DO COLLABORATION METRICS TRACK MULTI-AGENT SUCCESS? A CROSS-BENCHMARK MEASUREMENT AUDIT

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-agent LLM systems are increasingly deployed on complex, long-horizon tasks, but are usually evaluated through final outcomes or structural scale, neither of which tells us whether the collaboration itself was effective. Successful trajectories can contain poor coordination, and failed trajectories can still involve useful information exchange. We therefore investigate whether trajectory-level collaboration metrics provide stable signals of task success across domains, models, and system designs. We evaluate five metrics of collaboration quality, namely a holistic judged communication score, judged redundancy, judged downstream usage, judged contradiction rate, and semantic evolutionary distance, across four multi-agent benchmarks spanning algorithmic coordination, software engineering, and enterprise workflows. The study covers five language models and 18 model-benchmark configurations. Four of the five metrics fail to hold up. The narrower per-event judged measures survive multiplicity correction in at most six of 18 configurations, while evolutionary distance changes sign across settings and even within a single benchmark, so neither can be read as a directional quality score. The fifth, a holistic communication score, is associated with task success in 14 of 18 configurations ($r = 0.23$–0.77), every one surviving correction, but its rubric partly references task progress. We next ask whether the same signal transfers to a different outcome. In a secondary analysis of 1,500 trajectories from three privacy-constrained benchmarks, no general collaboration metric survives multiplicity correction as a predictor of information exposure, and this non-detection holds whether exposure is measured across the whole trajectory or restricted to failures that never reach the final output, so the privacy analysis can rule out large associations while smaller effects may still have gone undetected. Under this instrument, the measures that are cleanly separable from the task-success outcome carry little signal that transfers across benchmarks, and the one measure that does carry signal is not cleanly separable from it. Collaboration metrics should therefore be validated for a specified outcome and setting rather than treated as general indicators of collaboration quality.

1 INTRODUCTION

Large language model (LLM) agents are increasingly organised into multi-agent systems (MAS) that decompose tasks, exchange intermediate results, call tools, verify outputs, and coordinate across specialised roles (Wu et al., 2023; Hong et al., 2024). The central promise of MAS is not that several agents participate, but that their interaction produces coordinated and complementary work. Whether it does is contested. Additional agents can help through sampling, debate, or complementary reasoning (Li et al., 2024), yet single-agent systems can outperform MAS under equal thinking-token budgets (Tran & Kiela, 2026), and performance depends on how agents, communication steps, and roles are organised rather than on scale alone (Kim et al., 2026).

Task success alone cannot settle the question. A successful trajectory may contain redundant communication, ignored contributions, inconsistent assumptions, or poor handoffs that a later agent silently compensates for, while an unsuccessful one may contain useful exchange and verification that simply did not reach completion. Multi-agent designs cost real tokens, latency, and engineering

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.

complexity, so whether the interaction contributed determines whether those costs were warranted. When a system fails, outcome measures do not say which part of the collaboration to repair. We therefore distinguish collaboration process quality from collaborative task effectiveness, following human-team research in which communication, coordination, information sharing, and shared task understanding are process constructs that must be related empirically to outcomes rather than assumed equivalent to them (Marks et al., 2001; Hoegl & Gemuenden, 2001; LePine et al., 2008; Mesmer-Magnus & DeChurch, 2009; Mathieu et al., 2000).

Recent MAS benchmarks expose process failures that task success does not capture. Agents communicate without integrating distributed information (Zhang et al., 2026), delegation and context transfer fail in role-specialised enterprise tasks (Yu et al., 2026), coding agents interfere with one another when coordination is weak (Khatua et al., 2026), and individually capable agents become poor collaborators under constraints on instruction preservation, false-belief propagation, and leakage (Mazumder et al., 2026; Zhu et al., 2025). Each introduced its own collaboration metrics, and three limitations follow. Those metrics are typically evaluated only within the benchmark that introduced them, so whether their relationship with task success generalises across domains, models, and interaction structures is unclear. Structural properties such as team size and interaction length are discussed alongside collaboration-quality measures despite describing scale rather than quality. Many proposed measures need benchmark-specific instrumentation or injected signals, so they cannot be computed on an ordinary trajectory. Work on collaboration gain isolates the value of interaction from resource expenditure (Fan et al., 2026) but does not establish which trajectory properties explain when that gain occurs. We address this gap through a cross-benchmark study of three research questions.

RQ1 To what extent are trajectory-level collaboration metrics associated with task success across diverse multi-agent benchmarks, models, and interaction structures?

RQ2 Which metrics are stable under a fixed measurement instrument across domains, models, and coordination structures, and which are benchmark-specific?

RQ3 Do collaboration metrics remain informative when the outcome of interest extends beyond task success to trajectory-level risks that may not surface in the final output, such as privacy exposure?

We begin from a candidate pool of 25 process measures assembled across the seven collaboration dimensions this literature proposes, and screen it for measures computable on any ordinary trajectory (Section 3.2). The five that survive are communication score, redundancy rate, downstream usage, contradiction rate, and semantic evolutionary distance (Zhu et al., 2025; Fan et al., 2026). That the pool reduces this far is itself a result, and the screen makes explicit which dimensions no current measure reaches across benchmarks. We evaluate the five on AgentCollabBench, Silo-Bench, EntCollabBench, and CooperBench (Mazumder et al., 2026; Zhang et al., 2026; Yu et al., 2026; Khatua et al., 2026), spanning algorithmic coordination, software engineering, and enterprise workflows, each run with five language models across three providers and two capability tiers, giving 18 model–benchmark configurations.[1] A secondary analysis of 1,500 trajectories from PAC-BENCH, AgentSocialBench, and AgentLeak (Park et al., 2026; Wang & Jiang, 2026; El Yagoubi et al., 2026) then asks whether any signal transfers to a second outcome. Privacy constraints both motivate multi-agent designs in which agents hold different privilege layers, and introduce information exposure as an outcome distinct from completion, since a system may finish a task while leaking through messages, shared memory, tool arguments, or handoffs. The information-sharing that measures such as communication score and downstream usage reward is also the channel through which private context propagates, so a metric rewarding exchange or integration alone could favour riskier trajectories. RQ3 tests whether the retained measures track this risk or are blind to it.

These contributions carry two caveats stated at the outset. First, the communication rubric partly considers whether communication advances the task, so some overlap between this predictor and the task-success outcome is expected. Second, four retained measures are outputs of one GPT 5.5 judging instrument rather than independently validated human labels. A second judge reproduces

[1] Five models across four benchmarks would give 20. On CooperBench and EntCollabBench, Mistral Large did not engage with the agent harness, running without issuing tool calls or taking substantive actions. We excluded these two configurations rather than modify the harness for one model, which would have made those benchmarks incomparable to the rest.

the communication-score result (Appendix A.6.3), but the other three rest on that instrument alone, and Section 3.4 sets out what this does and does not license.

2 Related Work

Agent systems are commonly evaluated by aggregate task performance, meaning final success, answer correctness, pass rate, or task score. These establish whether a system completed a task but say little about the quality of the collaboration that produced it, which matters most for multi-agent systems (MAS), where execution spans decomposition, delegation, communication, handoffs, verification, and failure containment. Recent work therefore evaluates the collaborative process through the trajectory rather than inferring it from the final output.

We organise prior work by the analytical role each measure plays. *Process quality* captures how agents communicate, transfer and integrate information, maintain shared task understanding, verify work, and contain failures, which are explanatory trajectory features. *Robustness and privacy* capture whether interaction preserves instructions and information boundaries or amplifies errors, false beliefs, and leakage, making them secondary outcomes or stress-test criteria. Task success is not a collaboration measure but the outcome against which process measures are validated, and structural measures such as message count and team size describe interaction scale rather than quality. Keeping these roles distinct matters because a measure used as both explanatory feature and outcome makes the relationship trivially true.

Collaboration value and process quality: Attempts to measure collaboration value directly have produced collaboration gain, a comparison of MAS performance against a resource-equivalent single-agent baseline separating value created by interaction from value created by spending more tokens (Fan et al., 2026). Such comparisons establish *whether* collaboration paid off but not which trajectory properties produced the gain. Two systems with identical gain may differ substantially in how well their agents communicated or integrated information. Process-quality work instead evaluates the trajectory along seven dimensions spanning decomposition, handoffs, communication, verification, and failure containment, since a trajectory may communicate clearly but integrate poorly, or plan well but verify weakly. Three patterns recur. Interaction volume is a weak proxy for quality, since agents may exchange many messages while repeating prior content. Failures arise disproportionately at agent boundaries, during delegation, context transfer, and downstream use. Verification and containment determine whether local errors stay isolated. Section 3.2 states which measures within these dimensions are computable across the benchmarks studied here.

Human Collaboration Research: Human-team research supplies the constructs. Team-process frameworks describe collaboration as interdependent planning, coordination, monitoring, and adaptation (Marks et al., 2001). Teamwork Quality research identifies communication, coordination, balance of contributions, mutual support, effort, and cohesion as distinct facets (Hoegl & Gemuenden, 2001), and meta-analytic evidence supports multiple process measures over a single collaboration score (LePine et al., 2008). A central theme separates exchanging information from integrating it. Sharing predicts performance, but its value depends on relevance and uniqueness rather than volume (Mesmer-Magnus & DeChurch, 2009), and effective teams locate where knowledge sits and carry specialised contributions downstream (Lewis, 2003; Faraj & Sproull, 2000). Shared mental models add that members benefit from compatible understandings of goals, roles, and task state (Mathieu et al., 2000), though agreement is insufficient when the understanding is wrong (Mathieu et al., 2005). We use these as functional lenses rather than equivalences, so communication score and redundancy rate reflect information-sharing quality, downstream usage reflects integration, and contradiction rate reflects consistency of shared understanding. Evolutionary distance (Fan et al., 2026) has no established human-team analogue, so we treat it as agent-specific with no preferred direction.

Robustness and privacy: A system may produce a correct final output while corrupted information spreads internally or private context escapes through messages, memory, tool arguments, or handoffs, so task completion does not guarantee a reliable or privacy-preserving trajectory. AgentCollabBench targets collaboration-specific process failures with metrics such as instruction decay, radioactive tracer durability, consensus pollution, and cross-task leakage containment (Mazumder

et al., 2026). Because these require controlled perturbations, we treat them as robustness outcomes rather than process predictors. Privacy benchmarks extend this view, with AgentLeak instrumenting internal MAS pathways and showing sensitive data leaking through channels final-output audits do not inspect (El Yagoubi et al., 2026), PAC-BENCH evaluating coordination under explicit privacy constraints (Park et al., 2026), and AgentSocialBench studying agentic social networks where agents coordinate across users while protecting sensitive information (Wang & Jiang, 2026). MAST-Data identifies inter-agent misalignment and task-verification failures as recurring causes of MAS breakdown (Cemri et al., 2025), and causal attribution locates where failures originate (Ma et al., 2025). Because privacy constraints may independently alter task difficulty and agent behaviour, we analyse exposure separately from completion.

Positioning: Two gaps remain. The theoretical grounding of MAS collaboration metrics is often implicit, leaving it unclear whether a measure captures a general construct or a benchmark-specific heuristic, so Section 3.2 makes that grounding explicit. Process features, task outcomes, and robustness outcomes are also often reported together without distinguishing their roles, risking validation of a measure against an outcome it partly contains. We assign each candidate exactly one role and report where that separation is imperfect in our own design.

3 METHODOLOGY

3.1 STUDY DESIGN

The study combines a measure-selection stage with an empirical evaluation stage. Candidate MAS collaboration measures are first assigned analytical roles and screened for computability from ordinary trajectories (Section 3.2), after which we evaluate whether the retained process measures predict task success across benchmarks and models. AgentCollabBench, Silo-Bench, EntCollabBench, and CooperBench form the primary task-success testbed, while PAC-BENCH, AgentSocialBench, and AgentLeak form a secondary privacy testbed that assesses whether signals associated with task success transfer to information exposure. The two testbeds are analysed separately and share neither scenarios nor observations. The unit of analysis is the complete trajectory, meaning one execution of one scenario by one multi-agent configuration under one model, with every metric and outcome represented as a trajectory-level scalar and all analyses conducted within benchmark–model cells. The design is observational rather than experimental, and Section 3.8 states what this does and does not license.

3.2 METRIC SELECTION AND HARMONISATION

The measures we evaluate are not a new proposal but a screened subset of an existing pool of 25 process measures assembled from the seven process-quality dimensions in Section 2 and prior MAS evaluation frameworks (Zhu et al., 2025; Fan et al., 2026; Mazumder et al., 2026).

Measures were retained as *process features* only if they were trace-computable (C1), required no benchmark modification (C2), reduced to a trajectory-level scalar (C3), mapped to exactly one process dimension (C4), and were harmonisable across benchmarks without benchmark-specific ground truth (C5). These criteria encode one requirement, that a retained measure must be computable from what a deployed system already produces, without privileged ground truth or benchmark perturbation. C1–C2 ensure applicability outside a research harness, C3–C4 ensure one interpretable meaning per measure, and C5 ensures comparability across benchmarks rather than one-off validation.

Every candidate was assigned exactly one analytical role, either process feature, task outcome, structural descriptor, or robustness/privacy outcome. Five measures survived as process features, namely communication score, redundancy rate, downstream usage rate, contradiction rate, and evolutionary distance.

C5 was the main constraint, excluding measures that depend on benchmark-supplied ground truth and leaving three process dimensions without a retained measure, including verification quality. We also exclude collaboration gain (Fan et al., 2026), which requires matched single-agent baselines unavailable here. Table 2 in Appendix A reports the full pool and exclusion reasons.

3.3 Operational Definitions

All five measures are defined over the same object. For a trajectory τ, let the *eligible events* be the ordered sequence $E(\tau) = (e_1, \ldots, e_m)$ of inter-agent messages, artefacts, and final outputs in trajectory order. Tool traffic, execution logs, and memory writes are excluded as they are not agent-to-agent handoffs. This definition is applied identically in all seven benchmarks and is the most consequential choice in the pipeline, because it fixes the denominator of three of the five measures.

Communication score rescales the judge's rating $C \in \{1, \ldots, 5\}$ to the unit interval as $(C-1)/4$, with rubric bands running from poor (≤ 0.25) to strong (> 0.75). The three event-level rates are shares of $E(\tau)$ but take different denominators. *Redundancy rate* is the share of events repeating prior content without adding usable information, over the $m-1$ events having prior context. *Downstream usage rate* is the share a later event materially relies on, over the $m-1$ events having a successor. This is the most restrictive denominator, since a trajectory in which no event has a relied-upon successor contributes no observation at all, which is why downstream usage has the smallest analysed samples in Section 4.1. *Contradiction rate* is the share participating in a substantive contradiction unresolved at trajectory end, over all m events. A contradiction raised and then resolved does not count, so this measures residual inconsistency rather than disagreement. The three rates are therefore not interchangeable even where they appear numerically similar.

Evolutionary distance is the mean cosine distance between embeddings of consecutive eligible events,

$$\mathrm{ED}(\tau) = \frac{1}{m-1} \sum_{i=1}^{m-1} \left(1 - \frac{\phi(e_i) \cdot \phi(e_{i+1})}{\|\phi(e_i)\| \, \|\phi(e_{i+1})\|} \right),$$

with $\phi(\cdot)$ the embedding function. Because cosine distance ranges over $[0, 2]$ this measure has no preferred direction, and we treat it as a healthy-range quantity in which values near zero indicate near-duplicate handoffs rather than efficient ones and large values indicate topic drift or context fracture. Where that range lies depends on task type, embedding model, and message granularity, so we make no directional prediction and report the sign we observe.

3.4 Measurement Instrument

Communication score, redundancy, downstream usage, and contradiction rate come from a single LLM judge held fixed across all runs and pinned independently of the evaluated models, while evolutionary distance is computed using a single fixed embedding model (configuration in Appendix A.6). This consistency supports stability auditing, but it does not validate the instrument. If the judge applies labels noisily or differently across domains, any resulting benchmark dependence would arise from the measurement pipeline rather than from collaboration itself.

Three properties of the instrument constrain the interpretation of our results. First, the communication rubric partially distinguishes its middle bands by whether the task progresses, so communication score and task completion are not fully independent, and the measure with the greatest conceptual overlap with the outcome is also its strongest predictor. We retain the rubric unchanged because our objective is to evaluate existing measures within a common pipeline rather than redesign them, as a task-progress-free rubric would address a different question. We therefore read communication score as a diagnostic of process rather than an independent cause of success, and the privacy testbed gives a cleaner assessment because its rubric never references information exposure. As a check on how much the overlap drives the result, we re-scored every trajectory with all task-progress wording removed (Appendix A.6.1). The sign and rough size of the association held in most configurations, with a real but benchmark-concentrated attenuation in one of four benchmarks. Holding that wording and changing the judge changed nothing detectable (Appendix A.6.3), so the overlap is a rubric property, not an artefact of judge identity.

Second, all four judged measures come from a single batched judgment procedure. Communication score therefore aggregates facets the same judge has already labelled individually, so associations among these measures are not agreement between independent instruments. The second judge covers communication score only, where agreement was substantial (weighted $\kappa = 0.75$, $n = 2{,}461$). For the other three, and for human labels on any measure, inter-rater reliability remains unmeasured. We therefore treat these measures as a fixed pipeline whose associations can be audited, and read

their survival counts as evidence for the stability of one operationalisation rather than for established human validity.

Third, this same procedure means all four measures are judged with full-trajectory visibility. Because every event is serialised into one call, the judge has already read how the trajectory ends, including its outcome, when it labels redundancy, downstream usage, and contradiction at each message. The hindsight-bias risk noted above therefore applies pipeline-wide, which is why we ran the progress-blind check rather than treating it as specific to one measure. What does differ across measures is each rate's denominator, since redundancy excludes the first communication-eligible message, having no earlier message to repeat, and downstream usage excludes the last, having no later message to use it. Both narrow what is counted, not what the judge is shown.

3.5 Benchmarks, Models, and Data Preparation

The four primary benchmarks span algorithmic coordination, software engineering, and enterprise workflows, differing in coordination structure, agent count, and multi-agent design. They also operationalise task success differently. AgentCollabBench combines task-level outcomes with milestone progress (Zhu et al., 2025; Mazumder et al., 2026), Silo-Bench distinguishes partial correctness from complete coordination success (Zhang et al., 2026), EntCollabBench evaluates agent-, subtask-, and workflow-level completion from execution traces and database-state changes (Yu et al., 2026), and CooperBench scores whether interacting coding agents complete interdependent implementation tasks (Khatua et al., 2026). The three privacy benchmarks likewise score completion over two requirements in AgentLeak, five in PAC-BENCH, and one strict compound criterion in AgentSocialBench. Completion is therefore never pooled across benchmarks, a level difference is never read as evidence about relative difficulty, and all associations are estimated within a benchmark–model cell.

The grid is incomplete for the reason given in footnote 1. The exclusion was made on harness behaviour before any metric or outcome was inspected. We note the residual risk that non-engagement is partly a property of the model rather than the harness, in which case the surviving grid is mildly favourable to it, though Appendix A.5 finds its behaviour unremarkable where it does appear.

Three exclusions were applied to the raw per-trajectory files, and the primary table cannot be reproduced without them. Trajectories whose collaboration evaluation terminated in an error were removed as pipeline-side missing measurements rather than observed outcomes, and were not imputed. Trajectories carrying a task-completion score but no collaboration metrics were dropped from the correlations while retaining their task score in the reported mean. Per-metric sentinel values, written when a measure could not be computed for an otherwise valid trajectory, were excluded metric by metric rather than dropping the trajectory. The last rule is load-bearing, since treating the sentinel as observed in one cell (EntCollabBench with Haiku 4.5) moves evolutionary distance from $+0.13$ to $+0.54$, redundancy from -0.05 to $+0.49$, and downstream usage from $+0.03$ to $+0.51$, because a few low-completion trajectories share the same out-of-range value on several measures and act as high-leverage points. Because the rules apply per metric, analysed n varies across cells within a row. Counts appear in Table 3 and exact p-values in Appendix A.

3.6 Statistical Analysis

For each of the 18 primary cells and five measures we estimate the trajectory-level Pearson correlation between measure and task completion, with two-sided p-values from the t transform on $n - 2$ degrees of freedom. The estimand is a within-cell marginal association, not a causal effect and not an incremental contribution over other measures. The analysis comprises 90 tests, so we declare the correction structure rather than reporting nominal significance alone. We treat each metric column as a family of 18 tests, one per configuration, and apply Holm correction within column. This is the family matching the question asked of the table, namely whether a given measure holds up across configurations. Corrections over all 90 cells are also reported in Appendix A, and no individual cell is a confirmatory test.

Power varies substantially. Analysed n ranges from 45 to 213, so $|r| = 0.29$ is required for $p \leq .05$ at $n = 45$ whereas $|r| = 0.14$ suffices at $n = 213$. Nominal significance is therefore not monotone in effect size across a column, a null in a small cell is not evidence of absence, and we report

n beside every coefficient. Three further caveats apply. Communication score takes five distinct values and several completion distributions are bounded near the floor, so Pearson is not obviously the right default and we report Spearman as a robustness check, flagging cells where rank and linear analyses disagree. Trajectories within a benchmark share scenarios, so within-cell observations are not fully independent and the primary standard errors, which do not model this, are optimistic. The secondary analysis clusters by scenario and is the more conservative half of the paper. Finally, RQ1 compares marginal associations across benchmarks, models, and coordination structures rather than estimating incremental variance over a joint model, and team size, message count, and interaction length vary across our primary benchmarks by construction but are unavailable at trajectory level, so we do not include them as covariates here.

3.7 Secondary Analysis

The privacy testbed comprises 1,500 trajectories from 100 scenarios, five models, and three benchmarks. Task completion and information exposure are analysed as separate outcomes, and we do not use the formula-derived privacy-safe task score as an endpoint because it combines the two algebraically. Associations are estimated with fixed-effects linear models, and linear-probability models for binary exposure, including model and scenario fixed effects, with predictors standardised and sign-aligned so a positive coefficient denotes improvement. Standard errors are clustered by scenario with a CR1 correction, multiplicity is controlled by Holm over a declared family of 18 benchmark-by-metric comparisons, and interval estimates for leakage incidence use 50,000 scenario-cluster bootstrap resamples preserving all five model runs per sampled scenario. Measures are fitted one at a time, with a four-measure common-process block reported separately (Appendix B).

3.8 Scope of Claims

Four restrictions bound what the results support. The design is observational, with collaboration measured as it occurred rather than manipulated, so where a measure and an outcome covary the association is equally compatible with collaboration influencing the outcome, the outcome influencing the judged measure, and capability driving both. No perturbation experiment separates these. Predictors and outcome share method variance, since four of five measures come from one judge in one batched judgment and the rubric partly references task progress. This overlap is expected rather than excluded, and is not an artefact of judge identity, though correlated errors between two model judges remain possible. The privacy testbed is cleaner in construct but weakly powered (median minimum detectable effect 7.3 percentage points), constraining the primary result without settling it. Coverage is partial in two directions, four of seven process dimensions and 18 of 20 primary configurations. Neither gap is outcome-conditioned, but both narrow the claim. Nothing was pre-registered, since the screen and multiplicity families were fixed before the correlations were interpreted but not before they were computed, so we report them as declared rather than pre-specified.

4 Results

4.1 Collaboration Quality and Task Success

Table 1 summarises the trajectory-level Pearson associations between each collaboration metric and task completion across the 18 model–benchmark configurations. Each row gives one metric's behaviour across those configurations, and the Holm column gives how many of its tests are conclusive rather than merely suggestive, with the $|r|$ range covering only the cells that survive correction. The full coefficient grid, per-cell sample sizes, exact Pearson p-values, and Spearman robustness checks appear in Appendix A, which also plots all 90 coefficients with confidence intervals (Figure 1).

Communication score. All 14 significant cells survive Holm correction, the weakest being CooperBench with Sonnet 4.6 ($p = 3.1 \times 10^{-3}$). Communication score is the only measure that is both directionally stable and significant in most configurations, and it yields the largest absolute correlation in 12 of 18 cells. The four non-significant cells are EntCollabBench with GPT 5.4 Mini ($+0.15$), Nano ($+0.07$), and Haiku 4.5 (-0.00), together with CooperBench with Nano ($+0.13$, $p = .066$). Since EntCollabBench is the lowest-powered benchmark, needing roughly $|r| = 0.25$

Table 1: Summary of primary task-success associations across the 18 model–benchmark configurations. "Agreement" counts configurations whose coefficient matches the expected direction, omitted for evolutionary distance because that measure has no preferred direction. "Nominal" counts cells at $p \leq .05$ uncorrected, "Holm" those surviving Holm correction applied within each column over its 18 tests (Section 3.6), and "$|r|$ range" spans the Holm survivors. Full coefficients, per-cell n, and exact p-values are in Table 3.

Metric	Expected	Agreement	Nominal	Holm	$\lvert r \rvert$ range	Main pattern
Evolutionary distance	none	–	8	2	0.35–0.45	sign reverses within benchmarks
Communication score	+	17/18	14	14	0.23–0.77	strongest; overlaps outcome
Redundancy rate	−	14/18	8	6	0.29–0.46	one positive cell removed by Holm
Downstream usage	+	16/18	6	3	0.31–0.40	weakest; smaller analysed n
Contradiction rate	−	16/18	7	6	0.31–0.50	all significant cells negative

for nominal significance at $n = 45$–60, we read these as inconclusive rather than as benchmark-specific failures. No consistent model-level ordering emerges, and the association appears to depend on the model–benchmark pairing rather than on model scale. Rubric overlap nevertheless means this association cannot be cleanly separated from the outcome itself, as Section 3.4 notes.

Evolutionary distance. Evolutionary distance is significant in eight configurations, but only two survive correction, Silo-Bench with Haiku 4.5 (-0.45) and AgentCollabBench with Sonnet 4.6 (-0.35). The sign is unstable and reverses even within a benchmark, since Silo-Bench gives $+0.18$ for GPT 5.4 Nano but -0.45 for Haiku 4.5, while EntCollabBench gives $+0.29$ for Nano and -0.35 for Sonnet 4.6. Such reversals are inconsistent with a monotone quality ordering, matching the healthy-range interpretation of Section 3.3, where embedding distance alone cannot serve as a directional measure without first identifying the benchmark- and model-specific range associated with effective collaboration.

Event-level measures. The remaining three align with expectations but are not consistently significant. Contradiction rate is significant in seven cells and negative in all seven, with six surviving correction and reaching -0.50 for CooperBench with Sonnet 4.6, so unresolved contradiction is never beneficial. For redundancy rate, eight cells are significant and seven negative, reaching -0.46 in AgentCollabBench with Haiku 4.5, and six survive. Its only significant positive coefficient, Silo-Bench with GPT 5.4 Mini ($+0.21$, $p = .022$), does not survive correction, and that configuration has the study's second-lowest mean completion (0.071), where repeated content may reflect subtask retries rather than wasted effort. Downstream usage is weakest, with six significant cells that are all positive (largest $+0.40$ in Silo-Bench with GPT 5.4 Mini) but only three survivors, partly because its restrictive denominator reduces the analysed sample.

Across the 90 cells, 43 are nominally significant and 31 survive Holm correction, distributed 2/14/6/3/6 over evolutionary distance, communication score, redundancy, downstream usage, and contradiction rate. The Spearman analyses in Appendix A.2 give the same picture, agreeing with Pearson in sign for 84 of the 90 cells, in nominal significance for 80, and in Holm status for 83, and yielding 41 nominal and 30 surviving cells against 43 and 31. Section 4.2 examines whether this generalises to a different outcome.

4.2 Privacy-Constrained Collaboration

We next test whether the process–outcome distinction persists under privacy constraints in 1,500 trajectories from PAC-BENCH, AgentSocialBench, and AgentLeak (Park et al., 2026; Wang & Jiang, 2026; El Yagoubi et al., 2026). Completion and leakage are analysed separately because differences in completion criteria, scenario structure, sensitive-item denominators, and detection pipelines preclude cross-benchmark ranking (Section 3.5). Full methods and sensitivity analyses appear in Appendix B.

Task success and privacy preservation: Performance and privacy were not interchangeable (Table 8). The highest-completion model was not consistently the lowest-exposure model, and trajectory-level completion–leak correlations were small in AgentLeak (.021), PAC-BENCH (.044), and AgentSocialBench (.080). Task success therefore does not imply privacy preservation.

Collaboration measures and privacy outcomes: In direction-aligned scenario and model-fixed-effects models, no individual collaboration or robustness measure survived Holm correction for com-

pletion (smallest adjusted $p = .131$), and the four-measure common-process block met the threshold only in PAC-BENCH ($p = .048$). None survived correction for leakage either (smallest adjusted $p = .186$). One supplementary privacy-specific instruction-adherence measure did track exposure (-21.0 percentage points in PAC-BENCH and -10.7 in AgentSocialBench; adjusted $p < .001$ and $p = .027$), but we treat it as a negative control rather than a finding, because it counts violations of injected privacy constraints and the evaluator assigning those labels saw known disclosure evidence (Appendix B.5), making it procedurally non-independent of the leakage outcome. The power floor for a typical common-measure leakage model is roughly 7.3 percentage points per standard deviation (range 2.3–8.8; Appendix B.6), so large collaboration–privacy associations are excluded more confidently than small ones. The conclusion is unchanged when visible and hidden exposure are analysed separately, since no measure survives the 15-test Holm correction for leaks absent from the final output (smallest adjusted $p = .642$), although communication score is directionally negative in all three benchmarks (-0.014 to -0.039 per standard deviation). Conditional on trajectories that leaked, no measure is associated with hidden versus visible leakage ($p = .12$–$.59$; Appendix B.11).

Visibility of privacy failures: Only 53 of the 259 leaking trajectories exposed sensitive information in the final output, so a final-output-only audit would have missed the remaining 206 (79.5%; 95% CI 74.3–84.6). Terminal-answer evaluation therefore misses many failures occurring in inter-agent messages, tool arguments, memory writes, and shared artefacts.

5 Discussion and Conclusion

Returning to RQ1 and RQ2, the principal finding is negative and does not depend on the validity of our judging instrument. Of five measures surviving the analytical-role screen, four do not hold up across benchmarks and models, and the fifth cannot be separated from the outcome it tracks, so they are not interchangeable indicators of a single construct. Under this operationalisation the event-level rates survive correction in at most six of 18 configurations, too benchmark-dependent for standalone dashboard use, and evolutionary distance reverses sign within a benchmark, so it cannot be read directionally without a task-specific healthy range. This addresses RQ1 only at the level the data support, because the retained measures are trace-computable trajectory scalars whereas team size, message count, and interaction length are benchmark-level properties unavailable per trajectory here (Section 3.6). We therefore cannot decompose associations into components beyond versus within structure, and the two families answer different questions.

For RQ3, the leakage analyses test more cleanly whether the retained measures capture a general collaboration construct, since the communication rubric never references information exposure, and we find no reliably detectable relationship. Two readings fit. Either the signal is outcome-specific, or the primary association is largely the rubric's task-progress overlap and the privacy null is what remains without it. The progress-blind re-scoring (Appendix A.6.2) favours the first, because stripping every task-progress reference leaves the association largely intact in three of four benchmarks, with Holm survival falling only from 14 to 11. It does account for much of the AgentCollabBench association, so that caveat holds for one benchmark rather than the result as a whole. The visibility result in Section 4.2 bears on this independently, since most leakage never reaches the final output at all.

These findings suggest a checklist for validating a MAS collaboration metric before treating it as general-purpose. The requirements are (1) *outcome-specificity*, testing against more than one outcome such as task success and a risk outcome like privacy exposure, since validation on one need not transfer, (2) *cross-context stability*, testing across benchmarks, models, and coordination structures under one instrument, (3) *construct independence*, verifying that the rubric or computation does not reference the outcome it predicts or else disclosing the overlap, (4) *measurement reliability*, establishing agreement with a second judge or human subsample per configuration rather than pooled, and (5) *power disclosure*, reporting the minimum detectable effect alongside any non-detection. Applied to our own measures, only communication score meets (1) and (2) unconditionally and partially meets (4), and even it fails (3). Resource-matched single-agent baselines and benchmark-side instrumentation for verification and containment remain the next steps.

The practical conclusion is deliberately narrow. A collaboration measure earns validity for the analytical role it serves against a specified outcome, setting, and correction procedure.

6 GenAI Usage Disclosure

In line with our experimental design, LLMs and LLM-powered systems were used to generate and execute the agentic workflows evaluated in this paper, and an LLM judge scores four of the five collaboration metrics (Sections 3.4 and 3.6, Appendix A.6).

We also used LLMs in preparing this paper to assist with light editing, including grammar correction and sentence restructuring. All reported results were computed from the data and verified by the authors, who take full responsibility for the content of this paper.

References

Mert Cemri, Melissa Z. Pan, Shuyi Yang, Lakshya A. Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, Matei Zaharia, Joseph E. Gonzalez, and Ion Stoica. Why do multi-agent llm systems fail?, 2025. URL https://arxiv.org/abs/2503.13657.

Faouzi El Yagoubi, Godwin Badu-Marfo, and Ranwa Al Mallah. Agentleak: A benchmark for internal-channel privacy leakage in multi-agent llm systems, 2026. URL https://arxiv.org/abs/2602.11510.

Jingru Fan, Dewen Liu, Yufan Dang, Huatao Li, Yuheng Wang, Wei Liu, Feiyu Duan, Xuanwen Ding, Shu Yao, Lin Wu, Ruijie Shi, Wai-Shing Leung, Yuan Cheng, Zhongyu Wei, Cheng Yang, Chen Qian, Zhiyuan Liu, and Maosong Sun. Towards a science of collective ai: Llm-based multi-agent systems need a transition from blind trial-and-error to rigorous science, 2026. URL https://arxiv.org/abs/2602.05289.

Samer Faraj and Lee Sproull. Coordinating expertise in software development teams. *Management Science*, 46(12):1554–1568, 2000. doi: 10.1287/mnsc.46.12.1554.12072. URL https://doi.org/10.1287/mnsc.46.12.1554.12072.

Martin Hoegl and Hans Georg Gemuenden. Teamwork Quality and the success of innovative projects: A theoretical concept and empirical evidence. *Organization Science*, 12(4):435–449, 2001. doi: 10.1287/orsc.12.4.435.10635. URL https://doi.org/10.1287/orsc.12.4.435.10635.

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, 2024. URL https://arxiv.org/abs/2308.00352.

Arpandeep Khatua, Hao Zhu, Peter Tran, Arya Prabhudesai, Frederic Sadrieh, Johann K. Lieberwirth, Xinkai Yu, Yicheng Fu, Michael J. Ryan, Jiaxin Pei, and Diyi Yang. Cooperbench: Why coding agents cannot be your teammates yet, 2026. URL https://arxiv.org/abs/2601.13295.

Yubin Kim, Ken Gu, Chanwoo Park, Chunjong Park, Samuel Schmidgall, A. Ali Heydari, Yao Yan, Zhihan Zhang, Yuchen Zhuang, Yun Liu, Mark Malhotra, Paul Pu Liang, Hae Won Park, Yuzhe Yang, Xuhai Xu, Yilun Du, Shwetak Patel, Tim Althoff, Daniel McDuff, and Xin Liu. Towards a science of scaling agent systems, 2026. URL https://arxiv.org/abs/2512.08296.

Jeffery A. LePine, Ronald F. Piccolo, Christine L. Jackson, John E. Mathieu, and Jessica R. Saul. A meta-analysis of teamwork processes: Tests of a multidimensional model and relationships with team effectiveness criteria. *Personnel Psychology*, 61(2):273–307, 2008. doi: 10.1111/j.1744-6570.2008.00114.x. URL https://doi.org/10.1111/j.1744-6570.2008.00114.x.

Kyle Lewis. Measuring transactive memory systems in the field: Scale development and validation. *Journal of Applied Psychology*, 88(4):587–604, 2003. doi: 10.1037/0021-9010.88.4.587. URL https://doi.org/10.1037/0021-9010.88.4.587.

Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More agents is all you need, 2024. URL https://arxiv.org/abs/2402.05120.

Guoqing Ma, Jia Zhu, Hanghui Guo, Weijie Shi, Jiawei Shen, Jingjiang Liu, and Yidan Liang. Automatic failure attribution and critical step prediction method for multi-agent systems based on causal inference, 2025. URL https://arxiv.org/abs/2509.08682.

Michelle A. Marks, John E. Mathieu, and Stephen J. Zaccaro. A temporally based framework and taxonomy of team processes. *Academy of Management Review*, 26(3):356–376, 2001. doi: 10.5465/amr.2001.4845785. URL https://doi.org/10.5465/amr.2001.4845785.

John E. Mathieu, Tonia S. Heffner, Gerald F. Goodwin, Eduardo Salas, and Janis A. Cannon-Bowers. The influence of shared mental models on team process and performance. *Journal of Applied Psychology*, 85(2):273–283, 2000. doi: 10.1037/0021-9010.85.2.273. URL https://doi.org/10.1037/0021-9010.85.2.273.

John E. Mathieu, Tonia S. Heffner, Gerald F. Goodwin, Janis A. Cannon-Bowers, and Eduardo Salas. Scaling the quality of teammates' mental models: Equifinality and normative comparisons. *Journal of Organizational Behavior*, 26(1):37–56, 2005. doi: 10.1002/job.296. URL https://doi.org/10.1002/job.296.

Aritra Mazumder, Shubhashis Roy Dipta, Nusrat Jahan Lia, Tanzila Khan, Kainat Raisa Hossain, Nehaa Shri, Shubhrangshu Debsarkar, Humayra Tasnim, Gour Gupal Talukder Shawon, Debjoty Mitra, Sumaiya Ahmed Rani, Al Jami Islam Anik, and Al Nafeu Khan. Agentcollabbench: Diagnosing when good agents make bad collaborators, 2026. URL https://arxiv.org/abs/2605.08647.

Jessica R. Mesmer-Magnus and Leslie A. DeChurch. Information sharing and team performance: A meta-analysis. *Journal of Applied Psychology*, 94(2):535–546, 2009. doi: 10.1037/a0013773. URL https://doi.org/10.1037/a0013773.

Minjun Park, Donghyun Kim, Hyeonjong Ju, Seungwon Lim, Dongwook Choi, Taeyoon Kwon, Minju Kim, and Jinyoung Yeo. Pac-bench: Evaluating multi-agent collaboration under privacy constraints, 2026. URL https://arxiv.org/abs/2604.11523.

Dat Tran and Douwe Kiela. Single-agent llms outperform multi-agent systems on multi-hop reasoning under equal thinking token budgets, 2026. URL https://arxiv.org/abs/2604.02460.

Prince Zizhuang Wang and Shuli Jiang. Agentsocialbench: Evaluating privacy risks in human-centered agentic social networks, 2026. URL https://arxiv.org/abs/2604.01487.

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, 2023. URL https://arxiv.org/abs/2308.08155.

Tao Yu, Hao Wang, Changyu Li, Shenghua Chai, Minghui Zhang, Zhongtian Luo, Yuxuan Zhou, Haopeng Jin, Zhaolu Kang, Jiabing Yang, YiFan Zhang, Xinming Wang, Hongzhu Yi, Zheqi He, Jing-Shu Zheng, Xi Yang, Yan Huang, and Liang Wang. Beyond the all-in-one agent: Benchmarking role-specialized multi-agent collaboration in enterprise workflows, 2026. URL https://arxiv.org/abs/2605.08761.

Yuzhe Zhang, Feiran Liu, Yi Shan, Xinyi Huang, Xin Yang, Yueqi Zhu, Xuxin Cheng, Cao Liu, Ke Zeng, Terry Jingchen Zhang, and Wenyuan Jiang. Silo-bench: A scalable environment for evaluating distributed coordination in multi-agent llm systems, 2026. URL https://arxiv.org/abs/2603.01045.

Kunlun Zhu, Hongyi Du, Zhaochen Hong, Xiaocheng Yang, Shuyi Guo, Zhe Wang, Zhenhailong Wang, Cheng Qian, Xiangru Tang, Heng Ji, and Jiaxuan You. Multiagentbench: Evaluating the collaboration and competition of llm agents, 2025. URL https://arxiv.org/abs/2503.01935.

Table 2: The candidate pool and the screen. Sources: MAB, MultiAgentBench-style metrics (Zhu et al., 2025); SCI, collective-AI information dynamics (Fan et al., 2026); ACB, AgentCollabBench stress tests (Mazumder et al., 2026); Op., operational trace-level measures following from the seven dimensions. The screen column gives the retention decision or the first criterion (Section 3.2) a candidate fails; retained measures are defined in Section 3.3. Roles: PF, process feature, entering the primary analysis; RO, robustness or privacy outcome; SD, structural descriptor; –, not analysed.

Dimension	Candidate (source)	Screen	Role
Decomposition & role	Planning score (MAB); role–task fit, dependency coverage (Op.)	fails C5 (needs reference plan, assignment, or dependencies)	–
Handoff	**Evolutionary distance** (SCI)	**retained**: healthy range	**PF**
	Handoff schema, semantic completeness (Op.)	fails C5 (needs schema, labels)	–
	Tracer retention (ACB)	fails C2 (needs tagged facts)	RO
Communication	**Communication score** (MAB)	**retained**: higher better	**PF**
	Redundancy rate (Op.)	**retained**: lower better	**PF**
	Coordination-token ratio (Op.)	fails C5 (token accounting differs)	–
	Actionability rate (Op.)	fails C5 (not harmonised)	–
Semantic contribution	**Downstream usage rate** (Op.)	**retained**: higher better	**PF**
	Milestone contribution coverage (MAB)	fails C5 (needs milestone list)	–
	Validated novelty rate (Op.)	fails C5 (needs validation labels)	–
Consistency	**Contradiction rate** (Op.)	**retained**: lower better	**PF**
	State consistency rate (Op.)	fails C5 (needs ground-truth state)	–
	Entropy trajectory health (SCI)	fails C3 (a shape, not a scalar)	–
Verification	Detection recall, precision, miss rate, repair success, repair quality (Op.)	fails C5 (needs labelled issue set)	–
Containment	Failure propagation depth, recovery rate, containment latency (Op.)	fails C5 (needs error-origin labels)	–
	Consensus pollution rate (ACB)	fails C2 (needs seeded belief)	RO
Risk layer	Instruction decay, tracer durability, cross-task leakage (ACB)	fails C2 (needs injected constraint or marker)	RO
Structural	Message count, interaction length, team size	role reassigned	SD
Gain layer	Collaboration gain Γ, $\Delta\Gamma$ (SCI)	fails C1 (needs paired baseline)	–

A Primary Benchmarks: Samples, Measurement, and Reproducibility

This appendix documents the construction of the primary correlation analysis reported in Section 4.1: how the analysed samples were derived, exact significance levels, per-model task-completion levels, the judge and embedding configuration, and the artefacts the numbers were computed from.

A.1 Analysed Samples and Exclusion Rules

The three exclusion rules are stated in Section 3.5. This subsection records how many trajectories each rule removed and from which files.

The first rule, removal of trajectories whose collaboration evaluation terminated in an error, affected only the AgentCollabBench files, where such failures appear as a stored error string in place of a metrics record: 48 of 228 trajectories for GPT 5.4 Mini, 48 of 228 for GPT 5.4 Nano, 46 of 207 for Haiku 4.5, 53 of 204 for Sonnet 4.6, and 48 of 228 for Mistral Large. No trajectories in the Silo-Bench, EntCollabBench, or CooperBench files were affected.

Table 3: Full Pearson correlation grid between each collaboration metric and task completion, by benchmark and language model. Redundancy and contradiction rate are reported in their natural direction, so a negative coefficient denotes improvement in those two columns. Coefficients are given without significance markers because analysed n varies by metric within a row, so a column is not comparable at equal power ($|r| = 0.29$ is needed for $p \leq .05$ at $n = 45$ but $|r| = 0.14$ at $n = 213$); exact per-cell p-values and sample sizes are in Table 4. Of the 90 cells, 43 reach $p \leq .05$ uncorrected and 31 survive within-column Holm correction (Section 3.6); communication score is the only column in which every nominally significant cell survives. Mistral Large 1.0 could not be run on EntCollabBench or CooperBench.

Model	Mean task completion	Analysed n	Evol. distance	Comm. score	Redun. rate	Downstr. usage	Contra. rate
AgentCollabBench							
GPT 5.4 Mini	0.689	159–180	+0.07	+0.29	−0.08	+0.02	−0.04
GPT 5.4 Nano	0.706	160–180	−0.15	+0.37	+0.01	−0.02	−0.14
Haiku 4.5	0.410	161	−0.00	+0.53	−0.46	+0.08	−0.31
Sonnet 4.6	0.619	151	−0.35	+0.64	−0.29	+0.20	−0.32
Mistral Large	0.634	166–180	−0.09	+0.62	−0.19	+0.05	−0.20
Silo-Bench							
GPT 5.4 Mini	0.071	92–119	−0.19	+0.41	+0.21	+0.40	−0.38
GPT 5.4 Nano	0.134	164	+0.18	+0.61	−0.45	+0.20	−0.07
Haiku 4.5	0.417	118	−0.45	+0.77	−0.29	+0.11	−0.41
Sonnet 4.6	0.444	107–108	−0.04	+0.70	−0.15	+0.35	−0.19
Mistral Large	0.052	120	−0.13	+0.36	−0.34	+0.11	+0.02
EntCollabBench							
GPT 5.4 Mini	0.429	45	+0.14	+0.15	−0.28	+0.25	+0.02
GPT 5.4 Nano	0.390	59–60	+0.29	+0.07	−0.43	+0.02	−0.13
Haiku 4.5	0.628	58–60	+0.13	−0.00	−0.05	+0.03	−0.06
Sonnet 4.6	0.411	60	−0.35	+0.43	−0.25	−0.04	−0.23
CooperBench							
GPT 5.4 Mini	0.601	213	−0.15	+0.49	−0.11	+0.20	−0.11
GPT 5.4 Nano	0.402	191–203	+0.10	+0.13	+0.11	+0.31	−0.02
Haiku 4.5	0.676	181	−0.14	+0.32	+0.04	+0.07	−0.33
Sonnet 4.6	0.758	163	−0.05	+0.23	−0.11	+0.02	−0.50

The second rule, exclusion of trajectories carrying a task-completion score but no collaboration metrics, affected 5 CooperBench trajectories under GPT 5.4 Nano and 16 Silo-Bench trajectories under Sonnet 4.6. Their task scores are retained in the reported completion means.

The third rule, per-metric exclusion of the -1 sentinel, is the one whose effect on the coefficients is documented in Section 3.5. The affected cell is EntCollabBench with Haiku 4.5, where a small number of low-completion trajectories share the same out-of-range value on several metrics at once and act as high-leverage points.

Table 4 reports the exact two-sided p-value and the analysed n for every cell of Table 3. Downstream usage has the smallest samples because its denominator excludes eligible events with no later output to depend on, which removes whole trajectories when no such event exists: 159 of 180 for AgentCollabBench with GPT 5.4 Mini and 92 of 119 for Silo-Bench with GPT 5.4 Mini are the two most affected cells. The p-values in Table 4 are uncorrected, and are reported so that the multiplicity corrections declared in Section 3.6 can be recomputed from them; no individual cell is a confirmatory test.

As a stricter alternative to the within-column families of Section 3.6, we also pooled all 90 cells into a single Holm family. The count of survivors falls from 31 to 29: communication score loses one nominally significant cell (13 of 18 survive rather than 14) and redundancy rate loses one (5 of 18 rather than 6); evolutionary distance (2/18), downstream usage (3/18), and contradiction rate (6/18) are unchanged. The two cells that survive within-column correction but not the global family are Silo-Bench with Haiku 4.5 for redundancy rate ($r = -0.30$, $n = 118$, $p = .0012$, column-Holm $= .016$, global-Holm $= .073$) and CooperBench with Sonnet 4.6 for communication score

Table 4: Exact two-sided p-values for the correlations in Table 3, with the analysed n for each cell in parentheses. Sample sizes vary within a row because the three exclusion rules in Appendix A.1 are applied per metric.

Benchmark	Model	Evol. distance	Comm. score	Redun. rate	Downstr. usage	Contra. rate
AgentCollabBench	GPT 5.4 Mini	.321 (180)	<.001 (180)	.281 (180)	.827 (159)	.563 (180)
	GPT 5.4 Nano	.043 (180)	<.001 (180)	.852 (180)	.834 (160)	.057 (180)
	Haiku 4.5	.870 (161)	<.001 (161)	<.001 (161)	.316 (161)	<.001 (161)
	Sonnet 4.6	<.001 (151)	<.001 (151)	<.001 (151)	.016 (151)	<.001 (151)
	Mistral Large	.226 (180)	<.001 (180)	.011 (180)	.490 (166)	.006 (180)
Silo-Bench	GPT 5.4 Mini	.043 (119)	<.001 (119)	.022 (119)	<.001 (92)	<.001 (119)
	GPT 5.4 Nano	.024 (164)	<.001 (164)	<.001 (164)	.012 (164)	.401 (164)
	Haiku 4.5	<.001 (118)	<.001 (118)	.001 (118)	.243 (118)	<.001 (118)
	Sonnet 4.6	.683 (108)	<.001 (108)	.124 (108)	<.001 (107)	.053 (108)
	Mistral Large	.171 (120)	<.001 (120)	<.001 (120)	.238 (120)	.826 (120)
EntCollabBench	GPT 5.4 Mini	.367 (45)	.312 (45)	.067 (45)	.098 (45)	.912 (45)
	GPT 5.4 Nano	.025 (59)	.578 (60)	<.001 (59)	.904 (59)	.319 (60)
	Haiku 4.5	.341 (58)	.663 (60)	.734 (58)	.829 (58)	.633 (60)
	Sonnet 4.6	.007 (60)	<.001 (60)	.054 (60)	.773 (60)	.081 (60)
CooperBench	GPT 5.4 Mini	.029 (213)	<.001 (213)	.116 (213)	.004 (213)	.102 (213)
	GPT 5.4 Nano	.169 (191)	.066 (203)	.136 (191)	<.001 (191)	.830 (203)
	Haiku 4.5	.064 (181)	<.001 (181)	.592 (181)	.343 (181)	<.001 (181)
	Sonnet 4.6	.567 (163)	.003 (163)	.145 (163)	.843 (163)	<.001 (163)

($r = +0.23$, $n = 163$, $p = .0034$, column-Holm $= .017$, global-Holm $= .20$). Both readings agree that communication score is the most consistently significant measure and evolutionary distance the least; the choice of family changes two borderline cells but not the qualitative pattern reported in Section 4.1.

A.2 Spearman Robustness Check

Table 5 repeats the primary within-cell analysis using Spearman rank correlation. The six sign disagreements with Pearson (Section 4.1) all occur where at least one coefficient is close to zero.

A.3 Evolutionary Distance: A Nonlinearity Check

Because evolutionary distance changes sign across configurations (Section 4.1), we checked whether a quadratic term in an unclustered OLS regression of task completion on evolutionary distance better describes any of the 18 primary cells than the linear fit alone; no scenario identifier is available in the primary files, so this check cannot be clustered by scenario (Section 3.6). Across the 18 cells, the quadratic term is nominally significant in three: AgentCollabBench with Haiku 4.5 ($p = .001$), Silo-Bench with Haiku 4.5 ($p < .001$), and EntCollabBench with GPT 5.4 Nano ($p < .001$); all three survive Holm correction over the 18 tests. A joint test of the linear and quadratic terms is nominally significant in seven cells but survives Holm correction in three: AgentCollabBench with Sonnet 4.6, Silo-Bench with Haiku 4.5, and EntCollabBench with GPT 5.4 Nano. AgentCollabBench with Haiku 4.5 is a near-miss on the joint test (Holm $p = .063$) despite its quadratic term surviving alone. The remaining 15 cells show no evidence of curvature (quadratic $p \geq .05$ in 15 of 18, joint $p \geq .05$ in 11 of 18). This is consistent with, rather than an alternative to, the sign instability already reported: where curvature is detectable it coexists with a benchmark- and model-specific relationship rather than replacing the linear reading, and it does not recover evolutionary distance as a directionally interpretable measure.

A.4 Task Completion in the Primary Benchmarks

Task completion is not constructed identically across the four primary benchmarks, as set out in Section 3.5, so the mean-completion column of Table 3 describes each evaluation context and does

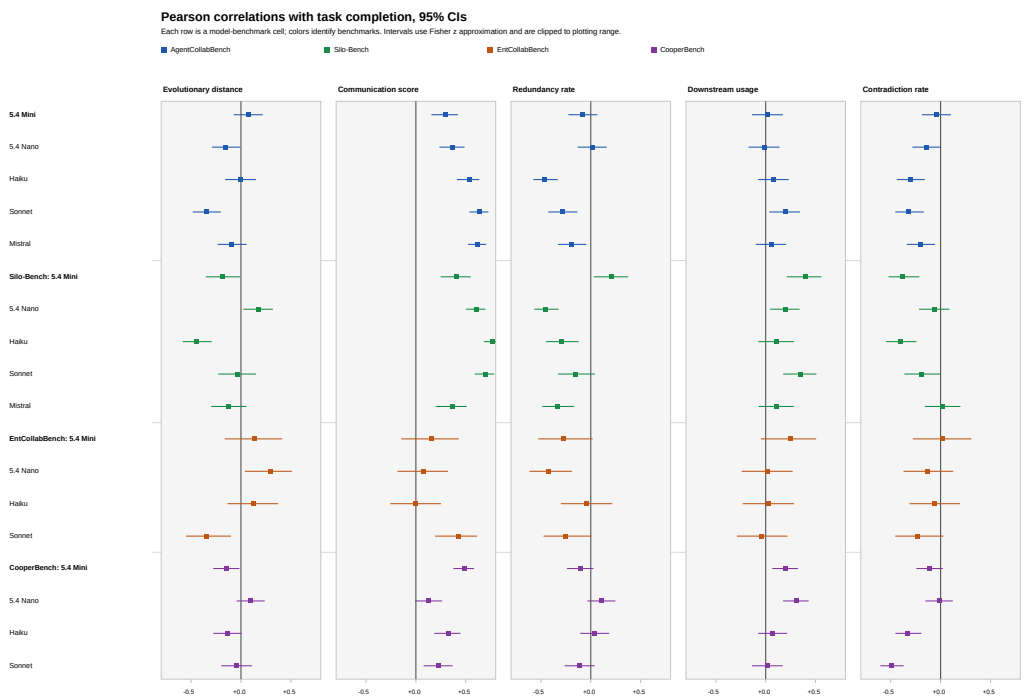


Figure 1: Forest plot of the 90 primary task-success associations. Each facet corresponds to one retained collaboration metric; points are within-cell Pearson correlations and horizontal intervals are approximate 95% confidence intervals from Fisher's z transform. Rows are the 18 benchmark–model configurations in the order of Table 3; colours identify benchmarks.

not rank benchmark difficulty. A mean of 0.41 in EntCollabBench and a mean of 0.41 in Silo-Bench are not comparable quantities.

The completion levels bear on the interpretation of individual cells. The two Silo-Bench configurations with near-floor completion, GPT 5.4 Mini (0.071) and Mistral Large (0.052), are where the anomalous positive redundancy coefficient and the largest downstream-usage coefficient both occur; at this level most trajectories fail, and metric variation is dominated by the difference between failing quickly and failing after extended effort. Conversely, CooperBench with Sonnet 4.6 has the highest completion in the study (0.758), compressing outcome variance and bounding the correlations attainable in that row.

A.5 Model Grid Coverage

The five language models evaluated are GPT 5.4 Nano, GPT 5.4 Mini, Claude Haiku 4.5, Claude Sonnet 4.6, and Mistral Large 1.0. The incomplete cell, described in Section 3.5, is Mistral Large 1.0, which could not be run to completion on the EntCollabBench or CooperBench harnesses, so those two benchmarks contribute four configurations each and the study totals 18 rather than 20. Where Mistral Large does appear, in AgentCollabBench and Silo-Bench, it behaves unremarkably relative to the other models, so we have no indication that its absence removes an outlying case. Its absence does mean that the two benchmarks contributing the fewest significant coefficients, EntCollabBench in particular, are also evaluated over a smaller model grid. Mistral Large 1.0 was included as the open-weights model available to us at the time of data collection, rather than selected to match the release cadence of the four closed-weight models; it is not intended as a representative frontier open-weights baseline.

Table 5: Spearman rank correlation between each collaboration metric and task completion, with two-sided p-values in parentheses. The analysed samples and -1 sentinel exclusions match Table 4.

Model	Evol. distance	Comm. score	Redun. rate	Downstr. usage	Contra. rate
AgentCollabBench					
GPT 5.4 Mini	$+0.07$ (.354)	$+0.30$ (<.001)	-0.10 (.202)	$+0.01$ (.862)	-0.02 (.764)
GPT 5.4 Nano	-0.14 (.070)	$+0.42$ (<.001)	$+0.01$ (.920)	-0.02 (.822)	-0.14 (.065)
Haiku 4.5	$+0.03$ (.735)	$+0.49$ (<.001)	-0.45 (<.001)	$+0.04$ (.613)	-0.30 (<.001)
Sonnet 4.6	-0.37 (<.001)	$+0.64$ (<.001)	-0.26 (.001)	$+0.23$ (.005)	-0.16 (.057)
Mistral Large	-0.12 (.114)	$+0.64$ (<.001)	-0.21 (.005)	$+0.08$ (.297)	-0.30 (<.001)
Silo-Bench					
GPT 5.4 Mini	-0.12 (.209)	$+0.20$ (.025)	$+0.04$ (.690)	$+0.07$ (.489)	-0.41 (<.001)
GPT 5.4 Nano	$+0.25$ (.001)	$+0.42$ (<.001)	-0.47 (<.001)	$+0.18$ (.018)	$+0.07$ (.367)
Haiku 4.5	-0.42 (<.001)	$+0.77$ (<.001)	-0.33 (<.001)	$+0.10$ (.280)	-0.48 (<.001)
Sonnet 4.6	-0.04 (.718)	$+0.73$ (<.001)	-0.12 (.220)	$+0.40$ (<.001)	-0.33 (<.001)
Mistral Large	-0.05 (.594)	$+0.35$ (<.001)	-0.31 (<.001)	-0.00 (.993)	$+0.13$ (.169)
EntCollabBench					
GPT 5.4 Mini	$+0.13$ (.383)	$+0.21$ (.176)	-0.28 (.064)	$+0.27$ (.074)	$+0.12$ (.427)
GPT 5.4 Nano	$+0.30$ (.020)	$+0.09$ (.512)	-0.39 (.003)	$+0.09$ (.514)	-0.15 (.267)
Haiku 4.5	$+0.15$ (.274)	$+0.06$ (.644)	-0.02 (.886)	-0.04 (.745)	-0.10 (.447)
Sonnet 4.6	-0.37 (.004)	$+0.41$ (.001)	-0.28 (.029)	-0.08 (.567)	-0.21 (.101)
CooperBench					
GPT 5.4 Mini	-0.13 (.068)	$+0.52$ (<.001)	-0.19 (.005)	$+0.17$ (.012)	-0.08 (.244)
GPT 5.4 Nano	$+0.13$ (.074)	$+0.13$ (.065)	$+0.08$ (.296)	$+0.31$ (<.001)	$+0.00$ (1.000)
Haiku 4.5	-0.11 (.129)	$+0.27$ (<.001)	$+0.10$ (.161)	$+0.10$ (.191)	-0.30 (<.001)
Sonnet 4.6	-0.04 (.625)	$+0.16$ (.044)	-0.16 (.038)	$+0.13$ (.097)	-0.38 (<.001)

A.6 Judge and Embedding Configuration

The judge used for communication score, redundancy rate, downstream usage, and contradiction rate is GPT 5.5, held fixed across every run and pinned independently of the model being evaluated, as described in Section 3.4. Evolutionary distance uses Titan Text Embeddings v2. The judged measures should be interpreted as outputs of this single instrument. We did not collect human labels. We did run a second judge over the full set of trajectories for communication score, reported in Appendix A.6.3; for the three event-level rates this appendix supports reproducibility of the measurement pipeline but not inter-rater reliability.

Benchmark variation in the event-level rates may reflect genuine differences in collaboration, judge noise, or domain-specific label interpretation; the design in Section 3.4 cannot separate these.

A.6.1 Judge Prompts

The judge is given the task description, agent profiles, and the conversation transcript (and, where indicated, the specific message under evaluation) and returns a numeric score together with a free-text justification, which is retained for audit but not scored. The four prompt templates below are reproduced verbatim, with {`task`}, {`agent_profiles`}, {`conversation`}, and {`message`} denoting the fields substituted at call time.

Communication score.

```
Task: {task}
Agent Profiles: {agent_profiles}

<conversation>
{conversation}
<conversation>

[System] You are tasked with evaluating the quality of communication among agents operating
     within a multiagent system. Evaluate whether agents made effective decisions based on the
```

```
     provided task results and whether their communication aligns with their agent profiles
     and social relationships. Consider the following:
1. Effective Decision-Making: Did agents use task results to guide their decisions effectively
     ?
2. Clarity and Precision: Were communications clear and unambiguous?
3. Adherence to Social Relationships: Did communications reflect the expected interactions
     based on the agents' social relationships?
4. Alignment with Agent Profiles: Were the messages consistent with the defined agent profiles
     ?
5. Overall Effectiveness: Did the communication facilitate task progress, considering both
     cooperative and competitive aspects?
Scoring Criteria (Communication):
- 5 (Exceptional): Outstanding communication with clear, precise messages fully aligned with
     agent profiles and social relationships.
Example: Every agent provided concise, accurate, and strategic information that directly
     advanced the task.
- 4 (Very Good): Mostly effective communication with only minor lapses and slight ambiguities.
Example: Occasional minor unclear messages, but overall effective.
- 3 (Adequate): Acceptable communication with moderate ambiguities or inconsistencies.
Example: Some messages were vague and did not fully meet required standards.
- 2 (Poor): Frequent unclear or misaligned communications causing significant miscommunication
     .
Example: Repeated incoherence negatively impacted task progress.
- 1 (Very Poor): Largely ineffective communication with confusing messages and complete
     misalignment.
Example: Chaotic communication with severely flawed decisions.
Return your answer as an integer between 1-5 as well as a string containing your reasoning
```

Redundancy rate.

```
Task: {task}

<conversation>
{conversation}
</conversation>

<current_message>
{message}
</current_message>

[System] You are tasked with evaluating the percentage of redundant information contained in
     an agents message
You will be given the conversation thus far as well as the current agent message. You should
     use this to estimate the percentage of the message that contains redundant information.

Redundant information is restated assertions that are already present in past messages, or
     repeated planning/execution of a task. Confirming what task should be performed is not
     considered redundant information as it is simply a confirmation

Return your answer as a float between 0 and 1 representing the % of redundant information in
     the message as well as a string containing your reasoning
```

Downstream usage rate.

```
Task: {task}

<conversation>
{conversation}
</conversation>

<current_message>
{message}
</current_message>

[System] You are tasked with determining how much of an agent's message was used downstream in
     a conversation. You will be given an agent's message as well as the task the agentic
     system was tasked with completing. You will also be given a list of all of the messages
     sent after that point.
You should evaluate what percentage of the information stored in the agent's message was used
     by the agents in the rest of the conversation. For example If 20% of the agents
     information is references later in the conversation the score would be 0.2

Return your answer as a float between 0 and 1 representing the % of information in the message
     used downstream as well as a string containing your reasoning
```

Contradiction rate.

```
Task: {task}

<conversation>
{conversation}
</conversation>

[System] You are tasked with determining the number of contradictions present in an agent
     conversation. You will be given a list of messages and must determine the total number of
      assertions made as well as the number of those assertions that are contradictions (so
     num_contradictions <= total_assertions). A contradiction occurs if a later message in a
     conversation contradicts an assertion made previously.
If an agent contradicts itself in its own message this is considered part of the reasoning
     process and not a contradiction.

For the purposes of this evaluation an assertion is a single statement containing some kind of
      factual information or a single confirmation of a proposed course of action

Return your answer as 2 integers, num_contradictions containing the number of contradictions
     and total_assertions containing the total number of assertions made in the conversation
```

A second communication-score variant, used when agent profiles are supplied without an explicit task string, is otherwise identical apart from omitting the `Task` field and the corresponding first evaluation criterion:

```
Agent Profiles: {agent_profiles}

<conversation>
{conversation}
<conversation>

[System] You are tasked with evaluating the quality of communication among agents operating
     within a multiagent system. Evaluate whether agents made effective decisions and whether
     their communication aligns with their agent profiles and social relationships.

Consider the following:
1. Effective Decision-Making: Did agents guide their decisions effectively?
2. Clarity and Precision: Were communications clear and unambiguous?
3. Adherence to Social Relationships: Did communications reflect the expected interactions
     based on the agents' social relationships?
4. Alignment with Agent Profiles: Were the messages consistent with the defined agent profiles
     ?
5. Overall Effectiveness: Considering both cooperative and competitive aspects, was the inter-
     agent communication effective.
Scoring Criteria (Communication):
- 5 (Exceptional): Outstanding communication with clear, precise messages fully aligned with
     agent profiles and social relationships.
Example: Every agent provided concise, accurate, and strategic information
- 4 (Very Good): Mostly effective communication with only minor lapses and slight ambiguities.
Example: Occasional minor unclear messages, but overall effective.
- 3 (Adequate): Acceptable communication with moderate ambiguities or inconsistencies.
Example: Some messages were vague and did not fully meet required standards.
- 2 (Poor): Frequent unclear or misaligned communications causing significant miscommunication
     .
Example: Repeated incoherence
- 1 (Very Poor): Largely ineffective communication with confusing messages and complete
     misalignment.
Example: Chaotic communication with severely flawed decisions.
Return your answer as an integer between 1-5 as well as a string containing your reasoning
```

The communication-score rubric's fifth criterion and its scoring examples reference whether communication "facilitate[s] task progress" or "advanced the task," which is the basis for the rubric-overlap caveat in Section 3.4.

A.6.2 TASK-PROGRESS-BLIND RE-SCORING

Because the rubric's overlap with the outcome is a caveat rather than a settled limitation, we re-scored every trajectory with a version of the same prompt in which every reference to task progress was removed, keeping the scoring scale, the profile and transcript inputs, and the rest of the criteria unchanged. This isolates how much of communication score's association with task completion comes from the rubric's own wording rather than from the communication being scored.

Table 6: Communication score's correlation with task completion under the original rubric and under the task-progress-blind rescoring, by benchmark and model. A checkmark indicates the coefficient survives within-column Holm correction (18 tests per column, matching Section 3.6).

Model	n	Original r	Holm	Blind r	Holm
AgentCollabBench					
GPT 5.4 Mini	180	$+0.29$	✓	-0.08	
GPT 5.4 Nano	180	$+0.37$	✓	$+0.07$	
Haiku 4.5	161	$+0.53$	✓	$+0.12$	
Sonnet 4.6	151	$+0.64$	✓	$+0.29$	✓
Mistral Large	180	$+0.62$	✓	$+0.31$	✓
Silo-Bench					
GPT 5.4 Mini	119	$+0.41$	✓	$+0.28$	✓
GPT 5.4 Nano	164	$+0.61$	✓	$+0.51$	✓
Haiku 4.5	118	$+0.77$	✓	$+0.76$	✓
Sonnet 4.6	108	$+0.70$	✓	$+0.63$	✓
Mistral Large	120	$+0.36$	✓	$+0.30$	✓
EntCollabBench					
GPT 5.4 Mini	45	$+0.15$		$+0.18$	
GPT 5.4 Nano	60	$+0.07$		$+0.10$	
Haiku 4.5	60	-0.00		$+0.13$	
Sonnet 4.6	60	$+0.43$	✓	$+0.54$	✓
CooperBench					
GPT 5.4 Mini	213	$+0.49$	✓	$+0.36$	✓
GPT 5.4 Nano	203	$+0.13$		$+0.16$	
Haiku 4.5	181	$+0.32$	✓	$+0.27$	✓
Sonnet 4.6	163	$+0.23$	✓	$+0.29$	✓

Table 6 gives the resulting correlation with task completion for both versions, in every configuration. The sign and rough size of the association held in 16 of 18 configurations; the two exceptions, AgentCollabBench with GPT 5.4 Mini and EntCollabBench with Haiku 4.5, both flip sign but only between coefficients already close to zero under the original rubric ($+0.29 \rightarrow -0.08$ and $-0.00 \rightarrow +0.13$). A paired comparison of the 18 coefficients (Fisher z transform, two-sided t-test) finds a small but detectable overall shift towards weaker association under the blind rubric ($p = .014$), and this shift is concentrated in AgentCollabBench, whose mean coefficient drops from $+0.49$ to $+0.14$; Silo-Bench, CooperBench, and EntCollabBench change little (mean shifts of -0.07, -0.02, and $+0.07$ respectively). The count of configurations surviving within-column Holm correction falls from 14 to 11, all three losses again in AgentCollabBench.

We read this as evidence that the rubric's task-progress wording is not the main source of communication score's association with task completion: removing it leaves the result largely intact in three of the four benchmarks. It does, however, account for a meaningful share of the association specifically in AgentCollabBench, so the overlap caveat in Section 3.4 should be read as a partial explanation for that one benchmark rather than for the result as a whole. This also gives us more confidence in communication score as a metric: a rubric that was simply relabelling task progress as communication quality would collapse everywhere once that wording was removed, not hold up in three benchmarks out of four.

A.6.3 Second-Judge Re-Scoring

The progress-blind check in Appendix A.6.2 varies the rubric while holding the judge fixed. This subsection varies the judge while holding the rubric fixed. We re-scored every trajectory's communication score with Claude Opus 5 under the original rubric, unchanged, with the same task, profile, and transcript inputs. Because the same trajectories carry both scores, the two judges can be compared on identical data rather than on separately derived samples. All coefficients in this subsection are computed on the intersection where both judges and the task-completion score are present, so the GPT 5.5 column here is recomputed on that intersection and differs from Table 3 in two cells where the second judge returned fewer usable records: AgentCollabBench with GPT 5.4

Mini ($n = 179$ against 180) and CooperBench with GPT 5.4 Nano ($n = 201$ against 203). Neither changes a rounded coefficient.

Table 7 reports both judges' correlations with task completion, their quadratic weighted agreement, and their difference. The headline result is judge-invariant. Communication score is nominally significant in 15 of 18 configurations under Opus against 14 under GPT 5.5, survives within-column Holm correction in 13 against 14, and carries the expected positive sign in 18 of 18 against 17 of 18. Holm-surviving coefficients span 0.21–0.78 under Opus and 0.23–0.77 under GPT 5.5. A paired comparison of the 18 coefficients (Fisher z transform, two-sided t-test) finds no detectable overall shift (mean $\Delta z = +0.020$, $p = .59$), against the small but detectable shift the same test found for the progress-blind rubric ($p = .014$). Per-cell, a Williams test for dependent correlations sharing task completion finds a nominal difference in 5 of 18 configurations, of which two survive Holm correction over the 18 comparisons: AgentCollabBench with Mistral Large ($+0.62 \rightarrow +0.42$, Holm $p = .001$) and Silo-Bench with GPT 5.4 Nano ($+0.61 \rightarrow +0.78$, Holm $p = .004$). These two run in opposite directions, so they do not indicate that one judge reads the rubric more conservatively than the other.

Two cells change significance status, in opposite directions and both near threshold. AgentCollabBench with GPT 5.4 Mini falls from $+0.29$ to $+0.15$ and loses nominal significance ($p = .052$), which is the sole source of the net change from 14 Holm survivors to 13. CooperBench with GPT 5.4 Nano rises from $+0.13$ to $+0.16$ and gains nominal significance ($p = .026$) without surviving correction. Under Spearman, these two cells are the only ones where rank and linear analyses disagree on nominal significance, and they disagree in opposite directions: AgentCollabBench with GPT 5.4 Mini is significant on ranks ($\rho = +0.17$, $p = .022$) where it is not on levels, and EntCollabBench with GPT 5.4 Mini is significant on levels ($+0.31$, $p = .035$) but not on ranks ($\rho = +0.27$, $p = .075$). The set of configurations where communication score is a reliable correlate of task success is therefore stable to judge identity at its core and unstable only at its margin.

Agreement between the two judges is substantial in aggregate and uneven across configurations. Pooled over all 2,461 paired trajectories, quadratic weighted $\kappa = 0.75$, with exact agreement on 51.7% of trajectories and agreement within one scale point on 95.9%. Per-configuration κ ranges from 0.18 (CooperBench with GPT 5.4 Nano) to 0.81 (Silo-Bench with Sonnet 4.6), and the benchmark means track how strongly the two judges' coefficients agree: Silo-Bench has both the highest mean κ (0.65) and the smallest mean coefficient shift, while CooperBench has the lowest mean κ (0.29). Agreement is weakest in the configurations where the coefficient is smallest, so what the second judge establishes most firmly is the upper part of the range, not the whole of it.

The two judges also differ systematically in calibration rather than only in noise. Opus scores lower on average (mean 2.44 against 2.90 on the 1–5 scale), assigning a strictly lower score in 45.3% of trajectories and a strictly higher one in 3.0%. Opus uses the top band almost not at all, awarding 5 to 6 trajectories against 110 for GPT 5.5. Pearson and Spearman correlations and weighted κ are all invariant to this kind of location shift, so it does not affect the coefficients, but it does mean the two instruments are not interchangeable for reporting absolute communication-score levels, only for the associations we report.

We read this as convergent validity between two instruments, not as validation against ground truth. Both judges are large language models and may share correlated failure modes that no comparison between them can detect. The re-scoring also covers communication score only; redundancy rate, downstream usage, and contradiction rate remain outputs of the single GPT 5.5 instrument, so the reliability caveat in Section 3.4 is narrowed to those three measures rather than removed. A human-annotated subsample remains the missing check.

A.7 Analysis Provenance

The correlations in Table 3 and the p-values in Table 4 were computed from 18 per-trajectory result files, one per model–benchmark configuration. Each file maps a trajectory identifier to a record holding the task-completion score and the five collaboration metrics. The derived correlation and p-value matrices are additionally stored as one worksheet per configuration in a companion spreadsheet, which is the artefact from which the reported tables were transcribed. All values reported here were recomputed from the per-trajectory files under the exclusion rules in Appendix A.1 and reconciled against that spreadsheet.

Table 7: Communication score's correlation with task completion under the GPT 5.5 judge and under second-judge re-scoring with Claude Opus 5, with the original rubric held fixed. Both columns are computed on the paired intersection where both judges scored the trajectory, so n differs from Table 3 in two cells. A checkmark indicates the coefficient survives within-column Holm correction (18 tests per column, matching Section 3.6). κ_w is the quadratic weighted agreement between the two judges' scores. Δ is the Opus coefficient minus the GPT 5.5 coefficient; a dagger marks the two cells whose difference survives Holm correction over 18 Williams tests for dependent correlations.

Model	n	**GPT 5.5** r	**Holm**	**Opus** r	**Holm**	κ_w	Δ
AgentCollabBench							
GPT 5.4 Mini	179	$+0.29$	✓	$+0.15$		0.28	-0.14
GPT 5.4 Nano	179	$+0.37$	✓	$+0.26$	✓	0.24	-0.11
Haiku 4.5	161	$+0.53$	✓	$+0.58$	✓	0.47	$+0.05$
Sonnet 4.6	151	$+0.64$	✓	$+0.63$	✓	0.65	-0.02
Mistral Large	179	$+0.62$	✓	$+0.42$	✓	0.51	$-0.20^{\dagger}$
Silo-Bench							
GPT 5.4 Mini	119	$+0.41$	✓	$+0.45$	✓	0.74	$+0.04$
GPT 5.4 Nano	164	$+0.61$	✓	$+0.78$	✓	0.48	$+0.17^{\dagger}$
Haiku 4.5	118	$+0.77$	✓	$+0.66$	✓	0.78	-0.11
Sonnet 4.6	108	$+0.70$	✓	$+0.72$	✓	0.81	$+0.02$
Mistral Large	120	$+0.36$	✓	$+0.54$	✓	0.44	$+0.18$
EntCollabBench							
GPT 5.4 Mini	45	$+0.15$		$+0.31$		0.37	$+0.16$
GPT 5.4 Nano	60	$+0.07$		$+0.16$		0.48	$+0.09$
Haiku 4.5	60	-0.00		$+0.12$		0.41	$+0.12$
Sonnet 4.6	60	$+0.43$	✓	$+0.54$	✓	0.53	$+0.11$
CooperBench							
GPT 5.4 Mini	213	$+0.49$	✓	$+0.41$	✓	0.35	-0.07
GPT 5.4 Nano	201	$+0.13$		$+0.16$		0.18	$+0.03$
Haiku 4.5	181	$+0.32$	✓	$+0.32$	✓	0.40	-0.00
Sonnet 4.6	163	$+0.23$	✓	$+0.21$	✓	0.23	-0.01

The two re-scoring checks are stored as parallel file sets, each record holding the original and re-scored communication score alongside the task-completion score for one trajectory, so that both versions are keyed to the same trajectory identifier and can be compared pairwise. The original score in these files was verified to be identical, record by record, to the communication score in the corresponding primary result file. The exclusion rules in Appendix A.1 apply unchanged, with the additional requirement for these two checks that both scores be present for a trajectory to enter the paired comparison.

B Privacy-Constrained Collaboration: Full Secondary Analysis

This appendix expands the privacy-constrained secondary analysis summarised in Section 4.2. It treats PAC-BENCH, AgentSocialBench, and AgentLeak (Park et al., 2026; Wang & Jiang, 2026; El Yagoubi et al., 2026) as a secondary analysis of collaboration in privacy-constrained settings.

B.1 Sample and Scope

The benchmark-level dataset comprised 1,500 completed trajectories: 100 scenarios evaluated with each of five language models in each benchmark. This aggregate was not exclusively multi-agent under a strict artificial-agent-count definition: 135 AgentSocialBench mediated-communication trajectories contained one artificial agent. We retain all 500 AgentSocialBench rows for benchmark-level description and report a strict-MAS sensitivity using 365 trajectories from 73 scenarios; the strict-MAS total across the three benchmarks is therefore 1,365 trajectories. The same 100 scenarios were paired across models within a benchmark, but scenarios were not paired between benchmarks. Task completion and privacy leakage were analysed as separate outcomes. We did not use the

Table 8: Task completion and information exposure in the three privacy-constrained benchmarks. "Mean compl." is the mean requirement-completion score. The 95% CI applies to any-leak incidence and uses 50,000 scenario-cluster bootstrap resamples, preserving all five model runs for each sampled scenario. "Item-leak rate" is a macro mean of per-trace ratios, not a pooled-item micro-rate. "Internal / final-out." gives the number of trajectories leaking through each channel; the two can overlap. The audit-miss denominator is the number of trajectories with any observed leak.

Benchmark	N	Mean compl.	Any-leak traj. (%)	95% CI (%)	Item-leak rate	Internal / final-out.	Missed by audit
AgentLeak	500	0.804	43 (8.6)	6.6–10.8	0.20%	43 / 1	42/43 (97.7%)
PAC-BENCH	500	0.694	133 (26.6)	20.6–32.8	11.57%	133 / 27	106/133 (79.7%)
AgentSocialBench	500	0.226	83 (16.6)	11.8–21.8	2.89%	66 / 25	58/83 (69.9%)

formula-derived privacy-safe task score as a primary endpoint because it combines the two outcomes algebraically.

Task completion was not constructed identically across the three benchmarks. AgentLeak awarded partial credit over two task requirements, PAC-BENCH awarded partial credit over five requirements, and AgentSocialBench used one strict compound success criterion. The absolute task-completion levels therefore describe each evaluation context but do not establish that one benchmark was intrinsically more difficult than another.

The privacy artefacts expose only a subset of the paper's full collaboration-quality framework at a common trajectory-level granularity: communication score, redundancy, downstream use as a measure of semantic contribution, and unresolved contradiction as a measure of coordination consistency. They do not provide harmonized measures of decomposition and role quality, handoff quality, verification quality, or failure containment. Radioactive tracer durability (RTD) and instruction decay rate (IDR) are therefore reported as supplementary robustness measures rather than components of the common collaboration construct; consensus pollution and cross-task leakage were not applicable in these runs and were not encoded as zero.

B.2 Task Completion and Privacy Exposure

Table 8 reports the task and privacy outcomes separately. Mean requirement-completion score was 0.804 in AgentLeak, 0.694 in PAC-BENCH, and 0.226 in AgentSocialBench. Exact full completion occurred in 342, 173, and 113 of the 500 trajectories, respectively; in AgentSocialBench, the strict binary criterion makes full completion identical to the reported task-completion mean. Information exposure was observed in all three settings, but its incidence and severity differed substantially. At least one privacy leak occurred in 43 of 500 AgentLeak trajectories (8.6%), 133 of 500 PAC-BENCH trajectories (26.6%), and 83 of 500 AgentSocialBench trajectories (16.6%). Macro mean trace-level item leakage was 0.20%, 11.57%, and 2.89%, respectively. Incident frequency and item-level severity are therefore reported separately throughout.

Performance and privacy were not interchangeable at either the benchmark or model level. Table 9 reports both outcomes for each of the 15 model–benchmark cells, each based on 100 scenarios. Mean requirement-completion scores ranged from 0.755 to 0.845 in AgentLeak, from 0.464 to 0.842 in PAC-BENCH, and from 0.140 to 0.390 in AgentSocialBench. Any-leak incidence ranged from 0% to 32%, 14% to 39%, and 8% to 21%, respectively. The model with the highest task completion was not consistently the model with the lowest information exposure. The clearest case is AgentLeak, where Haiku 4.5 attained the third-highest completion score in the benchmark (0.810, within 0.035 of the best model) while leaking in 32 of 100 trajectories, against 0–1 leaks for the two GPT models and Sonnet 4.6; a completion-only evaluation would have placed it mid-field rather than isolating it as the single largest privacy failure in the benchmark. The reverse dissociation appears in PAC-BENCH, where Haiku 4.5 has the lowest completion (0.464) and an intermediate leak rate (26%), while the highest-completing model in that benchmark, GPT 5.4 Mini (0.842), leaks in 22% of trajectories. Rank agreement between the two outcomes is therefore weak and inconsistent in direction across benchmarks. At the trajectory level, the unadjusted correlations between task completion and any-leak incidence were small in AgentLeak ($r = .021$), PAC-BENCH ($r = .044$), and AgentSocialBench ($r = .080$). These results support treating successful task completion and

Table 9: Task completion and any-leak incidence for each model in each privacy-constrained benchmark ($n = 100$ scenarios per cell). Models are ordered by mean requirement-completion score within each benchmark. The completion and exposure orderings do not agree, and the direction of their disagreement is not consistent across benchmarks.

Benchmark	Model	Mean requirement-completion score	Any-leak trajectories
AgentLeak	Sonnet 4.6	0.845	1%
	GPT 5.4 Mini	0.820	0%
	Haiku 4.5	0.810	32%
	GPT 5.4 Nano	0.790	0%
	Mistral Large	0.755	10%
PAC-BENCH	GPT 5.4 Mini	0.842	22%
	Mistral Large	0.820	39%
	GPT 5.4 Nano	0.794	32%
	Sonnet 4.6	0.552	14%
	Haiku 4.5	0.464	26%
AgentSocialBench	Sonnet 4.6	0.390	20%
	Haiku 4.5	0.260	21%
	GPT 5.4 Nano	0.180	20%
	GPT 5.4 Mini	0.160	8%
	Mistral Large	0.140	14%

Table 10: Pooled collaboration-profile means. RTD denotes radioactive tracer durability and IDR denotes instruction decay rate; both are supplementary robustness measures in this paper. AgentLeak lacked explicit radioactive tracers, so its evaluator used marked privacy-constraint clauses as proxy tracers. The AgentSocialBench row includes its 135 single-agent mediated-communication trajectories. Three AgentSocialBench trajectories lacked sufficient eligible events for redundancy and downstream-use scoring. RTD used benchmark-specific tracer semantics and is not a direct cross-benchmark ranking.

Benchmark	RTD ↑	IDR ↓	Comm. ↑	Redund. ↓	Downstr. use ↑	Contra. ↓
AgentLeak	0.875	0.227	0.804	0.063	0.997	0.017
PAC-BENCH	0.585	0.228	0.693	0.063	0.952	0.081
AgentSocialBench	0.962	0.115	0.634	0.166	0.917	0.035

privacy preservation as distinct empirical outcomes. Per-cell incidences rest on 100 trajectories each and are correspondingly imprecise; they are reported to show that the two outcomes separate, not to rank models.

Leakage levels should not be interpreted as a privacy ranking of the three benchmarks. Their scenario structures, sensitive-item denominators, and detection pipelines differ. In particular, PAC-BENCH used a conservative deterministic scan that left contextual candidates in 78 trajectories uncounted, whereas AgentSocialBench used GPT 5.5 semantic adjudication over event–item pairs. The experiment also lacks matched privacy-unconstrained runs, so it does not identify the causal effect of imposing a privacy constraint; it estimates associations among trajectories generated under privacy constraints.

B.3 Collaboration Profiles Differed across Privacy Settings

The three benchmarks produced different collaboration profiles (Table 10). AgentLeak had the highest mean communication score (0.804) and downstream usage (0.997), whereas PAC-BENCH had the highest unresolved-contradiction rate (0.081). AgentSocialBench had the highest redundancy rate (0.166) and the lowest mean instruction-decay rate (0.115). These pooled differences are descriptive and should not be interpreted as causal explanations of the benchmark-level task scores.

Table 11: Adjusted absolute percentage-point change in task completion per one-standard-deviation improvement in a process or robustness measure; 95% confidence intervals are in brackets. The models were fitted one measure at a time. Confidence intervals are not multiplicity-adjusted.

Process or robustness measure	AgentLeak	PAC-BENCH	AgentSocialBench
Communication score	$+3.6$ [0.4, 6.9]	$+5.0$ [1.4, 8.7]	$+1.0$ [-4.1, 6.1]
Non-redundancy	$+2.2$ [-0.7, 5.1]	$+2.9$ [0.3, 5.5]	$+1.8$ [-2.6, 6.3]
Downstream use	$+2.1$ [-0.03, 4.3]	$+2.3$ [-0.9, 5.6]	-0.6 [-5.0, 3.9]
Contradiction-free coordination	$+3.1$ [-0.7, 6.9]	-0.6 [-3.7, 2.5]	$+1.2$ [-2.8, 5.1]
Supplementary: tracer durability	$+3.3$ [-1.4, 8.0]	$+3.0$ [-0.7, 6.7]	$+5.6$ [0.9, 10.4]
Supplementary: instruction adherence (lower IDR)	$+2.5$ [-4.2, 9.2]	-3.5 [-6.7, -0.3]	-5.5 [-9.7, -1.3]

B.4 Estimator and Association between Process Measures and Task Completion

We estimated the association between each available common or supplementary process measure and task completion separately within each benchmark using fixed-effects OLS for fractional task completion and fixed-effects linear-probability models for binary completion and leakage. Predictors were standardised within each complete-case analysis sample and direction-aligned so that a positive coefficient denotes a one-standard-deviation improvement in the process measure. Each model included language-model and scenario fixed effects and scenario-clustered standard errors. PAC-BENCH additionally included message count; AgentSocialBench included message count and expected interaction length, the structural indicators that varied within their paired scenarios. AgentLeak had no varying structural control. Team size, category, and other scenario-invariant properties were absorbed by scenario fixed effects rather than estimated separately. Message count is realised trajectory length and may itself be affected by the collaboration process, so coefficients conditional on it should not be interpreted as total collaboration effects. The 12 AgentLeak scenarios flagged for task-relevant source-data contradictions were excluded using an outcome-independent source-quality rule applied identically to all five models, with the same 12 scenarios flagged in each. This left 440 AgentLeak trajectories from 88 aligned scenarios; the descriptive results in Table 8 retain all 500 trajectories. The coefficients in Table 11 are absolute percentage-point changes in task completion. Holm correction was applied across the family of 18 reported benchmark-by-metric comparisons.

AgentLeak did not contribute varying structural controls: every trajectory had two agents and four messages, while expected-turn counts were unavailable. Its adjusted estimates therefore condition on model and scenario but cannot independently estimate effects beyond team size or message count.

An all-500 AgentLeak sensitivity gave the same multiplicity-adjusted conclusion: communication was $+3.1$ percentage points [0.1, 6.1] and downstream use was $+1.9$ points [0.0, 3.8], but no individual task association survived Holm correction. The AgentLeak leakage analysis in Table 12 retains all 500 trajectories because the source-data exclusions apply to task-quality analysis rather than privacy detection.

Several coefficients excluded zero before multiplicity correction. Higher communication scores were associated with 3.6 percentage points greater task completion in AgentLeak and 5.0 points greater completion in PAC-BENCH. Downstream use in AgentLeak and non-redundancy in PAC-BENCH also had small positive point estimates. AgentSocialBench instead showed a positive estimate for tracer durability, while its general communication, redundancy, downstream-use, and contradiction measures had intervals spanning zero.

None of the task-completion associations remained statistically distinguishable from zero after Holm correction (smallest adjusted $p = .131$). The direction of the estimates also varied across benchmarks. In particular, lower instruction decay was associated with lower, rather than higher, task completion in PAC-BENCH and AgentSocialBench before correction. Because the injected hard constraints in these settings were privacy constraints, this direction is consistent with a possible privacy–utility tension: using restricted information can make a task easier while increasing expo-

Table 12: Adjusted absolute percentage-point change in the probability of any privacy leak per one-standard-deviation improvement in a process or robustness measure; 95% confidence intervals are in brackets. Negative values indicate lower exposure. An asterisk denotes Holm-adjusted $p < .05$ across the family of 18 reported comparisons.

Process or robustness measure	AgentLeak	PAC-BENCH	AgentSocialBench
Communication score	-3.8 [-7.4, -0.2]	-2.4 [-6.2, 1.4]	-5.0 [-8.9, -1.1]
Non-redundancy	$+1.3$ [-1.6, 4.2]	$+2.1$ [-2.0, 6.1]	$+1.4$ [-3.0, 5.8]
Downstream use	$+0.5$ [-0.7, 1.7]	$+2.6$ [-1.9, 7.1]	$+0.8$ [-2.4, 4.1]
Contradiction-free coordination	$+1.1$ [-0.3, 2.6]	-1.2 [-5.5, 3.2]	$+1.0$ [-1.4, 3.4]
Supplementary: tracer durability	-2.3 [-6.4, 1.7]	$+2.7$ [-4.0, 9.3]	-3.6 [-8.8, 1.5]
Supplementary: instruction adherence (lower IDR)	-6.4 [-12.6, -0.2]	-21.0 [-26.7, -15.3]*	-10.7 [-17.2, -4.2]*

sure. The result is associational, however, and the corrected task-completion tests do not establish such a trade-off conclusively.

The four common process measures (communication, redundancy, downstream use, and contradiction) were also entered as a block to test incremental explanatory value beyond the fixed-effect and structural baseline. The block increased in-sample R^2 by .012 in AgentLeak, .026 in PAC-BENCH, and .002 in AgentSocialBench. Scenario-cluster-robust joint Wald tests were evaluated as a three-test family. PAC-BENCH met the adjusted .05 threshold ($p = .048$), whereas AgentLeak ($p = .057$) and AgentSocialBench ($p = .899$) did not. These are modest in-sample explanatory gains; grouped out-of-sample validation is required before they can be described as predictive value.

The nominal communication effects should also be interpreted cautiously. The communication-scoring rubric partly considers whether communication advances the task, creating possible overlap between predictor and outcome. In addition, some event-derived measures had little within-scenario variation in AgentLeak; downstream use, for example, varied across models in only three of the 88 retained analysis scenarios. Overall, the privacy-constrained benchmarks did not provide evidence for one benchmark-invariant collaboration measure of task success.

B.5 Association between Process Measures and Information Exposure

None of the four common collaboration measures survived Holm correction for leakage. The supplementary instruction-adherence measure showed the expected within-trace association with privacy exposure (Table 12). After the same adjustments, a one-standard-deviation reduction in instruction decay was associated with a 21.0 percentage-point reduction in the probability of any leak in PAC-BENCH and a 10.7-point reduction in AgentSocialBench. Both associations survived correction across the family of 18 reported comparisons (Holm-adjusted $p < .001$ and $p = .027$, respectively). The corresponding AgentLeak estimate was smaller and did not survive correction.

Communication score had negative point estimates for leakage in all three benchmarks, but none survived family-wise correction. Non-redundancy, downstream use, and contradiction-free coordination were not consistently associated with exposure. A severity analysis produced the same central instruction-decay pattern: a one-standard-deviation improvement in instruction adherence was associated with reductions of 11.2 percentage points in the trace-level item-leakage rate in PAC-BENCH (95% CI 7.6–14.7) and 1.8 points in AgentSocialBench (95% CI 0.4–3.1). Thus, within the privacy-constrained subset, adherence to injected constraints was a more consistent indicator of information exposure than the broader collaboration measures were of task completion. The severity analysis was exploratory and was not included in the 18-comparison Holm family.

The AgentLeak communication–leakage association was also non-monotonic. Leaks occurred in 0/9, 6/70, 37/226, and 0/195 trajectories at communication scores of .25, .50, .75, and 1.00, respectively. Its negative linear coefficient should therefore not be interpreted as a dose-response or as evidence that raising the communication score would itself prevent leakage.

The instruction-decay result is partly construct-overlapping and should not be interpreted as independent predictive evidence. IDR counts turns that violate an injected hard constraint, whereas the

Table 13: Power floor for privacy null results. Values are 80%-power minimum detectable effects (absolute percentage points per one-standard-deviation predictor improvement) using a two-sided familywise threshold matching the first Holm step. Ranges summarise the four common collaboration measures.

Outcome family	**Tests**	**Smallest common adj. p**	**Median MDE**	**MDE range**
Task completion	18	0.131	6.7	3.8–10.1
Any privacy leak	18	0.186	7.3	2.3–8.8
Hidden leak only	15	0.642	7.3	2.3–9.0

privacy outcome counts protected items disclosed under those constraints in the same trace; the collaboration evaluator also received known disclosure evidence when assigning constraint-violation labels. The two measures are distinct but not procedurally independent. Likewise, preserving a privacy constraint as an RTD tracer does not imply compliance: an agent can retain or repeat a constraint after already violating it.

B.6 Power Floor for the Privacy Null Results

Table 13 reports 80%-power minimum detectable effects (MDEs) for the null privacy analyses. The calculation uses each fitted model's scenario-clustered CR1 standard error and a two-sided familywise threshold equal to the first Holm step ($.05/18$ for the task-completion and any-leak families; $.05/15$ for the hidden-leak robustness family). MDEs are therefore absolute percentage-point changes per one-standard-deviation improvement in the process measure. The ranges summarise the four common collaboration measures rather than the supplementary privacy-specific instruction-adherence measure.

The privacy nulls should therefore be read as bounded non-detections rather than equivalence results. In the main leakage family, the design is well powered to detect several-point to near-ten-point changes in leak probability, depending on benchmark and measure, but it cannot rule out smaller associations. This is why we describe the result as a lack of cross-benchmark evidence that general collaboration quality improves or worsens privacy, not as proof that the true effects are zero.

B.7 Interpreting the Collaboration–Privacy Relationship

A process measure associated with greater task completion and lower leakage would indicate privacy-preserving collaboration; one associated with greater completion and greater leakage would indicate a privacy–utility trade-off. Neither pattern was established for any of the four common measures after correction (Sections B.4–B.5). This absence of a uniform relationship is compatible with two opposing mechanisms: better coordination could reduce unnecessary disclosure and contain failures, or could instead propagate sensitive information more reliably once it enters a trajectory. Instruction adherence was associated with lower leakage in two benchmarks, but it remains a supplementary privacy-specific measure with partial construct overlap rather than evidence that general collaboration quality is inherently privacy-protective.

The any-leak outcome used here also combines leak initiation and subsequent propagation, so the present analysis cannot determine whether a collaboration process changed the probability of first disclosure, its downstream spread, or both. A mechanism-oriented extension would model these stages separately.

B.8 Evolutionary Distance

Evolutionary distance was analysed as an exploratory measure of semantic change rather than a directionally scored component of collaboration quality. Its linear association with task completion was small in AgentLeak (-0.9 percentage points per standard deviation), PAC-BENCH ($+1.0$ points), and AgentSocialBench (-3.0 points), with all confidence intervals spanning zero. AgentSocialBench evolutionary distance was available for 497 trajectories. The quadratic terms did not provide evidence of non-linearity for task completion (all $p \geq .085$) or leakage incidence (all $p \geq .310$). Joint tests of the linear and quadratic terms likewise did not reject the null for task

completion ($p = .224$, $.839$, and $.095$, respectively) or leakage incidence (all $p > .23$). These tests were exploratory and were not multiplicity-adjusted. The privacy-constrained results therefore do not support treating either greater or smaller semantic change as inherently better.

B.9 Scope and Robustness of the AgentSocialBench Sample

The benchmark-level AgentSocialBench results include 135 mediated-communication trajectories in which one artificial agent interacted with simulated human roles or recipients. These trajectories belong to the benchmark's broader social-agent setting but do not satisfy a strict definition requiring at least two artificial agents. Restricting AgentSocialBench to `num_agents` ≥ 2 left 365 trajectories from 73 aligned scenarios. In this strict-MAS subset, task completion was 29.9% and any-leak incidence was 18.6%. Communication, contradiction, RTD, and IDR used all 365 rows; redundancy, downstream use, and evolutionary distance used 364 because one retained trajectory lacked enough eligible events. Predictors were restandardized within the restricted sample, so its percentage-point effects are not directly comparable to the full-sample coefficients. This restriction was an exploratory sensitivity and its coefficients were not included in the full-sample Holm family. None of the general communication, redundancy, downstream-use, contradiction, or evolutionary-distance measures was associated with task completion after adjustment. The nominal full-sample association between communication score and leakage also weakened, whereas the association between greater instruction adherence and lower leakage remained (-13.4 percentage points per standard deviation; 95% CI -20.9 to -5.9). Greater instruction adherence was also associated with 8.4 points lower task completion in this subset (95% CI -13.7 to -3.1), remaining directionally consistent with the exploratory privacy–utility hypothesis without establishing causality. The instruction-adherence association was qualitatively robust after excluding the single-agent category, but the full 500-row AgentSocialBench aggregate should not be described as exclusively multi-agent under the strict artificial-agent-count definition.

B.10 Internal Leakage Was Largely Invisible in Final Answers

The per-benchmark breakdown behind the pooled 79.5% figure (Section 4): all 43 leaking AgentLeak and all 133 leaking PAC-BENCH trajectories contained an internal-channel leak, versus 66 of 83 in AgentSocialBench. This pooled value describes three deliberately equal-sized samples and should not be read as a population-weighted estimate; detection-pipeline differences further limit cross-benchmark comparison.

B.11 Robustness: Hidden Leaks as a Separate Outcome

Because most exposure never reached the final output, the null reported in Appendix B.5 could in principle be an artefact of aggregation: a measure that tracked only the hidden portion of exposure would be diluted by pooling it with visible leaks. We therefore re-estimated the exposure models with the outcome redefined as a leak that is present somewhere in the trajectory but absent from the final output. The specification is otherwise unchanged from the primary exposure analysis: a linear-probability model with model and scenario fixed effects, scenario-clustered CR1 standard errors, standardised direction-aligned predictors, and Holm correction over the family of 15 benchmark-by-metric comparisons.

No collaboration measure survives correction. The smallest Holm-adjusted p in the family is $.642$. Communication score is the only measure with a consistent direction, associated with fewer hidden failures in all three benchmarks: -3.7 percentage points per standard deviation in AgentLeak (unadjusted $p = .043$), -3.9 points in AgentSocialBench (unadjusted $p = .049$), and -1.4 points in PAC-BENCH (unadjusted $p = .494$). Taken on its own, without correction, that pattern is the shape a real but small effect would take. Two considerations argue against reading it as one. First, neither of the two nominally significant coefficients is close to the corrected threshold, and both sit at the boundary of nominal significance to begin with. Second, the effect does not survive the natural decomposition: among trajectories that leaked at all, communication score carries no association with whether the leak was hidden rather than visible, in any benchmark ($p = .12$–$.59$). The unconditional association is therefore attributable to communication score's weak relationship with leaking at all, rather than to any relationship with where a leak surfaces.

We report this analysis as a robustness check that rules out one alternative explanation of the main null, not as a finding. Collaboration quality does not predict where in a trajectory exposure appears any more than it predicts whether exposure occurs, which is what makes the process-level audit in Appendix B.10 necessary rather than substitutable by a better process metric.