Multi-Agent Training for Open Domains without Verifiers

Taeyoun Kim

Overview. Current research in reinforcement learning (RL) for language models has shown substantial progress in verifiable domains where oracle rewards are available (e.g., AIME). However, the recent success of strong frontier models such as Gemini [1] on IMO 2025 is shifting attention towards open-ended tasks where the verifiability of answers is not guaranteed [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. Open domains at this level of difficulty have a fundamental difference: there exists no strong capable "teacher" model to provide reward signals. In this research proposal, I address the problem through a multi-agent training framework where models of comparable performance collaboratively improve each other, incorporating solutions to mitigate misalignment in AI safety.

Problem Statement. Open-ended tasks have numerous solutions, making oracle verification difficult during RL even with an LLM judge or reference solution. These tasks span a wide range from generating math proofs [2, 3, 4, 5, 6, 7, 8, 9, 10, 11] to outputting harmless answers against jailbreak attacks [12, 13]. Most training methods *that scale* in open domains rely on a better capable "teacher" model to provide reference answers or reward signals. For example, methods that compare generations to reference solutions [2, 3, 4] require a model capable of creating the solutions beforehand. Similarly, methods that evaluate generations against handmade rubrics rely on a capable LLM judge for evaluation[5]. Also, using trained reward models implies access to a capable verifier. In contrast, some tasks are so difficult that there exists no capable verifier to provide training signals (e.g., proof for the Riemann Hypothesis). Current methods that try to train without a verifier aggregate online rollouts to approximate a reference solution or use model confidence as a reward [6, 7, 8, 9, 10, 11]. While such work on inventing new methodologies is important, no work tries to investigate open-ended generation from a more scientifically fundamental perspective on model capability: On extremely hard tasks where no model has the capability to generate or verify, how can we improve models? Answering this question is critical because benchmarks (e.g., IMO-Bench [14]) are rapidly approaching the difficulty of real-world tasks where models struggle. Traditional RL approaches that rely on rewards from better capable models may become infeasible. Through this proposal, I outline two interrelated research questions for harnessing the power of available low-capable models: (1) *How do we give models generation capabilities?* and (2) *How do we give models verification capabilities?* In the process of addressing these questions, I build a multi-agent training framework that trains each model to be both a generator and verifier to improve itself and help the improvement of other peer models. Furthermore, I address the no free lunch problem which is likely to occur when training without external signals.

The Multi-Agent Framework

Motivation of Framework. Reinforcement learning requires both generation and verification capabilities, which are essential: without a verifier, there will be no reward, and even with a verifier, there will be no reliable training signal if the generation capability is so poor that rewards are near zero. With only access to low capable models that lack both generation and verification capabilities, how do we train? More specifically, (1) how do we induce minimal generation capabilities to kick-start training and (2) how do we induce verification capabilities to continue training? This framework is built in two stages by answering each question.

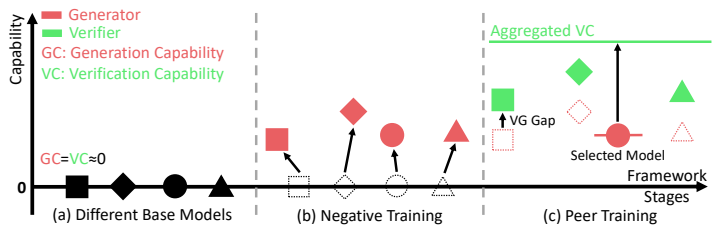

Figure 1: **The Multi-Agent Training Framework.** Different base models initially lack generation and verification capabilities on hard tasks. First, Negative Training pushes each model away from itself, giving minimal but diverse capabilities to kick-start training. Then, Peer Training uses other models to approximate a verifier by leveraging the verification–generation gap and model aggregation. The aggregated verifier exceeds the verification capability of individual models.

Concrete Problem Setup. For tasks where no existing model (e.g., Llama, Qwen, GPT, Claude) can reliably generate meaningful solutions or verify them, we effectively only have access to models of low capability. For research, this definition of "low capability" can be enforced by selecting a set of prompts where all models consistently fail over multiple rollouts. On this selected task, all models start with equally low capability (Figure 1.a).

RQ1: How do we induce minimal generation capabilities? The key assumption is that base models have poor generation capabilities. By leveraging this, a natural way to first improve models is to train away from the base policy. To do this, I would like to explore "negative training" where we treat base model rollouts as low reward generations (Figure 1.b). My work on reasoning for safety [12] demonstrates a form of verifier-based "negative training" where I train the model away from compliant behaviors. To extend this to non-verifiable domains, I plan to implement a REINFORCE-style [15] objective with a reward of -1 to penalize all samples, providing gradients that steer the model away from itself. While such negative training will not provide gradients in the oracle direction or guarantee convergence, it can push the generation quality above trivial answers, enabling a verifier to produce meaningful reward signals.

RQ2: How do we induce verification capabilities? Although we lack access to a high-capable verifier, we do have access to several low-capable models which we can ensemble to approximate a verifier. There are three insights into why aggregating models would work even when individual models have no verification capabilities. One, we can apply "negative training" to every model, which by improving generation capabilities also strengthens the internal representations needed for verification. Furthermore, different base models (e.g., Qwen, Llama) possess different cognitive behaviors [16] which would lead to diverse capabilities across models. Two, aggregation methods such as majority voting [17, 18, 19] or model ensembles [20, 21] surpass the capability of individual models, which would synergize even more once negative training improves each model. Three, models exhibit a verification-generation (VG) gap [22, 23] which may help models of similar capability verify each other's generations. Thus, with multiple models available to approximate a verifier and negative training which can increase initial generation capabilities, I would like to integrate the two solutions and build a two stage framework for multi-agent training where each model generates rollouts for on-policy training and also provides rewards for other models. The two stages: (1) Models first undergo individual "negative training" (Figure 1.b) to receive initial updates in meaningful directions. (2) Next, models go through "peer training" (Figure 1.c), where at each step a selected model generates rollouts and receives aggregated rewards from other models (e.g., majority-voted LLM judge). Each step in peer training improves one model, which serves as a verifier at a latter step to help improve another model. This framework is a fundamental training guideline to help a system of models improve on hard open domains, rather than a recipe confined to a single task. During my PhD, I plan to study its applications to several open domains such as math proofs, scientific discovery, and creative writing, and understand the core design choices to make it effective.

Risk Factors (No Free Lunch and Misalignment). The two main risks when training without external signals is that there is no free lunch (NFL) and misalignment can occur. Without an external verifier, it is unclear how a system of models can obtain new information to improve capabilities, and negative training can induce updates in any direction, potentially misaligned or degenerated. In my prior work on safety (TARS [12]), I encountered a relevant problem where the model degenerated by losing reasoning capabilities due to a large answer space. I mitigated the problem by adding an auxiliary task-completion reward on a separate task to encourage reasoning, which also improved reasoning on the main task. To prevent both the NFL problem and misalignment, I plan to extend this solution to the multi-agent framework by mixing in verifiable tasks during training. For example, if the main open domain task is to generate difficult math proofs, I will interleave math problems that are verifiable. The intuition is that the skills or abilities (e.g., sub-proofs, theorems) required to generate a difficult math proof overlap with the skills required to solve verifiable math problems. The verifiable problems will help utilize and connect those skills [23], which will in turn reinforce reasoning on the non-verifiable task. In this way, new information (e.g., skills) enters the system of models through the auxiliary verifiable task, addressing the NFL problem, while the verifiable reward also guides the model towards aligned and non-degenerate behavior.

Societal Impact. With difficult benchmarks such as IMO-Bench [14] sparking interest, future benchmarks may exceed the capability of existing models. The proposed framework provides a guideline to train a system of low capable models which initially cannot generate or verify solutions. This has significant societal implications. *One*, the framework would greatly expand the application of models to open domains (e.g., scientific discovery, law, medicine) that have no available verifier. Language models have been restricted to tasks that require some form of supervision, whether through curated data or external reward signals; this paradigm will change. *Two*, utilizing several small models to tackle open-ended domains has been underexplored. Recent work on math proofs leverage the strongest available models [24], but model capacity cannot scale infinitely. *Three*, the framework elevates AI safety to be more important. Since models will improve on their own, safeguards to prevent misalignment will be crucial. Overall, this framework of multi-agent training for open-ended tasks offers a promising and impactful agenda for better understanding the cooperation of low capable models.

References

[1] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[2] Xiangxin Zhou, Zichen Liu, Anya Sims, Haonan Wang, Tianyu Pang, Chongxuan Li, Liang Wang, Min Lin, and Chao Du. Reinforcing general reasoning without verifiers. *arXiv preprint arXiv:2505.21493*, 2025.

[3] Tianyu Yu, Bo Ji, Shouli Wang, Shu Yao, Zefan Wang, Ganqu Cui, Lifan Yuan, Ning Ding, Yuan Yao, Zhiyuan Liu, et al. Rlpr: Extrapolating rlvr to general domains without verifiers. *arXiv preprint arXiv:2506.18254*, 2025.

[4] Yunhao Tang, Sid Wang, Lovish Madaan, and Rémi Munos. Beyond verifiable rewards: Scaling reinforcement learning for language models to unverifiable data. *arXiv preprint arXiv:2503.19618*, 2025.

[5] Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*, 2025.

[6] Sheikh Shafayat, Fahim Tajwar, Ruslan Salakhutdinov, Jeff Schneider, and Andrea Zanette. Can large reasoning models self-train? *arXiv preprint arXiv:2505.21444*, 2025.

[7] Zizhuo Zhang, Jianing Zhu, Xinmu Ge, Zihua Zhao, Zhanke Zhou, Xuan Li, Xiao Feng, Jiangchao Yao, and Bo Han. Co-rewarding: Stable self-supervised rl for eliciting reasoning in large language models. *arXiv preprint arXiv:2508.00410*, 2025.

[8] Dulhan Jayalath, Shashwat Goel, Thomas Foster, Parag Jain, Suchin Gururangan, Cheng Zhang, Anirudh Goyal, and Alan Schelten. Compute as teacher: Turning inference compute into reference-free supervision. *arXiv preprint arXiv:2509.14234*, 2025.

[9] Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, et al. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*, 2025.

[10] Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*, 2025.

[11] Kongcheng Zhang, Qi Yao, Shunyu Liu, Yingjie Wang, Baisheng Lai, Jieping Ye, Mingli Song, and Dacheng Tao. Consistent paths lead to truth: Self-rewarding reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.08745*, 2025.

[12] Taeyoun Kim, Fahim Tajwar, Aditi Raghunathan, and Aviral Kumar. Reasoning as an adaptive defense for safety. *arXiv preprint arXiv:2507.00971*, 2025.

[13] Melody Y Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Helyar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, et al. Deliberative alignment: Reasoning enables safer language models. *arXiv preprint arXiv:2412.16339*, 2024.

[14] Minh-Thang Luong, Dawsen Hwang, Hoang H Nguyen, Golnaz Ghiasi, Yuri Chervonyi, Insuk Seo, Junsu Kim, Garrett Bingham, Jonathan Lee, Swaroop Mishra, et al. Towards robust mathematical reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35406–35430, 2025.

[15] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

[16] Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars. *arXiv preprint arXiv:2503.01307*, 2025.

[17] Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. Replacing judges with juries: Evaluating llm generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*, 2024.

[18] Siyuan Huang, Zhiyuan Ma, Jintao Du, Changhua Meng, Weiqiang Wang, and Zhouhan Lin. Mirror-consistency: Harnessing inconsistency in majority voting. *arXiv preprint arXiv:2410.10857*, 2024.

[19] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More agents is all you need. *arXiv preprint arXiv:2402.05120*, 2024.

[20] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR, 2022.

[21] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7959–7971, 2022.

[22] Yuda Song, Hanlin Zhang, Carson Eisenach, Sham Kakade, Dean Foster, and Udaya Ghai. Mind the gap: Examining the self-improvement capabilities of large language models. *arXiv preprint arXiv:2412.02674*, 2024.

[23] Amrith Setlur, Matthew YR Yang, Charlie Snell, Jeremy Greer, Ian Wu, Virginia Smith, Max Simchowitz, and Aviral Kumar. e3: Learning to explore enables extrapolation of test-time compute for llms. *arXiv preprint arXiv:2506.09026*, 2025.

[24] Yichen Huang and Lin F Yang. Gemini 2.5 pro capable of winning gold at imo 2025. *arXiv preprint arXiv:2507.15855*, 7, 2025.