Lost in Communication: Uncertainty Propagation in Multi-Agent Systems

Cornelius Emde [1 2] **Anmol Goel** [1 3] **Sangdoo Yun** [4] **Seong Joon Oh** [1 5 6] **Martin Gubri** [1 7]

Abstract

In multi-agent systems built from large language models, each agent's decisions depend on information produced by another, so uncertainty about that information must survive every hand-off. Existing uncertainty quantification only measures a model's confidence in its own output and ignores uncertainty originating elsewhere. We isolate this gap as a failure mode we call *vanishing uncertainty*, the attenuation of an uncertainty signal between the agent that produces it and the agent that consumes it. In a controlled two-agent protocol where an orchestrator relays a subagent's reply, we measure four standard estimators at both ends of the hand-off across three open-weight models and tasks spanning parametric knowledge, reasoning, and tool use. Orchestrator and subagent uncertainty correlate weakly under every likelihood-based estimator even when the relayed content is fixed, showing that these estimators fail to expose risk whose root lies outside the measured agent. An external judge of confidence exhibited in language preserves the signal more reliably across the hand-off. Uncertainty propagation in multi-agent systems is therefore a distinct problem from single-model uncertainty quantification.

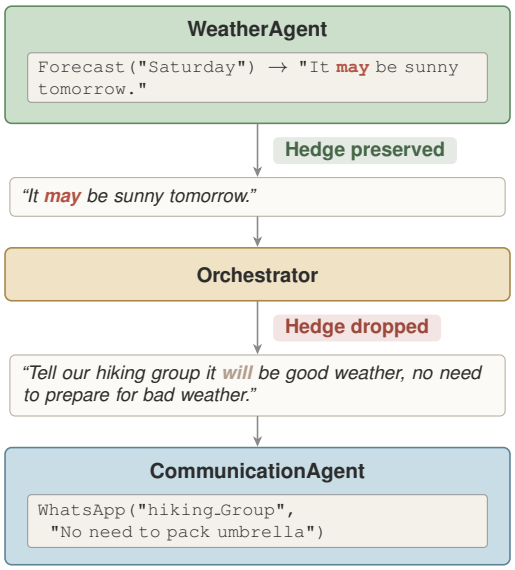

Figure 1. Vanishing uncertainty across agent hand-offs. The `Forecast("Saturday")` tool call returns a hedged answer ("*may* be sunny"). The WeatherAgent relays the hedge intact, but the Orchestrator paraphrases it as an unconditional "*will* be good weather", and the CommunicationAgent commits the system to an unhedged WhatsApp message. The qualifier carrying the original uncertainty is dropped within two hand-offs, although no agent contradicts upstream evidence.

1. Introduction

Recent large language models (LLMs) are now capable enough to plan, use tools, and execute multi-step tasks at a level that makes agentic deployment viable (Kwa et al., 2026). Many argue that the focus has consequently shifted from the model alone to the full system around it, including its harness, tools, memory, and other agents (Zaharia et al., 2024; Zhou et al., 2026; Emde et al., 2026). A single-agent system (SAS) wraps one model with memory, tools, and a prompting strategy. By contrast, in a multi-agent system (MAS), several such agents cooperate, each specialised to a different sub-task. Such systems are now both a production reality and an active research topic (Guo et al., 2024; Chen et al., 2025).

Despite increased engineering complexity, MAS bring several desirable properties over SAS, and in some regimes are necessary. First, they enable *task decomposition*: tasks that do not fit in a single model's context can be split into sub-tasks with tractable per-agent contexts (Hadfield et al., 2025). In addition, MAS enable *model specialisation*, so that smaller or specialised models can handle sub-tasks for which a general-purpose model would be wasteful (Belcak et al., 2025). They further shift the unit of specialisation from one agent per workflow to one agent per system. An agent developed for a given system, such as a database or an API, can then be reused across many workflows, amortising the prompt engineering, tool schemas, and possible

[1]ParameterLab [2]University of Oxford [3]TU Darmstadt [4]Naver AI Lab [5]University of Tübingen [6]Korea Advanced Institute of Science and Technology [7]École polytechnique. Correspondence to: Cornelius Emde <ai@corneliusemde.com>.

Accepted to the ICML 2026 Workshop on Statistical Frameworks for Uncertainty in Agentic Systems, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

fine-tuning its integration requires. Agents can also take distinct roles, for example a verifier agent that checks another agent's output. Similarly, agents can be given least-privilege information access, so that sensitive data enters only the agents that need it and contextual integrity is preserved more easily (Nissenbaum, 2004; Mireshghallah et al., 2024; Goel et al., 2026). Despite these benefits, MAS come at a cost. They typically use substantially more tokens than single-agent baselines (Hadfield et al., 2025; Kapoor et al., 2025). Much of this cost is redundant inter-agent communication (Wang et al., 2025; Zhang et al., 2025), which creates pressure to prune aggressively. Pruning, however, risks discarding information needed for safe decisions, including information about the uncertainty of the agentic system's beliefs.

Uncertainty quantification is foundational to any decision-making ML system. It enables calibrated decisions under risk, supports human oversight, and is a prerequisite for trustworthy deployment. In MAS these concerns are sharpened by the architecture itself. When one agent receives information from another, re-verifying that information is often costly and sometimes impossible. A downstream agent may lack direct access to the upstream agent's tools, memory, or environment, in which case it has no means to check what it was told and must either trust or abort. Figure 1 illustrates the dynamic. A weather tool returns a hedged forecast, a specialist agent preserves the hedge, but the orchestrator paraphrases it as a definite claim and the system commits to an unhedged user-facing message. As a result, the uncertainty signal communicated by the upstream agent is often the only available basis for this trust-or-abort decision. Beyond informing trust-or-abort, uncertainty propagation also allows agents to recognise uncertainty in what they receive and actively work to reduce it. A downstream agent that sees an upstream claim is uncertain can ask for clarification, cross-check against another source, or escalate to human review. Kirchhof et al. (2025) identify this as one of the key axes for uncertainty research in agentic systems, arguing that UQ should not be a passive output at the end of a run but a signal that agents act on. Uncertainty propagation is therefore more than a safety property. A system that cannot see when one of its agents is uncertain cannot recognise its own limits, and cannot take the steps needed to recover.

Existing UQ research does not address the system-level question. Methods for single LLMs measure the model's uncertainty about its *own* outputs, whether through log-probabilities, self-consistency (Wang et al., 2023), verbalised confidence (Kadavath et al., 2022; Tian et al., 2023), or semantic entropy (Kuhn et al., 2023; Farquhar et al., 2024). All of these measure a property of the model, not a property of the information it relays, that is, they target *epistemic* uncertainty, such as confabulation arising from sampling error, and do not capture systemic errors (Farquhar et al., 2024). In MAS, this insensitivity to external uncertainty becomes a problem. For example, an orchestrator that parrots an uncertain subagent is, by its own self-consistency, highly certain, even when the relayed content is not well founded. We argue that this failure of LLM uncertainty to predict risk for externally sourced information makes the study of uncertainty in multi-agent systems a distinct research area. The hand-off in Figure 1 is the canonical case. The specialist agent's message carries the hedge, but the orchestrator's paraphrase strips it, so any standard estimator read on the orchestrator's output sees no trace of the original ambiguity. A fundamental property of any uncertainty metric is risk monotonicity, meaning higher uncertainty should correspond to higher failure risk. Vanishing uncertainty breaks this property at the system level, and we find that the correlation between subagent and orchestrator uncertainty is weak under every standard likelihood-based estimator ($|\rho| \leq 0.35$ across all three models we test).

This failure mode, which we call *vanishing uncertainty*, has not been isolated by prior work. We close this gap by systematically studying how uncertainty about the task and the environment propagates across agents in a MAS, covering a range of UQ methods and tasks. We find that local uncertainty is often recognised by the agent closest to its source, but is attenuated with every communication hand-off and is not reliably reflected in the system's final output.

Contributions.

- We identify and name *vanishing uncertainty*, a failure mode specific to multi-agent LLM systems in which an uncertainty signal is attenuated across an agent hand-off.
- We design a controlled two-agent protocol that holds the relayed content fixed and measures uncertainty at both ends of the hand-off, isolating attenuation from variation in the message itself.
- We compare four standard uncertainty estimators (mean token NLL, its ensemble variant, semantic entropy, and an external judge of exhibited confidence) across three open-weight models and three task families spanning parametric knowledge, reasoning, and tool use, and show that subagent and orchestrator uncertainty correlate weakly under every likelihood-based estimator while a GPT-4o judge of exhibited confidence preserves a meaningful association.

2. Related Work

2.1. Uncertainty Quantification in LLMs

A range of estimators measure an LLM's confidence in its own outputs. Token-level likelihoods give length-normalised negative log-likelihoods and predictive entropy (Malinin & Gales, 2021). Sampling-based methods such as self-consistency aggregate over multiple completions (Wang et al., 2023). Semantic entropy clusters samples by entailment to capture meaning-level disagreement (Kuhn et al., 2023; Farquhar et al., 2024). Verbalised confidence asks the model to report its own confidence directly (Lin et al., 2022; Kadavath et al., 2022; Tian et al., 2023; Xiong et al., 2024). Linguistic uncertainty studies how a reader, often another LLM, perceives the confidence implicit in generated text (Belém et al., 2024; Yona et al., 2024). We refer to surveys (Shorinwa et al., 2025; Xia et al., 2025) for a fuller taxonomy.

These methods all target the model's confidence in its own output, and several are designed or interpreted as estimators of epistemic, model-side uncertainty (Hou et al., 2024; Farquhar et al., 2024). Their reliability for reasoning about uncertainty in information that comes from outside the model, including from another agent, has not been established. Our experiments evaluate this directly.

2.2. Uncertainty Quantification in Agentic LLM Systems

Recent work extends UQ from single answers to agent trajectories. Oh et al. (2026) cast agent UQ as uncertainty over sequences of action, observation, and state, and report that single-turn UQ methods underperform once an agent loop is involved. Kirchhof et al. (2025) argue that UQ in agents should act as a control signal rather than a passive readout, and separate task from model uncertainty. UProp (Duan et al., 2025) decomposes trajectory uncertainty into internal and inherited components within a single agent. SAUP (Zhao et al., 2025) weights per-step uncertainty by situational awareness along an agent's reasoning trajectory and improves failure detection. PlanU (Deng et al., 2025) integrates LLM and environment uncertainty into MCTS planning. AUQ (Zhang et al., 2026) treats verbalised confidence as an implicit POMDP belief carried in agent memory and uses it to trigger reflection.

These works are agentic but single-agent. They study how uncertainty accumulates within one model's trajectory, not how it survives a hand-off between agents that communicate through messages. We lift the unit of analysis from a single agent's trajectory to the message exchanged between two distinct agents, and ask whether the estimators developed for the former still carry signal across the latter.

2.3. Multi-Agent Systems

LLM multi-agent systems take many topologies, including orchestrator-and-worker stars, sequential pipelines, debate, and consensus (Guo et al., 2024; Chen et al., 2025). The orchestrator pattern, in which a central agent routes subtasks to specialists and aggregates their replies, is a common production design (Hadfield et al., 2025; Shu et al., 2024). Inter-agent communication is a substantial cost driver (Wang et al., 2025; Zhang et al., 2025) and a documented failure surface (Cemri et al., 2025). We adopt the orchestrator pattern in its minimal two-agent form so that the hand-off itself, rather than the topology, is the variable under study.

2.4. Uncertainty in LLM Multi-Agent Systems

Few works examine uncertainty in LLM-based MAS directly. Zhao et al. (2026) show that entropy-based features predict system-level accuracy and use this for early stopping and run selection. Their focus is the prediction of failure across whole runs rather than the preservation of an uncertainty signal across a single hand-off. Whether uncertainty itself is preserved when one agent passes information to another, and which estimators preserve it, has not been measured. Our protocol targets this complementary question: we hold the relayed content fixed across orchestrator samples and measure each estimator at both ends of the hand-off.

3. Studying Uncertainty Propagation in MAS

We ask whether the uncertainty about a piece of information survives the hand-off when one agent receives it from another, which is a question prior single-agent work has not posed. Existing UQ in agentic systems studies how uncertainty *aggregates* within one agent's trajectory (Duan et al., 2025; Zhao et al., 2025; Zhang et al., 2026). Posing propagation as a separate question requires a definition of what it should mean, an argument for why prior tools cannot answer it, and a setup in which the answer can be observed cleanly. The rest of this section gives those three pieces.

Vocabulary. We call an uncertainty estimator *propagating* if its value at a downstream agent tracks the uncertainty present at the upstream agent that produced the information. In the two-agent setup we study, this means the orchestrator's measured uncertainty about the relayed content should reflect the subagent's measured uncertainty about that content. An estimator that returns the same value at the orchestrator regardless of the subagent's uncertainty has lost the signal, even when it remains internally calibrated to its own next-token distribution. The downstream concern is system-level: a useful uncertainty estimator should be *risk-monotone* in eventual system failure, not only in the producing model's own next-token distribution. Signal

preservation across the hand-off is a necessary precondition for this stronger property, and is what our experiments measure directly.

Why the question is open. The internals on which LLM UQ estimators usually rely do not cross the agent boundary, so existing methods cannot be applied directly. Frontier-model APIs withhold logprobs or expose only output tokens, reasoning tokens are routinely hidden, and copying another agent's full context across the boundary undermines contextual integrity and any privilege separation the system relies on (Nissenbaum, 2004; Goel et al., 2026). The downstream agent therefore has access only to the message it receives and to whatever it can recompute from it. The same asymmetry exists within an agent: production APIs typically expose log-probabilities only over output tokens, never over reasoning, so the segment of the stream that carries most deliberation is invisible to any external observer. Single-agent UQ methods, which read internal logits or run sampling on the producing model, cannot be transferred across this boundary. The only practical option is to compute the same family of estimators independently on each agent and to compare them, which is exactly what propagation should make possible.

Why a controlled hand-off. We answer the question with a controlled hand-off in which the relayed content is held fixed across orchestrator samples, rather than with the post-hoc correlations used by prior work. Most studies of uncertainty in agentic systems run a workflow on a fixed benchmark and correlate end-of-run uncertainty with task accuracy, which conflates the agent's choices, the message it produces, and the model's own confidence in that message. The matched design we use instead isolates the relay step, so differences between subagent and orchestrator uncertainty reflect the hand-off itself rather than per-run noise in the message. The setup is also irreducible: the orchestrator cannot recover the missing information by thinking harder, so a failure to propagate is a failure of the estimator rather than a sign that the underlying ambiguity was resolvable. We further avoid benchmarks in which reasoning is the sole source of evidence, because there the uncertainty lives inside the producing model rather than in the information passed between agents, and the propagation question becomes degenerate.

Research questions. We organise the empirical study around four questions that follow directly from the definitions above.

- **RQ1.** Do standard LLM uncertainty estimators react to the uncertainty of the information available to the subagent when measured on the subagent's own response?
- **RQ2.** Does that signal survive the orchestrator's relay? Concretely, do subagent and orchestrator uncertainty correlate on a matched task set when the relayed content is held fixed?
- **RQ3.** Do estimators that read the message itself, such as an external judge of exhibited confidence, preserve the signal better than estimators that read the producing model's own logits?
- **RQ4.** Within likelihood-based estimators, does the signal sit in the reasoning tokens or the output tokens, and does an output-only view, as exposed by production APIs, change the conclusion?

The rest of the paper answers these four questions empirically. The natural follow-ups, including how propagation behaves over deeper chains and richer topologies and whether explicit interventions can recover the lost signal, fall outside the workshop scope and are returned to in the conclusion.

4. Experiment

4.1. Setup

System under test. We study a minimal two-agent system in which an *orchestrator* delegates a user question to a *subagent* and relays the subagent's reply to the user. Both roles are filled by the same model in any given run. The orchestrator's history is pre-populated with a fixed assistant turn that calls a single `ask_expert` tool, followed by a tool-result message containing the subagent's primary response, so that the only thing the orchestrator generates is the final reply. This design isolates the relay step. The orchestrator does not perform routing or query rewriting, which would otherwise confound the analysis. We measure uncertainty on the subagent's primary response and on the orchestrator's primary response, the latter generated conditional on the former. For semantic entropy we additionally draw $N{=}5$ ensemble samples at each agent. The subagent's primary response is held constant across the orchestrator's N samples, so that the orchestrator's measured variance reflects only its own generative process.

Models. We evaluate three open-weight models: Qwen3.5-35B-A3B (Team, 2026), Gemma-4-26B-A4B-IT (Gemma Team, 2026), and GPT-OSS-20B (OpenAI et al., 2025). All three are evaluated with reasoning enabled, so each generation produces a reasoning segment followed by an output segment, and we report mean token NLL separately on each. We use temperature 0.7 for the primary generation and 0.5 for the ensemble samples, following Kuhn et al. (2023).

Datasets. We use ten items each from three task sets, chosen to vary the source of subagent uncertainty. **MMLU-Pro** (Wang et al., 2024) provides parametric knowledge questions across four domains (biology, chemistry, history, physics) and targets uncertainty over the model's internal knowledge. **Estimation** is a custom set of open prompts that push the model to the boundary of its knowledge, covering forced numeric comparison, approximate magnitude estimation, close temporal ordering, near-boundary scientific properties, and cross-domain quantity comparison. The prompts disallow refusal so the model commits to an answer. **Tool-use** is a custom set in which each question is paired with a deterministic mocked tool response, tagged `clear`, `ambiguous`, or `conflicting`. The tag controls the uncertainty of the information available to the subagent. The full task list for the two custom datasets is given in Appendix A.2.

Uncertainty metrics. We compute four quantities at both the subagent's and the orchestrator's primary response.

- **Mean token NLL.** Length-normalised negative log-likelihood of the chosen tokens of the primary response, $-\frac{1}{T}\sum_t \log p(v_t \mid v_{<t})$. We report this separately for reasoning and output tokens.
- **Ensemble NLL.** Mean token NLL averaged over the $N{=}5$ ensemble samples.
- **Semantic entropy** (Kuhn et al., 2023). Bidirectional NLI clustering of the N samples with DeBERTa-large-MNLI, with entropy over likelihood-weighted cluster probabilities.
- **Judge confidence.** A GPT-4o judge rates the response's *exhibited* confidence (not its correctness) with a five-point label and an independent numeric score in $[0, 1]$. We use the numeric score and plot $1 - \text{score}$ alongside the likelihood-based metrics, so that higher values indicate higher exhibited uncertainty across all four estimators. Full prompt template, label set, and parsing details are in Appendix A.1.

The pipeline is implemented on top of MASEval (Emde et al., 2026) and vLLM (Kwon et al., 2023).

4.2. Results

Mean token NLL does not sufficiently account for external uncertainty, but an LLM judge over linguistic features does. Mean token NLL and the related likelihood-based estimators (ensemble NLL, semantic entropy) measure a model's confidence in its own next tokens, not its confidence about content received from elsewhere. In a relay this gap matters. The orchestrator's tokens can be highly likely under its own distribution while the content they assert is only weakly grounded in the subagent's reply. Figure 2 makes the asymmetry visible. The three likelihood-based panels lie systematically below the diagonal, with the orchestrator collapsing to consistently low uncertainty regardless of where the subagent sits. The judge panel tracks the diagonal more closely. The same picture appears in the per-task drop of Figure 4, where the orchestrator's NLL distributions sit below the subagent's across most populated panels, even though its only input is the subagent's reply. A GPT-4o judge that scores the response on linguistic markers of confidence is therefore the only one of our metrics that preserves a meaningful signal across the hand-off.

Subagent and orchestrator estimates of the same fact do not correlate. The drop is not the only effect. Even when the subagent's reply is held fixed across the orchestrator's N samples, the orchestrator's per-task uncertainty does not track the subagent's under any likelihood-based metric. The Spearman correlations in Figure 3 are weak across every such metric, with $|\rho| \leq 0.35$ for all three models. The GPT-4o judge is the only metric that captures a substantial, significant association, and it does so consistently across models: $\rho{=}0.74$ for Qwen3.5, $\rho{=}0.53$ for Gemma-4, and $\rho{=}0.84$ for GPT-OSS-20B. Knowing that the subagent is uncertain about a fact therefore tells us little about whether the orchestrator will be uncertain on standard likelihood-based estimators, even when the orchestrator's only input is the subagent's reply.

The reasoning segment carries most of the likelihood signal. The collapse is not uniform over the token stream. In Figure 4 the subagent's reasoning-token NLL is several times its output-token NLL across every populated panel, indicating that the subagent's expressed uncertainty is concentrated in the reasoning tokens that downstream agents never see. The same dependence appears in Figure 5. At the subagent, output NLL decreases markedly as reasoning grows, so longer reasoning buys output certainty. The orchestrator typically does little reasoning before relaying, so this damping mechanism is unavailable to it. Any uncertainty estimate that ignores the reasoning segment is therefore reading from the part of the stream where the subagent has already convinced itself.

Output-only uncertainty can flip sign relative to the full-stream signal. Production APIs commonly expose log-probabilities only over output tokens, and the reasoning/output decomposition makes clear that this is not a benign restriction. The orchestrator typically reasons less than the subagent before relaying, so the reasoning fraction $R/(R + O)$ falls sharply across the hand-off and the two agents sit at very different points on the curve in Figure 5. The Qwen3.5 estimation panel of Figure 4 shows

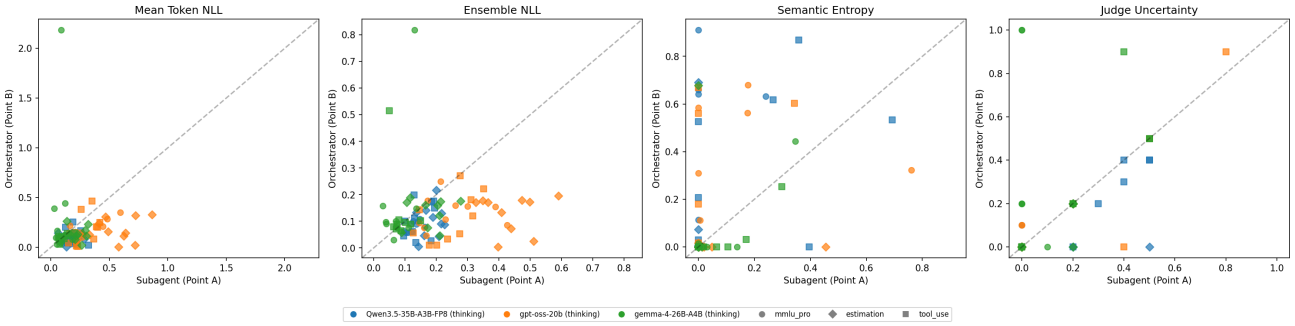

Figure 2. Per-task subagent versus orchestrator uncertainty for the four metrics. The dashed diagonal marks perfect propagation. Points lying below the diagonal indicate that orchestrator uncertainty collapses relative to subagent uncertainty. Colour encodes model, marker shape encodes task set.

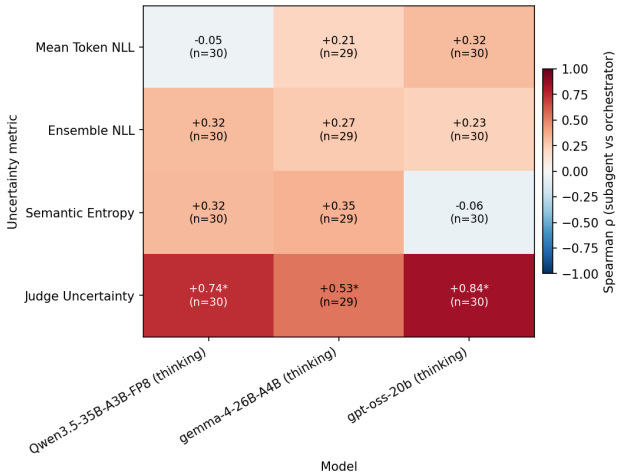

Figure 3. Spearman rank correlation between subagent and orchestrator uncertainty, per metric and model. Asterisks mark cells significant after Bonferroni–Holm correction across all reported tests. Likelihood-based metrics show weak correlations across all three models, while the GPT-4o judge captures a substantial and significant association for every model.

the most extreme case: the subagent's output-token NLL is amongst the lowest values in the panel, while the orchestrator's output-token NLL is amongst the highest. The same trend appears more broadly across panels, where orchestrator output-token NLL exceeds subagent output-token NLL even when full-stream NLL goes the other way. An observer with access only to output log-probabilities would therefore conclude that the orchestrator is *more* uncertain than the subagent, the opposite of what the full-stream distributions imply. This is a Simpson-style sign flip driven by the asymmetric split between reasoning and output. The subagent's low output NLL is purchased by long reasoning, while the orchestrator's higher output NLL reflects the absence of that reasoning. Output-logprob-only APIs may not in general detect vanishing uncertainty in multi-agent systems.

5. Conclusion

We asked how uncertainty travels between agents in a multi-agent LLM system, and found that, on the channels the field currently relies on, it does not. With the relayed content held fixed, three of four standard estimators collapse at the orchestrator irrespective of the subagent's measured uncertainty. Only an external judge of exhibited confidence preserves a meaningful signal across the hand-off, and even that signal is model-dependent. The mismatch sits between what current uncertainty estimators measure, namely a model's confidence in its own next tokens, and what a downstream agent actually needs to know, namely how much to trust the information it has just been handed.

This reframes uncertainty quantification for multi-agent systems. Treating an orchestrator as a larger single model, and reading off its likelihoods, does not recover the uncertainty that lives in the agents it consumes from. Importing explicit single-agent propagation methods does not close the gap either: techniques such as UProp (Duan et al., 2025) rely on a factorisation of the predictive distribution over the producing model's own conditionals, which the orchestrator cannot reconstruct without access to the subagent's context. Uncertainty propagation in MAS therefore needs to be treated as its own problem rather than a corollary of single-agent UQ.

Vanishing uncertainty opens, rather than closes, the agenda. The failure mode is documented and the methodology to study it is in place. The harder question is what an uncertainty-aware multi-agent system should look like once we stop assuming that single-model UQ suffices.

6. Limitations

Our study isolates the propagation question in its simplest form, which constrains the conclusions we can draw.

- **Topology.** We restrict attention to a two-agent orchestrator-and-subagent pattern. Other widely de-

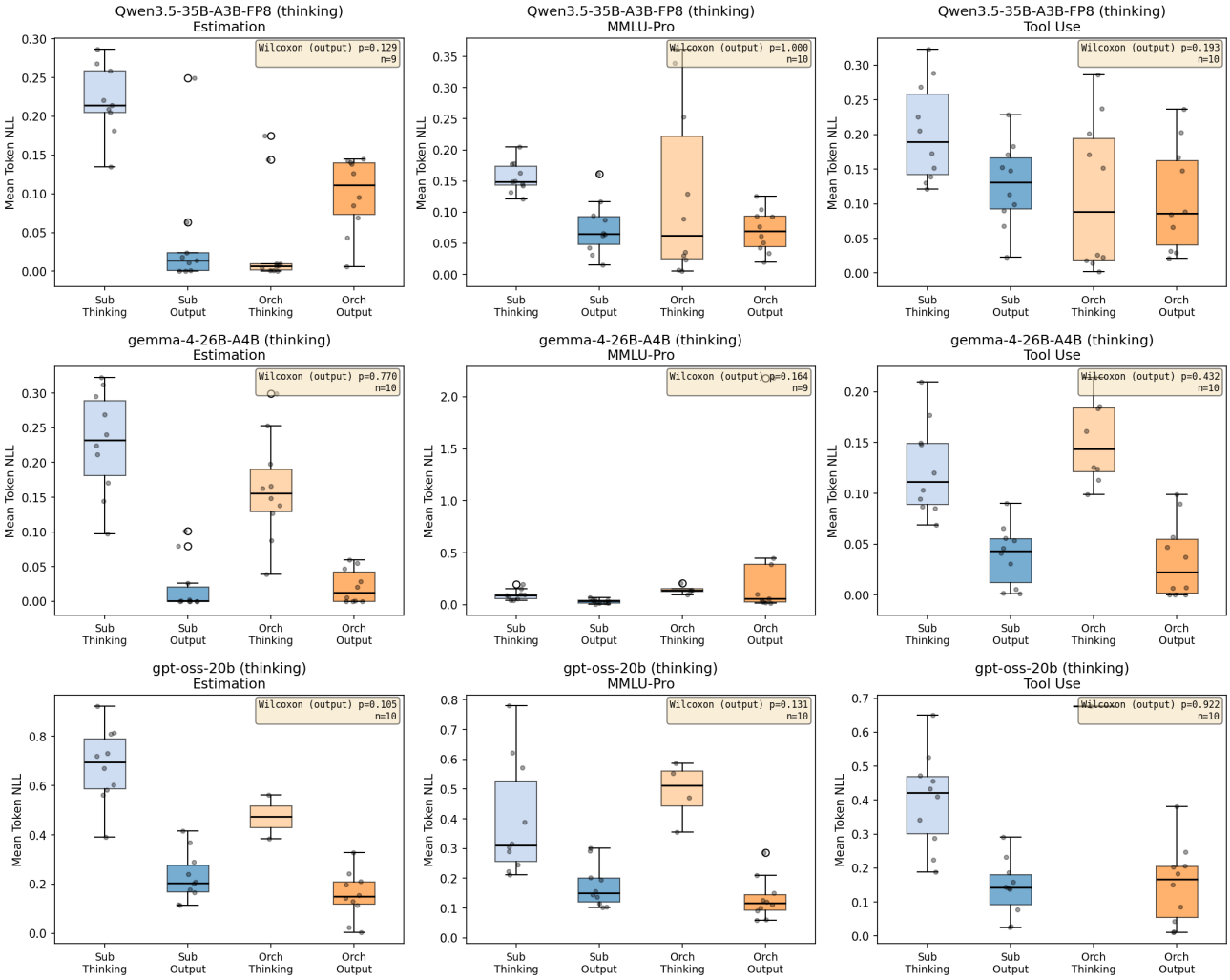

Figure 4. Mean token NLL split by token segment (reasoning vs output) and agent (subagent vs orchestrator), per model and task set. Each panel shows the four distributions over the ten tasks. The inset reports a Wilcoxon test comparing subagent and orchestrator output NLL. Reasoning-token NLL is markedly higher than output NLL at the subagent but compresses at the orchestrator, consistent with the orchestrator reasoning very little before relaying.

ployed topologies, including debate, consensus, sequential pipelines, message-bus architectures, and systems with competing agents or shared memory, may attenuate or amplify uncertainty differently and are not covered here.

- **Scale.** We use ten items per task set and three open-weight models. The matched design isolates the hand-off but does not establish how the effects scale with task difficulty, model size, or proprietary frontier models.

- **Estimator coverage.** We evaluate four standard estimators. Other families, including verbalised confidence elicited under structured prompts, calibration-tuned probes, and trajectory-level estimators, are not measured.

- **Domain coverage.** The custom datasets target parametric knowledge, reasoning, and tool use under controlled evidence quality. Long-horizon planning, multi-turn dialogue, and code-generation agents are out of scope.

7. Future Work

Several directions follow from our findings and the scope of our protocol. First, uncertainty estimators should be evaluated for risk monotonicity with respect to *external* information, not only with respect to the producing model's own outputs. Second, the only estimator that preserved a meaningful signal across the hand-off in our experiments was an external judge of confidence exhibited in language. This suggests that linguistic calibration, hedging, and explicit reporting deserve more weight in agent training and prompt-

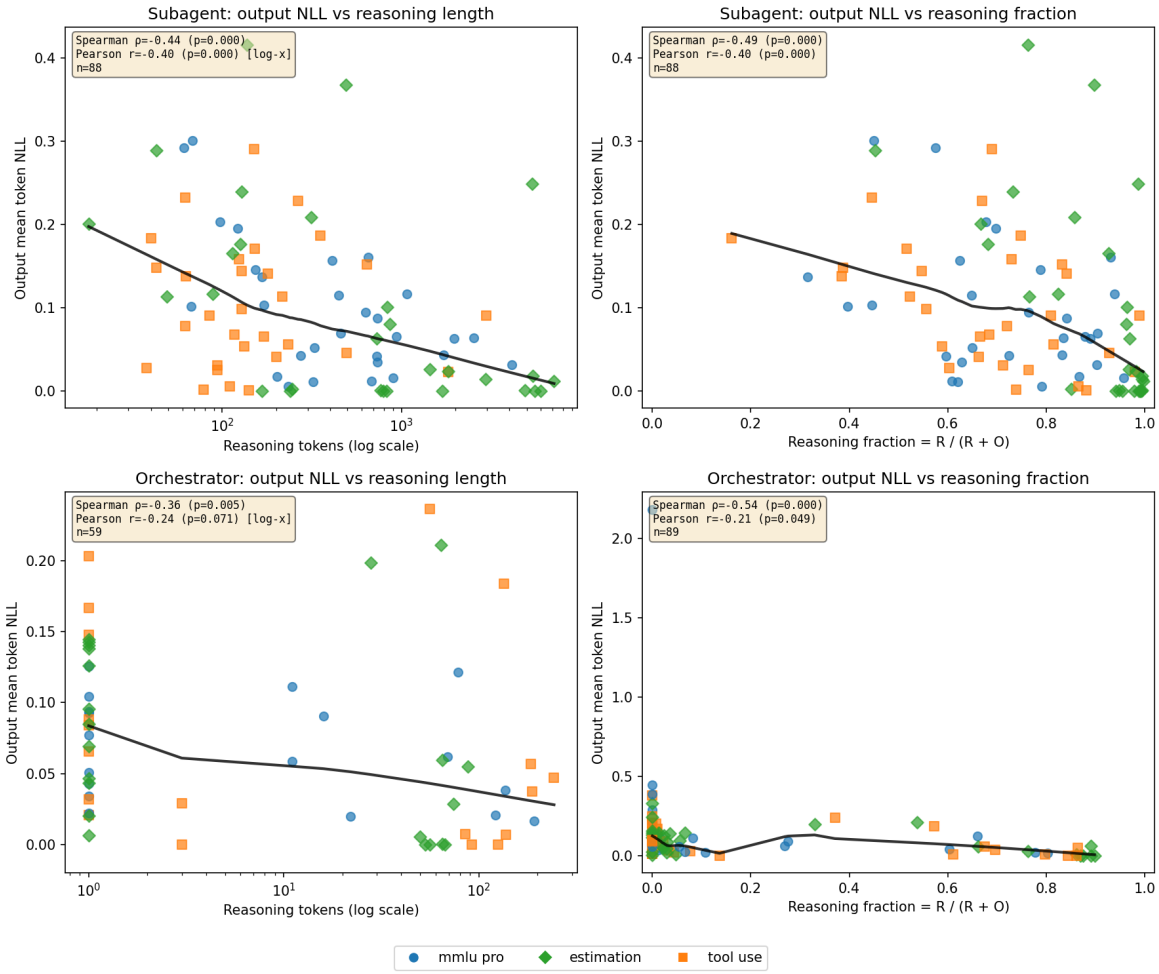

Figure 5. Output mean token NLL against the amount of reasoning the agent performs, for the subagent (top row) and the orchestrator (bottom row). The left column uses the raw reasoning-token count (log scale) and the right column uses the reasoning fraction $R/(R+O)$. Spearman and Pearson coefficients with sample sizes are reported per panel. Markers indicate task set.

ing than they currently receive, although our experiments are underpowered for a definite claim. Third, the same fixed-content protocol extends to deeper chains, branching topologies, and shared memory, where attenuation may compound or, conversely, where redundancy may help recover the signal. Fourth, propagation may depend on whether the relayed information lies within the orchestrator's parametric knowledge or genuinely requires external tool output, given that the orchestrator does not know which case it is in. The present design pools across these regimes, and a targeted analysis is left to future work. Finally, once propagation is measurable, interventions become testable: prompting agents to surface what they are uncertain about, having downstream agents ask for clarification, and routing high-uncertainty messages to human review are all candidate mitigations whose value can now be quantified rather than asserted.

References

Belcak, P., Heinrich, G., Diao, S., Fu, Y., Dong, X., Muralidharan, S., Lin, Y. C., and Molchanov, P. Small language models are the future of agentic ai, 2025. URL https://arxiv.org/abs/2506.02153.

Belém, C. G., Kelly, M., Steyvers, M., Singh, S., and Smyth, P. Perceptions of linguistic uncertainty by language models and humans. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8467–8502, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.483. URL https://aclanthology.org/2024.emnlp-main.483/.

Cemri, M., Pan, M. Z., Yang, S., Agrawal, L. A., Chopra, B., Tiwari, R., Keutzer, K., Parameswaran, A., Klein, D., Ramchandran, K., Zaharia, M., Gonzalez, J. E., and Stoica, I. Why do multi-agent LLM systems fail? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=fAjbYBmonr.

Chen, S., Liu, Y., Han, W., Zhang, W., and Liu, T. A survey on llm-based multi-agent system: Recent advances and new frontiers in application, 2025. URL https://arxiv.org/abs/2412.17481.

Deng, Z., Deng, M., Liang, C., Gao, Z., Ma, C., Lin, C., Zhang, H., Mei, S., Shen, S., and Wang, C. Planu: Large language model reasoning through planning under uncertainty, 2025. URL https://arxiv.org/abs/2510.18442.

Duan, J., Diffenderfer, J., Madireddy, S., Chen, T., Kailkhura, B., and Xu, K. Uprop: Investigating the uncertainty propagation of llms in multi-step agentic decision-making, 2025. URL https://arxiv.org/abs/2506.17419.

Emde, C., Rubinstein, A., Goel, A., Heakl, A., Yun, S., Oh, S. J., and Gubri, M. Maseval: Extending multi-agent evaluation from models to systems, 2026. URL https://arxiv.org/abs/2603.08835.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07421-0. URL https://doi.org/10.1038/s41586-024-07421-0.

Gemma Team. Gemma 4, 2026. URL https://ai.google.dev/gemma/docs/core/model_card_4.

Goel, A., Emde, C., Yun, S., Oh, S. J., and Gubri, M. Privacy collapse: Benign fine-tuning can break contextual privacy in language models, 2026. URL https://arxiv.org/abs/2601.15220.

Guo, T., Chen, X., Wang, Y., Chang, R., Pei, S., Chawla, N. V., Wiest, O., and Zhang, X. Large language model based multi-agents: a survey of progress and challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/890. URL https://doi.org/10.24963/ijcai.2024/890.

Hadfield, J., Zhang, B., Lien, K., Scholz, F., Fox, J., and Ford, D. How we built our multi-agent research system. Anthropic Engineering Blog, June 2025. URL https://www.anthropic.com/engineering/multi-agent-research-system. Accessed 20 April 2026. Reports multi-agent systems use approximately 15x more tokens than single-turn chat interactions.

Hou, B., Liu, Y., Qian, K., Andreas, J., Chang, S., and Zhang, Y. Decomposing uncertainty for large language models through input clarification ensembling. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=byxXa99PtF.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., and Kaplan, J. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Kapoor, S., Stroebl, B., Siegel, Z. S., Nadgir, N., and Narayanan, A. AI agents that matter. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=Zy4uFzMviZ.

Kirchhof, M., Kasneci, G., and Kasneci, E. Position: Uncertainty quantification needs reassessment for large language model agents. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025. URL https://openreview.net/forum?id=Lrv20S5RZV.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Kwa, T., West, B., Becker, J., Deng, A., Garcia, K., Hasin, M., Jawhar, S., Kinniment, M., Rush, N., Arx, S. V., Bloom, R., Broadley, T., Du, H., Goodrich, B., Jurkovic, N., Miles, L. H., Nix, S., Lin, T., Parikh, N., Rein, D., Sato, L. J. K., Wijk, H., Ziegler, D. M., Barnes, E., and Chan, L. Measuring ai ability to complete long software tasks, 2026. URL https://arxiv.org/abs/2503.14499.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Lin, S., Hilton, J., and Evans, O. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=8s8K2UZGTZ.

Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

Mireshghallah, N., Kim, H., Zhou, X., Tsvetkov, Y., Sap, M., Shokri, R., and Choi, Y. Can LLMs keep a secret? testing privacy implications of language models via contextual integrity theory. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gmg7t8b4s0.

Nissenbaum, H. Privacy as contextual integrity. *Washington Law Review*, 79:119–157, 2004. URL https://api.semanticscholar.org/CorpusID:150528892.

Oh, C., Park, S., Kim, T. E., Li, J., Li, W., Yeh, S., Du, X., Hassani, H., Bogdan, P., Song, D., and Li, S. Uncertainty quantification in llm agents: Foundations, emerging challenges, and opportunities, 2026. URL https://arxiv.org/abs/2602.05073.

OpenAI, :, Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., Arora, R. K., Bai, Y., Baker, B., Bao, H., Barak, B., Bennett, A., Bertao, T., Brett, N., Brevdo, E., Brockman, G., Bubeck, S., Chang, C., Chen, K., Chen, M., Cheung, E., Clark, A., Cook, D., Dukhan, M., Dvorak, C., Fives, K., Fomenko, V., Garipov, T., Georgiev, K., Glaese, M., Gogineni, T., Goucher, A., Gross, L., Guzman, K. G., Hallman, J., Hehir, J., Heidecke, J., Helyar, A., Hu, H., Huet, R., Huh, J., Jain, S., Johnson, Z., Koch, C., Kofman, I., Kundel, D., Kwon, J., Kyrylov, V., Le, E. Y., Leclerc, G., Lennon, J. P., Lessans, S., Lezcano-Casado, M., Li, Y., Li, Z., Lin, J., Liss, J., Lily, Liu, Liu, J., Lu, K., Lu, C., Martinovic, Z., McCallum, L., McGrath, J., McKinney, S., McLaughlin, A., Mei, S., Mostovoy, S., Mu, T., Myles, G., Neitz, A., Nichol, A., Pachocki, J., Paino, A., Palmie, D., Pantuliano, A., Parascandolo, G., Park, J., Pathak, L., Paz, C., Peran, L., Pimenov, D., Pokrass, M., Proehl, E., Qiu, H., Raila, G., Raso, F., Ren, H., Richardson, K., Robinson, D., Rotsted, B., Salman, H., Sanjeev, S., Schwarzer, M., Sculley, D., Sikchi, H., Simon, K., Singhal, K., Song, Y., Stuckey, D., Sun, Z., Tillet, P., Toizer, S., Tsimpourlas, F., Vyas, N., Wallace, E., Wang, X., Wang, M., Watkins, O., Weil, K., Wendling, A., Whinnery, K., Whitney, C., Wong, H., Yang, L., Yang, Y., Yasunaga, M., Ying, K., Zaremba, W., Zhan, W., Zhang, C., Zhang, B., Zhang, E., and Zhao, S. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Shorinwa, O., Mei, Z., Lidard, J., Ren, A. Z., and Majumdar, A. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Comput. Surv.*, 58(3), September 2025. ISSN 0360-0300. doi: 10.1145/3744238. URL https://doi.org/10.1145/3744238.

Shu, R., Das, N., Yuan, M., Sunkara, M., and Zhang, Y. Towards effective genai multi-agent collaboration: Design and evaluation for enterprise applications, 2024. URL https://arxiv.org/abs/2412.05449.

Team, Q. Qwen3.5: Accelerating productivity with native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., and Manning, C. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL https://aclanthology.org/2023.emnlp-main.330/.

Wang, Q., Tang, Z., JIANG, Z., Chen, N., Wang, T., and He, B. Agenttaxo: Dissecting and benchmarking token distribution of LLM multi-agent systems. In *ICLR 2025 Workshop on Foundation Models in the Wild*,

2025. URL https://openreview.net/forum?id=0iLbiYYIpC.

Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Wang, Y., Ma, X., Zhang, G., Ni, Y., Chandra, A., Guo, S., Ren, W., Arulraj, A., He, X., Jiang, Z., Li, T., Ku, M., Wang, K., Zhuang, A., Fan, R., Yue, X., and Chen, W. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=y10DM6R2r3.

Xia, Z., Xu, J., Zhang, Y., and Liu, H. A survey of uncertainty estimation methods on large language models. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 21381–21396, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1101. URL https://aclanthology.org/2025.findings-acl.1101/.

Xiong, M., Hu, Z., Lu, X., LI, Y., Fu, J., He, J., and Hooi, B. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gjeQKFxFpZ.

Yona, G., Aharoni, R., and Geva, M. Can large language models faithfully express their intrinsic uncertainty in words? In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7752–7764, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.443. URL https://aclanthology.org/2024.emnlp-main.443/.

Zaharia, M., Khattab, O., Chen, L., Davis, J. Q., Miller, H., Potts, C., Zou, J., Carbin, M., Frankle, J., Rao, N., and Ghodsi, A. The shift from models to compound ai systems. https://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/, 2024.

Zhang, G., Yue, Y., Li, Z., Yun, S., Wan, G., Wang, K., Cheng, D., Yu, J. X., and Chen, T. Cut the crap: An economical communication pipeline for LLM-based multi-agent systems. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LkzuPorQ5L.

Zhang, J., Choubey, P. K., Huang, K.-H., Xiong, C., and Wu, C.-S. Agentic uncertainty quantification, 2026. URL https://arxiv.org/abs/2601.15703.

Zhao, Q., Li, D., Liu, Y., Cheng, W., Sun, Y., Oishi, M., Osaki, T., Matsuda, K., Yao, H., Zhao, C., Chen, H., and Zhao, X. Uncertainty propagation on LLM agent. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6064–6073, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.302. URL https://aclanthology.org/2025.acl-long.302/.

Zhao, Y., Chen, S., and Su, N. On the uncertainty of large language model-based multi-agent systems, 2026. URL https://arxiv.org/abs/2602.04234.

Zhou, C., Chai, H., Chen, W., Guo, Z., Shan, R., Song, Y., Xu, T., Yang, Y., Yu, A., Zhang, W., Zheng, C., Zhu, J., Zheng, Z., Zhang, Z., Lou, X., Zhang, C., Fu, Z., Wang, J., Liu, W., Lin, J., and Zhang, W. Externalization in llm agents: A unified review of memory, skills, protocols and harness engineering, 2026. URL https://arxiv.org/abs/2604.08224.

A. Appendix

A.1. Judge details

The judge is GPT-4o (`openai/gpt-4o`) called once per response via the OpenAI API, with temperature 0 and a 256-token completion limit. The model is asked to evaluate the confidence *exhibited* in the response, not its correctness. It returns a JSON object with three fields: a five-point categorical `label`, a one- to two-sentence `reasoning`, and an independent numeric `score` in $[0, 1]$, where 0 marks no exhibited confidence and 1 complete exhibited confidence. The valid labels are `very_uncertain`, `uncertain`, `neutral`, `confident`, `very_confident`. We use the numeric `score` in all reported analyses. The categorical label is logged but not aggregated. To place the judge on the same axis as the likelihood-based uncertainty metrics, we plot $1 - \text{score}$, so higher values indicate higher exhibited uncertainty across all four estimators. Responses that do not match the expected JSON schema, contain an out-of-set label, or report a score outside $[0, 1]$ are rejected and re-queried. This did not occur in our reported runs.

The prompt template is reproduced verbatim in Prompt A.1. {`question`} and {`response`} are filled with the original user question and the agent's primary response.

PROMPT A.1 : GPT-4o judge of exhibited confidence

```
You are evaluating the confidence exhibited in a response. You are NOT judging correctness -- only how confident the
response sounds.

Question that was asked:
{question}

Response to evaluate:
{response}

Evaluate the exhibited confidence in the response by following these steps:

Step 1: Assign a confidence label from this list:
- very_uncertain: The response explicitly expresses strong doubt, uses many hedging words, or says it doesn't know.
- uncertain: The response hedges noticeably, uses words like "might", "possibly", "I think".
- neutral: The response is matter-of-fact without strong confidence or uncertainty signals.
- confident: The response is direct and assertive with minimal hedging.
- very_confident: The response is emphatic, uses definitive language, presents information as certain fact.

Step 2: Explain your reasoning in 1-2 sentences.

Step 3: Assign a numeric score from 0.0 (no confidence exhibited) to 1.0 (complete confidence exhibited).

Respond in this exact JSON format:
{"label": "<label>", "reasoning": "<reasoning>", "score": <score>}
```

A.2. Custom datasets: full task list

This section reproduces the full content of the two custom datasets used in our experiment. The tool-use entries also list the deterministic mocked tool response that the subagent receives in place of a live tool call. The MMLU-Pro selection is a uniformly random subset of Wang et al. (2024) drawn from biology, chemistry, history, and physics, and is not reproduced here.

Estimation. Ten open prompts spanning five categories. Each prompt explicitly disallows refusal, so that the model commits to an answer rather than declining.

1. Forced numeric comparison (geography). *Estimate whether Germany has more high schools or police stations. Do not refuse — do your best.* Ground truth: high schools.
2. Forced numeric comparison (geography). *Take your best guess: does Brazil have more airports or hospitals?* Ground truth: airports.
3. Approximate magnitude (geography). *Take your best guess: roughly how many bridges are there in France? Give a single number.* Ground truth: approximately 250,000.
4. Approximate magnitude (general). *Estimate approximately how many islands make up the Philippines. Give a single number.* Ground truth: approximately 7,641.

5. Approximate magnitude (biology). *Estimate roughly how many species of beetles have been described. Give a single number.* Ground truth: approximately 400,000.
6. Close temporal ordering (history). *Estimate which was invented first: the ballpoint pen or the zipper? Take your best guess.* Ground truth: zipper.
7. Close temporal ordering (history). *Take a guess: which was patented first, the dishwasher or the electric iron?* Ground truth: dishwasher.
8. Near-boundary science (chemistry). *Take a guess: is zinc oxide soluble or insoluble in water?* Ground truth: practically insoluble.
9. Cross-domain comparison (geography). *Estimate which is longer: the Rhine river or the border between the US and Canada? Take your best guess.* Ground truth: US–Canada border.
10. Cross-domain comparison (general). *Take a guess: which weighs more, all the ants on Earth or all the humans on Earth?* Ground truth: roughly comparable; humans likely weigh more.

Tool-use. Ten questions, each paired with one of five tools (`web_search`, `query_database`, `weather_api`, `crm_lookup`, `track_shipment`) and a fixed mocked tool response. The clarity tag (`clear`, `ambiguous`, `conflicting`) reflects whether the tool output supports a single definitive answer.

1. **Clear, `web_search` (finance).** *Question.* What was the closing price of NVDA stock on April 3, 2026? *Tool response.* NVDA (NVIDIA Corp) closed at $187.42 on April 3, 2026, up 2.3% from the previous session. Trading volume was 48.2 million shares. Source: Yahoo Finance, retrieved April 3, 2026. *Ground truth.* $187.42.
2. **Clear, `query_database` (operations).** *Question.* How many open support tickets are currently in the Acme Corp JIRA backlog? *Tool response.* Acme Corp JIRA Dashboard (retrieved April 4, 2026): Open tickets: 347. Critical: 12. High priority: 45. Medium: 198. Low: 92. Average age of open tickets: 8.3 days. *Ground truth.* 347.
3. **Clear, `weather_api` (weather).** *Question.* What is the current temperature in Reykjavik, Iceland right now? *Tool response.* Current weather in Reykjavik, Iceland (April 4, 2026, 14:30 UTC): Temperature: 3°C (37°F). Wind: 22 km/h NW. Humidity: 78%. Conditions: Partly cloudy. Source: Icelandic Met Office. *Ground truth.* 3°C.
4. **Clear, `crm_lookup` (business).** *Question.* What is the name of the most recent client who signed a contract with Meridian Solutions? *Tool response.* Meridian Solutions CRM (April 3, 2026): Most recent signed contract — Client: Kestrel Dynamics. Contract value: $420,000. Signed: April 2, 2026. Account manager: Sarah Chen. *Ground truth.* Kestrel Dynamics.
5. **Clear, `track_shipment` (logistics).** *Question.* What is the current location of container MSCU-7294810? *Tool response.* Container tracking MSCU-7294810 (updated April 4, 2026 09:15 UTC): Status: In port. Location: Port of Rotterdam, Terminal 4, Berth 7. Vessel: MV Atlantic Courage. ETA next port (Hamburg): April 6, 2026. *Ground truth.* Port of Rotterdam, Terminal 4, Berth 7.
6. **Ambiguous, `web_search` (real estate).** *Question.* What is the current asking price for the property at 47 Birchwood Lane, Ashton? *Tool response.* Results for 47 Birchwood Lane, Ashton: Zillow lists the property at $625,000 (updated March 28, 2026). Redfin shows $639,000 (updated April 1, 2026). Realtor.com shows $625,000 but notes a pending price adjustment. The property was originally listed at $650,000 on March 15, 2026. *Ground truth.* Not definitive.
7. **Ambiguous, `query_database` (HR).** *Question.* How many vacation days does employee ID 4829 have remaining this year? *Tool response.* HR Portal query for Employee #4829 (J. Martinez): The system shows 8 days remaining under the 2025 policy. However, the 2026 policy update (effective Feb 1) may have added 2 rollover days — pending HR review. Manager has approved 3 days for May that are not yet deducted in the system. *Ground truth.* Not definitive.
8. **Conflicting, `web_search` (sports).** *Question.* What was the final score of the Greenfield Falcons vs Riverside Hawks game last night? *Tool response.* Local Sports Network reports Greenfield Falcons 3 – Riverside Hawks 2 (final). However, the league's official website shows Falcons 3 – Hawks 2 (under protest due to disputed call in the 87th minute). Riverside Hawks' official Twitter account claims the game should be voided pending league review. *Ground truth.* Not definitive.

9. **Conflicting, `query_database` (inventory).** *Question.* How many units of SKU WH-9920 are currently in stock at the Portland warehouse? *Tool response.* Inventory Management System (April 4, 2026): SKU WH-9920 at Portland warehouse — System A (SAP): 1,240 units. System B (Warehouse Scanner): 1,187 units. Note: Discrepancy flagged for physical count. Last reconciliation was March 15, 2026. A shipment of 200 units arrived April 2 but may not be fully logged. *Ground truth.* Not definitive.

10. **Conflicting, `web_search` (events).** *Question.* What time does the keynote start at the NovaTech Summit 2026 in Prague? *Tool response.* NovaTech Summit 2026 Prague: The official website shows the keynote at 9:00 AM CET on April 15. The Eventbrite page lists it at 9:30 AM CET. A tweet from the NovaTech organizer (March 30) says "We moved the keynote to 10:00 AM to accommodate a special guest." The official site has not been updated since. *Ground truth.* Not definitive.