Unleashing Diverse Thinking Modes in LLMs through Multi-Agent Collaboration

Anonymous ACL submission

Abstract

Large language models(LLMs) often achieve strong benchmark performance, yet their reasoning remains brittle, hard to audit, and difficult to compare fairly across multi-agent settings. We present DiMo, a multi-agent collaboration framework that operationalizes two interaction protocols: Divergent and Logical thinking modes. Rather than treating debate as a single monolithic technique, DiMo makes protocol choices explicit and auditable: agents externalize hypotheses, supporting knowledge, reasoning paths, and localized revisions as inspectable traces. Under a unified token budget and open-source setup, DiMo improves accuracy on six reasoning benchmarks over single-model and debate baselines, with the largest gain on mathematical reasoning. Beyond raw performance, we use DiMo as an analytic lens to characterize a protocol-task affinity: commonsense tasks tend to benefit more from divergent interaction, whereas mathematical tasks benefit more from logical verification. We present this as an empirical regularity under fixed models, prompts, judges, and budgets, rather than a claim about model-internal mechanisms. Code is available at https://anonymous.4open.science/r/DiMo-D2F0.

1 Introduction

Large language models(LLMs) have achieved strong performance across a wide range of tasks (Meta AI, 2024; Liu et al., 2024; Kimi Team et al., 2025), yet single-model prompting can remain brittle on multifaceted reasoning problems. Multi-agent debate systems address this issue by allowing multiple LLM instances to critique, revise, and aggregate candidate answers (Guo et al., 2024), and have shown promise in complex reasoning, fact-checking, and hypothesis evaluation (Wang et al., 2023). However, most existing debate frameworks treat interaction as a generic procedure: the intermediate states being produced, the critique operations being applied, and the stopping rules are often implicit. This makes it difficult to compare different collaboration protocols fairly under a fixed compute budget.

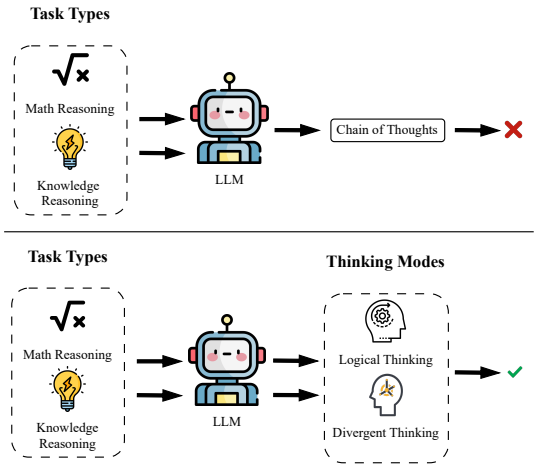

Figure 1: LLMs prefer to use different thinking modes to solve different reasoning tasks and lead to higher accuracy.

A second challenge is interpretability. Since LLMs are black-box models, their internal problem-solving mechanisms cannot be directly observed (Zhao et al., 2024). In this work, we therefore focus on process transparency: instead of claiming access to model-internal mechanisms, we aim to expose external traces of intermediate hypotheses, critiques, and revisions. Such traces make the reasoning process more auditable for humans and verifier agents, even when the base model remains opaque.

In this work, we propose DiMo, a multi-agent collaboration framework that makes interaction protocols explicit and auditable. Rather than treating debate as a monolithic procedure, DiMo operationalizes two protocols(thinking modes): a Divergent Thinking Mode that expands candidate knowl-

edge and reasoning paths, and a Logical Thinking Mode that emphasizes step-wise verification and localized refinement. In our instantiation, role-specicalized agents propose supporting knowledge, construct reasoning paths, evaluate candidate answers, and revise them through iterative deliberation. The resulting hypotheses, knowledge snippets, critiques, and refinements form auditable traces of the reasoning process.

Motivation. Large language models excel across benchmarks, yet their reasoning remains brittle and opaque: **single-model prompting can swing between correct solutions and confident mistakes, and intermediate steps are rarely auditable. Inspired by cognitive accounts of problem solving that distinguish hypothesis generation from hypothesis testing, we make this separation explicit as an operational, auditable protocol for multi-agent collaboration. Debate-style multi-agent systems improve robustness by letting agents critique and revise one another, but they often lack operational definitions of the "modes" of thinking being exercised, blur process transparency with mechanistic interpretability, and are hard to compare fairly across compute budgets.** Our goal is to provide a reproducible, open-source protocol that makes these design choices explicit. DiMo constrains collaboration into two operational modes—divergent (parallel hypothesis/knowledge proposal) and logical (step-wise verification with localized refinement)—so we can ask not only "does debate help?" but "which protocol helps which task under a fixed budget?" **This lens lets us study protocol–task affinity while improving answer accuracy and exposing auditable traces, without making claims about the base models' internal mechanisms.** We therefore treat multi-agent debate not as a single monolithic technique, but as a design space of interaction protocols that can be made explicit, audited, and compared under matched budgets. We therefore treat multi-agent debate not as a single monolithic technique, but as a design space of interaction protocols that can be made explicit, audited, and compared under matched budgets.

Our contributions include: 1)We formalize divergent and logical thinking modes as operational interaction protocols for multi-agent reasoning, specified by explicit roles, intermediate states, critique operations, and stopping rules. 2)We instantiate DiMo as an auditable reasoning framework that externalizes hypotheses, supporting knowledge, reasoning paths, and localized revisions as process-level traces. 3)Under a unified token budget and open-source setup, we provide a controlled empirical study of accuracy, cost, and component contributions across six reasoning benchmarks. 4)We use DiMo as an analytic lens to characterize protocol-task affinity as an empirical regularity conditioned on models, prompts, judges, and budgets, and discuss its implications for future routing and protocol selection.

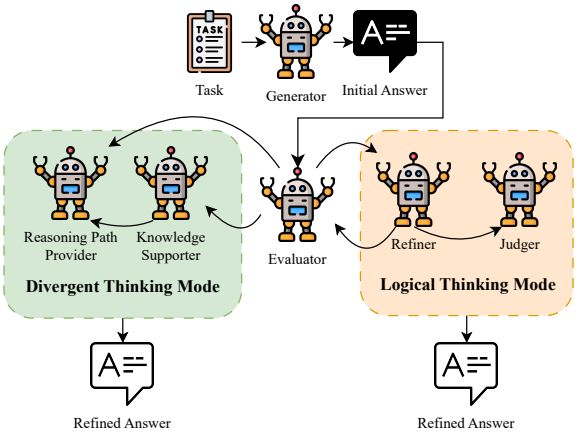

Figure 2: An Overview of Multi-Agent Collaboration Framework for Diverse Thinking Modes (DiMo). Role-specialized agents debate and refine solutions, yielding auditable intermediate reasoning traces.

2 Related Work

LLM reasoning and self-improvement. LLMs exhibit strong but uneven reasoning abilities across tasks, motivating methods that elicit more reliable multi-step inference (Brown et al., 2020; Wei et al., 2022; Kojima et al., 2022). Representative prompting paradigms include Chain-of-Thought (Wei et al., 2022), Tree-of-Thoughts (Yao et al., 2024), and self-consistency (Wang et al., 2022), while recent post-training efforts further strengthen mathematical reasoning with long-form traces (DeepSeek-AI et al., 2025; Qwen Team, 2024). Complementary work studies iterative self-improvement via critique-and-revision (Madaan et al., 2024; Renze and Guven, 2024; Shinn et al., 2024). DiMo is orthogonal: we set up two interaction protocols (thinking modes) and coordinate role-specialized agents to generate, critique, and refine intermediate states under a fixed budget.

Multi-agent debate and process transparency. Multi-agent debate frameworks improve robustness by encouraging agents to propose alternatives, cri-

tique one another, and consolidate diverse perspectives (Xiong et al., 2023; Du et al., 2023; Liang et al., 2023). In parallel, process-level interpretability has been pursued via explainability surveys and structured reasoning systems that externalize intermediate traces (Luo and Specia, 2024; Wang et al., 2024; Creswell et al., 2022). DiMo targets this notion of interpretability by explicitly recording intermediate hypotheses, supporting knowledge, and step-wise refinements produced during collaboration.

Cognitive inspiration. Neuroimaging evidence suggests deductive and probabilistic reasoning recruit partially distinct neural systems (Osherson et al., 1998; Castañeda et al., 2023). We use this only as high-level motivation: our thinking modes are engineering protocols for auditable collaboration rather than claims about base models' internal mechanisms.

Overall, prior work has shown that prompting, self-refinement, and multi-agent debate can improve reasoning. DiMo differs in its focus: we treat the interaction protocol itself as the object of study. Rather than asking only whether multiple agents help, DiMo asks which intermediate states should be externalized, how they should be critiqued, and which protocol is most effective for which task under a matched budget. This protocol-centric perspective distinguishes DiMo from prior debate frameworks.

3 Methodology

We propose an LLM-based multi-agent debate framework, **Multi-Agent Collaboration Framework for Diverse Thinking Modes (DiMo)**. An overview is shown in fig. 2. DiMo introduces two operational thinking modes to handle heterogeneous reasoning tasks. We use cognitive and neuroscience findings only as high-level motivation; the framework does not claim any correspondence between agents and biological mechanisms.

Definitions of Modes. We define a *mode* as an interaction protocol that restricts what intermediate states an agent may propose and how other agents may critique or revise them. Divergent mode encourages proposing alternative hypotheses, supportive knowledge snippets, and candidate reasoning paths. Logical mode enforces stepwise derivation with verification and refinement.

Generator. The Generator receives an input question and produces an initial answer, which

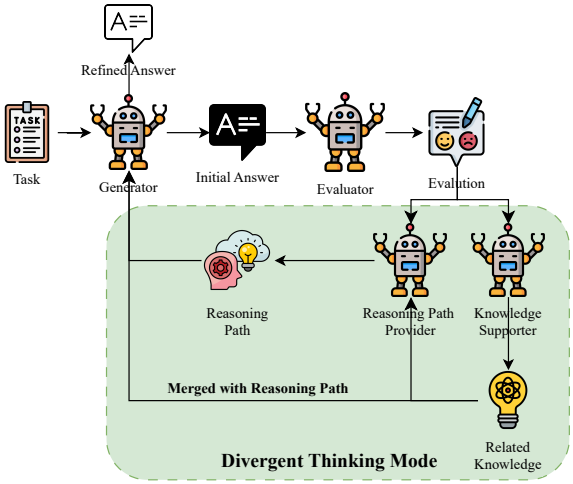

Figure 3: An Overview of Divergent Thinking Mode. Parallel proposals of knowledge and reasoning paths are synthesized into answers, benefiting commonsense tasks.

is then passed to the Evaluator. We design distinct prompts for different task types: for commonsense/knowledge-based tasks, the Generator outputs a direct initial answer; for mathematical reasoning tasks, it outputs a step-by-step solution. Formally,

$$A = \text{Generator}(Q), \tag{1}$$

where A is the initial answer, Generator denotes the Generator, and Q is the input question. Following the generation of the initial answers, the output is directed to the Evaluator for assessment.

Evaluator. The Evaluator assesses the initial answer and produces structured feedback. For knowledge-dependent tasks, it highlights logical deficiencies and missing knowledge. For logic-intensive tasks, it performs step-by-step verification to locate inconsistencies and computational errors, emphasizing both correctness and coherence. Based on the task type, the Evaluator's feedback routes the solution into the corresponding thinking mode, where multiple agents iteratively debate and refine the answer.

Prior work suggests that LLMs may employ different processing strategies for factual-knowledge tasks (e.g., commonsense reasoning) versus rule-based derivation tasks (e.g., mathematical reasoning) (Brown et al., 2020). Motivated by this empirical distinction, we instantiate two operational thinking modes and study whether different tasks exhibit preference for different protocols.

W

3.1 Divergent Thinking Mode

We present **Divergent Thinking Mode** in fig. 3. Beyond the Generator and Evaluator introduced above, this mode adds two role-specialized agents: a Knowledge Supporter and a Reasoning Path Provider.

Knowledge Supporter. The Knowledge Supporter retrieves task-relevant, domain-specific knowledge, validates its correctness and applicability, and consolidates evidence from multiple sources. This knowledge serves as explicit support for revising the current solution.

Reasoning Path Provider. The Reasoning Path Provider constructs and validates an explicit reasoning path, emphasizing logical completeness and traceability. Together, the retrieved knowledge and the proposed reasoning path are fed back to the Generator, which integrates them to revise the initial answer and produce a refined answer. The refined answer is then passed to the Discussion module.

Formally, we define Divergent Thinking Mode as:

$$K = \text{KnowledgeSupporter}(E, A), \tag{2}$$

$$R = \text{PathProvider}(E, K), \tag{3}$$

where E is the evaluation draft, A is the initial answer, K denotes the retrieved knowledge, and R denotes the reasoning paths. The Generator then produces the refined answer:

$$O = \text{Generator}(R, K), \tag{4}$$

where O is the refined answer.

After obtaining O, we run a **Discussion** module as a collaborative debate verifier. It consists of the Evaluator, Knowledge Supporter, and Reasoning Path Provider, who debate the validity and soundness of O. The discussion follows a binary decision rule: if the agents reach consensus that O satisfies the correctness criteria, the process terminates and O is accepted; otherwise, the system proceeds to the next debate round ("Next Round Debate" in fig. 3) to generate another refinement. This iterative procedure continues until consensus is reached or the maximum number of debate rounds is met.

3.2 Logical Thinking Mode

We provide an overview of the **Logical Thinking Mode** of DiMo in fig. 4. In this mode, we also set up the Generator to receive the task inputs and generate the initial answer and the Evaluator to assess the initial answers alongside two agents tasked with debate.

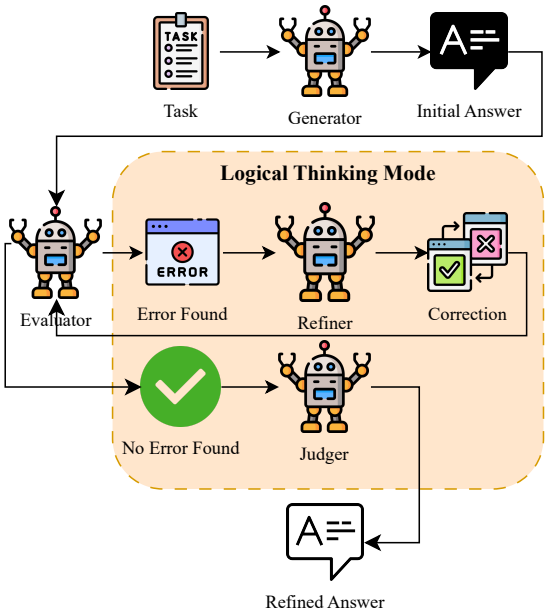

Figure 4: An Overview of Logical Thinking Mode. An evaluate–refine–judge loop enforces step-wise verification, effective for mathematical reasoning.

Refiner. The Refiner is a correction agent that performs the refinement of problematic steps identified by the Evaluator. This agent employs a localized rewriting methodology, maintaining consistency with preceding and subsequent steps while implementing necessary corrections. The refinement corrects errors while preserving the reasoning path.

Judger. The Judger is a comprehensive assessment agent that conducts a holistic judgment of the refined reasoning path and final solution if it demonstrates both logical consistency and computational accuracy or initiates a single iteration of the Evaluator-Refiner cycle if deficiencies are detected. The judgment process is bounded by a predetermined iteration limit to prevent infinite recursion, ensuring computational efficiency while maintaining solution quality.

Based on the above framework, we can construct the **Logical Thinking Mode** with the specific formulas as follows:

$$e = \text{Evaluator}(A), \tag{5}$$

Where e represents the error status, A represents the initial answer. Then we can have conditions:

$$R = \begin{cases} \text{Refiner}(A) & \text{if } e = 1 \\ \text{Judger}(A) & \text{if } e = 0 \end{cases}, \tag{6}$$

Table 1: Dataset Statistics. Six benchmarks spanning commonsense and math reasoning (CSQA, OpenBookQA, ARC-C, StrategyQA, GSM8K, GSM-Hard), with dataset sizes and task characteristics summarized.

DATASET	TASK TYPE	TASK INFO	SIZE
CSQA(TALMOR ET AL., 2018)	COMMONSENSE	5 CHOICES	1,221
ARC-CHALLENGE(CLARK ET AL., 2018)	COMMONSENSE	4 CHOICES	1,170
STRATEGYQA(MIHAYLOV ET AL., 2018)	COMMONSENSE	TRUE/FALSE	687
OPENBOOKQA(GEVA ET AL., 2021)	COMMONSENSE	4 CHOICES	500
GSM8K(COBBE ET AL., 2021)	MATH	QUESTIONNAIRE	1,319
GSM-HARD(GAO ET AL., 2022)	MATH	QUESTIONNAIRE	1,319

Table 2: The Overall Performance of DiMo and Baselines on Commonsense Reasoning Tasks. **Bold** numbers represent the performance of our method DiMo. Underlined numbers represent the best performance.

METHOD	CSQA	ARC-CHALLENGE	STRATEGYQA	OPENBOOK QA
LLAMA-3.1-8B	76.1%	79.1%	63.4%	79.2%
LLAMA-3.1-8B(COT)	72.5%	77.5%	74.2%	77.8%
LLM MAD(LLAMA-3.1-8B)	70.6%	78.3%	84.4%	74.3 %
MAD(LLAMA-3.1-8B)	46.2%	57.7%	42.3%	26.4%
DOV(LLAMA-3.1-8B)	74.3%	65.4%	66.5%	81.4%
DIMO(LLAMA-3.1-8B)	**80.02%**	**84.1%**	**92.7%**	**84.5%**
QWEN-2.5-32B	87.6%	93.0%	93.4%	93.20%
QWEN-2.5-32B(COT)	86.5%	94.0%	78.0%	95.40%
QWQ-32B-PREVIEW	80.0%	87.3%	81.3%	87.2%
LLM MAD(QWEN-2.5-32B)	86.3%	94.1%	91.4%	94.9%
MAD(QWEN-2.5-32B)	79.1%	91.4%	58.3%	23.4%
DOV(QWEN-2.5-32B)	85.3%	87.4%	76.1%	95.6%
DIMO(QWEN-2.5-32B)	**88.4%**	**90.5%**	**90.8%**	**96.0%**

Where R represent the refined answer. Finally, the entire mode can be formalized through the following formula:

$$R = \text{Evaluator}(A)^n \ \text{Refiner}(A)^n + \left(1 - \text{Evaluator}(A)^n\right) \ \text{Judger}(A) \tag{7}$$

Where n is a number of Evaluator-Refiner correction cycles.

4 Experiment

In our experiment, we evaluate our multi-agent debate system using different types of QA tasks, which test whether our multi-agent debate system improves LLM reasoning.

4.1 Datasets

We evaluate DiMo on six benchmarks: four commonsense and knowledge-based QA datasets (CommonsenseQA(Talmor et al., 2018), ARC-Challenge(Clark et al., 2018), OpenBookQA(Geva et al., 2021), and StrategyQA(Mihaylov et al., 2018)) and two mathematical reasoning datasets (GSM8K(Cobbe et al., 2021) and GSM-Hard(Gao et al., 2022)). The commonsense set tests whether multi-agent deliberation can surface missing implicit knowledge and improve answer selection, while the math set tests whether collaboration can identify and correct calculation errors through explicit reasoning traces. All datasets statistics are listed in table 1.

4.2 Baselines

We compare DiMo with (i) single-model prompting, (ii) single-model Chain-of-Thought (CoT) prompting with deterministic answer extraction, (iii) an o1-like open-source reasoning model (QwQ-32B-preview(Qwen Team, 2024)), and (iv) debate-style multi-agent baselines (LLM-MAD(Du et al., 2023), MAD(Liang et al., 2023), and Debate-or-Vote(Choi et al., 2025)). For controlled comparison, we align decoding settings and per-instance token budgets across methods and count all rationale/debate tokens; tool use and external code execution are disabled.

4.3 Evaluation Metrics

We report Exact Match (EM)(Jiang et al., 2023). We enforce dataset-specific output formats via prompting to enable strict string-level evaluation.

Table 3: The Overall Performance of DiMo and Baselines on Mathematical Reasoning Tasks. **Bold** numbers represent the performance of our method DiMo. Underlined numbers represent the best performance.

METHOD	GSM8K	GSM-HARD
LLAMA-3.1-8B	49.8%	42.3%
LLAMA-3.1-8B(COT)	84.0%	44.7%
LLM MAD(LLAMA-3.1-8B)	75.8%	34.6%
MAD(LLAMA-3.1-8B)	43.7%	15.6%
DOV(LLAMA-3.1-8B)	83.4%	0.45%
DIMO(LLAMA-3.1-8B)	**90.7%**	**71.4%**
QWEN-2.5-32B	37.3%	28.9%
QWEN-2.5-32B(COT)	95.2%	72.4%
QWQ-32B-PREVIEW	75.4%	51.4%
LLM MAD(QWEN-2.5-32B)	94.7%	60.9%
MAD(QWEN-2.5-32B)	89.8%	30.9%
DOV(QWEN-2.5-32B)	94.5%	0.68%
DIMO(QWEN-2.5-32B)	**98.4%**	**84.1%**

Table 4: The Results of Using Different Modes on Mathematical Reasoning Tasks. Logical mode outperforms Divergent on math, supporting a protocol–task affinity for step-wise reasoning.

METHOD	GSM8K	GSM-HARD
DIMO (DIVERGENT) (LLAMA-3.1-8B)	79.9%	57.2%
DIMO (LOGICAL) (LLAMA-3.1-8B)	90.7%	71.4%
LLAMA-3.1-8B(COT)	84.0%	44.7%

4.4 Experiments Settings

LLMs Used in the Agents The experimental implementation utilized consistent model architecture across both our agents and the baselines, specifically employing LLaMA-3.1-8B(Meta AI, 2024) and Qwen-2.5-32B(Yang et al., 2024).

4.5 Main Results

Commonsense Reasoning Tasks. DiMo uses **Divergent Thinking Mode** for commonsense and knowledge-based benchmarks. As shown in table 2, DiMo delivers consistent gains on the multiple-choice tasks (CSQA, ARC-Challenge, and OpenBookQA): DiMo (LLaMA-3.1-8B) surpasses the 80% accuracy threshold on all three, while many baselines fall below 80% on at least one of them. On StrategyQA, DiMo further achieves 92.7%, outperforming strong multi-agent baselines.

Mathematical Reasoning Tasks. DiMo adopts **Logical Thinking Mode** for math reasoning. In table 3, DiMo improves over baselines with both LLaMA-3.1-8B and Qwen-2.5-32B, including the more challenging GSM-Hard benchmark. Notably, the Debate-or-Vote collapses to near-zero accuracy on GSM-Hard across debate rounds, indicating that LLM-only debate/vote aggregation is brittle when tasks demand high-precision numeric answers (large magnitudes, negatives and fractions). This failure mode motivates the verifier-guided component in DiMo, which targets high numerical entropy and strict correctness requirements.

Comparison with o1-like Reasoning LLMs. DiMo demonstrates superior performance not only on general LLMs but also when evaluated against Large Reasoning Language Models that employ post-training techniques, including reinforcement learning and Chain-of-Thought prompting. As shown in table 2 and table 3, As shown in tables 2 and 3, DiMo with Qwen2.5-32B compares favorably with QwQ-32B-preview across both commonsense and mathematical reasoning tasks, with particularly strong gains on mathematical benchmarks.

Dataset	Method	Acc. (%)	AvgTok/Q
CSQA	Direct	89.4	1140
CSQA	CoT	88.4	1701
ARC-Challenge	Direct	98.2	849
ARC-Challenge	CoT	97.2	1380
GSM8K	Direct	98.8	1207
GSM8K	CoT	97.4	1853
GSM-Hard	Direct	78.0	2926
GSM-Hard	CoT	77.4	3560

Table 5: Qwen3.5-Plus commercial reasoning reference with default thinking enabled.

Strong Reasoning Model Reference. To contextualize DiMo against newer strong reasoning models, we additionally evaluate Qwen3.5-Plus with its default thinking mode enabled. We treat this model as an external reference rather than a matched-backbone DiMo baseline, since its reasoning is performed internally through hidden thinking tokens. As table 5 shows, Qwen3.5-Plus achieves strong direct-prompting accuracy: 89.4% on CSQA, 98.2% on ARC-Challenge, 98.8% on GSM8K, and 78.0% on GSM-Hard. Explicit CoT prompting does not further improve performance and instead slightly decreases accuracy while increasing token cost. This suggests that Qwen3.5-Plus already performs substantial internal reasoning under direct prompting. In contrast, DiMo ex-

Table 6: Ablation of Divergent Thinking Mode (accuracy %).

Setting	CSQA	ARC-Challenge
DiMo (full)	80.02%	84.1%
w/o knowledge	46.7%	48.1%
w/o reasoning path	51.5%	51.4%

Table 7: Ablation of Logical Mode (accuracy %).

Setting	GSM8K Acc.	GSM-Hard Acc.
DiMo (full)	90.7%	71.4%
w/o evaluator	82.8%	36.7%
w/o refiner	77.4%	35.4%
w/o judger	82.5%	48.07%

ternalizes intermediate hypotheses, critiques, and refinements as auditable multi-agent traces. Notably, on GSM-Hard, DiMo with Qwen2.5-32B surpasses the Qwen3.5-Plus, suggesting that explicit multi-agent verification can still provide benefits over stronger single-model hidden reasoning on high-precision arithmetic tasks.

5 Discussion

5.1 Ablation Study

We ablate key components to isolate their contributions in both thinking modes.

Divergent Thinking Mode. Removing either the knowledge supporter (w/o knowledge) or the reasoning-path provider (w/o reasoning path) causes a clear performance drop on both CSQA and ARC-Challenge (table 6), indicating that explicit knowledge injection and structured reasoning paths are both necessary for robust commonsense reasoning; knowledge contributes slightly more in our current setup.

Logical Thinking Mode. Ablating the evaluator, refiner, and judger(table 7) shows that the refiner is the primary driver of GSM8K accuracy (largest drop when removed, while also saving tokens), highlighting the importance of iterative refinement for math. On GSM-Hard, removing the judger increases accuracy but substantially raises token consumption, revealing a steep cost-accuracy trade-off; this motivates our default choice of keeping the judger to control computation under budgets.

5.2 Different Tasks Require Different Thinking Modes

Our results suggest a protocol–task affinity: commonsense and knowledge-based tasks benefit more

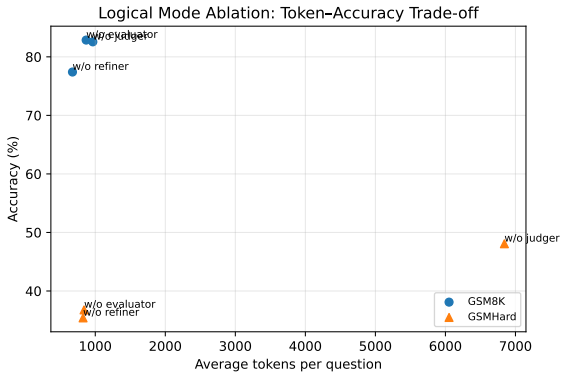

Figure 5: Logical Mode token–accuracy scatter.

Table 8: The Accuracy of DiMo on CSQA by Using Different Prompts. No-CoT initialization exceeds CoT on CSQA, highlighting prompt design's impact on convergence and accuracy.

METHOD	PROMPTS	CSQA
DIMO (LLAMA-3.1-8B)	COT	75.81%
	NO COT	80.02%

from a Divergent Thinking Mode that expands candidate knowledge and hypotheses, whereas mathematical tasks benefit more from a Logical Thinking Mode that emphasizes step-wise deduction. When we force GSM8K and GSM-Hard into the Divergent Thinking Mode, accuracy drops noticeably compared to the Logical Mode (table 4), even falling below a single-model CoT baseline on GSM8K in our setting. This indicates that outcomes are sensitive to interaction rules and selecting an appropriate thinking mode can improve accuracy under fixed models, prompts, and budgets.

5.3 Different Initialization Prompts

Prompt design also matters: on CSQA, initializing the Generator with CoT decreases final accuracy compared to direct-answer initialization (table 8). A possible explanation is that longer intermediate rationales introduce noisier anchors for later refinement, while concise initial proposals are easier for agents to verify and correct.

5.4 Number of Debate Rounds

Varying debate rounds (fig. 6) shows diminishing returns beyond a small number of rounds. OpenBookQA and StrategyQA stabilize after about one round, while GSM8K and GSM-Hard improve more consistently and peak around three rounds in our setting.

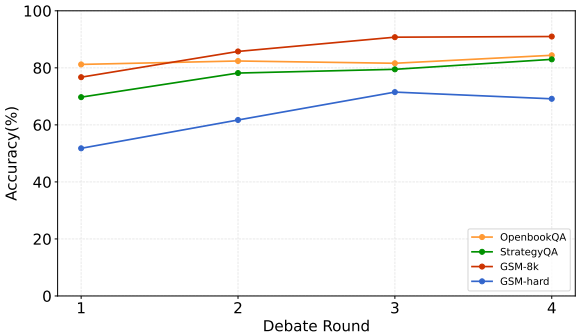

Figure 6: Accuracy of DiMo with Different Debate Rounds. Accuracy improves up to about three rounds, and the subsequent upward trend has leveled off.

Table 9: Accuracy–token trade-off of DiMo across debate rounds (500 examples). We report accuracy and average tokens per question (input+output, summed across agents and rounds). Round 0 indicates the initial answer before any debate.

Dataset	Backbone	Acc. (%) r=0/1/3	Avg Tokens/Q r=0/1/3
CSQA	LLaMA-3.1-8B Instruct	74.6/85.2/86.6	3897/5987/9111
CSQA	Qwen2.5-32B Instruct	81.4/86.2/86.4	3992/5557/8370
GSM8K	LLaMA-3.1-8B Instruct	76.0/69.6/79.8	684/4419/7854
GSM8K	Qwen2.5-32B Instruct	82.0/97.2/99.2	730/1334/1504

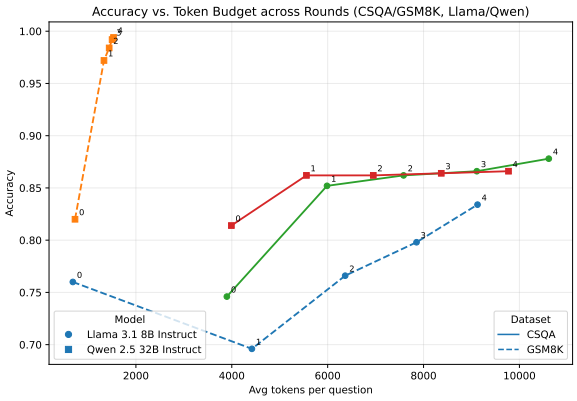

Figure 7: Pareto Curves of Accuracy versus Average Token Per Question.

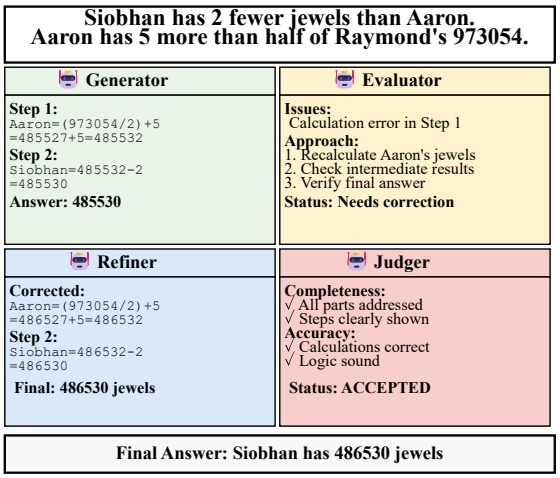

Figure 8: Case study of DiMo on GSM-hard dataset. The Evaluator flags errors, the Refiner applies targeted fixes, and the Judger finalizes a transparent solution.

5.5 Tasks Token Counting for Computational Consumption

We use token usage as a hardware-agnostic proxy for inference cost, counting both input and output tokens and summing over all agents and rounds to reflect the true compute of multi-agent deliberation. We treat initial independent generation as round 0 (no interaction), and start debate interactions from round 1; fig. 7 shows the accuracy–token Pareto curves across rounds, and table 9 reports representative rounds. Overall, commonsense tasks obtain most gains from one debate round and then saturate, whereas math tasks exhibit stronger accuracy–cost scaling across rounds. Based on this trade-off, we set default rounds to 3 in our main experiments to capture most gains while avoiding the marginal improvement and higher token cost at round 4.

5.6 Case Study

We present a GSM-Hard example in fig. 8: the Evaluator localizes an error, the Refiner applies a targeted fix, and the Judger finalizes a transparent solution. This staged interaction provides auditable intermediate states; accordingly, we use "interpretability" to mean process transparency rather than mechanistic interpretability of base models. A natural next step is developing task-agnostic quantitative metrics for evaluating the quality of traces.

6 Conclusion

We propose DiMo, a multi-agent collaboration framework that instantiates two operational thinking modes:a Divergent Thinking Mode for commonsense reasoning and a Logical Thinking Mode for mathematical reasoning. Across benchmarks under a unified budget, DiMo improves accuracy on most datasets while producing explicit intermediate traces that make hypotheses and revisions inspectable. Our analyses highlight that protocol choice, module design, and round budgets are task-dependent.

7 Limitations

DiMo introduces additional compute and latency due to multi-agent interaction, and its effectiveness can vary with the backbone model, prompting, and the quality of judging/verifying components. We focus on a limited set of datasets and answer formats, and we do not claim mechanistic interpretability—only process-level auditability. Future work includes learning automatic mode routing, developing more principled metrics for trace quality, and extending evaluation to harder benchmarks and formats (e.g., MATH (Saxton et al., 2019) and GPQA (Rein et al., 2023)), as well as other reasoning types such as symbolic reasoning.

References

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Lupita Estefania Gazzo Castañeda, Benjamin Sklarek, Dennis E Dal Mas, and Markus Knauff. 2023. Probabilistic and deductive reasoning in the human brain. *NeuroImage*, 275:120180.

Hyeong Kyu Choi, Xiaojin Zhu, and Sharon Li. 2025. Debate or vote: Which yields better decisions in multi-agent large language models? *arXiv preprint arXiv:2508.17536*.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Antonia Creswell, Murray Shanahan, and Irina Higgins. 2022. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv preprint arXiv:2205.09712*.

DeepSeek-AI, D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, and 1 others. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv:2501.12948. ArXiv preprint.

Y. Du, S. Li, A. Torralba, J. B. Tenenbaum, and I. Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig. 2022. PAL: Program-aided language models. *arXiv preprint arXiv:2211.10435*.

M. Geva, D. Khashabi, E. Segal, T. Khot, D. Roth, and J. Berant. 2021. Did Aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

T. Guo, X. Chen, Y. Wang, R. Chang, S. Pei, N. V. Chawla, O. Wiest, and X. Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*.

Z. Jiang, F. F. Xu, L. Gao, Z. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, J. Callan, and G. Neubig. 2023. Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*.

Kimi Team, A. Du, B. Gao, B. Xing, C. Jiang, C. Chen, C. Li, and 1 others. 2025. Kimi K1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*.

T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35:22199–22213.

T. Liang, Z. He, W. Jiao, X. Wang, Y. Wang, R. Wang, Y. Yang, S. Shi, and Z. Tu. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, and 1 others. 2024. Deepseek-V3 technical report. *arXiv preprint arXiv:2412.19437*.

Haoyan Luo and Lucia Specia. 2024. From understanding to utilization: A survey on explainability for large language models. *arXiv preprint arXiv:2401.12874*.

A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, and 1 others. 2024. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46547.

Meta AI. 2024. Llama 3 model card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Daniel Osherson, Daniela Perani, Stefano Cappa, Tatiana Schnur, Franco Grassi, and Ferruccio Fazio. 1998. Distinct brain loci in deductive versus probabilistic reasoning. *Neuropsychologia*, 36(4):369–376.

Qwen Team. 2024. QwQ: Reflect deeply on the boundaries of the unknown. https://qwenlm.github.io/blog/qwq-32b-preview/.

D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. 2023. GPQA: A graduate-level google-proof Q&A benchmark. *arXiv preprint arXiv:2311.12022*.

M. Renze and E. Guven. 2024. Self-reflection in LLM agents: Effects on problem-solving performance. *arXiv preprint arXiv:2405.06682*.

D. Saxton, E. Grefenstette, F. Hill, and P. Kohli. 2019. Analysing mathematical reasoning abilities of neural models. *arXiv preprint arXiv:1904.01557*.

N. Shinn, F. Cassano, A. Gopinath, K. Narasimhan, and S. Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652.

A. Talmor, J. Herzig, N. Lourie, and J. Berant. 2018. CommonsenseQA: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

H. Wang, X. Du, W. Yu, Q. Chen, K. Zhu, Z. Chu, L. Yan, and Y. Guan. 2023. Apollo's oracle: Retrieval-augmented reasoning in multi-agent debates. *arXiv preprint arXiv:2312.04854*.

Ruochen Wang, Si Si, Felix Yu, Dorothea Wiesmann, Cho-Jui Hsieh, and Inderjit Dhillon. 2024. Large language models are interpretable learners. *arXiv preprint arXiv:2406.17224*.

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

J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

K. Xiong, X. Ding, Y. Cao, T. Liu, and B. Qin. 2023. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. *arXiv preprint arXiv:2305.11595*.

A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

S. Yao, D. Yu, J. Zhao, I. Shafran, T. Griffiths, Y. Cao, and K. Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822.

H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, and M. Du. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38.

A Role Prompts

Table 10: The Role Descriptions and Prompts for Different Agents in Divergent Thinking Mode of MA4DTM

Role	Prompt
Evaluator	You are an answer evaluator for multiple-choice commonsense reasoning questions. Analyze the given answer for: 1. Logical reasoning 2. Consideration of all options 3. Understanding of common sense principles 4. Clear justification for the chosen answer Please point out any errors, omissions, or areas that need improvement. Focus on whether the reasoning is sound and leads to a logical conclusion.
Knowledge Supporter	You are a knowledge supporter for multiple-choice commonsense reasoning questions. Your role is to: 1. Identify relevant commonsense knowledge needed for the question 2. Provide real-world examples and context 3. Highlight any missing or incorrect information 4. Support or challenge the reasoning with factual information Focus on practical, everyday knowledge that relates to the question and choices.
Reasoning Path Provider	You are a reasoning path provider for multiple-choice commonsense reasoning questions. Create clear logical paths that: 1. Break down the question into key components 2. Analyze each option systematically 3. Show clear steps leading to the answer 4. Explain why incorrect options are eliminated 5. Connect evidence to conclusions Ensure each step follows logically from the previous one and leads to a clear choice.

table 10 presents the role sample prompts for agents in the Divergent Thinking Mode of MA4DTM. These sample prompts are used for the CommonsenseQA dataset.

B Dataset Format

B.1 Commonsense Reasoning Datasets Details

Table 11: CommonsenseQA Sample Format

Components	Content
Question	A revolving door is convenient for two direction travel, but it also serves as a security measure at a what?
Question Concept	revolving door
Choices	{ "label": ["A", "B", "C", "D", "E"], "text": ["bank", "library", "department store", "mall", "new york"] }
Answer Key	A

Table 12: ARC-Challenge Sample Format

Components	Content
Question	An astronomer observes that a planet rotates faster after a meteorite impact. Which is the most likely effect of this increase in rotation?
Choices	{ "text": ["Planetary density will decrease.", "Planetary years will become longer.", "Planetary days will become shorter.", "Planetary gravity will become stronger."], "label": ["A", "B", "C", "D"] }
Answer Key	C

Table 13: OpenbookQA Sample Format

Components	Content
Question	When food is reduced in the stomach
Choices	{ "text": ["the mind needs time to digest", "take a second to digest what I said", "nutrients are being deconstructed", "reader's digest is a body of works"], "label": ["A", "B", "C", "D"] }
Answer Key	C
Fact	digestion is when stomach acid breaks down food
Humanscore	1
Clarity	1.6

Table 14: StrategyQA Sample Format

Components	Content
Question	Was ship that recovered Apollo 13 named after a World War II battle?
Answer	True
Term	Apollo 13
Description	A failed crewed mission to land on the Moon
Facts	Apollo 13 was recovered by the USS Iwo Jima. Iwo Jima was captured from the Imperial Japanese Army during World War II by the US in a conflict called the Battle of Iwo Jima.

tables 11 to 14 present the commonsense reasoning datasets sample that used in our experiments. Our datasets consist of three multiple-choice dataset and one binary judgment dataset.

B.2 Mathematical Reasoning Datasets Details

Table 15: GSM8K Sample Format

Components	Content
Question	Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?
Answer	Janet sells 16 - 3 - 4 = «16-3-4=9»9 duck eggs a day. She makes 9 * 2 = $«9*2=18»18 every day at the farmer's market. #### 18

Table 16: GSM-hard Sample Format

Components	Content
Input	Every day, Wendi feeds each of her chickens three cups of mixed chicken feed, containing seeds, mealworms and vegetables to help keep them healthy. She gives the chickens their feed in three separate meals. In the morning, she gives her flock of chickens 6887483 cups of feed. In the afternoon, she gives her chickens another 25 cups of feed. How many cups of feed does she need to give her chickens in the final meal of the day if the size of Wendi's flock is 20 chickens?
Target	-6,887,448

tables 15 and 16 present the mathematical reasoning datasets sample used in our experiments.