q \;=\; \frac{1}{1^3}+\frac{1}{2^3}+\frac{1}{3^3}+\frac{1}{4^3}+\frac{1}{5^3}+\frac{1}{6^3}+\frac{1}{7^3}+\frac{1}{8^3}+\frac{1}{9^3}+\frac{1}{10^3}+\cdots$$ Now, we need to find $$\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(j+k)^3}.$$ This is a big sum. I will try to break it down. Let's say $j=1$. Then we need to find $\sum_{k=1}^{\infty}\frac{1}{(1+k)^3}$, which is a sum like $\frac{1}{2^3}, \frac{1}{3^3}, \frac{1}{4^3}, \ldots$ Let's say $j=2$. Then we need $\sum_{k=1}^{\infty}\frac{1}{(2+k)^3}$, which is a sum like $\frac{1}{3^3}, \frac{1}{4^3}, \frac{1}{5^3}, \ldots$ We can see a pattern. When $j=1$, we have the same fractions as q but we need to add $\frac{1}{2^3}$ to every fraction. When $j=2$, we have the same fractions as q but we need to add $\frac{1}{3^3}$ to every fraction. So, we can say that $$\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(j+k)^3}$$ is like q, but we need to add $\frac{1}{2^3}, \frac{1}{3^3}, \frac{1}{4^3}, \ldots$ to every fraction in q. The sum of $\frac{1}{2^3}, \frac{1}{3^3}, \frac{1}{4^3}, \ldots$ is equal to $q-\frac{1}{1^3}$. So the double sum is $q+(q-\frac{1}{1^3})$. We know that $q=\sum_{k=1}^{\infty}\frac{1}{k^3}$ and $\frac{1}{1^3}=\frac{1}{1^3}$. So the double sum is $2q-\frac{1}{1^3}$. We know that $p=\sum_{k=1}^{\infty}\frac{1}{k^2}$ and $\frac{1}{1^3}=\sqrt{\frac{1}{1^2}}$. So the double sum is $2q-\sqrt{p}$. But we need to write it in terms of p and q. We can see that $\sqrt{p}$ is like $\sqrt{\sum_{k=1}^{\infty}\frac{1}{k^2}}$, so $\sqrt{p}$ is like $\sqrt{p}$. Therefore, $$\boxed{2q-\sqrt{p}}.$$

Table 7: Child-like response with Llama-3.1-8B-Instruct (non-RLVR distilled) model given kindergartener prompt.

Response	Text
`robust response`	Alright, so I have this math problem here where I need to find a way to express the double sum $\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(j+k)^3}$ in terms of p and q, where p is the sum of reciprocals of squares and q is the sum of reciprocals of cubes. Hmm, okay, let's break this down step by step. First, let me recall what p and q are. p is $\sum_{k=1}^{\infty}\frac{1}{k^2}$, which I know converges to $\frac{\pi^2}{6}$, but maybe I don't need the numerical value right now. Similarly, q is $\sum_{k=1}^{\infty}\frac{1}{k^3}$, which converges to $\zeta(3)$, but again, maybe the exact value isn't necessary here. The problem is asking about a double sum: $\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(j+k)^3}$. So, this is summing over all pairs of positive integers j and k, adding $\frac{1}{(j+k)^3}$ for each pair. I need to express this in terms of p and q. Let me think about how to approach this. Maybe I can change variables or find a way to relate this double sum to the single sums p and q. I remember that sometimes with double sums, especially infinite ones, it's helpful to switch the order of summation or to use some form of series expansion. Wait, another idea: perhaps I can express $\frac{1}{(j+k)^3}$ in terms of integrals since integrals can sometimes turn reciprocals of powers into something more manageable. For example, I know that $\frac{1}{n^3}$ can be written as an integral, but I'm not sure if that will help here. Maybe not directly. Alternatively, perhaps I can consider expanding $\frac{1}{(j+k)^3}$ as a power series or use generating functions. Let me think: $\frac{1}{(j+k)^3}$ can be expressed as an integral of exponentials or something similar. Wait, maybe a better approach is to consider swapping the order of summation. So, currently, the double sum is over all j and k. Maybe if I fix $n = j + k$, I can rewrite the double sum as a sum over n and then sum over j and k such that $j + k = n$. Let me try that. Set $n = j + k$. Then for each n, j can range from 1 to $n-1$, and $k = n - j$. So, the double sum becomes: $$\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(j+k)^3} = \sum_{n=2}^{\infty}\sum_{j=1}^{n-1}\frac{1}{n^3}.$$ The inner sum is just $\sum_{j=1}^{n-1}1 = n-1$. Therefore, $$\sum_{n=2}^{\infty}\frac{n-1}{n^3} = \sum_{n=2}^{\infty}\left(\frac{1}{n^2}-\frac{1}{n^3}\right).$$ Splitting the sums and adjusting indices using $p = \sum_{n=1}^{\infty}\frac{1}{n^2}$ and $q = \sum_{n=1}^{\infty}\frac{1}{n^3}$, we get $$\sum_{n=2}^{\infty}\frac{1}{n^2} - \sum_{n=2}^{\infty}\frac{1}{n^3} = (p-1)-(q-1) = p-q.$$ So the double sum simplifies to $\boxed{p-q}$. **Final Answer.** The double sum can be expressed in terms of p and q as $\boxed{p-q}$.

Table 8: Robust response from DeepSeek-R1-Distill-Llama-8B (RLVR-distilled) model given kindergartener prompt.