Extended Abstract

Motivation This project aims to improve the capabilities of small language models Qwen 2.5 0.5B Base model) through reinforcement learning (RL) and knowledge distillation. We investigate whether preference-based optimization methods such as Direct Preference Optimization (DPO) and Reward Learning with Online Optimization (RLOO) can enhance small model performance on instruction following and math reasoning tasks. Additionally, we examine the effectiveness of distilling from medium-sized instruction-tuned teacher models (Qwen 2.5 Math 1.5B and 3B Instruct) and assess how teacher size influences student performance, particularly on mathematical reasoning benchmarks. This study is motivated by the broader goal of understanding the trade-offs between model efficiency and performance, evaluating the role of instruction-tuned models as strong teachers, and revealing scenarios where RL methods may or may not benefit small-scale models compared to distillation approaches.

Method We structure the study around two core tasks: instruction following and math reasoning. Starting with supervised fine-tuning (SFT) on task-specific datasets, we apply DPO to optimize preference alignment in instruction-following and RLOO to improve reward-guided reasoning in math tasks. In parallel, we explore offline distillation by generating teacher responses from larger models and training the 0.5B student to mimic these outputs. We then compare student models trained via distillation versus those trained with RLOO alone across standardized evaluation benchmarks.

Implementation All pipelines are built using PyTorch and Hugging Face Transformers, with inference accelerated via vLLM. RLOO leverages vLLM's distributed sampling for response generation and a learned reward model for gradient updates. Evaluation follows two strategies: instruction responses are scored by the Llama 3.1 Nemotron-70B reward model, while math reasoning outputs are evaluated by a verifier using rule-based correctness checks.

Results Distillation consistently improves performance in both instruction following and math reasoning. In contrast, DPO fails to outperform SFT, possibly due to the influence of negative samples. RLOO improves correctness in math tasks but slightly underperforms compared to distilled models. Prompt format significantly affects SFT performance. We find that system prompts yield better alignment than single-turn prompts.

Discussion The results highlight the promise of distillation from strong supervised teachers, especially when RL-finetuning does not yield additional gains. DPO's limitations may stem from the inclusion of poor-quality responses during training. Limitations of distillation include the use of only supervised teacher models, fixed student size, and limited teacher diversity. Future work should explore RL-finetuned teachers, larger student scales, and ensemble distillation. Overall, our results highlight the nuanced trade-offs between imitation-based and reward-based training strategies for scaling small LLMs.

Conclusion Our results suggest that while prompt-aware SFT can produce strong baselines, DPO may introduce instability, likely due to harmful gradients from negative samples. RLOO, on the other hand, effectively leverages reward signals for structured tasks like math reasoning. Distillation proves to be a stable and scalable alternative, offering moderate gains without the overhead of online RL. However, its effectiveness appears bounded by teacher diversity and student capacity. These findings highlight the importance of task characteristics in choosing between RL and imitation-based alignment strategies.

Can Small LLMs Learn from Medium Ones?

Charlie Jiang
Department of Bioengineering
Stanford University
cjiang3@stanford.edu

Yixing Jiang
Department of Biomedical Data Science
Stanford University
jiang6@stanford.edu

Yi Jing
Department of Statistics
Stanford University
jingi@stanford.edu

Abstract

Large language models (LLMs) achieve strong performance on instruction-following and reasoning tasks, but their size imposes limitations on cost-effective deployment. This work investigates whether a small LLM (Qwen 2.5 0.5B) can be improved via reinforcement learning and knowledge distillation from medium-sized instruction-tuned models. We study two tasks—instruction following and math reasoning—and evaluate the effects of supervised fine-tuning (SFT), Direct Preference Optimization (DPO), Reward Learning with Online Optimization (RLOO), and distillation from frozen 1.5B and 3B teacher models. We find that while prompt design plays a critical role in instruction-following SFT, DPO does not significantly improve over strong SFT baselines. In contrast, RLOO leads to notable gains in math reasoning. Distillation from teacher models yields moderate improvements, with performance influenced by teacher size. Our results highlight when and how small models benefit from larger ones, and provide practical guidance for aligning small LLMs efficiently.

1 Introduction

Large language models (LLMs) have achieved impressive results in instruction following and reasoning tasks Ouyang et al. (2022); Chung et al. (2024); Lewkowycz et al. (2022); Brown et al. (2020). However, their large parameter counts make them expensive to train and deploy, especially in real-world or edge settings Kai et al. (2023); Alizadeh et al. (2024). This motivates a central question: *Can small LLMs acquire stronger reasoning and alignment capabilities by learning from medium-sized models through reinforcement learning or knowledge distillation?*

In this work, we examine whether a small-scale model (Qwen 2.5 0.5B Base) can be effectively enhanced through two major strategies: reinforcement learning and knowledge distillation. We focus on two distinct tasks—instruction following and math reasoning—and structure our study accordingly.

As part of the project requirement, we first fine-tune the 0.5B model using supervised fine-tuning (SFT) on relevant datasets. For instruction following, we explore different prompt formats and observe that prompt engineering, including the use of system prompts, plays a critical role in determining SFT performance. On top of SFT, we apply Direct Preference Optimization (DPO) Rafailov et al. (2023) for instruction following and Reward Learning with Online Optimization (RLOO) Bai et al. (2022) for math reasoning. Both DPO and RLOO are initialized from the respective SFT checkpoints. While RLOO successfully enhances math reasoning capabilities, we find that DPO does not consistently

outperform well-optimized SFT in instruction following, suggesting that reinforcement learning may not always provide gains beyond prompt-aware supervision Gao et al. (2023).

Our extension explores knowledge distillation from instruction-tuned medium-sized teacher models (Qwen 2.5 Math 1.5B and 3B Instruct) Bai et al. (2023). For the math reasoning task, we generate teacher responses and train the 0.5B student model on these outputs, following initial SFT. We further investigate the effect of applying RLOO after distillation to assess the complementary benefits of offline and online learning. This extension allows us to study how teacher size, alignment quality, and downstream fine-tuning interact in shaping the performance of small models. This extension allows us to study whether learning from teacher models can improve performance, and how teacher size shapes the learning outcomes of small models.

Our results demonstrate that both RL and distillation can improve small model performance, but their effectiveness is task-dependent. Instruction following benefits more from prompt engineering and SFT, while math reasoning sees stronger gains from reward optimization via RLOO. Knowledge distillation from larger models provides moderate improvements, though not as strong as RL in the math domain. These findings offer practical insights into when and how small models can best learn from their larger counterparts.

2 Related Work

Alignment of Language Models As language models scale in size and capability, aligning them with human intent has become a central objective. Early approaches rely on supervised fine-tuning (SFT) on instruction datasets to adapt pretrained models for downstream tasks Sanh et al. (2022); Wei et al. (2021). More recent methods leverage reinforcement learning from human feedback (RLHF) to directly optimize for human preferences Ouyang et al. (2022). However, RLHF is often resource-intensive and unstable. To address this, Direct Preference Optimization (DPO) has been proposed as a more efficient and stable alternative that sidesteps reward model training while preserving alignment benefits Rafailov et al. (2023). For task-specific feedback like math reasoning, Reward Learning with Online Optimization (RLOO) offers a fine-grained signal by jointly updating the reward model and policy Bai et al. (2022).

Our work builds on these developments by applying DPO and RLOO to a small model (0.5B), initialized from SFT, and evaluating their effectiveness in instruction following and mathematical reasoning tasks. In contrast to prior work which focuses on large models, we examine whether such alignment techniques remain effective for compact models.

Instruction Tuning and Mathematical Reasoning Instruction-tuned models such as FLAN-T5 Chung et al. (2024), InstructGPT Ouyang et al. (2022), and the Qwen series Bai et al. (2023) have shown that high-quality instructions substantially improve task generalization. For mathematical reasoning, models like Minerva Lewkowycz et al. (2022) and Qwen2.5 Math Yang et al. (2024) fine-tune on domain-specific datasets to improve symbolic and numerical reasoning.

Unlike these prior efforts targeting large models, our work studies the same alignment paradigms at the small model scale and investigates how reinforcement learning and distillation affect their performance on instruction-following and reasoning tasks.

Knowledge Distillation for Language Models Knowledge distillation is widely used to compress large models into smaller students by training on softened outputs or latent representations Hinton et al. (2015); Sanh et al. (2019). In the LLM context, instruction-tuned teacher models have been shown to produce high-quality outputs for supervision Wei et al. (2024). However, prior work rarely compares distillation directly with reinforcement learning, or isolates the role of teacher size.

Our extension addresses this gap by distilling from medium-sized (1.5B and 3B) instruction-tuned teachers into a 0.5B student. We compare distillation directly with RLOO to better understand their relative impact on math reasoning performance under constrained model capacity.

3 Method

This section describes our experimental framework for aligning a small-scale language model (Qwen 2.5 0.5B) using reinforcement learning and knowledge distillation. As shown in Figure 1, our

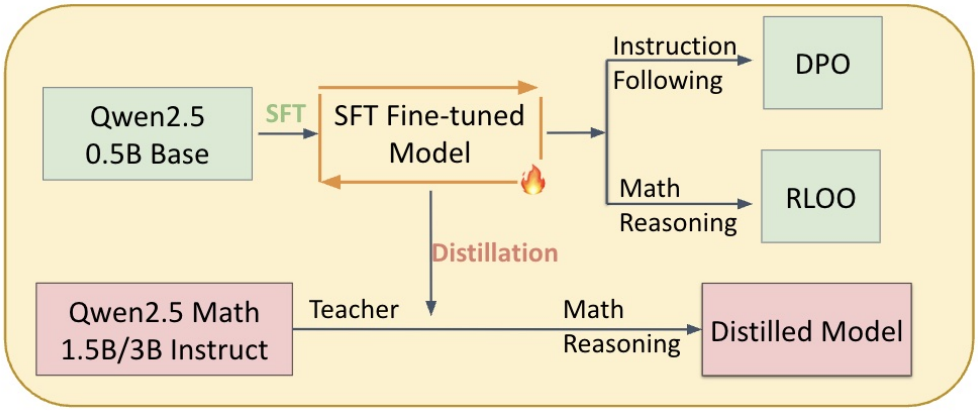

Figure 1: Training pipeline overview. We first fine-tune the Qwen2.5 0.5B Base model with SFT. For instruction following, we apply DPO. For math reasoning, we either apply RLOO or distill from Qwen2.5 Math Instruct teacher models (1.5B/3B).

approach begins with supervised fine-tuning (SFT), followed by either reinforcement learning via DPO/RLOO or offline distillation from medium-sized teacher models.

3.1 Model Setup

We use the Qwen 2.5 0.5B Base model as the foundation for all training pipelines. Supervised fine-tuning (SFT) is applied to this base model to create task-specific initializations. For reinforcement learning, both DPO and RLOO are initialized from their respective SFT checkpoints. In the distillation setup, the student model is also initialized from the SFT checkpoint, while the Qwen 2.5 Math Instruct models (1.5B and 3B) serve as frozen teachers. These models are used to generate target responses for training but are not updated during the process.

All models are trained and compared on instruction-following or math reasoning tasks. Further details on evaluation metrics and results are presented in Section 4.

3.2 Supervised and Reinforcement Learning Fine-Tuning

All pipelines begin with supervised fine-tuning. For instruction following, we fine-tune the 0.5B base model using the SmolTalk dataset. We experiment with prompt format variations (e.g., single-turn prompts, system prompts) and find that prompt design significantly affects performance. For math reasoning, we fine-tune on the Countdown dataset, which involves structured arithmetic tasks. Further details about datasets and experimental setups are provided in Section 4.

After SFT, we apply reinforcement learning based on task type:

Direct Preference Optimization (DPO) For instruction following, we use DPO Rafailov et al. (2023), which uses pairwise human preference data. Given a preferred output y^+ and a rejected one y^- for the same prompt x, the DPO loss maximizes the following log-probability ratio:

$$\mathcal{L}_{\text{DPO}} = -\log\left(\frac{\exp\left(\beta \log \pi_\theta(y^+|x)\right)}{\exp\left(\beta \log \pi_\theta(y^+|x)\right) + \exp\left(\beta \log \pi_\theta(y^-|x)\right)}\right)$$

where π_θ is the policy model and β is a temperature parameter that controls the sharpness of the preference.

Reward Learning with Online Optimization (RLOO) For math reasoning, we apply RLOO Bai et al. (2022), which jointly optimizes a reward model r_ϕ and a policy π_θ. Given a trajectory (i.e., generated solution) y conditioned on prompt x, the policy is optimized to maximize the expected reward:

$$\mathcal{L}_{\text{RL}} = -\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[r_\phi(x, y)]$$

The reward model r_ϕ is trained using correctness-labeled outputs with a binary objective, and both r_ϕ and π_θ are updated iteratively during training. We use Countdown task scores to guide the reward signal.

3.3 Knowledge Distillation

As an extension, we study offline knowledge distillation. For each prompt in the Countdown dataset, we use the frozen Qwen2.5 Math Instruct models (1.5B and 3B) to generate response outputs. The student model is then trained to match these responses using the standard maximum likelihood objective:

$$\mathcal{L}_{\text{distill}} = -\sum_{t=1}^{T} \log \pi_\theta(y_t | y_{<t}, x)$$

where y is the teacher output sequence and x is the prompt.

This approach enables the small model to inherit reasoning behavior from a stronger teacher without additional reward signals or preference data.

4 Experiments

4.1 Task Descriptions

We evaluate our methods on two core tasks that require different reasoning and alignment capabilities:

Instruction Following. This task involves generating helpful, coherent, and well-structured responses to a wide variety of natural language instructions. It reflects the model's alignment with user intent and sensitivity to prompt formatting.

Math Reasoning. This task tests symbolic reasoning capabilities through a numerical target-matching game called Countdown. Given a set of integers and a target value, the model must output a valid arithmetic expression that computes to the target. The task requires planning, verification, and structured output.

4.2 Datasets

We use distinct datasets aligned with each learning objective in our pipeline.

Instruction Following. For supervised fine-tuning (SFT), we use the SmolTalk dataset Ben Allal et al. (2025), a high-quality collection of GPT-4o chat responses. We adopt a filtered subset from HuggingFaceTB/smol-smoltalk [1]. For preference-based reinforcement learning (DPO), we use the **UltraFeedback** dataset Dubois et al. (2024), which provides binary preferences over model completions. Prompts are drawn from the shared `train_sft` split, and `test_gen` is used for evaluation.

Math Reasoning. For verifier-based learning tasks, we use the WarmStart dataset Gandhi et al. (2025), which presents structured arithmetic problems requiring symbolic reasoning. A rule-based reward function checks the correctness of generated responses. For RLOO and distillation, we use the TinyZero dataset Pan et al. (2025), which provides curated reasoning trajectories demonstrating backtracking and verification strategies.

Table 1 summarizes the datasets used for each task and training stage.

4.3 Experimental Setup

We evaluate models on two tasks: instruction following and math reasoning. For each task, we begin with supervised fine-tuning (SFT) on task-specific datasets, followed by either reinforcement learning (DPO or RLOO) or knowledge distillation. All models are initialized from the Qwen 2.5 0.5B Base model.

[1] `https://huggingface.co/datasets/HuggingFaceTB/smol-smoltalk`

Task	Dataset	Size (Train/Test)	Training
Instruction Following	SmolTalk	460k / 24k	SFT
	UltraFeedback	61k / 1k	DPO
Math Reasoning	Warmstart	1k	SFT
	TinyZero	490k / 1k	RLOO

Table 1: Overview of datasets used for training and evaluation.

Supervised Fine-Tuning (SFT). SFT is performed on the SmolTalk dataset using PyTorch FSDP with hybrid sharding. Inputs are truncated to 1024 tokens, and only the assistant response is used for supervision. Training uses bfloat16 precision, AdamW optimizer (learning rate 7×10^{-6}, weight decay 0.1), and linear learning rate decay to 7×10^{-7}. We train with a batch size of 32, gradient accumulation of 8, and distributed data loaders across 8 A100 GPUs.

Direct Preference Optimization (DPO). DPO is applied to the SFT checkpoint using UltraFeedback, which contains prompt pairs labeled by preference. Training uses a batch size of 16, gradient accumulation of 4, and 3.8k steps. The policy and frozen reference models are both initialized from SFT. Optimization is performed with AdamW (learning rate 1×10^{-6}, weight decay 0.01, $\beta = 0.1$).

Reward Learning with Online Optimization (RLOO). RLOO is used for math reasoning with the Countdown dataset. The policy model is initialized from the DPO checkpoint. Sampling is done via vLLM (16 completions per prompt), scored by a frozen reward model trained separately. The policy is updated via AdamW using the same hyperparameters as DPO. We synchronize weights across sampling and training using 'torch.distributed'.

Knowledge Distillation. We distill teacher outputs from Qwen 2.5 Instruct (1.5B and 3B) models on Countdown prompts. Prompts include '<think>' and '<answer>' tags to encourage structured reasoning. Generation is performed with vLLM using top-p sampling (p=0.8, temperature=0.7). The student (0.5B) is fine-tuned on these teacher-generated responses using standard SFT.

4.4 Evaluation Metrics

Instruction Following. To evaluate instruction-following performance, we adopt a parametric reward model: LLaMA 3.1 Nemotron-70B Reward Model [2]. This model assigns a scalar reward score to a prompt-response pair, where a higher score indicates better alignment and response quality for that specific prompt. However, because absolute scores across different prompts are not directly comparable, we follow a win-rate-based evaluation strategy.

Given a set of evaluation prompts, we generate responses using both the trained model and a reference model (Qwen 2.5 0.5B Instruct). For each prompt, we compute the Nemotron reward scores for both responses. We assign a binary label: 1 if the trained model receives a higher reward, and 0 otherwise. The final win rate is computed as the average of these binary labels across all prompts:

$$\text{Win Rate} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1} \left[R^{(i)}_{\text{trained}} > R^{(i)}_{\text{ref}} \right]$$

where $R^{(i)}_{\text{trained}}$ and $R^{(i)}_{\text{ref}}$ are Nemotron reward scores for the trained and reference models on prompt i, respectively.

Math Reasoning. For evaluating math reasoning performance on the Countdown dataset, we follow the evaluation protocol from TinyZero Pan et al. (2025). Specifically, we use a two-stage reward function:

1. Format score: Binary indicator of whether the model provides an output in a valid format.
2. Verification score: Binary indicator of whether the response contains a correct final answer to the arithmetic problem.

[2] `https://huggingface.co/nvidia/Llama-3.1-Nemotron-70B-Reward`

The overall task accuracy is computed as the product of the two components, averaged across all evaluation examples.

5 Results

5.1 Quantitative Evaluation

We evaluate performance on two core tasks: instruction following and math reasoning. For instruction following, we report win-rates using the Nemotron 70B reward model, comparing responses across model variants on identical prompts. For math reasoning, we measure exact answer correctness on the Countdown dataset using a verifier that checks both reasoning format and final numerical answer.

As shown in Table 5.1, instruction following performance increases significantly through prompt design and supervised tuning alone. Starting from a baseline SFT model (win-rate 0.695), introducing structured single-turn prompts yields a large gain (+12.0%), and further gains are achieved through hyperparameter tuning and careful prompt formatting. The best instruction-following result (0.905 win-rate) is obtained with a hand-crafted system-level prompt. These improvements illustrate that prompt optimization—when combined with a high-quality supervised dataset—can encode useful behavioral priors without the need for expensive RL steps. This is especially effective for instruction following, where the output space is highly structured and model alignment is sensitive to prompting.

However, we observe that applying Direct Preference Optimization (DPO) to a strong SFT model actually hurts performance (win-rate drops to 0.395 against Qwen-0.5B-Instruct). We hypothesize that this degradation arises from the inclusion of negative samples during DPO training, which may introduce harmful gradients that overwrite helpful behaviors learned during SFT. Additionally, since DPO training is guided by implicit reward differences between response pairs, its effect can be unstable on tasks where preferences are subtle or dataset signal is noisy.

For math reasoning, SFT alone yields a correctness score of 0.316. Applying RLOO raises this to 0.400, demonstrating the value of reward-guided fine-tuning in tasks that involve symbolic reasoning and format-sensitive verification. Offline distillation from medium-sized teacher models (Qwen 1.5B and 3B Instruct) improves over SFT (0.348 and 0.338 respectively), but falls short of RLOO performance. Interestingly, the 1.5B teacher slightly outperforms the 3B model, suggesting diminishing returns with increased model size and highlighting the importance of teacher selection beyond just parameter count.

We also examine distillation from larger instruction-tuned models. Distilling from Qwen 1.5B and 3B teachers produces models with intermediate performance (0.348 and 0.338 respectively), higher than the SFT baseline but below RLOO. This confirms that knowledge distillation is an effective but bounded strategy—students can mimic teacher outputs, but are limited by teacher diversity and the lack of online feedback. Notably, the 1.5B teacher slightly outperforms the 3B teacher, suggesting that decoding strategy, prompt formatting, or training data alignment, not model size alone, may dictate teacher quality.

Overall, these results highlight the following insights:

- Prompt tuning is highly effective for instruction following, especially when combined with supervised fine-tuning on curated data.
- DPO may degrade performance if applied after high-quality SFT, due to noisy or harmful supervision from negative samples.
- Math reasoning benefits significantly from reward-based optimization, as symbolic correctness is difficult to learn from demonstrations alone.
- Distillation is competitive, especially when compute is constrained, but lacks the iterative refinement of online RL.
- Larger teacher size does not guarantee better student performance—effective teacher behavior depends on the alignment between the task, prompt structure, and decoding style.

Method	Instruction Win-Rate		Math Reasoning Score
	Leaderboard rnd. 1	Qwen-0.5B-Instruct	
Baseline (SFT)	0.695	–	0.316
+ Single Prompts	0.815	–	–
+ Hyperparam Tuning	0.885	0.458	–
+ System Prompts	**0.905**	**0.535**	–
DPO + Hyperparam Tuning	–	0.395	–
RLOO	–	–	**0.400**
Distillation from 1.5B	–	0.143	0.348
Distillation from 3B	–	0.4900	0.338

Table 2: Performance Comparison on Instruction Following (Win-Rate) and Math Reasoning (Score) **Leaderboard rnd. 1** denotes scores submitted to the evaluation leaderboard, while **Qwen-0.5B-Instruct** represents win-rates against the Qwen 0.5B Instruct baseline. Math reasoning is evaluated on the Countdown dataset using format and correctness checks.

5.2 Qualitative Analysis

Prompt Format Comparison. We explore two key prompting strategies in instruction-following: single prompts and system prompts. Single prompts are constructed using only a user message. For example:

> <|*im_start*|>*user*
> *What's the difference between supervision and reinforcement learning in AI?*
> <|*im_end*|>
> <|*im_start*|>*assistant*

This format relies solely on the user input to elicit a response. In contrast, system prompts prepend a detailed instructional message to guide generation behavior. The following is an example of our default system prompt, which consistently led to the best win-rate in our experiments:

> <|*im_start*|>*system*
> *Expert AI: Deliver superior, helpful, detailed, accurate, clear, well-structured answers.*
> *Key directives:*
> *- Comprehensive & Insightful: Explain reasoning; avoid superficiality.*
> *- Accurate & Clear: Ensure correctness and easy understanding.*
> *- Concise Completeness: Be thorough yet efficient.*
> <|*im_end*|>
> <|*im_start*|>*user*
> *What's the difference between supervision and reinforcement learning in AI?*
> <|*im_end*|>
> <|*im_start*|>*assistant*

This setup provides high-level behavioral constraints, enabling more structured and informative completions. Empirically, system prompts boost win-rates from 0.695 (baseline) to 0.905, indicating their effectiveness in aligning small models to preferred response patterns. We theorize that due to smaller models having more limited capacity, they may require more through guidance in order to reach a desired output.

Response Comparison. Diving into the responses generated by distilled models, we observe some characteristics. Notably, we find that distilled models often have difficulty terminating. Here we see in a response generated by the 1.5B distilled model, despite generating detailed steps for the task required, the model continues on to provide repeated instructions:

> *.... 7. Click on ""OK"" to start the conversion process. 8. Wait for the conversion to complete. This may take some time depending on the size of your Eevee materials. 9. After the conversion is complete, you should be able to use the converted Cycles materials in your Blender scene. ..... <themap> <table> <tr> <th>Step</th>*

<th>Command</th> </tr> <tr> <td>1</td> <td>Select the Eevee materials file and import it into Blender.</td> ....

This may reflect the limitations of smaller models in their learning capacity, where they are unable to effectively learn the longer reasoning chains that larger models are capable of.

6 Discussion

This study investigates the potential of enhancing small language models through two complementary approaches: reinforcement learning (RL) and knowledge distillation. Our experiments show that while supervised fine-tuning (SFT) provides a strong foundation, the effectiveness of further training varies by task and methodology. In instruction-following tasks, Direct Preference Optimization (DPO) yields mixed results and sometimes fails to outperform strong SFT baselines, indicating that prompt-aware supervision can already capture much of the alignment signal. In contrast, math reasoning tasks benefit more substantially from Reward Learning with Online Optimization (RLOO), where iterative sampling and reward-guided optimization encourage more structured and verifiable outputs.

In our extension, we explore knowledge distillation as an alternative to RL. Distillation from instruction-tuned teacher models—specifically Qwen 2.5 1.5B and 3B Instruct—yields promising results, particularly when applied to math reasoning tasks. This suggests that small models can absorb valuable reasoning patterns and alignment strategies from larger models via imitation, without needing to engage in complex reward optimization procedures.

Despite these findings, our work is subject to several limitations. First, the teacher models employed during distillation were not fine-tuned using reinforcement learning techniques. It remains an open question whether distilling from RL-enhanced teachers could lead to stronger downstream performance in student models. Second, distilled responses are often highly imitative, which may limit response diversity and robustness compared to models directly optimized with RL-based objectives. Third, our experiments fix the student model size at 0.5B parameters. The impact of distillation or RL on models of different scales is unexplored and may not generalize in a straightforward manner.

Moreover, the base model used for student training may have already incorporated forms of alignment or weak distillation during pretraining, which could reduce the observed marginal gains from further distillation or preference optimization. Finally, we restricted our teacher models to two variants of the Qwen family. The effects of alternative architectures or alignment strategies remain to be tested.

7 Conclusion

This work explores the feasibility and effectiveness of enhancing small language models through two complementary strategies: reinforcement learning (RL) and knowledge distillation. Using the Qwen 2.5 0.5B base model as our student, we conduct systematic experiments across two core tasks, instruction following and math reasoning, to evaluate performance improvements derived from Direct Preference Optimization (DPO), Reward Learning with Online Optimization (RLOO), and supervised distillation from larger teacher models.

Our findings show that RLOO effectively improves performance in math reasoning, likely due to the structured reward feedback in such tasks. However, DPO fails to yield improvement over SFT in instruction following. One plausible explanation is that the inclusion of negative samples in DPO training may introduce harmful gradients from poor-quality responses, ultimately hindering the model's learning. In contrast, distillation from stronger teacher models provides a robust and scalable approach to transfer capabilities, often outperforming RL-based alternatives.

Future work may address these limitations through several directions. One promising avenue is to compare the effectiveness of distilling from RL-finetuned teachers versus traditional instruction-tuned ones. Such comparisons could clarify whether the benefits of reinforcement learning propagate effectively through distillation. Another direction is to explore multi-teacher or ensemble distillation, where the student learns from a diverse pool of teacher responses. This could promote both generalization and diversity in student behavior. Scaling experiments across different student sizes would also help elucidate whether our findings hold consistently across parameter regimes. Lastly, evaluating distillation performance using teacher models trained with varied reward models, pretraining corpora,

or alignment objectives would provide a more comprehensive understanding of how teacher diversity affects student learning.

Overall, our study provides empirical evidence that even modestly sized models can benefit significantly from both reinforcement and imitation-based training regimes. These findings contribute to the broader goal of democratizing access to capable language models by making small, efficient models more competitive through strategic training interventions.

8 Team Contributions

- **Charlie Jiang** led the implementation of the RLOO and distillation pipelines, including model training and large-scale inference for teacher-student distillation.
- **Yixing Jiang** implemented the SFT and DPO pipelines, conducted training for these models, and contributed to visualization and interpretation of experimental results.
- **Yi Jing** developed the evaluation framework, including reward model integration and win-rate computation. Led the writing of the project proposal and took responsibility for writing the final report and designing the poster.

Changes from Proposal. The work distribution evolved during the project, driven by computational resource constraints and individual team members' strengths. Initially, Charlie was expected to lead RL finetuning and large-teacher training; instead, efforts focused on distillation and small-scale RLOO due to compute limitations. Yi Jing transitioned from teacher model training to evaluation and reporting. Yixing's proposed focus on comparing RL-finetuned and supervised teachers shifted to pipeline development for SFT and DPO. All team members collaboratively participated in experiment tracking, analysis, and final deliverables.

References

Keivan Alizadeh, Seyed Iman Mirzadeh, Dmitry Belenko, S Khatamifard, Minsik Cho, Carlo C Del Mundo, Mohammad Rastegari, and Mehrdad Farajtabar. 2024. Llm in a flash: Efficient large language model inference with limited memory. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 12562–12584.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609* (2023).

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback, 2022. *arXiv preprint arXiv:2212.08073* 8, 3 (2022).

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. 2025. SmolLM2: When Smol Goes Big – Data-Centric Training of a Small Language Model. *arXiv preprint arXiv:2502.02737* (2025). `https://arxiv.org/abs/2502.02737`

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research* 25, 70 (2024), 1–53.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2024. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback. *arXiv preprint arXiv:2402.09677* (2024). `https://arxiv.org/abs/2402.09677`

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. 2025. Cognitive Behaviors That Enable Self-Improving Reasoners, or, Four Habits of Highly Effective STARs. *arXiv preprint arXiv:2503.01307* (2025). https://arxiv.org/abs/2503.01307

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*. PMLR, 10835–10866.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

Anda Kai, Lin Zhu, and Jiangchuan Gong. 2023. Efficient Compression of Large Language Models with Distillation and Fine-Tuning. *Journal of Computer Science and Software Applications* 3, 4 (2023), 30–38.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems* 35 (2022), 3843–3857.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

Jiayi Pan, Junjie Zhang, Xingyao Wang, Lifan Yuan, Hao Peng, and Alane Suhr. 2025. TinyZero. https://github.com/Jiayi-Pan/TinyZero. Accessed: 2025-01-24.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2023), 53728–53741.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *ICLR 2022-Tenth International Conference on Learning Representations*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652* (2021).

Wei Wei, Jiabin Tang, Lianghao Xia, Yangqin Jiang, and Chao Huang. 2024. Promptmm: Multi-modal knowledge distillation for recommendation with prompt-tuning. In *Proceedings of the ACM Web Conference 2024*. 3217–3228.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. 2024. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122* (2024).