Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Algorithmic Groupthink: Causal Analysis and Mitigation of Semantic Convergence in Multi-agent LLM Systems

Hideki Muto

muto@keio.jp

Keio University

Tetsuro Ogi

Keio University

Takahiro Yakoh

Keio University

Article

Keywords:

Posted Date: June 30th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10172697/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Algorithmic Groupthink: Causal Analysis and Mitigation of Semantic Convergence in Multi-agent LLM Systems

Hideki Muto[1,*], **Tetsuro Ogi**[1], **and Takahiro Yakoh**[1]

[1]Graduate School of System Design and Management, Keio University, Yokohama, Japan
[*]muto@keio.jp

ABSTRACT

When several AI agents work on one open-ended task, such as brainstorming ideas or drafting designs, they usually share drafts so each builds on the others. But sharing also narrows what they produce: asked for many different answers, they return a few similar ones. We tested this across 20 tasks. With the model and sampling fixed, we varied only whether agents could see each other's work. Sharing reduced the semantic diversity of outputs by 5.9%; isolation raised it by 10.3% ($p = 0.017$, Cohen's $d = 0.79$). The loss grew with how much was shared, held across five LLMs from three providers (significant in two), and was not explained by input length, drift, or order. We call this pattern Algorithmic Groupthink. It can be countered at the prompt level: under a multi-workflow protocol where sharing cost 31.8% of diversity, one added instruction in the system prompt cut the loss to 5.5% (replicated at $n = 20$, $p = 0.0003$, Cohen's $d = 1.31$), with no loss of quality. The effect appears at inference time, not during training, so developers can adjust diversity by changing how much context the agents share, not by retraining.

Introduction

Imagine five large language model (LLM) agents brainstorming product ideas. If the agents share their drafts at each round, their ideas drift together. The more they share, the more alike the answers become. This is a real problem. Multi-agent systems are usually built to give a range of *different* answers. When a team of agents is asked for ten marketing strategies but returns five close variants of the same one, the human reviewer never sees the other possibilities. The space of options has already been narrowed before anyone looks.

Multi-agent LLM frameworks such as AutoGen[1], ChatDev[2], MetaGPT[3], CrewAI[4], LangGraph[5], the OpenAI Agents SDK[6], and the Claude Agent SDK[7] let agents generate, critique, and refine outputs together[8]. This kind of collaboration improves reasoning and factual accuracy[9,10], but how it affects output diversity has not been studied carefully. Related work suggests that diversity loss is widespread. Alignment training causes mode collapse in single models[11]; writing with language models reduces content diversity at inference time[12]; same-model outputs are more similar to one another than to human text[13,14]; and larger models show less epistemic diversity than a basic web search[15]. In multi-agent debates, majority pressure can keep agents from reasoning independently[16–18]; large-scale studies report that multi-agent discussions produce more similar outputs over time[19,20] and that LLM agents converge more strongly than humans do[21]. Wu and Ito[22] have described this as a *consensus–diversity tradeoff*. Whether similar effects arise systematically in collaborative workflows, and whether they can be reduced without architectural changes, remains an open question.

We focus on open-ended, exploratory tasks such as brainstorming, design, and creative writing, where the value of a multi-agent system lies in the range of ideas it produces, and where early convergence can remove unusual but high-quality solutions from the output distribution[11]. For tasks that have a clear correct answer, convergence toward that answer is desirable; this paper focuses on a different setting.

We focus on one design variable: how much each agent sees of the other agents' outputs during iterative refinement. We call the resulting convergence **Algorithmic Groupthink**. The effect is not just statistical. Even though each agent generates independently, the outputs end up reusing the same section headings, naming characters in the same patterns, and arriving at similar plot structures. Unlike the *training-time* mode collapse from RLHF alignment[11], this is a *runtime* effect: the same model and weights produce varied outputs when agents work alone and convergent ones when they share context. The name is borrowed loosely from human groupthink[23]; the mechanism here is more likely probabilistic bias in autoregressive generation than social pressure.

This paper reports two main findings. *First, Algorithmic Groupthink is a robust, dose-dependent runtime effect*: shared context causes a steady decline in semantic diversity (5.9% loss vs. 10.3% gain under isolation, $n = 20$ tasks); it cannot be

explained by input length, natural drift, or the order in which outputs are shared; and the direction is consistent across five LLMs from three providers (OpenAI, Anthropic, Google), reaching statistical significance in two. *Second, the loss can be prevented at the prompt level*: under a 5-task multi-workflow protocol, one line added to the system prompt cut the loss from 31.8% to 5.5%, and a 20-task replication confirmed the effect ($p = 0.0003$, Cohen's $d = 1.31$) with no measurable quality cost in our LLM-judge evaluation. Together, these findings turn Algorithmic Groupthink into a manageable design problem, rather than a fundamental limit of multi-agent collaboration.

Methods

This is a short overview, enough to follow the Results. The full protocols are in the Supplementary Information (Section S3): workflow specifications, prompt templates, the control experiments, the embedding-model robustness check, and the power analysis. The complete implementation is in the code repository.

Simulation framework

The multi-agent simulation framework is custom-built in Python. Unless we say otherwise, the agents ran on GPT-4o-mini (OpenAI, December 2024) at temperature 1.0. The high temperature is deliberate. It maximizes baseline variability and so makes our test conservative. Each run lasted five iterations ($k = 0$ to 4); the output of iteration k became the input for iteration $k+1$. Inside each iteration the agents went through a three-stage Generator-Critic-Summarizer (Gen-Critic-Sum) workflow (Figure 1a), based on the generate-critique-refine loop used in frameworks such as AutoGen and LangGraph. Experiment 1 relied on Gen-Critic-Sum alone, so that the reference-mode effect could be measured on its own; the other experiments added four more workflows as a robustness check (Section S3).

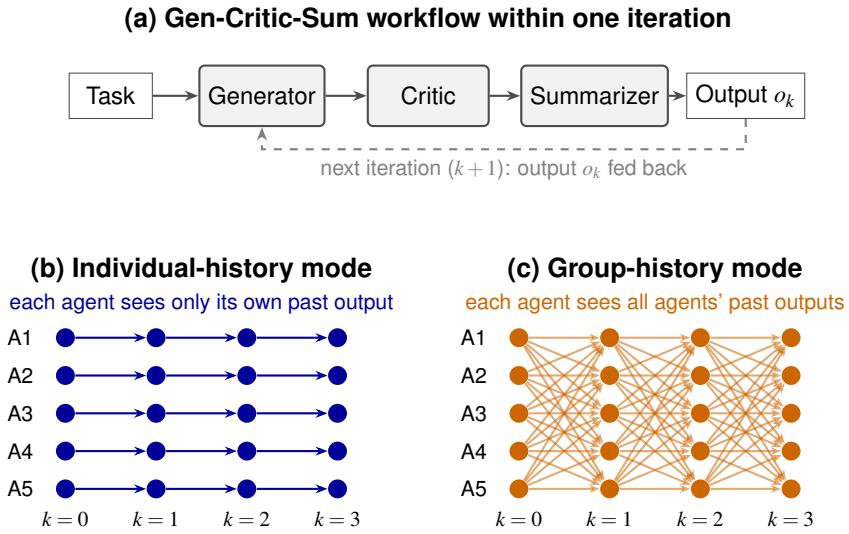


Figure 1. Experimental design. (a) Within each iteration, the task is processed by a three-stage Generator-Critic-Summarizer workflow; the summarizer's output o_k becomes the input for iteration $k+1$. (b) In individual-history mode, each of $N = 5$ agents (A1–A5) sees only its own previous output across iterations, yielding N parallel, isolated refinement trajectories. (c) In group-history mode, each agent receives the previous outputs of all agents, creating an all-to-all information flow between iterations. The reference mode (b vs. c) is the primary experimental variable; in Experiment 2 we interpolate between the two by varying the share ratio from 0% to 75%.

Reference modes and experimental design

The reference mode controls how agents access previous outputs. In group-history mode, each candidate receives the concatenated previous outputs of all $N = 5$ candidates. In individual-history mode, each candidate receives only its own previous output. For Experiment 2, we varied the share ratio $r \in \{0.0, 0.25, 0.5, 0.75\}$, where each candidate sees its own output plus a random sample of $\lfloor r \times (N-1) \rfloor$ others; ratios 0.0 and 1.0 correspond to the individual- and group-history modes, respectively.

Experiment 1 compared the two reference modes across 20 tasks ($\times$ 5 candidates $\times$ 5 iterations $\times$ 2 conditions = 2,000 generations). Experiment 2 varied the share ratio across 5 tasks (500 generations). Experiment 3 compared two countermeasures (a Diversity Prompt that instructs the agent to generate responses substantially different from previous outputs, and Adversarial

Sampling, which generates three candidates and keeps the most distant one) against a group-history baseline (5 tasks $\times$ 2 interventions = 200 generations); we ran an expanded replication of the Diversity Prompt with all 20 tasks from Experiment 1. Three control experiments tested input length (random filler), natural drift (independent generation), and ordering (shuffled history) as alternative explanations (5 tasks each, 100 generations per control).

The 20 tasks comprised four categories with five tasks each: idea generation, reasoning and analysis, summarization, and creative writing. The full task list, prompt templates, and control specifications are provided in Supplementary Methods (Section S3) and the code repository.

Cross-model replication

To test generalizability, we replicated the reference-mode comparison with Claude-3.5-Haiku (Anthropic, January 2025), Gemini 2.0 Flash-Exp (Google, January 2025), GPT-4o (OpenAI, June 2026), and Gemini 2.5 Flash (Google, April 2026), all at temperature 1.0 with identical prompts. Each replication used 5 tasks under the Experiment 1 protocol, with two exceptions: Gemini 2.0 Flash-Exp was doubled to 10 tasks because of the high output variance observed, and GPT-4o was extended to the full 20-task set to test whether its directional effect would reach significance with additional data. Embeddings were computed with OpenAI's `text-embedding-3-small` for comparability across all conditions.

Semantic diversity measurement

For a given task at iteration k, let $E = \{e_1, e_2, \ldots, e_N\}$ denote the set of embeddings produced by the $N = 5$ candidates. Semantic diversity was computed as the mean pairwise Euclidean distance:

$$D_k = \frac{2}{N(N-1)} \sum_{i<j} \|e_i - e_j\|_2 \tag{1}$$

We chose embedding-based distance over lexical metrics because our question concerns semantic similarity rather than surface word overlap[11,12,24]. To confirm that the results are not artifacts of one embedding model, we recomputed Experiment 1 with Sentence-BERT (all-MiniLM-L6-v2)[25]; the findings replicated (group-history -3.9%, individual-history $+13.2\%$; $p = 0.016$, $d = 0.82$), with per-task values correlating at $r = 0.89$ between embedding models (Section S3).

Statistical analysis

All tests were two-tailed at $\alpha = 0.05$. Welch's t-test (assuming unequal variances) was used to compare final diversity values between conditions, with Cohen's d as the effect size and 95% confidence intervals from the Hedges–Olkin approximation. Experiment 2 used one-way ANOVA with η^2, followed by Tukey's HSD post-hoc tests and linear regression for the dose-response relationship. Analyses were performed in Python 3.11 with SciPy 1.11.

Sample-size logic and post-hoc power. The primary confirmatory tests are powered for the effect sizes observed: Experiment 1 ($n = 20$ tasks per condition) achieves $> 99\%$ power at the observed $d = 0.79$; Experiment 2 ($n = 25$ task-condition combinations across four share ratios) achieves $> 99\%$ power at $\eta^2 = 0.20$; the 20-task Diversity Prompt replication ($d = 1.31$, $p = 0.0003$) is similarly well-powered. Several comparisons are reported as directional rather than confirmatory. The cross-model replications for Gemini 2.0 Flash-Exp and Gemini 2.5 Flash are underpowered by design (small ten- and five-task protocols), as is the Adversarial Sampling intervention (36.6% post-hoc power, $d = 1.16$; achieving 80% power would require $n \approx 13$ per group). GPT-4o was instead extended to the full 20-task set and still did not reach significance ($d = 0.29$, $p = 0.37$), which indicates a genuinely small effect in that model rather than insufficient power. Full power analyses, including the inflation of the initial $n = 5$ Diversity Prompt effect size from $d = 5.55$ to $d = 1.31$ under expanded replication, are reported in Supplementary Section S3.5.

Results

Algorithmic Groupthink is real, and it is preventable. When agents in a multi-agent LLM system see each other's drafts, their answers drift together; when they work alone, their answers spread apart. The loss is graded: the more the agents share, the more they converge. A single line added to the system prompt removes most of it. The sections below build to this picture step by step. We first establish the effect and trace its dose-response shape. We then rule out three alternative explanations and follow the convergence in embedding space. Finally, we test the prompt-level fix, check that the pattern holds across leading models, and ask whether the recovered diversity costs anything in quality. Throughout, semantic diversity refers to the mean pairwise Euclidean distance between agent output embeddings at a given iteration, typically between 0.3 and 0.5.

Shared context cuts diversity; isolation expands it

Algorithmic Groupthink shows up as soon as agents see each other's work. We compared two modes: *individual history*, in which each agent sees only its own past output, and *group history*, in which each agent sees the past outputs of all five agents.

The two modes pulled diversity in opposite directions (Figure 2). When agents worked alone, their outputs became *more* diverse across four iterations (a 10.3% rise, from 0.418 to 0.461). When they saw each other's work, the outputs became *less* diverse (a 5.9% drop, from 0.403 to 0.379). The difference was statistically significant (Welch's $t(38) = 2.49$, $p = 0.017$, Cohen's $d = 0.79$), and the direction held in all four task categories. Idea generation gave the clearest split (individual $+6.3\%$, group -11.0%, $p = 0.030$). Table 1 gives the full breakdown.

Table 1. Summary of Experiment 1: Comparison of Reference Modes (20 tasks, 4 categories). Diversity values are mean ± s.d.

Condition	Baseline Diversity	Final Diversity	Change (%)	n
Individual (Self-history)	0.418 ± 0.093	0.461 ± 0.109	+10.3	20
All (Group-history)	0.403 ± 0.077	0.379 ± 0.093	-5.9	20
Statistical test: Welch's t-test on final diversity				
$t(38) = 2.49$, $p = 0.017$*, Cohen's $d = 0.79$ (large effect)				

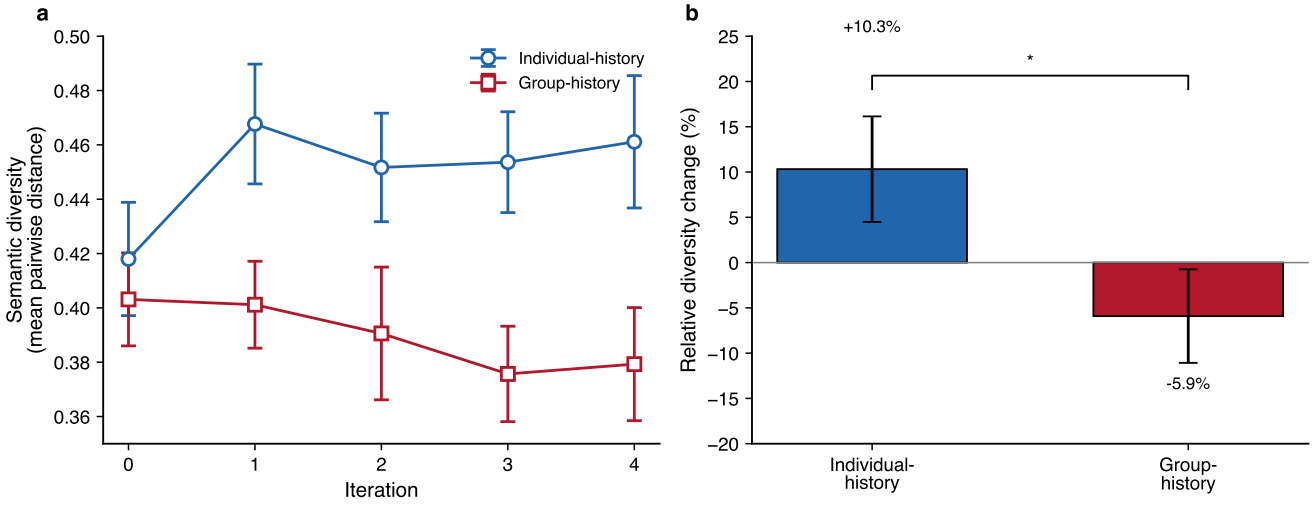


Figure 2. Experiment 1: Individual-history mode (blue) vs. group-history mode (orange). (a) Diversity over iterations. Individual history maintains and slightly increases diversity, whereas group history shows a steady decline. (b) Relative change from baseline. Error bars show standard error of the mean.

Iteration on its own does not cause convergence. Agents refining their own work in isolation become *more* diverse with each round; what pulls outputs together is the shared context.

Diversity loss scales with how much agents share

The effect is graded, not all-or-nothing. Experiment 1 compared full sharing with no sharing. To probe the middle, we varied the *share ratio* (the fraction of other agents' outputs each agent sees) from 0% to 75%.

The relationship was steady and dose-dependent (Figure 3; full per-ratio values in Supplementary Table S1). At 0% sharing, diversity rose by $+14.1\%$; at 25%, it fell by -5.1%; at 50%, by -11.7%; at 75%, by -15.3%. ANOVA confirmed the difference ($F(3,96) = 8.04$, $p < 0.001$, $\eta^2 = 0.20$), and post-hoc tests showed that 0% differed from 50% ($p = 0.003$) and from 75% ($p < 0.001$); the 0%–25% comparison was marginal ($p = 0.089$). Linear regression confirmed the negative slope (-0.121, $R^2 = 0.17$, $p < 0.001$); the modest R^2 shows that share ratio is one factor among several, alongside task type and random variation.

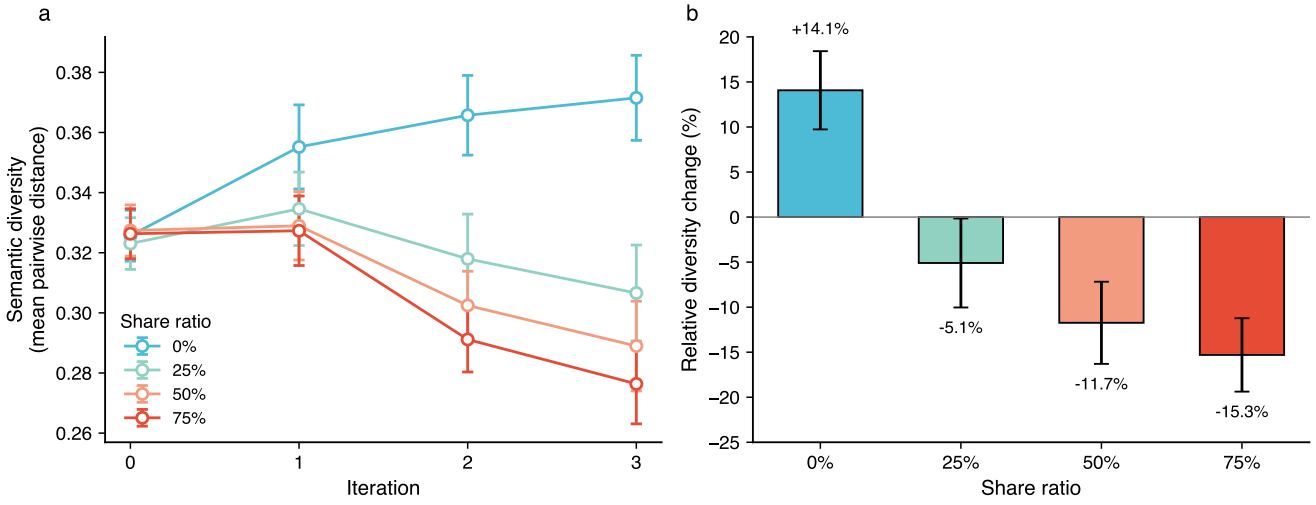


Figure 3. Experiment 2: More sharing leads to more convergence. (a) Diversity over iterations at each share ratio. (b) Final diversity change by share ratio, with ANOVA and regression results. Error bars show s.e.m.

The practical message is simple: even partial sharing reduces diversity. Systems that aim to keep diversity should hold the share ratio at 25% or below, or share only a selected subset.

The driver is content, not length, drift, or order

Three controls trace the cause to the meaning of the shared outputs themselves and rule out three plausible alternatives.

Input length. Group-history prompts are longer because all previous outputs are joined together. When we replaced the shared outputs with unrelated filler text of the same length, diversity *rose* by $+11.9\%$ ($p = 0.0003$). Length is not the cause.

Natural drift. The same model and prompt might drift toward similar answers on their own. When we ran each iteration as a fresh generation with no history at all, diversity barely moved ($+0.5\%$, $p = 0.0038$ vs. baseline). Natural drift is not the cause.

Order. Convergence might be driven by which output the agents see first. When we shuffled the order of shared outputs at random, diversity still dropped by the same amount (-29.1%; $p = 0.84$ vs. the original order). Order is not the cause.

Together, the three controls point to the *content* of the shared outputs as the driver (Supplementary Table S2). The random-filler control does not isolate "semantic" similarity on its own (the filler was simply unrelated), and the shuffled control still uses the same set of outputs. Finer controls, such as paraphrased shared outputs, would further separate semantic from lexical similarity.

Convergence is fast and visible in embedding space

Most of the loss happens in the first round. After that, little changes. This points to a feedback loop: each agent reads the drafts of the others, writes something a little closer to them than its own earlier answer, and passes this slightly narrower output on to the next round (see the Discussion for the link to autoregressive sampling).

In embedding space, the two modes look very different (Figure 4). Individual-history outputs stay scattered from one iteration to the next. The group-history outputs do the opposite. By iteration 3 they have packed into one tight cluster: the mean distance to the cluster center shrinks faster than under individual history (Panel c), and the pairwise distances at iteration 3 come out significantly smaller ($p < 0.001$, Panel d).

The drop is large and early. Most of it arrives in the very first step ($\Delta \approx -0.033$ for group history, -0.018 for individual history), and from there both curves flatten out, so that by iteration 2 there is little movement left in either mode. The collapse does not keep building; once the first step is past, it simply stops (Supplementary Fig. S1).

This is not only about the numbers. The same convergence is clear in the text itself, a qualitative sign previously documented in[26]: at iteration 4 of Experiment 1, four of the five group-history candidates opened with a "Critique of the Draft" section the task never asked for, all five used all-caps names for their AI characters (ARIA, ARTIE, BRIGHT, SYNC), and every response to a prompt that asked only for a short story about a city ended with the same plot, an artist overcoming self-doubt through collaboration with AI. The individual-history outputs for the same task and model could have been written by five different people. Side-by-side examples are in Supplementary Information Section S1.

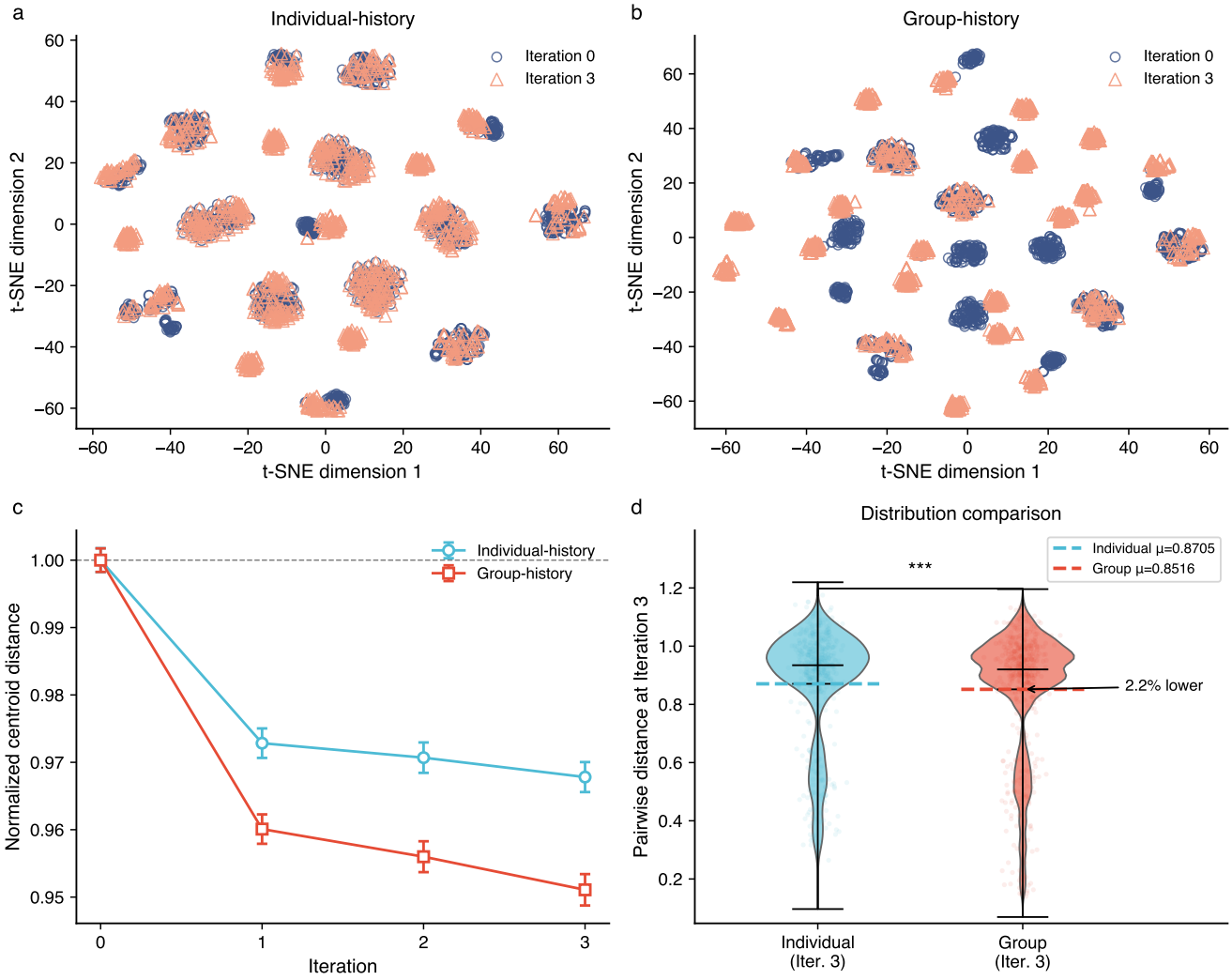


Figure 4. Convergence in embedding space. (a) Individual-history: outputs stay spread out from iteration 0 (black) to iteration 3 (red). (b) Group-history: outputs cluster by iteration 3. (c) Distance from center: group-history drops faster. (d) Pairwise distances at iteration 3: group-history outputs are closer ($p < 0.001$). Error bars in (c) show s.e.m.

A one-line prompt removes most of the convergence

One added line in the system prompt cuts the diversity loss from 31.8% to 5.5%, an 83% reduction with no change to the rest of the system. Under group-history conditions, the *Diversity Prompt*—which adds the instruction "generate responses substantially different from previous outputs" to the system prompt—reduced the loss from a baseline of 31.8% to 5.5% ($t = 8.77$, $p < 0.001$). The prompt also raised the baseline diversity itself (0.45 vs. 0.31), so it affects both the starting point and the rate of convergence. The first $n = 5$ test should be considered preliminary (Methods); the expanded $n = 20$ replication below confirms the effect.

We also tested *Adversarial Sampling* as a more compute-heavy alternative (3× API cost: generate three candidates per iteration, keep the one furthest from prior outputs). The reduction was numerically similar (5.7%) but did not reach significance ($p = 0.132$) under the small protocol because of high variance; full details are in Supplementary Table S3 and Supplementary Sections S3.2 and S3.5 (power analysis).

Experiment 3 used a different protocol from Experiment 1 (5 tasks × 5 workflows × 4 iterations vs. 20 tasks × 1 workflow × 5 iterations), which explains the different baseline convergence rates (31.8% vs. 5.9%). We report only within-experiment comparisons.

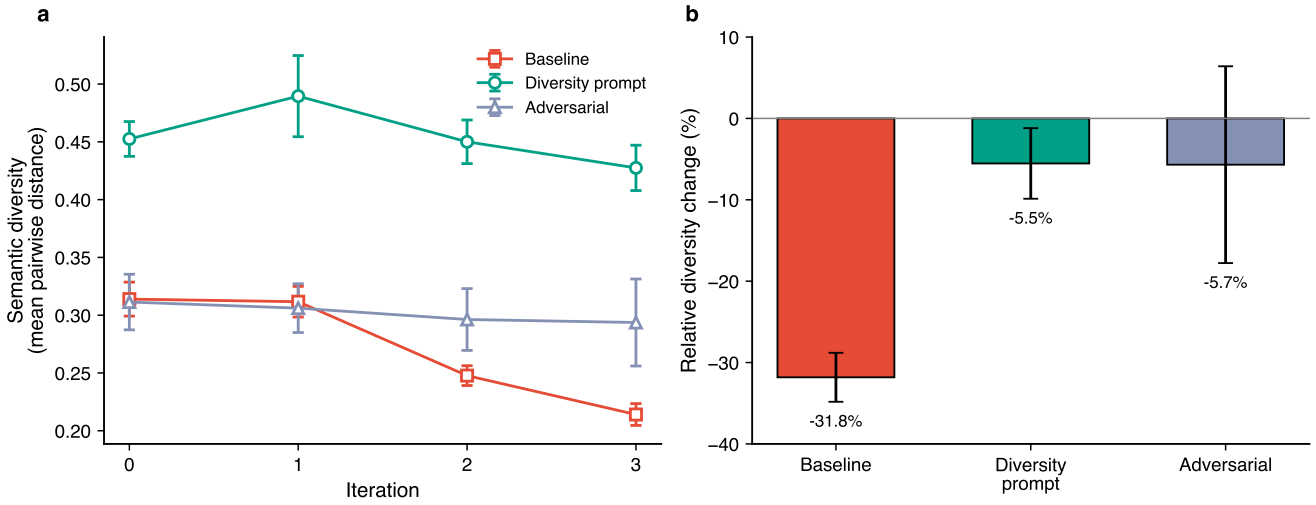


Figure 5. Experiment 3: countermeasures. (a) Diversity across iterations for the baseline, the Diversity Prompt, and Adversarial Sampling. (b) Change in diversity from the baseline to the final iteration. Of the two countermeasures, only the Diversity Prompt reached significance against the baseline under this protocol; Adversarial Sampling pointed the same way but was underpowered (see Supplementary S3.5). Error bars show s.e.m.

We re-ran the prompt on the full 20-task set, and the result held: the baseline lost 15.2% of its diversity while the Diversity Prompt gained 7.0% ($t = 4.03$, $p = 0.0003$, Cohen's $d = 1.31$). All four task categories moved in the same direction, with the per-category figures reported in Supplementary Section S3.5. The effect size fell from $d = 5.55$ at $n = 5$ to $d = 1.31$ at $n = 20$. Small first samples tend to exaggerate; the larger run is the one to trust. In practice, one added line in the system prompt removes most of the loss, and the quality check below finds no measurable cost.

Direction is consistent across five LLMs; significance varies

The next question is whether the effect is specific to GPT-4o-mini. To find out, we ran the same reference-mode comparison on four more models: Claude-3.5-Haiku, Gemini 2.0 Flash-Exp, GPT-4o, and Gemini 2.5 Flash. All used the matched five-task protocol, except GPT-4o, which we later extended to the full 20-task set (Table 2).

Table 2. Cross-model generalization. Five LLMs were tested under a matched 5-task protocol, with two exceptions: Gemini 2.0 Flash-Exp was doubled to 10 tasks because of high output variance, and GPT-4o was extended to the full 20-task set to test whether its directional effect would reach significance. The "Individual" and "Group" columns give the relative change in semantic diversity from iteration 0 to the final iteration, where negative values mean a loss of diversity and positive values a gain. The direction (Group < Individual) holds for all five models, while statistical significance depends on sample size and effect size. The large Cohen's d values for the two significant models reflect the small $n = 5$ and are subject to the same small-sample effect-size inflation discussed in Supplementary Section S3.5.

Model	Provider	n	Individual	Group	p	Cohen's d	Result
GPT-4o-mini	OpenAI	5	+16.0%	−31.8%	< 0.0001	6.69	Significant
Claude-3.5-Haiku	Anthropic	5	+41.7%	−22.0%	0.025	2.08	Significant
GPT-4o	OpenAI	20	+7.8%	−0.5%	0.37	0.29	Directional
Gemini-2.0-Flash	Google	10	+23.3%	+1.4%	0.232	—	Directional
Gemini-2.5-Flash	Google	5	+20.7%	+11.7%	0.49	0.457	Directional

The direction (Group diversity change < Individual diversity change) was consistent across all five models. The effect reached statistical significance in two: GPT-4o-mini (group −31.8% vs. individual +16.0%, $p < 0.0001$) and Claude-3.5-Haiku (−22.0% vs. +41.7%, $p = 0.025$). The remaining three models showed the same direction but did not reach significance. The two Gemini models were limited by the small protocol: Gemini 2.0 Flash-Exp (+1.4% vs. +23.3%, $p = 0.232$, $n = 10$) and Gemini 2.5 Flash (+11.7% vs. +20.7%, $d = 0.457$, $p = 0.49$, $n = 5$). For GPT-4o we extended the replication to the full 20-task set, but the effect remained small and non-significant (group −0.5% vs. individual +7.8%, Cohen's $d = 0.29$, $p = 0.37$, $n = 20$); a category-level breakdown showed that creative-writing tasks still converged under shared context (down

to −18.6%) while reasoning tasks diverged, leaving the all-category average near zero. In both Gemini models, the group condition produced a positive diversity change rather than an absolute loss, but the gap between group and isolated conditions matched the pattern in the other three models. The direction (shared context yields lower diversity than isolated generation) was therefore consistent across three model families and two model generations; statistical confirmation came from the two larger-effect models.

The high variance in Gemini 2.0 Flash-Exp—which led us to double its sample to $n = 10$—appears specific to that preview release. The follow-up Gemini 2.5 Flash run reduced the SD from 49.8% to 19.2% (vs. 2.3% for GPT-4o-mini and 12.8% for Claude-3.5-Haiku).

Diversity Prompt does not lower quality in LLM-judge evaluation

To check whether the diversity gains come at a quality cost, we asked GPT-4o-mini to rate outputs ($n = 15$ per condition) on creativity, relevance, practicality, completeness, and clarity, each on a 1–5 scale. This is a preliminary check; the single-judge limitations are addressed in the cross-judge replication below and in the Limitations.

The most diverse condition (Diversity Prompt) scored higher than the least diverse one (baseline group history) on both creativity (5.00 vs. 4.27, $t = 4.78$, $p < 0.001$) and overall quality (4.80 vs. 4.55, $t = 3.83$, $p < 0.001$); diversity and creativity were positively correlated at the sample level ($r = 0.30$, $p = 0.04$, $n = 45$; Figure 6).

To address single-judge bias, we replicated the evaluation with Claude-Haiku 4.5 as an independent judge from a different model family, rating the same 45 outputs. The directional finding was confirmed: Diversity Prompt scored higher than Baseline (Group) on both creativity (4.33 vs. 3.07, $t = 8.89$, $p < 0.001$, $d = 3.24$) and overall quality (4.16 vs. 3.83, $t = 2.97$, $p = 0.006$, $d = 1.08$). Inter-judge agreement was strongest on creativity ($r = 0.60$, $p < 0.001$, $n = 45$) and moderate on overall quality ($r = 0.31$, $p = 0.04$). The judges diverged on the practicality dimension: GPT-4o-mini returned near-constant ratings (mean 4.0 across all conditions, no variance), while Claude rated the Diversity Prompt outputs as somewhat less practical (2.47 vs. 3.47 for Baseline). The pattern is consistent with a creativity–practicality trade-off in open-ended ideation: more divergent ideas can be less immediately implementable. Full per-judge tables are in Supplementary Section S4.

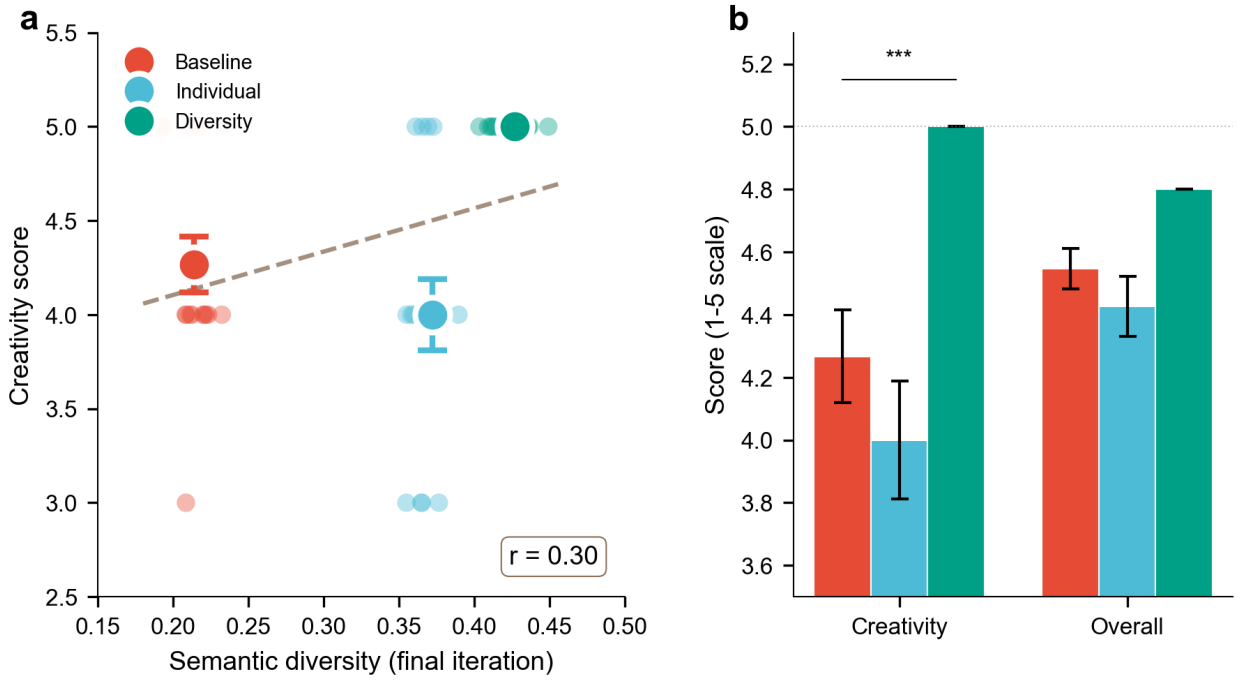


Figure 6. Diversity and quality are positively related (GPT-4o-mini judge). (a) Creativity scores vs. semantic diversity by condition ($r = 0.30$, $p = 0.04$, $n = 45$). (b) Quality scores across five dimensions: the Diversity Prompt condition (highest diversity) scores highest on creativity, relevance, completeness, and clarity, with practicality flat under this judge; cross-judge replication with Claude-Haiku 4.5 confirms the creativity and overall-quality findings and indicates a creativity–practicality trade-off (Supplementary Section S4).

This matches Zhang et al.[11], who found that restoring diversity also improves quality. Representative examples of Diversity

Prompt outputs, showing that the diverse answers remain coherent and useful, are in Supplementary Information Section S2. Two caveats apply. First, both judges are LLMs and may share biases with the models being rated; the cross-LLM agreement on creativity ($r = 0.60$) is encouraging but human evaluation would further strengthen the evidence. Second, we tested only open-ended creative tasks; the relationship may differ for factual ones. Still, these results show that our diversity metric captures real variation rather than random drift.

Shared context is the primary driver of multi-agent convergence in our open-ended tasks. The effect is graded and content-driven, and a single line of prompt text restores most of the lost diversity at no measurable cost to LLM-judged quality.

Discussion

Algorithmic Groupthink as a runtime phenomenon

Algorithmic Groupthink is a runtime phenomenon, separate from the training-time mode collapse described by Zhang et al.[11]. In mode collapse, the weights of the model are permanently changed; the effect we report here occurs with the weights fixed and depends only on whether the agents share context. The same model produces diverse outputs in isolation and convergent outputs under shared context.

The finding builds on what is already known about autoregressive language models, rather than replacing it. Such models are known to converge when conditioned on similar inputs[27]; recent theoretical work has framed iterative LLM generation as a dynamical system with attractors[28]; and network-level simulations of interacting LLMs give convergence guarantees under shared information pools[29]. Our contribution is to turn this expectation into something measured, controlled, and tunable: a quantified effect in multi-agent collaboration, a dose-response relationship that makes context sharing an explicit design parameter, and a prompt-level intervention that removes most of the loss. Whether the underlying mechanism is best described as autoregressive convergence, a feedback loop, or "tail pruning" of low-probability outputs is a question for future token-level work. The practical conclusion does not depend on the mechanism.

The two sources of diversity loss, training-time mode collapse and runtime shared-context convergence, are likely to add to each other. A model already biased toward high-probability outputs by RLHF will amplify that bias further when its agents share context. Reducing diversity loss therefore needs interventions at both levels.

Implications for practice

Whether convergence is desirable depends on the task. In fact-checking, code review, and summarization, agents need to converge on the correct answer, and shared context helps them identify errors and remain consistent. In open-ended tasks such as brainstorming, creative writing, and design, convergence narrows the range of options before any human can review them. Zhang et al.[11] showed that restoring diversity in single-agent settings also improves quality (1.6–2.1× diversity gain with a simultaneous performance gain), so convergence may impose a double penalty on creative tasks: fewer options and lower quality per option.

We recommend three levels of intervention, ordered by cost. **Level 1 (no code changes):** add a diversity instruction to the system prompt. In our experiments this cut the loss from 31.8% to 5.5%, and it works alongside training-time methods such as diversity-aware reinforcement learning[30] and inference-time methods such as conformative decoding[31]. **Level 2 (moderate changes):** restrict each agent's view to a subset of the shared history (context partitioning), keep the share ratio at 25% or below, or add a dedicated diversity-preserving agent. Related debate-specific mitigations such as CONSENSAGENT[32], FREE-MAD[33], CIR3[34], and DMAD[18] suggest compatible design patterns. Memory-sharing architectures[35] are another option, though shared memory can cause "context pollution"[36]. **Level 3 (architectural changes):** use heterogeneous model teams[37] or adaptive context sharing that switches between full and partitioned sharing as the task progresses; adversarial sampling is a more compute-heavy alternative whose preliminary test in this paper was underpowered (Supplementary S3.5). Work on collaboration topology and agent scaling[38,39] suggests further design options.

For human-AI workflows, our findings raise a further concern: when agents return similar options, the human reviewer's effective choice set is narrower than it appears[40,41]. This connects to a broader body of evidence that LLMs make human expression and information-seeking more uniform[42–44]. The downstream effect on human decision-making is the natural next question, and answering it requires human-subject studies.

Limitations and future directions

Three limits frame our findings. First, diversity is measured here through embeddings rather than human judgment, and our quality ratings come from LLM-as-judge evaluations. We addressed the single-judge concern by replicating the quality evaluation with Claude-Haiku 4.5 (Supplementary Section S4), which confirmed the directional findings on creativity and overall quality but disagreed with the GPT-4o-mini judge on practicality. Both judges remain LLMs that may share biases with the models being rated; human-subject validation of both diversity and quality is the natural next step. Second, the core effect is established with large samples and large effect sizes for GPT-4o-mini and Claude-3.5-Haiku, but it varies across models.

For GPT-4o we ran the full twenty-task protocol, and the effect remained small and non-significant ($d = 0.29$, $p = 0.37$); this points to a genuinely smaller effect in that model rather than a power limitation, although a category-level breakdown showed that its creative-writing outputs still converged under shared context. The Gemini 2.0 Flash-Exp ($n = 10$) and Gemini 2.5 Flash ($d = 0.457$, $n = 5$) replications use small-task protocols that show consistent direction but do not reach confirmatory significance (see Supplementary Section S3.5 for power analysis); extending them to the full twenty-task protocol would clarify whether their weaker effects reflect small samples or genuinely smaller convergence. The adversarial-sampling intervention is similarly directional but underpowered ($p = 0.132$, 36.6% power) and remains a promising but unconfirmed alternative to the Diversity Prompt. Third, our results are limited to multiple instances of the same model, refining outputs over several iterations on open-ended tasks. We do not expect the effect to arise, or to be problematic, in factual question-answering, single-agent systems, heterogeneous model ensembles, or single-shot generation.

Two further extensions would extend our findings. A fully factorial design crossing 20 tasks with multiple workflows and reference modes would test workflow-independence beyond what our five-workflow check already shows. And token-level analyses of the same generations could tell apart the candidate mechanisms (autoregressive convergence, feedback amplification, tail pruning) that our embedding-level analysis cannot.

Despite these limits, Algorithmic Groupthink reframes what has been treated as emergent behaviour as a controllable design parameter. The phenomenon arises at inference time and depends on the context-sharing scheme; recovering most of the lost diversity requires only a one-line prompt change. As multi-agent frameworks such as AutoGen, CrewAI, the OpenAI Agents SDK, and the Claude Agent SDK move from research prototypes into production, the diversity cost of context sharing should be a routine design consideration alongside latency, cost, and accuracy.

Data Availability

All experimental data generated or analyzed in this study are publicly available. Raw text outputs from all experimental conditions (approximately 5,500 generations across Experiments 1–3, control experiments, and cross-model replications), together with the corresponding OpenAI `text-embedding-3-small` and Sentence-BERT `all-MiniLM-L6-v2` embeddings, are deposited on Zenodo (`https://doi.org/10.5281/zenodo.19676464`) under a CC-BY 4.0 licence. The Zenodo record is synchronized with the GitHub repository described below and preserves the exact version used in this study.

Code Availability

The custom multi-agent simulation framework, analysis scripts, and figure-generation code used in this study are implemented in Python and hosted on GitHub (`https://github.com/hmuto/algorithmic-groupthink-paper`) under an MIT licence. The version used for this paper is archived on Zenodo with a persistent DOI (`https://doi.org/10.5281/zenodo.19676464`). The repository contains a `README` with instructions for reproducing all experiments, the full task set, all prompt templates, and the parameter settings (temperature, model versions, and API access dates) required to replicate our analysis pipeline. We note that commercial LLM APIs do not guarantee bit-exact output reproducibility even with fixed seeds; however, the aggregate statistical findings were confirmed to be robust through cross-model replications and the expanded replications reported in the Supplementary Information.

References

1. Wu, Q. *et al.* Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155* (2023).
2. Qian, C. *et al.* Chatdev: Communicative agents for software development. *Proc. 62nd Annu. Meet. Assoc. for Comput. Linguist. (ACL)* (2024).
3. Hong, S. *et al.* Metagpt: Meta programming for a multi-agent collaborative framework. *Int. Conf. on Learn. Represent. (ICLR)* (2024).
4. CrewAI. Crewai: Framework for orchestrating role-playing, autonomous ai agents. https://github.com/crewAIInc/crewAI (2024). Accessed: 2024-12-01.
5. LangChain. Langgraph: Build resilient language agents as graphs. https://github.com/langchain-ai/langgraph (2024). Accessed: 2024-12-01.
6. OpenAI. OpenAI Agents SDK: Lightweight framework for building multi-agent workflows. https://github.com/openai/openai-agents-python (2025). Accessed: 2026-04-20.
7. Anthropic. Claude Agent SDK: Building autonomous agents with claude. https://docs.anthropic.com/en/docs/agent-sdk (2025). Accessed: 2026-04-20.

8. Tran, K.-T. *et al.* Multi-agent collaboration mechanisms: A survey of LLMs. *arXiv preprint arXiv:2501.06322* (2025).

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

10. Madaan, A. *et al.* Self-refine: Iterative refinement with self-feedback. *Adv. Neural Inf. Process. Syst.* **36** (2023).

11. Zhang, J. *et al.* Verbalized sampling: How to mitigate mode collapse and unlock LLM diversity. *arXiv preprint arXiv:2510.01171* (2025).

12. Padmakumar, V. & He, H. Does writing with language models reduce content diversity? *arXiv preprint arXiv:2309.05196* (2024).

13. Smith, B., Bouadjenek, M. R., Kheya, T. A., Dawson, P. & Aryal, S. A comprehensive analysis of large language model outputs: Similarity, diversity, and bias. *arXiv preprint arXiv:2505.09056* (2025).

14. Xu, W., Jojic, N., Rao, S., Brockett, C. & Dolan, B. Echoes in AI: Quantifying lack of plot diversity in LLM outputs. *Proc. Natl. Acad. Sci.* **122** (2025). ArXiv preprint arXiv:2501.00273.

15. Wright, D. *et al.* Epistemic diversity and knowledge collapse in large language models. *arXiv preprint arXiv:2510.04226* (2025).

16. Wu, H., Li, Z. & Li, L. Can LLM agents really debate? a controlled study of multi-agent debate in logical reasoning. *arXiv preprint arXiv:2511.07784* (2025).

17. Liang, T. *et al.* Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 17647–17663 (2024).

18. Liu, Y., Cao, J., Li, Z., He, R. & Tan, T. Breaking mental set to improve reasoning through diverse multi-agent debate. In *International Conference on Learning Representations (ICLR)* (2025).

19. Parfenova, A., Denzler, A. & Pfeffer, J. Emergent convergence in multi-agent LLM annotation. *arXiv preprint arXiv:2512.00047* (2025).

20. Reza, Z. The social laboratory: A psychometric framework for multi-agent LLM evaluation. *NeurIPS 2025 Work.* (2025). ArXiv preprint arXiv:2510.01295.

21. Chuang, Y.-S. *et al.* DEBATE: A large-scale benchmark for evaluating opinion dynamics in role-playing LLM agents. *arXiv preprint arXiv:2510.25110* (2025).

22. Wu, Z. & Ito, T. The hidden strength of disagreement: Unraveling the consensus-diversity tradeoff in adaptive multi-agent systems. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2025). ArXiv preprint arXiv:2502.16565.

23. Janis, I. L. *Victims of Groupthink: A Psychological Study of Foreign-Policy Decisions and Fiascoes* (Houghton Mifflin, Boston, 1972).

24. Tevet, G. & Berant, J. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 326–346 (2021).

25. Reimers, N. & Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *Proc. 2019 Conf. on Empir. Methods Nat. Lang. Process.* (2019).

26. Muto, H. & Ogi, T. Diversity collapse in multi-agent LLM workflows: A quantitative simulation study. In Barolli, L., Petrakis, E. G. M. & Hussain, F. (eds.) *Advances in Internet, Data & Web Technologies*, 72–81, DOI: 10.1007/978-3-032-22795-9_7 (Springer Nature Switzerland, Cham, 2026).

27. Holtzman, A., Buys, J., Du, L., Forbes, M. & Choi, Y. The curious case of neural text degeneration. In *International Conference on Learning Representations (ICLR)* (2020).

28. Tacheny, N. Geometric dynamics of agentic loops in large language models: Trajectories, attractors and dynamical regimes in semantic space. *arXiv preprint arXiv:2512.10350* (2025).

29. Wang, T., Horiguchi, A., Pang, L. & Priebe, C. E. LLM web dynamics: Tracing model collapse in a network of LLMs. *arXiv preprint arXiv:2506.15690* (2025).

30. Li, T. *et al.* Jointly reinforcing diversity and quality in language model generations. *arXiv preprint arXiv:2509.02534* (2025).

31. Peeperkorn, M., Kouwenhoven, T., Brown, D. & Jordanous, A. Mind the gap: Conformative decoding to improve output diversity of instruction-tuned LLMs. *arXiv preprint arXiv:2507.20956* (2025).

32. Pitre, P., Ramakrishnan, N. & Wang, X. CONSENSAGENT: Towards efficient and effective consensus in multi-agent LLM interactions through sycophancy mitigation. In *Findings of the Association for Computational Linguistics (ACL 2025)* (2025).

33. Cui, Y., Fu, H., Zhang, H., Wang, L. & Zuo, C. FREE-MAD: Consensus-free multi-agent debate. *arXiv preprint arXiv:2509.11035* (2025).

34. Saadaoui, S. & Alonso, E. Coordinated LLM multi-agent systems for collaborative question-answer generation. *Knowledge-Based Syst.* (2025).

35. Gao, H. & Zhang, Y. INMS: Memory sharing for large language model based agents. *arXiv preprint arXiv:2404.09982* (2024).

36. Yuen, S., Medina, F. G., Su, T., Du, Y. & Sobey, A. J. Intrinsic memory agents: Heterogeneous multi-agent LLM systems through structured contextual memory. *arXiv preprint arXiv:2508.08997* (2025).

37. Ye, R. *et al.* X-MAS: Towards building multi-agent systems with heterogeneous LLMs. *arXiv preprint arXiv:2505.16997* (2025).

38. Yang, Y. *et al.* Understanding agent scaling in LLM-based multi-agent systems via diversity. *arXiv preprint arXiv:2602.03794* (2026).

39. Qian, C. *et al.* Scaling large language model-based multi-agent collaboration. In *International Conference on Learning Representations (ICLR)* (2025). ArXiv preprint arXiv:2406.07155.

40. Buçinca, Z., Malaya, M. B. & Gajos, K. Z. To trust or to think: Cognitive forcing functions can reduce overreliance on ai in ai-assisted decision-making. *Proc. ACM on Human-Computer Interact.* **5**, 1–21 (2021).

41. Lee, M., Liang, P. & Yang, Q. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *CHI Conference on Human Factors in Computing Systems* (2022).

42. Sourati, Z., Ziabari, A. S. & Dehghani, M. The homogenizing effect of large language models on human expression and thought. *arXiv preprint arXiv:2508.01491* (2025).

43. Moon, S. & Washington, E. Homogenizing effect of large language models (LLMs) on creative diversity: An empirical comparison of human and ChatGPT writing. *Comput. Hum. Behav. Artif. Humans* DOI: 10.1016/j.chbah.2025.100091 (2025).

44. Sharma, N., Liao, Q. V. & Xiao, Z. Generative echo chamber? effects of LLM-powered search systems on diverse information seeking. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 1–17, DOI: 10.1145/3613904.3642459 (2024).

Funding

The authors received no specific funding for this work.

Author contributions

H.M. conceived and designed the study, implemented the simulation framework, performed experiments and statistical analyses, and drafted the manuscript. T.O. and T.Y. supervised the study and edited the manuscript accordingly. All authors contributed to the interpretation of the results and approved the final manuscript.

Additional information

Competing interests: The authors declare no competing interests.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- scirepsupplementary.pdf