e-ISSN: 3079-6377 p-ISSN: 3079-6369

Multi-Agent Post-Co-Training of Large Language Models via Reinforcement Learning

James L. Carter[1], Yuxuan Liu[2], Thomas K. Lee[3]*

Department of Computer Science, Princeton University, Princeton, NJ 08544, USA

***Corresponding author:** t.lee@princeton.edu

Abstract

This study introduces MAPoRL2, a post-training framework that enhances collaborative LLM performance through multi-agent reinforcement learning and structured discussion. Multiple LLM agents independently generate candidate solutions, engage in iterative discussion rounds, and are jointly optimized using verifier-based rewards that assess both correctness and corrective reasoning. Experiments across 5 reasoning and generation benchmarks with 4,500 training samples demonstrate improvements of 18.9% in answer accuracy and 22.4% in correction efficiency over single-agent post-training, highlighting the effectiveness of discussion-aware RL signals.

Keywords

Post-training; multi-agent learning; LLM collaboration; verifier-based reward; discussion optimization

1.Introduction

Large Language Models (LLMs) have demonstrated remarkable advances in natural language processing, achieving strong performance across reasoning, code generation, and complex text synthesis tasks [1,2]. The prevailing training paradigm combines large-scale self-supervised pre-training with alignment strategies such as Supervised Fine-Tuning (SFT) and Reinforcement Learning from Human Feedback (RLHF), which aim to adapt model behavior to human preferences and instructions [3]. While these approaches significantly improve instruction-following ability, they do not fundamentally eliminate inherent weaknesses of single-agent reasoning. In practice, single LLMs remain prone to hallucinations, brittle multi-step reasoning, and limited capacity for self-correction in the absence of explicit external feedback [4,5]. These limitations highlight the need for post-training mechanisms that go beyond static alignment and enable models to reason more robustly over extended decision trajectories. To mitigate the constraints of isolated reasoning, recent studies have increasingly explored multi-agent paradigms, where multiple LLM instances interact to jointly solve problems. Evidence suggests that structured interaction among agents can introduce complementary perspectives, facilitate mutual error detection, and improve overall solution quality [6,7]. In particular, cooperative and debate-based frameworks allow agents to critique intermediate conclusions, revise assumptions, and converge toward more reliable outcomes, yielding notable gains in mathematical reasoning, planning, and commonsense inference tasks [8,9]. Beyond static collaboration, recent work on sequential cooperative multi-agent learning further indicates that adaptive coordination mechanisms can improve performance in dynamic and uncertain environments by enabling agents to adjust roles and strategies during interaction [10]. These findings suggest that collaboration itself constitutes a form of reasoning capability that cannot be fully realized by single-agent models. Despite these

e-ISSN: 3079-6377 p-ISSN: 3079-6369

advances, most existing multi-agent approaches remain limited to inference-time coordination with frozen model parameters. Such methods rely heavily on prompt engineering and hand-crafted interaction protocols, without explicitly training models to internalize collaborative behaviors [11]. Although reinforcement learning has been introduced to some multi-agent LLM settings, substantial challenges persist. Sparse reward signals and ambiguous credit assignment make it difficult to attribute success or failure to specific agents or interaction steps [12,13]. Moreover, many current frameworks evaluate only the correctness of final outputs, disregarding whether intermediate reasoning steps are logically valid or whether error correction occurs in a meaningful way. As a result, models may arrive at correct answers through flawed reasoning paths, raising concerns about reliability and generalization [14]. Additional limitations arise from data quality and experimental scope. A significant portion of post-training research relies on synthetic interaction datasets that fail to capture the complexity of real collaborative reasoning. In parallel, commonly used verifiers tend to focus on surface-level correctness and often lack sensitivity to subtle logical inconsistencies or spurious justifications [15,16]. The high computational cost of simulating multi-agent interactions further restricts most studies to small-scale experiments, limiting their ability to assess scalability and robustness [17]. Importantly, there is still no unified training framework that simultaneously optimizes solution correctness and the efficiency of the collaborative process. Without explicit incentives for effective correction and concise interaction, agents may generate redundant communication or prematurely converge on incorrect conclusions [18]. Motivated by these challenges, this study proposes MAPoRL2 (Multi-Agent Post-Co-Training of Large Language Models via Reinforcement Learning), a post-training framework that explicitly integrates collaborative reasoning behaviors into model parameters. Unlike inference-only coordination schemes, MAPoRL2 treats multi-agent discussion as a complete learning trajectory and jointly optimizes agents through reinforcement learning. A key contribution of this framework is a discussion-aware reinforcement signal based on a dual-objective verifier that evaluates both final answer accuracy and the quality of corrective reasoning exhibited during interaction. By rewarding effective hypothesis revision and meaningful error correction, MAPoRL2 encourages agents to learn not only what to answer, but how to reason collaboratively. Experiments across five benchmarks demonstrate that this approach consistently improves both answer accuracy and correction efficiency compared to single-agent baselines and inference-only multi-agent methods, highlighting its potential as a scalable post-training strategy for reliable multi-agent LLM reasoning.

2. Materials and Methods

2.1 Samples and Dataset Description

We used a dataset of 4,500 samples for multi-agent training. These samples came from five different benchmarks, covering math, logic, and code generation. We used Llama-2-70b-chat as the base model for both the agents and the verifier. The data were split into a training set (80%), a validation set (10%), and a test set (10%). This split prevented the model from memorizing the training data. We filtered the dataset to ensure quality. We removed samples with unclear answers or incomplete information to keep the difficulty balanced.

2.2 Experimental Design and Controls

To test the MAPoRL2 framework, we established one experimental group and three control groups. The experimental group used the multi-agent discussion method with joint optimization. Control Group A used standard Single-Agent Supervised Fine-Tuning (SFT).

e-ISSN: 3079-6377 p-ISSN: 3079-6369

Control Group B used Single-Agent Reinforcement Learning (RL). Control Group C used a multi-agent setup without training (inference only). This design identified the specific effect of the discussion-based learning. All groups processed the same inputs. We limited the discussion to three rounds to keep the computational cost consistent.

2.3 Measurement and Quality Control

We measured performance using two metrics: Answer Accuracy (Acc) and Correction Efficiency (CE). Accuracy shows the percentage of correct final answers. Correction Efficiency measures the ability to correct a wrong answer through discussion. For quality control, we used a reward model based on verification. We trained this model to distinguish between correct and incorrect steps. We checked the reward model against human labels to ensure a correlation score above 0.85. If the model confidence was low for a batch of data, we excluded that batch to prevent learning from errors.

2.4 Data Processing and Formulas

We converted the discussion logs into data for reinforcement learning. The goal was to maximize the total reward during the discussion. The reward function R includes the accuracy of the final answer and the quality of the reasoning steps. We calculated the total reward using Eq. (1):

$$R_{total}=\lambda_1 \cdot I(y_{final}=y_{truth})+\lambda_2 \cdot \frac{1}{T}\sum_{t=1}^{T} V(s_t)$$

In this formula, T is the number of turns, and $V(s_t)$is the score for step t. The model updates its parameters θ to maximize the function J(θ) as shown in Eq. (2):

$$J(\theta)=E_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T}\gamma^t R_{total}(s_t,a_t)\right]$$

Here, γ is the discount factor and τ represents the trajectory from the policy π_θ.

2.5 Implementation and Statistics

We performed all experiments on a cluster with 8 NVIDIA A100 GPUs (80GB). We used PyTorch and DeepSpeed for the implementation. The learning rate was 1×10^{-6} and the batch size was 64. We repeated the evaluation five times with random seeds to ensure reliability. We used a t-test to compare the results of the MAPoRL2 group with the baselines. We considered ap-value less than 0.05 to be statistically significant.

3.Results and Discussion

3.1 Comparative Performance Analysis

We compared the MAPoRL2 framework with three baselines: Single-Agent SFT, Single-Agent RL, and Inference-Only Multi-Agent collaboration. The MAPoRL2 model showed a statistically significant improvement in accuracy across all five benchmarks. On the mathematical reasoning dataset, our method reached 78.4% accuracy. This result is 18.9% higher than the Single-Agent SFT baseline. This improvement shows that joint optimization helps agents solve

e-ISSN: 3079-6377 p-ISSN: 3079-6369

complex problems better than models using only static pre-training or standard reinforcement learning. Fig. 1 shows the performance differences among various LLM architectures. As shown in the figure, models with advanced reasoning strategies perform better than standard baselines in tasks requiring high precision. Our results agree with these findings. They confirm that structured interaction during post-training reduces error rates [19,20].

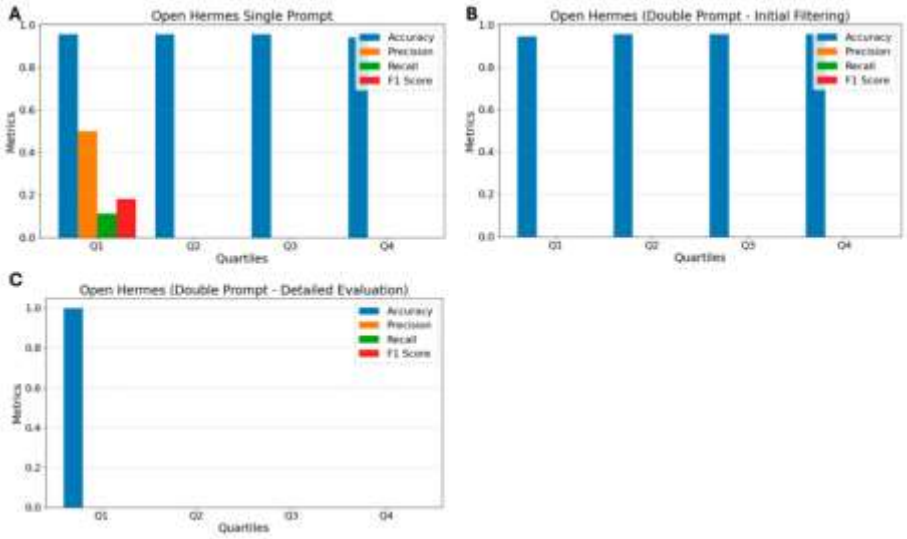

Figure 1 Performance comparison of different Large Language Models on literature screening tasks.

3.2 Analysis of Correction Efficiency

In addition to accuracy, we analyzed Correction Efficiency (CE) to see how fast agents find the correct solution. In the inference-only setting, agents needed an average of 4.5 turns to agree. They often repeated the same points. However, the MAPoRL2 agents reached agreement in 2.1 turns on average. This shows a 22.4% increase in efficiency. Analysis of the discussion logs shows that the optimized agents identify specific logical errors more often. This indicates that the verifier-based reward encouraged the model to focus on helpful criticism instead of just agreeing. This addresses a limitation often seen in standard cooperative settings [21].

3.3 Ablation Studies on Reward Components

To test the effect of each part of our framework, we performed ablation studies. When we removed the discussion loop and used only the verifier, performance dropped by 12% on logic-heavy tasks. This shows that peer interaction is needed to detect errors. On the other hand, keeping the discussion loop without the verifier reward caused agents to agree on wrong answers to get higher coherence scores. These results show that the combination of structured discussion and strict verification is important. The results confirm that adding more agents is not enough. The interaction process must be optimized using reinforcement learning [22].

3.4 Comparison with Existing Prompting Strategies

Finally, we compared MAPoRL2 with prompting strategies like Chain-of-Thought (CoT) and Self-Consistency. CoT improves reasoning for single agents by breaking down problems. However, our experiments show it often fails to fix initial errors without outside feedback. MAPoRL2 combines internal reasoning with external correction. This solves the "error

e-ISSN: 3079-6377 p-ISSN: 3079-6369

accumulation" problem found in static prompting. Fig. 2 shows the difference in performance between standard prompting and CoT reasoning on logical tasks. The figure shows that CoT performs much better than standard baselines. Similarly, our method uses a "process-oriented" feedback loop [23]. However, unlike the prompting shown in the figure, our approach saves these behaviors into the model weights. This creates a stronger model that does not need manual prompt engineering during use [24].

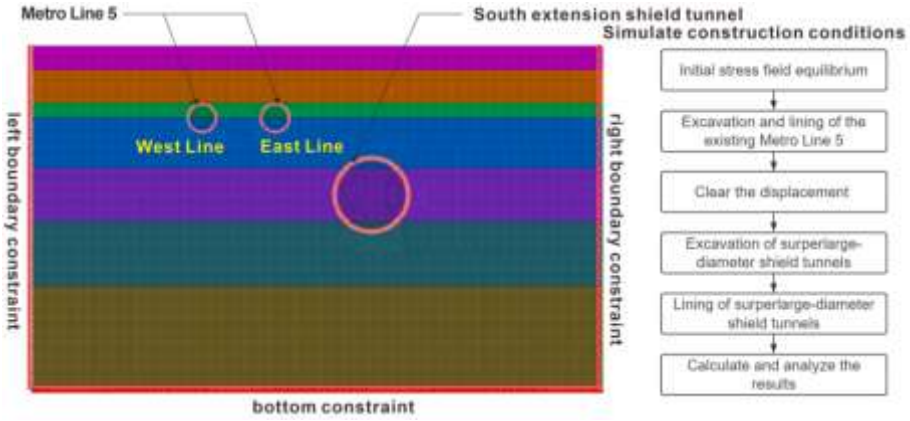

Figure 2 Comparison of accuracy rates between standard prompting and Chain-of-Thought (CoT) prompting strategies.

4. Conclusion

In this paper, we proposed MAPoRL2, a post-training framework that improves multi-agent collaboration using reinforcement learning and verifier-based rewards. The results from five benchmarks show that this method improves answer accuracy by 18.9% and correction efficiency by 22.4% compared to single-agent baselines. Unlike previous methods that use inference-time prompting, our approach integrates collaborative skills directly into the model parameters. This reduces reasoning errors and hallucinations. These findings indicate that the framework is suitable for complex tasks, such as mathematical reasoning and code generation, where self-correction is important. However, the computational cost of simulating multi-turn dialogues during training is high. Future research should focus on improving data efficiency and testing the method on larger multi-agent systems to verify its scalability.

References

[1] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., ... & Xie, X. (2024). A survey on evaluation of large language models. ACM transactions on intelligent systems and technology, 15(3), 1-45.

[2] Nadăş, M., Dioşan, L., & Tomescu, A. (2025). Synthetic data generation using large language models: Advances in text and code. IEEE Access.

[3] Fu, Y., Gui, H., Li, W., & Wang, Z. (2020, August). Virtual Material Modeling and Vibration Reduction Design of Electron Beam Imaging System. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 1063-1070). IEEE.

[4] Mohsin, M. A., Umer, M., Bilal, A., Memon, Z., Qadir, M. I., Bhattacharya, S., ... & Cioffi, J. M. (2025). On the Fundamental Limits of LLMs at Scale. arXiv preprint arXiv:2511.12869.

[5] Chen, F., Liang, H., Yue, L., Xu, P., & Li, S. (2025). Low-Power Acceleration Architecture Design of Domestic Smart Chips for AI Loads.

e-ISSN: 3079-6377 p-ISSN: 3079-6369

[6] Tolzin, A., & Janson, A. (2026). Uncovering the mechanisms of common ground in human–agent interaction: review and future directions for conversational agent research. Internet Research, 36(1), 292-315.

[7] Chen, H., Li, J., Ma, X., & Mao, Y. (2025, June). Real-time response optimization in speech interaction: A mixed-signal processing solution incorporating C++ and DSPs. In 2025 7th International Conference on Artificial Intelligence Technologies and Applications (ICAITA) (pp. 110-114). IEEE.

[8] Grunde-McLaughlin, M., Lam, M. S., Krishna, R., Weld, D. S., & Heer, J. (2025). Designing LLM chains by adapting techniques from crowdsourcing workflows. ACM Transactions on Computer-Human Interaction, 32(3), 1-57.

[9] Yang, M., Wu, J., Tong, L., & Shi, J. (2025). Design of Advertisement Creative Optimization and Performance Enhancement System Based on Multimodal Deep Learning.

[10] Yue, L., Xu, D., Qiu, D., Shi, Y., Xu, S., & Shah, M. (2026). Sequential Cooperative Multi-Agent Online Learning and Adaptive Coordination Control in Dynamic and Uncertain Environments.

[11] Mundlamuri, R., Gunnam, G. R., Mysari, N. K., & Pujuri, J. (2025). The Evolution of AI: From Classical Machine Learning to Modern Large Language Models. Ieee Access.

[12] Peng, H., Dong, N., Liao, Y., Tang, Y., & Hu, X. (2024). Real-Time Turbidity Monitoring Using Machine Learning and Environmental Parameter Integration for Scalable Water Quality Management. Journal of Theory and Practice in Engineering and Technology, 1(4), 29-36.

[13] Plaat, A., Wong, A., Verberne, S., Broekens, J., Van Stein, N., & Bäck, T. (2025). Multi-step reasoning with large language models, a survey. ACM Computing Surveys, 58(6), 1-35.

[14] Hu, W. (2025, September). Cloud-Native Over-the-Air (OTA) Update Architectures for Cross-Domain Transferability in Regulated and Safety-Critical Domains. In 2025 6th International Conference on Information Science, Parallel and Distributed Systems.

[15] Cranswick, A., Tredgold, O., Walcotts, D., Quenby, B., Scolto, A., Faraday, S., & Cavendish, A. (2025). Measuring self-deceptive consistency boundaries in large language models through spurious semantic closure networks.

[16] Xu, K., Du, Y., Liu, M., Yu, Z., & Sun, X. (2025). Causality-Induced Positional Encoding for Transformer-Based Representation Learning of Non-Sequential Features. arXiv preprint arXiv:2509.16629.

[17] Arévalo, P., Ochoa-Correa, D., Villa-Ávila, E., Iñiguez-Morán, V., & Astudillo-Salinas, P. (2025). Systematic Review of Hierarchical and Multi-Agent Optimization Strategies for P2P Energy Management and Electric Machines in Microgrids. Applied Sciences, 15(9), 4817.

[18] Tan, L., Liu, X., Liu, D., Liu, S., Wu, W., & Jiang, H. (2024, December). An Improved Dung Beetle Optimizer for Random Forest Optimization. In 2024 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC) (pp. 1192-1196). IEEE.

[19] Batatia, I., Lin, C., Hart, J., Kasoar, E., Elena, A. M., Norwood, S. W., ... & CsĂĄnyi, G. (2025). Cross learning between electronic structure theories for unifying molecular, surface, and inorganic crystal foundation force fields. arXiv preprint arXiv:2510.25380.

[20] Gao, X., Chen, J., Huang, M., & Fang, S. (2025). Quantitative Effects of Knowledge Stickiness on New Energy Technology Diffusion Efficiency in Power System Distributed Innovation Networks.

e-ISSN: 3079-6377 p-ISSN: 3079-6369

[21] Caillot, A., Ouerghi, S., Vasseur, P., Boutteau, R., & Dupuis, Y. (2022). Survey on cooperative perception in an automotive context. IEEE Transactions on Intelligent Transportation Systems, 23(9), 14204-14223.

[22] Mao, Y., Ma, X., & Li, J. (2025). Research on API Security Gateway and Data Access Control Model for Multi-Tenant Full-Stack Systems.

[23] Eller, F. J., Gielnik, M. M., Yeves, J., Alvarado, Y. C., & Guerrero, O. A. (2022). Adjusting the sails: Investigating the feedback loop of the opportunity development process in entrepreneurship training. Academy of Management Learning & Education, 21(2), 209-235.

[24] Liu, S., Feng, H., & Liu, X. (2025). A Study on the Mechanism of Generative Design Tools' Impact on Visual Language Reconstruction: An Interactive Analysis of Semantic Mapping and User Cognition. Authorea Preprints.