COMMUNICATION AS EVIDENCE ROUTING: ADAPTIVE RELATIONAL GRAPHS VIA OPTIMAL TRANSPORT FOR MULTI-AGENT LLM REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-agent large language model (LLM) systems offer a promising paradigm for complex reasoning by integrating complementary expertise across multiple models. However, existing agent communication strategies rely on either free-form interaction or answer-level aggregation, making reasoning susceptible to dominant-answer collapse, where prevailing answers prematurely suppress valuable dissenting evidence. To address this, we reinterpret multi-agent communication as evidence routing and propose Adaptive Relational Graphs via Optimal Transport (ARGOT), a relational graph framework for multi-agent LLM reasoning. ARGOT represents agents' initial responses as evidence distributions and uses optimal transport to construct a task-adaptive graph comprising support and challenge relations. These typed relations enable conflict-aware message passing, allowing consistent and conflicting evidence to be propagated and integrated differently during reasoning. A trajectory-level verifier then selects the final answer by evaluating routed evidence and reasoning trajectories, rather than relying solely on answer frequency. Across six reasoning benchmarks, ARGOT achieves the highest average performance and ranks first on four tasks. Further analyses show it preserves informative disagreement, improves evidence aggregation, and mitigates premature consensus. These results establish structured evidence routing as an effective, principled approach to more reliable multi-agent LLM reasoning.

1 INTRODUCTION

Large language model (LLM) reasoning research shifts from single-model prompting toward agentic systems that decompose tasks, invoke tools, and coordinate actions Wu et al. (2025b); Ferrag et al. (2026); Xu et al. (2025). Multi-agent reasoning advances this by coordinating agents with distinct domain expertise, assumptions, and reasoning styles Tran et al. (2025); Guo et al. (2024); Wu et al. (2024); Chen et al. (2024b); Hong et al. (2024). Such collaboration is valuable when individual agents' reliability varies across tasks, domains, and problem settings, as heterogeneous agents contribute complementary capabilities and specialized expertise.

Existing multi-agent systems employ chain-based Wei et al. (2022); Hong et al. (2024), tree-based Wang et al. (2023); Yao et al. (2023), multi-path Du et al. (2024); Liang et al. (2024); Wang et al. (2024a), or graph-based structures Besta et al. (2024); Yao et al. (2024); Liu et al. (2024); Zhuge et al. (2024); Yun et al.

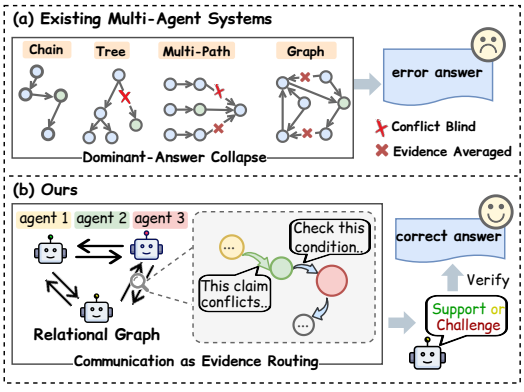

Figure 1: From answer aggregation to evidence routing. (a) Existing Systems: Uniform communication overlooks conflicts and leads to dominant-answer collapse. (b) Ours: Relational communication preserves useful disagreement through support–challenge reasoning.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(2026); Zhang et al. (2025), as summarized in Figure 1(a). Despite their different topologies, communication can remain *conflict-blind*, treating supporting and opposing messages alike, or simply *average evidence* without assessing the reliability of individual claims. This lack of distinction can obscure informative disagreement and amplify misleading consensus. Consequently, agents may converge on a dominant but incorrect answer, while a decisive minority trajectory with stronger supporting evidence is overlooked or suppressed. We refer to this failure mode as *dominant-answer collapse* Smit et al. (2024); Wu et al. (2025a).

Figure 1(b) illustrates our alternative: agents communicate over a relational graph that treats communication as evidence routing. Rather than uniformly forwarding complete answers, agents expose claim-level relations, such as "this claim conflicts," and request targeted checks, such as "check this condition." The verifier then interprets the routed evidence through support and challenge roles, reinforcing consistent evidence while retaining disagreement for inspection. This process allows a correct answer to be selected from validated evidence rather than from the dominant response alone.

We instantiate this paradigm with Adaptive Relational Graphs via Optimal Transport (ARGOT), a relational framework for multi-agent LLM reasoning. ARGOT selects task-relevant agents, represents their responses as task-adaptive evidence distributions, and uses optimal transport to construct support and challenge relations. Support routes shared evidence for consolidation, whereas challenge preserves confident disagreement for inspection. After relation-conditioned message passing, ARGOT constructs evidence cards and verifies the answer from routed evidence rather than vote count. This lets a precise, task-grounded minority trajectory influence the final decision.

In summary, this work makes the following contributions:

- We propose ARGOT, a task-local relational graph framework for multi-agent LLM reasoning that selects relevant agents, constructs support and challenge relations over responses, and routes evidence to the final decision rather than relying solely on answer counts.
- We introduce an OT-guided relational communication mechanism that distinguishes supportive and challenging evidence and enables conflict-aware message passing.
- Across six general-domain and professional-domain benchmarks, ARGOT achieves the best average performance and ranks first on four of six tasks, with its strongest gains arising when agent outputs are noisy, inconsistent, or dominated by misleading majority answers.

2 Related Work

Interaction-based reasoning. Chain-based, tree-based, and multi-path structures are widely used in multi-agent reasoning. Chain-of-thought and self-consistency elicit or sample trajectories, with the latter voting over their answers Wei et al. (2022); Wang et al. (2023). Multi-agent debate, ReConcile, and Mixture-of-Agents instead exchange critiques or aggregate layered responses Du et al. (2024); Liang et al. (2024); Chen et al. (2024a); Wang et al. (2024a); however, debate can remain vulnerable to conformity and majority pressure Smit et al. (2024); Wu et al. (2025a). ARGOT retains the full trajectories but explicitly labels incoming evidence as support or challenge before verification.

Thought-graph-based reasoning. Graph-based structures have been introduced for more flexible reasoning. Graph of Thoughts represents thoughts as vertices and dependencies as edges, supporting generation, aggregation, refinement, and feedback Besta et al. (2024). GoT encodes a thought graph and integrates it with the input through a learned gate Yao et al. (2024). Both model reasoning states and dependencies. ARGOT instead treats agents and trajectories as nodes, using transport costs to infer support/challenge edges that guide communication. Thus, ARGOT's graph serves as a communication policy over agents rather than a graph over thoughts.

Adaptive-graph-based collaboration. Recent work further explores adaptive graph construction for multi-agent systems. DyLAN ranks agents, GPTSwarm optimizes computational graphs, GoA prunes a relevance graph, and G-Designer generates task-adaptive structures Liu et al. (2024); Zhuge et al. (2024); Yun et al. (2026); Zhang et al. (2025). Although these approaches improve structural adaptivity, they mainly determine which agents or connections should participate without explicitly distinguishing consistent from conflicting evidence. ARGOT instead constructs task-specific support and challenge relations to enable conflict-aware evidence routing during collaboration.

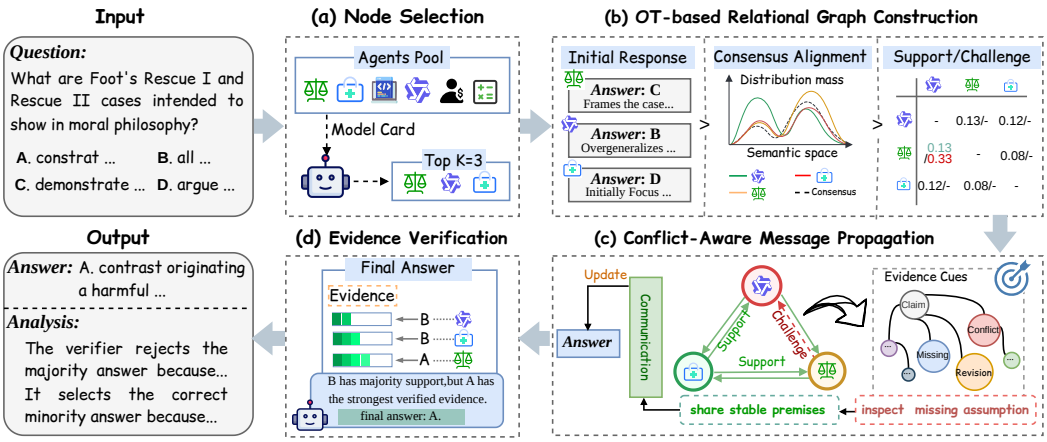

Figure 2: Overview of ARGOT. (a) Node Selection: Selects task-relevant agents using model cards to form an adaptive agent pool. (b) OT-Based Relational Graph Construction: Converts agent responses into evidence distributions and builds support/challenge relations via optimal transport. (c) Conflict-Aware Message Propagation: Routes evidence through relation-conditioned communication while preserving useful disagreement. (d) Evidence Verification: Aggregates and verifies routed evidence trajectories to generate the final answer.

3 Methodology

3.1 Overview and Task Formulation

Given a question x and heterogeneous agent pool $\mathcal{A} = \{a_1, \ldots, a_m\}$, ARGOT formulates reasoning as task-local relational evidence selection. Each agent has a model card describing its domain, specialization, and capabilities. ARGOT selects a query-dependent subset $\mathcal{S} \subseteq \mathcal{A}$ with $|\mathcal{S}| = k$, obtains an initial trajectory y_i^0 from each agent, and parses an answer $\hat{z}_i$ and confidence estimate c_i. As shown in Figure 2, it constructs a relational graph, propagates support and challenge evidence, and verifies the trajectories to produce a traceable final answer.

Definition 1 (Task-local Relational Agent Graph)**.** A Task-local Relational Agent Graph is defined as $\mathcal{G}_x = (\mathcal{S}, \mathcal{E}, \mathcal{W}, \mathcal{T})$, where $\mathcal{S}$ denotes the agents selected for question x, $\mathcal{E}$ denotes directed communication edges, $\mathcal{W}$ denotes relation scores, and $\mathcal{T}$ denotes semantic relation types.

This task-local graph supports subsequent communication and decision making. ARGOT makes the final prediction by comparing trajectories on $\mathcal{G}_x$ rather than simply counting answer strings.

3.2 Node Selection

Following the Node Sampling step in GoA Yun et al. (2026), the first module selects a small set of task-relevant agents from the candidate pool. Each agent is described by a model card, and a meta agent ranks candidates by domain match, task specialization, and generalist coverage. This produces a selected set $\mathcal{S} = \{a_i\}_{i=1}^{k}$ and serves as the task-aware front end of ARGOT. The selection is formulated as maximizing a prior-regularized utility:

$$\mathcal{S}^\star = \underset{\mathcal{S}\subseteq\mathcal{A},|\mathcal{S}|=k}{\operatorname{argmax}} \sum_{a\in\mathcal{S}} u(a,x) + \lambda_{\text{cov}}\text{Cov}(\mathcal{S}), \tag{1}$$

where $u(a,x)$ denotes the relevance of agent a to question x, and $\text{Cov}(\mathcal{S})$ encourages coverage by including a generalist agent when appropriate. The top-k agents are queried independently before inter-agent communication, producing an initial trajectory y_i^0, parsed answer $\hat{z}_i$, and confidence score c_i. These independent trajectories serve as the evidence inputs for OT graph construction.

Confidence source and calibration. In all experiments, c_i is *self-reported*: the prompt explicitly requires JSON containing a rationale, a parseable answer, and a scalar confidence score in $[0, 1]$. The parser accepts valid values and otherwise substitutes 0.5. No labels, held-out calibration set,

temperature scaling, or post-hoc calibrator are used; thus, c_i serves as a bounded uncertainty proxy rather than a calibrated probability of correctness. It shapes the evidence distribution and diversity term, while correctness is assessed later from the complete reasoning trajectories.

3.3 OT-BASED RELATIONAL GRAPH CONSTRUCTION

This module converts agent responses into comparable evidence distributions and constructs a task-adaptive relational graph in OT space. For multiple-choice tasks, the selected option receives confidence c_i, while others share $1 - c_i$; unparsed answers use uniform distributions. For open-ended tasks, responses are represented as normalized token distributions from task-specific vocabularies. The ground-cost matrix $\mathcal{D}$ is derived from option similarity or token overlap, enabling ARGOT to interpret OT costs as semantic divergence and uncertainty-aware signals Oliver et al. (2026).

Definition 2 (Agent Evidence Distribution)**.** An Agent Evidence Distribution $\mathcal{P}_i$ represents the initial response of agent a_i in a task-adaptive OT space. For a multiple-choice item with $\mathcal{L}$ options, each probability mass is defined as

$$\mathcal{P}_i(\ell) = \begin{cases} c_i, & \ell = \hat{z}_i, \\ \frac{1-c_i}{\mathcal{L}-1}, & \ell \neq \hat{z}_i. \end{cases} \tag{2}$$

This definition separates answer identity from confidence. Agents with different parsed answers can remain close when their mass lies on semantically related options, while agents with the same answer remain distinguishable by confidence or supporting content. For code and open-ended responses, identifiers, answer-bearing phrases, symbolic operations, and task-relevant tokens form normalized feature distributions. The ground cost measures the cost of moving evidence between features rather than treating all disagreements equally.

For clarity, we distinguish entropy-regularized transport optimization from the nonnegative graph distance. Given distributions $\mathcal{P}$ and $\mathcal{Q}$, Sinkhorn iterations solve

$$\Gamma^\star_\epsilon(\mathcal{P}, \mathcal{Q}) = \operatorname*{argmin}_{\Gamma \in U(\mathcal{P},\mathcal{Q})} \langle \Gamma, \mathcal{D} \rangle + \epsilon \sum_{u,v} \Gamma_{uv}(\log \Gamma_{uv} - 1), \tag{3}$$

where $U(\mathcal{P}, \mathcal{Q})$ imposes $\sum_v \Gamma_{uv} = \mathcal{P}_u$ and $\sum_u \Gamma_{uv} = \mathcal{Q}_v$. The graph then uses only the transport component

$$C_{\epsilon,\mathcal{D}}(\mathcal{P}, \mathcal{Q}) = \langle \Gamma^\star_\epsilon(\mathcal{P}, \mathcal{Q}), \mathcal{D} \rangle \geq 0. \tag{4}$$

The implementation uses $\mathcal{K} = \exp(-\mathcal{D}/\epsilon)$ with $\epsilon = 0.05$. Thus, the potentially negative entropy term affects $\Gamma^\star_\epsilon$ but is not itself treated as a nonnegative distance. $C_{\epsilon,\mathcal{D}}$ is a regularized transport *cost*, not a metric: it need not satisfy identity of indiscernibles, and ARGOT uses only its relative magnitude under fixed ϵ to compare response alignment Peyré & Cuturi (2019). The plan acts as a soft alignment between evidence units; low-cost mass indicates similar answer-bearing content, whereas high-cost mass indicates substantive disagreement. OT is therefore a relational signal rather than a correctness estimator.

The transport costs serve two roles. First, a weighted reference distribution measures each agent's divergence from the task-local consensus. Let $\bar{b}_i$ denote the normalized response-relevance score from peer scoring. We define the reference distribution and agent-wise transport cost:

$$\mathcal{P}_{\text{ref}} = \sum_i \bar{b}_i \mathcal{P}_i, \qquad d_i = C_{\epsilon,\mathcal{D}}(\mathcal{P}_i, \mathcal{P}_{\text{ref}}). \tag{5}$$

These reference-relative costs yield consensus reliability and a confidence-weighted diversity bonus:

$$r_i = \bar{b}_i e^{-\lambda d_i}, \qquad q_i = \bar{b}_i c_i (1 - e^{-\lambda d_i}), \qquad \rho_i = r_i + \beta q_i, \quad 0 < \beta < 1. \tag{6}$$

Here r_i is consensus reliability, q_i is a confidence-weighted diversity bonus, β is the diversity-retention coefficient, and ρ_i is their combined node reliability. A small d_i indicates alignment with the reference; a large d_i together with high c_i identifies a potentially useful challenger. The normalized communication score is $s_i = \text{norm}(0.70\bar{b}_i + 0.30\,\text{norm}(\rho_i))$.

The subsequent edge construction uses ρ_i to retain reliable disagreement, so its behavior must be established before it can serve as a graph score. In particular, transport divergence should expose a possible challenge without increasing overall node reliability, and the diversity contribution must remain bounded. Proposition 1 proves these properties, thereby justifying the use of ρ_i in constructing $\mathcal{W}$ and the downstream challenge relations in $\mathcal{E}$.

Proposition 1 (Divergence-aware score behavior). *Assume $\bar{b}_i > 0$, $c_i \in [0,1]$, $\lambda > 0$, $0 < \beta < 1$, and $d_i \geq 0$. For the scores in Equation (6), r_i and ρ_i are strictly decreasing functions of d_i, whereas q_i is nondecreasing and is strictly increasing when $c_i > 0$. Moreover,*

$$\beta \bar{b}_i c_i \leq \rho_i \leq \bar{b}_i, \qquad \frac{q_i}{r_i} = c_i\big(e^{\lambda d_i} - 1\big). \tag{7}$$

Consequently, larger transport divergence increases the relative contribution of diversity while keeping the combined node reliability bounded.

Proof. Differentiating the score definitions with respect to d_i gives

$$\frac{\partial r_i}{\partial d_i} = -\lambda \bar{b}_i e^{-\lambda d_i} < 0, \quad \frac{\partial q_i}{\partial d_i} = \lambda \bar{b}_i c_i e^{-\lambda d_i} \geq 0, \quad \frac{\partial \rho_i}{\partial d_i} = -\lambda \bar{b}_i (1 - \beta c_i) e^{-\lambda d_i} < 0. \tag{8}$$

Because $c_i \in [0,1]$ and $0 < \beta < 1$, the derivatives establish the stated monotonicity. Also, $\rho_i = \bar{b}_i[\beta c_i + (1 - \beta c_i)e^{-\lambda d_i}]$; hence $e^{-\lambda d_i} \in (0,1]$ gives the claimed bounds. Finally,

$$\frac{q_i}{r_i} = c_i\big(e^{\lambda d_i} - 1\big), \qquad \square$$

Proposition 1 shows that disagreement cannot increase an agent's overall reliability. Instead, transport divergence shifts the score composition toward a bounded diversity term, allowing confident minority evidence to remain visible without treating disagreement as evidence of correctness.

Second, pairwise costs quantify support salience, whereas agent-to-reference costs identify challenge relations. For active agents with $\bar{b}_i > t$,

$$e_{ij}^{\mathrm{sup}} = \frac{\sqrt{r_i r_j}}{1 + C_{\epsilon,\mathcal{D}}(\mathcal{P}_i, \mathcal{P}_j)}, \qquad e_{i \to c}^{\mathrm{cha}} = \rho_i (1 + d_i), \tag{9}$$

where $a_c = \arg\min_i d_i$ is the consensus anchor. Active agents are connected by base score, with e_{ij}^{sup} denoting support salience. The most diverse disagreeing agent is linked to a_c by a protected challenge edge, preserving informative disagreement under pruning. The resulting query-dependent graph avoids forcing full consensus or diversity. These e scores are graph metadata for relation strength, not message coefficients.

This construction instantiates every component of Definition 1: $\mathcal{S}$ contains the selected agents; $\mathcal{E}$ contains the retained directed support and challenge links; $\mathcal{W}$ stores the corresponding scores e_{ij}^{sup} or $e_{i \to c}^{\mathrm{cha}}$; and $\mathcal{T}$ assigns each edge its support, challenge, or protected-challenge type. Thus, $\mathcal{G}_x$ is the explicit output of OT-based graph construction and the input to message propagation.

3.4 Conflict-Aware Message Propagation

Message propagation is performed over $\mathcal{G}_x$ through relation-conditioned prompts. For each target $a_j \in \mathcal{S}$, its retained in-neighborhood is induced by $\mathcal{E}$, the edge label $\tau_{ij} \in \mathcal{T}$ specifies how incoming evidence is interpreted, and the corresponding entry in $\mathcal{W}$ records its relational salience. Relation scores determine edge retention and remain graph metadata, whereas the strengths of surviving messages are normalized from the source-node scores:

$$\alpha_{ij} = \frac{s_i}{\sum_{a_k \in \mathcal{N}(j)} s_k}, \tag{10}$$

where $\mathcal{N}(j)$ is the retained in-neighborhood. Therefore, OT affects communication through edge existence and type and through the OT component of s_i, but the pairwise score e_{ij} is not multiplied a second time. This distinction separates relation construction from message weighting. The first phase updates each target using its initial response and weighted relational evidence:

$$y_j^1 = \mathbf{LLM}_j(x, y_j^0, \{(\alpha_{ij}, \tau_{ij}, y_i^0)\}_{i \in \mathcal{N}(j)}). \tag{11}$$

The reverse phase then returns the refined target responses to their source agents:

$$y_i^2 = \mathbf{LLM}_i(x, y_i^0, \{(\alpha_{ji}, \tau_{ji}, y_j^1)\}_{j \in \mathcal{N}(i)}). \tag{12}$$

These equations describe the actual execution pattern. No neural graph network is trained. Support edges consolidate compatible premises, whereas challenge edges prompt the receiver to inspect disputed claims, missing conditions, or revision cues. When the two conflict, the prompt favors task-grounded justification over vote count. Protected challenges remain visible in the reverse phase, allowing potentially useful minority trajectories to be reconsidered.

3.5 Evidence Verification

After propagation, evidence verification replaces direct voting by assessing candidate-level trajectories. The verifier uses Qwen2.5-7B-Instruct Qwen Team et al. (2025), the same meta-agent as GoA, and bases final decisions on trajectory quality, stability, and graph support.

Definition 3 (Trajectory Evidence Card)**.** A Trajectory Evidence Card is defined as

$$\mathcal{C}_i = (a_i, s_i, c_i, \hat{z}_i^0, \hat{z}_i^2, \delta_i, y_i^2), \tag{13}$$

where a_i denotes the agent, s_i denotes the graph weight, c_i denotes confidence, $\hat{z}_i^0$ and $\hat{z}_i^2$ denote the initial and post-message answers, $\delta_i = \mathbf{1}[\hat{z}_i^0 = \hat{z}_i^2]$ denotes answer stability, and y_i^2 denotes the final trajectory.

For multiple-choice tasks, candidate-level evidence is further summarized as

$$\mathcal{N}_z = \sum_i \mathbf{1}[\hat{z}_i^2 = z], \quad \mathcal{M}_z = \sum_i s_i \mathbf{1}[\hat{z}_i^2 = z], \tag{14}$$

where $\mathcal{N}_z$ is the support count and $\mathcal{M}_z$ is graph-weighted support. Neither statistic alone decides the output.

The final decision stage first synthesizes an anchor response through graph pooling, then asks a task-family-specific verifier to compare the anchor with all trajectory evidence cards:

$$z^\star = \mathbf{Verifier}(x, \bar{y}, \{\mathcal{C}_i\}_{i=1}^k). \tag{15}$$

For choice tasks, the verifier selects among anchor-supported or post-communication candidates; for math, finance, and code, it checks algebraic, unit, signature, syntax, and logic constraints. Parsing failures use task-specific deterministic fallbacks. Verification compares candidate groups by correctness, premise specificity, answer stability, and relation effects, allowing precise minority corrections while rejecting unsupported claims. This separates correctness assessment from routing and avoids treating confidence or OT cost as correctness.

4 Experiments

4.1 Experiment Setup

Benchmarks and Metrics. ARGOT is evaluated on six benchmarks: MMLU-Pro Wang et al. (2024b), GPQA Rein et al. (2023), AIME Dekoninck et al. (2026) from AIME 2023–2024, HumanEval Chen et al. (2021a), FinQA Chen et al. (2021b), and MedMCQA Pal et al. (2022). MMLU-Pro samples 70 examples from each of 14 categories, MedMCQA samples 1,000 examples with a fixed seed, and other datasets use their original splits. Metrics are accuracy for choice tasks, exact match for AIME and FinQA, and pass@1 for HumanEval.

Baselines. The comparison includes SC Wang et al. (2023), Refine Madaan et al. (2023), ReConcile Chen et al. (2024a), Debate Du et al. (2024); Liang et al. (2024), MoA Wang et al. (2024a), and GoA Yun et al. (2026). GoA and ARGOT dynamically select agents from the full pool. Fixed-agent baselines use {`qwen`, `mathstral`, `qwen_coder`}, replacing `qwen_coder` with `biomedical_llama` for MedMCQA and `finance_llama` for FinQA.

Implementation Details. The agent pool contains six LLMs with 7B–8B parameters: Qwen2.5-7B-Instruct Qwen Team et al. (2025), Qwen2.5-Coder-7B-Instruct Hui et al. (2024), Mathstral-7B-v0.1 Mistral AI (2024), Bio-Medical-Llama-3-8B ContactDoctor (2024), finance-Llama3-8B Instruction Pre-Training Team (2024), and Saul-7B-Instruct Colombo et al. (2024). ARGOT uses top-k selection with $k = 3$, graph threshold $t = 0.05$, one communication round, temperature 0.7, maximum output length 800, seed 0, OT-causal graph construction, and trajectory-verification pooling. As clarified in Section 3.2, all reported runs use uncalibrated self-reported confidence.

4.2 Main Results

Table 1 summarizes the main comparison across six benchmarks covering general-domain and professional-domain reasoning. The best result in each column is shown in bold, and the second-best result is underlined. Three key observations emerge from these results:

Table 1: Main results on general- and professional-domain benchmarks. All numbers are percentages. Best in **bold**; second-best underlined. The GoA row reports the $\mathrm{GoA}_{\mathrm{Mean}}$ variant with a 3-model pool; $\mathrm{GoA}_{\mathrm{Max}}$ is lower on every dataset and 4.19 points lower on average (see supplementary material). GoA and ARGOT use dynamic agent selection from the full heterogeneous pool, while fixed-agent baselines use task-dependent triples described in the experimental setup.

	General Domain				Professional Domain	
Benchmark	MMLU-Pro	GPQA	AIME	HumanEval	FinQA	MedMCQA
Default Quality Metric (%)	Accuracy	Accuracy	Exact Match	Pass@1	Exact Match	Accuracy
SC (ICLR '23)	49.69	34.34	8.89	55.49	6.63	50.80
Refine (NeurIPS '23)	45.51	33.84	6.67	84.76	17.52	48.30
ReConcile (ACL '24)	53.06	32.32	**15.56**	65.85	16.48	51.20
Debate (ICML '24)	53.27	37.88	6.67	83.54	17.52	53.00
MoA (ICLR '25)	54.08	33.84	11.11	85.37	17.44	54.08
GoA (ICLR '26)	50.61	35.35	8.89	**87.80**	18.31	53.30
ARGOT (ours)	**55.71**	**39.90**	13.33	85.98	**18.74**	**55.20**

1) Strong overall performance under conflicting evidence. ARGOT achieves the highest average score across all tasks (44.81%), surpassing the strongest baseline average achieved by MoA (42.65%), and ranks first on four benchmarks. The largest gains occur on MMLU-Pro and GPQA. These results align with ARGOT's design: OT-based relations distinguish broad consensus from high-confidence counter-evidence, while trajectory verification preserves precise minority reasoning during final selection rather than merging it into the majority answer group.

2) Consistent gains across general and professional domains. ARGOT leads on MMLU-Pro, GPQA, FinQA, and MedMCQA, and ranks second on AIME and HumanEval. Its gains on FinQA and MedMCQA are 0.43 and 1.12 points, respectively, suggesting that routing domain-specific biomedical or financial evidence can recover useful constraints overlooked by the majority.

3) Clear limitations on strictly verifiable tasks. ReConcile leads on AIME and GoA on HumanEval, with ARGOT second on both. These tasks require exact symbolic or executable correctness, where direct refinement or graph pooling can better preserve strong candidates. ARGOT remains competitive but is limited by language-level verification, showing its strongest advantage when conflicting trajectories must be judged from evidence rather than execution feedback.

4.3 Ablation Study

To further evaluate the contribution of each component in ARGOT, a single-module ablation study is conducted on GPQA and HumanEval. Four variants are constructed by removing one key design choice at a time. *w/o agent selection* replaces prior-regularized node selection with random selection to examine the effect of task-aware agent choice. *w/o OT graph* replaces the OT-based adaptive communication graph with a deterministic fully connected directed graph. *w/o challenge edges* retains the consensus-supporting edges in the OT graph but removes challenge and protected-challenge edges. *w/o evidence verification* removes the final trajectory-level evidence verification module and uses majority voting over agent outputs for final selection. Except for the ablated component, all other settings, including the agent configuration, communication rounds, meta LLM, and decoding parameters, are kept unchanged.

Table 2: Ablation study on ARGOT modules.

Variant	GPQA	HumanEval
ARGOT	**39.90**	**85.98**
w/o agent selection	33.33	82.32
w/o OT graph	36.87	84.15
w/o challenge edges	36.36	83.54
w/o evidence verification	34.34	69.51

The ablation results reveal three observations. (1) Removing any module lowers performance on both benchmarks, confirming that agent selection, OT graph construction, challenge edges, and evidence verification provide complementary signals. The largest GPQA drop occurs without agent selection, from 39.90% to 33.33%, suggesting that task-relevant specialists are important for difficult

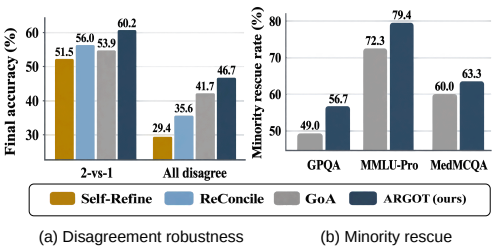

Figure 3: Evidence routing under (a) disagreement and (b) minority-correct cases.

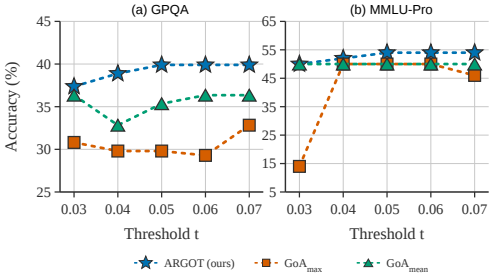

Figure 4: Threshold sensitivity on GPQA and MMLU-Pro.

scientific questions. (2) Evidence verification is the most critical component on HumanEval, where pass@1 falls from 85.98% to 69.51% when it is replaced by majority voting. This indicates that code tasks require trajectory-level checking because a majority-supported implementation can still fail exact syntax, signature, or logic constraints. (3) Removing the OT graph or challenge edges causes smaller but consistent degradation, showing that conflict-aware prompts and the verifier can partly compensate, while OT-based relation weights and protected disagreement still provide useful routing signals before communication.

4.4 Effectiveness of Evidence Routing

This auxiliary experiment evaluates whether evidence routing helps in cases where simple aggregation is unreliable and majority signals may be misleading. Figure 3(a) tests disagreement robustness on MMLU-Pro by separating questions according to the initial answer pattern, which exposes whether a method can reason under weak consensus. Figure 3(b) tests minority rescue, where the correct answer appears in the initial agent set but is not the majority response.

As shown in Figure 3(a), ARGOT is most effective when initial agreement is weak. It achieves 60.2% accuracy in the two-against-one setting and 46.7% in the all-disagree setting, outperforming Self-Refine, ReConcile, and GoA in both cases. The advantage is larger for all-disagree inputs, where majority cues are unavailable and communication must determine which trajectory contains stronger evidence. This result supports the role of typed support and challenge relations, as messages are used to compare evidence quality rather than simply amplify the largest answer group.

Figure 3(b) further indicates that these gains arise from preserving useful minority signals. ARGOT improves the minority-rescue rate over GoA on GPQA, MMLU-Pro, and MedMCQA, achieving its highest value on MMLU-Pro (79.4%). These datasets contain close distractors or domain-specific clues, meaning that a correct minority trajectory may provide decisive evidence despite being initially unpopular. This result supports the intended mechanism in which disagreement is routed and verified as evidence rather than treated as noise to be averaged away.

4.5 Hyperparameter Sensitivity

The threshold t is a general control parameter for deciding which agents remain active for inter-agent communication. A smaller t keeps more low-relevance agents and enables broader information exchange, while a larger t prunes uncertain participants and makes the communication graph more selective. This parameter therefore mainly affects graph sparsity, evidence exposure, and the risk of propagating noisy disagreement, making it a useful probe of whether a method depends on a narrow communication setting.

Figure 4 shows that ARGOT remains stable as t varies on both GPQA and MMLU-Pro. In contrast, $\mathrm{GoA}_{\mathrm{Max}}$ shows abrupt accuracy changes at some thresholds, while $\mathrm{GoA}_{\mathrm{Mean}}$ still fluctuates despite averaging over the pool. This difference arises because GoA's aggregation is more directly affected by which messages survive thresholding; a small change in t can switch the active neighborhood and alter the pooled trajectory. ARGOT uses t mainly to control relation exposure, while OT-based support/challenge typing, protected challenge edges, and trajectory-level verification keep reliable evidence available after pruning. The final decision is therefore less sensitive to the exact threshold.

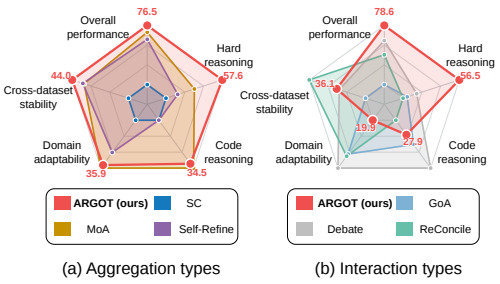

Figure 5: Comprehensive capability profiles for (a) aggregation and (b) interaction methods. Higher standardized scores indicate stronger relative capability.

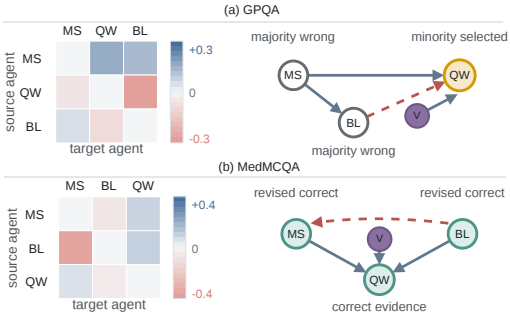

Figure 6: Conflict resolution on GPQA and MedMCQA. (a) GPQA: Correct minority evidence is preserved against majority errors. (b) MedMCQA: Interaction refines conflicting responses toward correct evidence.

4.6 Comprehensive Capability Profile

The capability profiles summarize overall performance, hard and code reasoning, domain adaptability, and cross-dataset stability. Scores are standardized within each method group, centered at 50, and clipped to $[20, 90]$, so the plots show relative rather than raw performance.

Figure 5 gives ARGOT the strongest aggregation profile, including overall performance of 76.5 and hard reasoning of 57.6. This broad gain is consistent with assigning support/challenge roles and verifying routed trajectories instead of directly merging answers. Among interaction methods, ARGOT also leads in overall performance (78.6) and hard reasoning (56.5). GoA remains stronger on code reasoning, while Debate or ReConcile can be steadier under repeated critique. ARGOT's advantage is therefore clearest when conflicting trajectories must be judged and smaller on exact-execution or conservative-refinement tasks.

4.7 Case Study

Figure 6 illustrates how ARGOT handles two minority-correct examples. In the GPQA case, the initial responses form an incorrect majority against a single correct minority trajectory. The OT graph preserves the minority agent as a high-confidence challenger rather than suppressing it under majority pressure, enabling the verifier to select the more reliable reasoning path. In the MedMCQA case, the initial responses again contain one correct minority signal, but communication propagates the corrective evidence to the other agents and converts the post-communication group into a correct consensus; the verifier then selects from the aligned evidence group rather than relying on a single agent. These examples demonstrate two complementary effects of evidence routing: ARGOT can either preserve an informative minority trajectory or propagate its evidence when other agents can be corrected through relation-aware messages.

5 Conclusion

This paper introduced ARGOT, a relational graph framework that uses optimal transport to build task-adaptive support and challenge relations, route evidence through conflict-aware communication, and verify trajectories before aggregation. Across six benchmarks, ARGOT achieved the best average performance and ranked first on four tasks; ablations showed complementary gains from graph construction, challenge edges, and verification. Minority-rescue analyses further support preserving useful disagreement as structured evidence. ARGOT remains limited by uncalibrated self-reported confidence and language-level verification. Future work will calibrate uncertainty and incorporate task-specific validators for strictly verifiable tasks.

AI USE STATEMENT

In this work, we did not use generative AI tools for research tasks requiring disclosure; we used generative AI tools only for language polishing, grammar checking, and organizing the presentation of the authors' research ideas. We reviewed all AI-assisted work to ensure that it accurately reflects the original methodology and results. We take responsibility for the final content of this work, including text produced with the aid of generative AI.

ETHICS STATEMENT

This work involves no human subjects or private personal data. Multi-agent LLM systems may still produce errors, biases, and overconfident outputs, particularly in medical and financial settings; therefore, our results should not be interpreted as support for high-stakes deployment without expert validation.

REPRODUCIBILITY STATEMENT

All experiments were run on four NVIDIA A100 GPUs. The complete code and execution environment are provided in the uploaded code supplement.

REFERENCES

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17682–17690, 2024. doi: 10.1609/aaai.v38i16.29720.

Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. ReConcile: Round-table conference improves reasoning via consensus among diverse LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7066–7085. Association for Computational Linguistics, 2024a. doi: 10.18653/v1/2024.acl-long.381.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021a.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. In *International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=EHg5GDnyq1.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3697–3711. Association for Computational Linguistics, 2021b. doi: 10.18653/v1/2021.emnlp-main.300.

Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, André F. T. Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, and Michael Desa. Saullm-7b: A pioneering large language model for law, 2024.

ContactDoctor. Bio-medical-llama-3-8b. Hugging Face Model Hub, 2024. URL https://huggingface.co/ContactDoctor/Bio-Medical-Llama-3-8B.

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: Matharena as an evaluation platform for mathematics with llms, 2026. URL https://arxiv.org/abs/2605.00674.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *International Conference on Machine Learning*, 2024.

Mohamed Amine Ferrag, Norbert Tihanyi, and Merouane Debbah. From llm reasoning to autonomous ai agents: A comprehensive review. *IEEE Access*, 2026.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*, 2024.

Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VtmBAGCN7o.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, Kai Dang, Fan Yang, Yichang Zhang, An Yang, Rui Men, Fei Huang, Bo Zheng, Yibo Miao, Shanghaoran Quan, Yunlong Feng, Xingzhang Ren, Xuancheng Ren, Jingren Zhou, and Junyang Lin. Qwen2.5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Instruction Pre-Training Team. finance-llama3-8b. Hugging Face Model Hub, 2024. URL https://huggingface.co/instruction-pretrain/finance-Llama3-8B.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17889–17904. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.992.

Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. A dynamic llm-powered agent network for task-oriented agent collaboration. In *First Conference on Language Modeling*, 2024.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, 2023.

Mistral AI. Mathstral. https://mistral.ai/news/mathstral/, 2024.

Mary Chriselda Antony Oliver, Lan Jiang, Aaron Bundi Anampiu, Elaf Almahmoud, Francesco Quinzan, and Umang Bhatt. Learning to orchestrate agents under uncertainty, 2026.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pp. 248–260. PMLR, 2022.

Gabriel Peyré and Marco Cuturi. Computational optimal transport with applications to data science. *Foundations and Trends in Machine Learning*, 11(5–6):355–607, 2019. doi: 10.1561/2200000073.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023.

Andries P. Smit, Nathan Grinsztajn, Paul Duckworth, Thomas D. Barrett, and Arnu Pretorius. Should we be going MAD? a look at multi-agent debate strategies for LLMs. In *International Conference on Machine Learning*, 2024.

Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O'Sullivan, and Hoang D Nguyen. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint arXiv:2501.06322*, 2025.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. Mixture-of-agents enhances large language model capabilities. *arXiv preprint arXiv:2406.04692*, 2024a.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2024b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Haolun Wu, Zhenkun Li, and Lingyao Li. Can LLM agents really debate? a controlled study of multi-agent debate in logical reasoning, 2025a.

Junde Wu, Jiayuan Zhu, Yuyuan Liu, Min Xu, and Yueming Jin. Agentic reasoning: A streamlined framework for enhancing llm reasoning with agentic tools. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28489–28503, 2025b.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation. In *First Conference on Language Modeling*, 2024.

Fengli Xu, Qianyue Hao, Chenyang Shao, Zefang Zong, Yu Li, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, et al. Toward large reasoning models: A survey of reinforced reasoning with large language models. *Patterns*, 6(10), 2025.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822, 2023.

Yao Yao, Zuchao Li, and Hai Zhao. Got: Effective graph-of-thought reasoning in language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2901–2921. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-naacl.183.

Sukwon Yun, Jie Peng, Pingzhi Li, Wendong Fan, Jie Chen, James Zou, Guohao Li, and Tianlong Chen. Graph-of-agents: A graph-based framework for multi-agent LLM collaboration. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=34cANdsHKV.

Guibin Zhang, Yanwei Yue, Xiangguo Sun, Guancheng Wan, Miao Yu, Junfeng Fang, Kun Wang, Tianlong Chen, and Dawei Cheng. G-designer: Architecting multi-agent communication topologies via graph neural networks. In *International Conference on Machine Learning*, 2025.

Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. Gptswarm: Language agents as optimizable graphs. In *International Conference on Machine Learning*, 2024.

A Appendix

B Experimental Configuration

All experiments were conducted on a single Ubuntu 24.04.3 LTS GPU server with $4\times$ NVIDIA A100-PCIE-40GB GPUs, dual Intel Xeon Gold 5218R CPUs, and 251 GiB system memory.

C Algorithm

Algorithm 1 summarizes ARGOT's test-time inference pipeline. It selects task-relevant agents, builds OT-based evidence relations, and verifies relation-aware communication to produce the final answer. No agent model is trained or updated during this procedure.

Algorithm 1: Overall Inference Pipeline of ARGOT

Input: Question x, agent pool $\mathcal{A}$, model cards $\mathcal{M}$, top-k, threshold t, OT coefficient ϵ

Output: Final prediction $z^\star$

Phase I: Agent Selection
Rank agents by task relevance and coverage.
Select $\mathcal{S} \leftarrow \mathrm{TopK}(\mathcal{A}, x, k)$.
Query selected agents; parse $(y_i^0, \hat{z}_i^0, c_i)$.
Phase II: OT Relational Graph
Map responses to evidence distributions $\{\mathcal{P}_i\}$.
Build ground cost $\mathcal{D}$ and reference $\mathcal{P}_{\mathrm{ref}}$.
Compute OT distances, node reliabilities, and anchor a_c.
Derive support and challenge scores.
Retain edges above t plus protected challenges.
Form $\mathcal{G}_x = (\mathcal{S}, \mathcal{E}, \mathcal{W}, \mathcal{T})$.
Phase III: Message Propagation
Normalize incoming relational weights.
Revise targets with support/challenge evidence.
Return refined target responses to source agents.
Obtain final trajectories $\{y_i^2\}$.
Phase IV: Evidence Verification
Build trajectory evidence cards $\{\mathcal{C}_i\}$.
Synthesize graph-pooled anchor $\bar{y}$.
$z^\star \leftarrow \mathbf{Verifier}(x, \bar{y}, \{\mathcal{C}_i\}_{i=1}^k)$.
return $z^\star$

D Theoretical Guarantees

We establish two complementary properties of relation-aware communication: normalized incoming influence and guaranteed exposure to selected minority evidence.

Proposition D.1 (Normalized and scale-invariant incoming influence)**.** *For a target agent a_j, let $\mathcal{N}(j)$ be its nonempty retained in-neighborhood and assume $s_i > 0$ for every $a_i \in \mathcal{N}(j)$. Define*

$$\alpha_{ij} = \frac{s_i}{\sum_{a_k \in \mathcal{N}(j)} s_k}. \tag{D.1}$$

Then the incoming weights satisfy

$$\alpha_{ij} > 0, \qquad \sum_{a_i \in \mathcal{N}(j)} \alpha_{ij} = 1, \qquad \frac{\alpha_{ij}}{\alpha_{\ell j}} = \frac{s_i}{s_\ell} \quad (a_i, a_\ell \in \mathcal{N}(j)). \tag{D.2}$$

Moreover, replacing every incoming score by $s_i' = \gamma s_i$ for any $\gamma > 0$ leaves all α_{ij} unchanged. Hence, the total incoming influence assigned to a target is conserved, while the relative priority among its retained sources is preserved.

Proof. Because all incoming scores are positive and $\mathcal{N}(j)$ is nonempty, the denominator in equation D.1 is positive, so $\alpha_{ij} > 0$. Summing over the neighborhood gives

$$\sum_{a_i \in \mathcal{N}(j)} \alpha_{ij} = \frac{\sum_{a_i \in \mathcal{N}(j)} s_i}{\sum_{a_k \in \mathcal{N}(j)} s_k} = 1. \tag{D.3}$$

For any two incoming agents, cancellation of the common denominator yields $\alpha_{ij}/\alpha_{\ell j} = s_i/s_\ell$. Finally,

$$\frac{s'_i}{\sum_{a_k \in \mathcal{N}(j)} s'_k} = \frac{\gamma s_i}{\gamma \sum_{a_k \in \mathcal{N}(j)} s_k} = \alpha_{ij}, \tag{D.4}$$

which proves invariance to a common positive rescaling. □

Proposition D.1 ensures that graph density or a common change in score scale cannot increase the total normalized communication mass received by an agent. The graph determines which evidence is exposed, whereas normalized node scores determine its relative priority within that retained context.

Proposition D.2 (Protected minority preservation)**.** *Let $\mathcal{A}_t = \{a_i : \bar{b}_i > t\}$ be the active agents, let a_c be the consensus anchor, and suppose the disagreeing set $\mathcal{D}_c = \{a_i \in \mathcal{A}_t : \hat{z}_i \neq \hat{z}_c\}$ is nonempty. If $a_h \in \arg\max_{a_i \in \mathcal{D}_c} q_i$ is assigned a protected challenge edge (a_h, a_c) and $s_h > 0$, then threshold pruning cannot remove this edge. Consequently, $a_h \in \mathcal{N}(c)$, $\alpha_{hc} > 0$, and the initial trajectory y_h^0 is included in the first-phase communication context of the consensus anchor.*

Proof. The retained edge set has the form

$$\mathcal{E} = \{(a_i, a_j, \tau_{ij}) : e_{ij}^{\tau} \geq t\} \cup \mathcal{E}_{\text{prot}}. \tag{D.5}$$

Because $(a_h, a_c) \in \mathcal{E}_{\text{prot}}$, equation D.5 retains the edge independently of its thresholded score, so $a_h \in \mathcal{N}(c)$. The normalized message weight satisfies

$$\alpha_{hc} = \frac{s_h}{\sum_{a_k \in \mathcal{N}(c)} s_k} > 0, \tag{D.6}$$

where positivity follows from $s_h > 0$ and the nonnegative node scores. Therefore, the anchor update explicitly contains $(\alpha_{hc}, \tau_{hc}, y_h^0)$ in its incoming relational context. Thus, protected-challenge pruning guarantees exposure to the selected minority trajectory, although the verifier must still determine whether that trajectory is correct. □

E Detailed Experimental Setup

E.1 Backbones

ARGOT uses six instruction-tuned backbone models. Table 3 lists the model pool and the role of each model in the system.

Table 3: Backbone language models used in ARGOT.

Model	Params	Role
Qwen2.5-7B-Instruct	7.61B	general
Qwen2.5-Coder-7B-Instruct	7.61B	code
Mathstral-7B-v0.1	7B	math
Bio-Medical-Llama-3-8B	8B	biomedical
finance-Llama3-8B	8B	finance
Saul-7B-Instruct-v1	7B	legal

E.2 Benchmarks and Metrics

ARGOT is evaluated on six benchmarks covering mathematical reasoning, code generation, general scientific question answering, graduate-level reasoning, financial question answering, and medical reasoning. The benchmark suite is designed to test whether evidence routing remains useful across both general-domain and professional-domain multi-agent settings. Table 4 summarizes the split and metric used for each benchmark.

Table 4: Benchmarks and metrics used in ARGOT.

Benchmark	Split	Metric
MMLU-Pro	70×14 subset	Accuracy
GPQA	Test split	Accuracy
AIME 2023–2024	AIME 2023–2024	Match
HumanEval	Test split	Pass@1
FinQA	Test split	Match
MedMCQA	1,000 examples	Accuracy

MMLU-Pro evaluates broad multiple-choice reasoning across academic and professional subjects. We include it because it requires stronger selection among close options, which makes it suitable for testing whether ARGOT can resolve disagreement with evidence rather than with majority vote alone.

GPQA evaluates difficult graduate-level scientific question answering. We include it because the questions are intentionally high-uncertainty and often contain plausible distractors, which makes naive agreement less reliable.

AIME 2023–2024 evaluates open-ended mathematical problem solving: each problem requires a numerical answer rather than a choice among options. We use exact match, making it a strict test of whether ARGOT can preserve precise reasoning under a harder verification setting.

HumanEval evaluates code generation ability. We include it because the benchmark checks whether the final output is executable, which is useful for testing trajectory-level verification rather than answer aggregation alone.

FinQA evaluates financial question answering with numerical reasoning over financial text. We include it because it contains domain-specific constraints and calculations, which make specialist evidence useful when general agents are not sufficient.

MedMCQA evaluates medical multiple-choice reasoning. We include it because the benchmark benefits from domain knowledge and provides a good setting for checking whether task-specific evidence can improve final selection.

E.3 BASELINES

Table 5: Baseline families and their roles.

Method	Type	Role
Self-Consistency	sampling	vote
Refine	refinement	self-feedback
ReConcile	consensus	synthesis
Debate	critique	challenge
Mixture-of-Agents	layering	stacking
GoA	graph	message passing

Table 5 summarizes the principal coordination families considered in our comparison, covering sampling aggregation, trajectory refinement, debate-style interaction, layered synthesis, and graph-based communication. These baselines span different levels of interaction, from independent trajectory sampling to explicit multi-agent communication and structured message passing, providing a broad comparison across representative coordination strategies. All methods are evaluated on the same benchmark splits under consistent experimental settings. For fixed-agent baselines, we further use the same task-dependent backbone pool whenever their original pipelines permit, reducing performance differences caused by model selection rather than coordination mechanisms.

Self-Consistency serves as the simplest aggregation baseline. It samples multiple candidate reasoning trajectories and selects the most frequent final answer, providing a direct measure of how effective answer-level voting can be in the absence of explicit inter-agent communication. Refine provides the closest single-trajectory refinement baseline. It iteratively revises a reasoning trajectory through self-feedback, allowing us to examine whether repeated self-correction within a single reasoning path can achieve gains comparable to those obtained through explicit multi-agent coordination.

ReConcile and Debate represent two distinct interaction styles. ReConcile focuses on building consensus through iterative discussion, whereas Debate emphasizes explicit challenge and rebuttal among agents. Mixture-of-Agents instead follows a layered synthesis strategy, combining multiple responses through successive aggregation stages rather than simple voting. GoA serves as the primary graph-based coordination baseline in our comparison. It organizes agents through a directed

communication graph and applies pooling over selected agents, making it the closest structural counterpart to ARGOT among the evaluated baselines.

E.4 Implementation Details

All models are served through vLLM using bf16 weights and a tensor-parallel size of 1 on the same Ubuntu 24.04.3 LTS server used throughout the experiments. Unless otherwise specified, candidate generation uses a temperature of 0.7, top-p of 1.0, a maximum output length of 800, top-k selection with $k = 3$, a graph threshold of $t = 0.05$, one communication round, and seed 0. These settings are applied consistently to ARGOT and the baselines whenever the corresponding method supports the same decoding controls. For baselines with fixed coordination pipelines, we preserve their original interaction structures while aligning only the agent pool and compatible decoding parameters, so that their method-specific coordination procedures remain unchanged.

Table 6: Default inference settings.

Setting	Value	Role
Temperature	0.7	generation diversity
Top-p	1.0	nucleus sampling
Max output length	800	response budget
Selected agents k	3	graph size
Graph threshold t	0.05	edge pruning
Communication rounds	1	message passing depth
Seed	0	reproducibility

F Additional Experiment Results

F.1 Comparison with GoA (Graph-of-Agents)

GoA is a graph-based multi-agent collaboration framework that organizes coordination around a selected agent set, a directed communication graph, and graph-level pooling. It first picks task-relevant agents, then connects them with relevance-guided edges, and lets agents exchange messages along the graph before pooling the final trajectories into one output. In our comparison, GoA and ARGOT use the same heterogeneous agent pool and dynamic agent selection setting. We report both GoA pooling variants: $\mathrm{GoA}_{\mathrm{Mean}}$, where the final response is synthesized from graph-weighted agent outputs, and $\mathrm{GoA}_{\mathrm{Max}}$, where the highest-scored agent output is selected by max pooling.

Table 7 reports a controlled comparison between ARGOT and two GoA variants. The $\mathrm{GoA}_{\mathrm{Mean}}$ row corresponds to the GoA configuration reported in the main results, whereas $\mathrm{GoA}_{\mathrm{Max}}$ provides an additional variant that changes only the pooling rule. This comparison helps disentangle the contribution of GoA's aggregation strategy from that of its graph construction, making the source of performance differences more interpretable. All methods are evaluated on the same benchmark splits with percentage-scale metrics and otherwise matched experimental settings, ensuring that the comparison primarily reflects differences in coordination and aggregation design.

Table 7: Direct comparison with GoA variants. All numbers are percentages. AIME combines AIME 2023 and AIME 2024.

Method	MMLU	GPQA	AIME	HEval	FinQA	MedMCQA	Avg.
$\mathrm{GoA}_{\mathrm{Mean}}$	50.61	35.35	8.89	**87.80**	18.31	53.30	42.38
$\mathrm{GoA}_{\mathrm{Max}}$	43.88	33.33	6.67	84.15	12.99	48.10	38.19
ARGOT (ours)	**55.71**	**39.90**	**13.33**	85.98	**18.74**	**55.20**	**44.81**

ARGOT achieves the best average score, improving over $\mathrm{GoA}_{\mathrm{Mean}}$ by 2.43 points and over $\mathrm{GoA}_{\mathrm{Max}}$ by 6.62 points. The gain is most consistent on MMLU-Pro, GPQA, AIME, FinQA, and MedMCQA, where conflicting or domain-specific evidence often determines the final answer. HumanEval is the only benchmark where $\mathrm{GoA}_{\mathrm{Mean}}$ is higher, suggesting that graph pooling can preserve a strong candidate program when exact executable behavior is the main bottleneck. Across all six benchmarks, $\mathrm{GoA}_{\mathrm{Max}}$ is lower than $\mathrm{GoA}_{\mathrm{Mean}}$ and 4.19 points lower on average, indicating that selecting a single highest-scored trajectory is brittle under heterogeneous-agent outputs. ARGOT differs from both GoA variants by routing support and challenge evidence before verification, which reduces reliance on either a dominant pooled response or a single selected trajectory.

F.2 Hyperparameter Sensitivity

The query-selection coefficient λ_{sel} controls how many task-relevant agents are retained before communication. The horizontal axis in Figure 7(a)–(b) abbreviates λ_{sel} as λ. This coefficient is distinct from the divergence-decay coefficient λ in Equation (6) and the coverage coefficient λ_{cov} in Equation (1). ARGOT remains stable across the tested range on GPQA and HumanEval, with the best performance appearing at moderate values of λ_{sel}. This suggests that retaining too few agents weakens complementary evidence, while retaining too many agents introduces redundant or noisy trajectories. In other words, ARGOT benefits from a balanced selection budget rather than an extreme one.

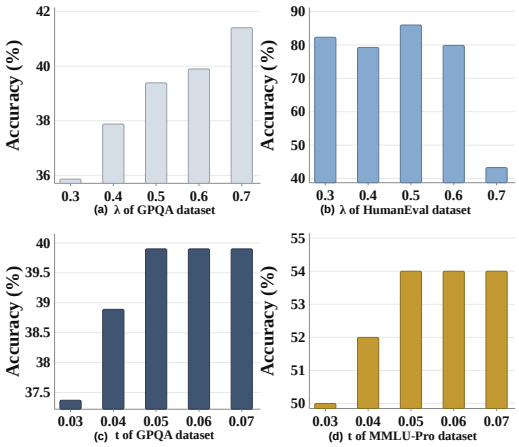

Figure 7: Sensitivity to the query-selection coefficient λ_{sel} (shown as λ in the plot) and graph threshold t.

The graph threshold t controls which inter-agent relations are kept in the relational graph. Figure 7(c)–(d) shows a similarly smooth trend on GPQA and MMLU-Pro: performance changes gradually as t varies, and no sharp collapse appears when the graph becomes slightly denser or sparser. Low thresholds preserve more weak relations but can also carry extra noise, whereas high thresholds prune useful disagreement together with uncertain links. Overall, the results indicate that ARGOT is robust to reasonable settings of either parameter, and that moderate filtering is sufficient to preserve useful evidence while suppressing weak relations.

F.3 Case Study

The following cases are taken from the recorded test runs and identified by their dataset indices. They illustrate three distinct outcomes of evidence routing: preserving a correct minority, completing a numerical computation, and propagating a correction to initially wrong agents.

The first record contains an incorrect two-agent majority. The correct trajectory remains visible as a challenge and is selected from its task-grounded derivation rather than its vote count.

GPQA case study (idx 4)

Question: For a spin-1/2 muon measured along a direction in the x–z plane, which option gives the normalized eigenvector associated with $+\hbar/2$? Option B is $(\cos(\theta/2), \sin(\theta/2))$.
Initial answers: Mathstral = A; Qwen = B; Bio-Medical-Llama = A.
Interaction shift: The B trajectory remains distinct under a challenge relation while the two A trajectories retain their majority. Evidence verification identifies the half-angle spinor form supporting B.
Final answer: B (gold: B).

The second record shows how a correct numerical trace can survive alongside an unevaluated expression and an incorrect alternative. Communication turns the expression into the correct value, after which verification retains the supported result.

FinQA case study (idx 0)

Question: Entergy reports net revenue of 5829 million in 2015 and 5735 million in 2014. What is the net change in 2015?
Initial answers: Finance-Llama = 6; Qwen = $5829 - 5735$; Mathstral = 94.
Interaction shift: Qwen evaluates its expression and changes to 94, while Mathstral remains at 94. The remaining incorrect trajectory does not override their consistent arithmetic evidence.
Final answer: 94 (gold: 94).

The third record demonstrates corrective propagation. A single correct answer initially opposes a two-agent wrong majority, but both incorrect trajectories revise after receiving the routed evidence.

MedMCQA case study (idx 190)

Question: Which tooth-numbering system depicts the 32 permanent teeth consecutively as 1--32? Options: A Universal; B Palmer; C Haderup; D diagrammatic depiction.
Initial answers: Bio-Medical-Llama = B; Qwen = A; Qwen-Coder = B.
Interaction shift: Both B trajectories revise to A, producing the post-communication answers A, A, and A. The verifier links consecutive numbering to the Universal system.
Final answer: A (gold: A).

F.4 Qualitative Error Analysis

The recorded failures reveal two different limitations: the verifier can prefer an incorrect explanation even when the correct candidate is available, and communication can consolidate an incorrect candidate when no reliable correction is present. A final HumanEval record illustrates a low-diversity boundary rather than an error.

The first failure is a verifier-selection error. None of the three answers changes during communication, and the correct candidate has the largest graph weight, yet the verifier accepts an incorrect uncertainty-principle calculation.

GPQA boundary case (idx 0): wrong verification

Question: Two energy levels have lifetimes 10^{-9} and 10^{-8} seconds. Which separation clearly resolves them? Options: A 10^{-8} eV; B 10^{-4} eV; C 10^{-9} eV; D 10^{-11} eV.
Initial answers: Mathstral = B; Qwen = A; Bio-Medical-Llama = D. The post-communication answers remain B, A, and D.
Interaction shift: The verifier incorrectly treats A as the closest scale derived from its calculation, despite the correct B trajectory remaining available with the largest graph weight.
Final answer: A (gold: B).

The second failure shows incorrect consolidation on a mathematical problem. One trajectory changes toward an already stable wrong answer, after which the verifier accepts their agreement without recovering the gold construction.

AIME24 boundary case (idx 23): wrong convergence

Question: A list of positive integers sums to 30, has unique mode 9, and has an integer median that does not appear in the list. Find the sum of the squares of its entries.
Initial answers: The agents independently generate the numerical answers 175 and 244; the remaining response does not yield a usable final number. These are free-form answers, not predefined options.
Interaction shift: The 175 trajectory revises to 244, while another trajectory remains at 244. Their agreement is then accepted even though it is incorrect.
Final answer: 244 (gold: 236).

The final record is a low-diversity boundary rather than a failure. Two candidates are textually identical all-pairs implementations, and the third uses the equivalent strategy of sorting before comparing adjacent values. Their small OT separation leaves little substantive disagreement to route.

HumanEval/0 boundary case: near-duplicate programs

Question: has_close_elements(numbers, threshold) returns whether any pair of numbers is closer than the threshold.
Initial answers: Candidates 0 and 2 use identical pairwise comparisons; candidate 1 sorts the input and compares adjacent elements.
Interaction shift: Because all three programs implement the same core condition, the graph provides little corrective signal and the verifier primarily acts as a selector.
Final answer: Candidate 0 is selected and passes the HumanEval tests.

G PROMPTS

This section summarizes the actual ARGOT call chain. In the implementation, selection must return indices, generation must return a parseable answer and confidence, graph construction must expose typed relations, and the later stages must preserve the task output format. The same instruction skeleton is reused across benchmark families, so the wording has to be specific enough to guide the model, but still generic enough to survive changes in task type, answer format, and evidence layout.

Each agent first produces an initial response together with a confidence signal. This is the only point where ARGOT asks the model to answer directly before any communication begins, so the goal is to capture an early trajectory and a confidence estimate that later stages can reuse as evidence.

Initial Response and Confidence

Input: Question x, task type, and a model-specific instruction.
Output: A brief rationale, a parseable answer, and confidence.
Provide only a few key reasoning sentences. Derive confidence from the model's output uncertainty. Keep the answer machine-readable so later parsing can recover the final label or numerical value.

The selector then chooses a small but diverse working set of agents. In practice, this filters the full pool down to a manageable subgraph. Relevance matters most, but the set still needs enough diversity that OT-based relations have something useful to compare and route.

Agent Selection

Input: Question x, agent pool $\mathcal{A}$, model cards $\mathcal{M}$, top-k.
Output: Selected indices $\mathcal{S}$.
Rank agents by task relevance, expertise match, and coverage. Prefer complementary specialists and generalists over near-duplicate choices. If the question is difficult, keep at least one broad-coverage model in the pool. Return indices only, in ranked order, without prose.

The selected responses are converted into a task-local OT graph before communication starts. This is the point where raw answer strings become structured evidence objects, and where typed relations start to matter more than surface agreement.

OT Relational Graph

Input: Parsed responses, task evidence, OT distance, threshold t.
Output: A relational graph with support and challenge edges.
Use label-space transport for choice tasks and feature-space transport for open responses. Type agreement as support, conflict as challenge, and reliable disagreement as protected challenge. Prune weak edges with t, but keep an anchor and the evidence needed to explain corrections.

Communication then runs in two directions so corrections can circulate. The first pass lets a target model absorb evidence from its neighbors, while the second pass lets source-side models recover information from the updated targets. That first pass is where the most local repairs happen, because the target sees which neighboring traces actually support its own reasoning and which ones only look similar on the surface. The second pass then keeps those repairs from being lost when the information flows back to the source side. The two directions are intentionally separated so that each prompt specifies one refinement stage at a time and follows the actual order in which evidence is moved and reused.

Source-to-Target Refinement

Input: Question x, target response, weighted relation messages.
Output: A revised target trajectory.
Use support evidence to reinforce shared premises. Use challenge evidence to inspect contradictions, missing assumptions, and wrong steps. Revise only when the evidence actually changes the target answer.

The extra pass is also useful because a correction often becomes clearer only after the target response has already absorbed its neighbors. Once that local revision is made, the source models can revisit the same evidence with a cleaner view of what changed, which helps preserve the strongest revised trace instead of letting it disappear in the next round.

Target-to-Source Consolidation

Input: Question x, refined target responses, the same relation context.
Output: Source-side revised trajectories.
Let the source models absorb useful corrections from the target side. Keep the strongest trace when the graph is sparse. Avoid turning the second pass into a majority vote over the revised answers.

The final answer is assembled from the refined trajectories and then checked once more against the task format. This stage is intentionally narrow: it should not restart the whole reasoning process, only select the best-supported trajectory and package it in the expected output format. At this point, the prompt is less about generating new reasoning and more about keeping the synthesis faithful to the routed evidence. The stage therefore acts like a checkpoint: it compresses the interaction history without erasing the decisive correction that made the final answer possible. The transition from synthesis to verification is also where ARGOT stops expanding the evidence set and starts compressing it into a task-specific reference for final checking.

Anchor Synthesis

Input: Refined trajectories, weights, and the current evidence set.
Output: A graph-pooled reference anchor.
Synthesize the best insights into one coherent anchor. Be critical of weak derivations and preserve decisive corrections. Keep the anchor compact enough for downstream verification.

The anchor is not the final decision; it is a compressed reference point that summarizes the strongest routed evidence before the last check. This extra pass matters because a trajectory can look persua-

sive locally but still fail when the task-specific output format or a final consistency check is applied. This separation keeps the anchor useful for verification without making it a hidden second answer. The verifier can then compare the anchor against the trajectory cards directly and decide whether the correction is strong enough to survive the last formatting step.

Final Verification and Formatting

Input: Question x, graph-pooled anchor, trajectory evidence cards.
Output: Final prediction $z^\star$.

```
Compare evidence quality, stability, and correction strength.  Allow a
minority trajectory to win if it is better justified.  If verification is
uncertain, fall back to the best-parsed candidate or anchor.  Use the
task-specific final format:  boxed answer for math, letter choice for
multiple choice, and completed code for HumanEval.
```

Taken together, these prompts specify ARGOT's test-time behavior. The selection stage supplies diversity, the graph stage separates support from challenge, the two message-passing blocks carry corrections in both directions, and the final blocks keep the output grounded and parseable. More importantly, the prompts mirror the information flow of the implementation: confidence enters with the first answer, relations enter with the OT graph, corrections move through communication, and the final prediction is checked only after the evidence has been consolidated.

H Limitation and Future Work

ARGOT is evaluated mainly on reasoning benchmarks with clear and verifiable answers, including multiple-choice reasoning, mathematical reasoning, code generation, finance QA, and medical QA. This scope matches the main claims of the paper: evidence routing is most effective when disagreement can be checked against a concrete target, and the strongest gains appear when a minority trajectory carries a better-supported correction. The error analysis also shows the current boundary more clearly. When candidates are nearly identical, the graph has little signal to route, and when a task is dominated by a long symbolic chain or a final arithmetic slip, the verifier can still miss the last correction.

These results suggest a natural direction for future work. One option is to make the evidence model more adaptive to task family, so that the graph and verifier can weight symbolic consistency, execution checks, or domain priors more explicitly when the answer space is fragile. Another direction is to expand beyond closed-form reasoning into open-ended collaboration, where disagreement may represent creativity, preference diversity, or strategic alternatives rather than noise. In that setting, ARGOT would need a broader definition of evidence quality, for example by modeling novelty, user intent, and constructive conflict alongside correctness. A further extension would be to study whether larger agent pools, multiple communication rounds, or task-dependent stopping rules can improve robustness without making the routing graph too dense.