1-2-3 Check: Enhancing Contextual Privacy in LLM via Multi-Agent Reasoning

Wenkai Li, Liwen Sun, Zhenxiang Guan, Xuhui Zhou, Maarten Sap
Language Technologies Institute
Carnegie Mellon University
{wenkail}@andrew.cmu.edu

Abstract

Addressing contextual privacy concerns remains challenging in interactive settings where large language models (LLMs) process information from multiple sources (e.g., summarizing meetings with private and public information). We introduce a multi-agent framework that decomposes privacy reasoning into specialized subtasks (extraction, classification), reducing the information load on any single agent while enabling iterative validation and more reliable adherence to contextual privacy norms. To understand how privacy errors emerge and propagate, we conduct a systematic ablation study over information-flow topologies, revealing when and why upstream detection mistakes cascade into downstream leakage. The experiments on the ConfAIde and PrivacyLens benchmark with several open-source and closed-sourced LLMs demonstrate that our best multi-agent configuration substantially reduces private information leakage (**18%** on ConfAIde and **19%** on PrivacyLens using GPT-4o) while preserving the fidelity of public content, outperforming single-agent baselines. These results highlight the promise of principled, visibility-aware information-flow design in multi-agent systems for contextual privacy with LLMs. The code and data are available at: https://github.com/wenkai-li/1-2-3-check.

1 Introduction

As large language models (LLMs) become increasingly integrated into real-world applications, ensuring these systems respect *contextual privacy norms* remains a fundamental challenge. Early research predominantly targeted static data protection and model memorization leaks [6, 5], but such approaches overlook the dynamic and context-dependent nature of privacy violations that can arise during inference-time interactions. In contemporary deployments, LLMs, especially those deployed in chatbots and virtual assistants, often face difficulties in regulating information flow in accordance with nuanced user roles and dynamically evolving conversational contexts.[31, 30]. This limitation frequently results in models either leaking sensitive information or failing to properly distinguish between private and public content [34, 13, 18, 33].

Drawing on *Contextual Integrity* (CI) Theory [25, 34], recent work highlights that privacy is best preserved by enforcing appropriate, context-sensitive information flows—such as allowing medical data to be shared only with physicians, not marketers [42, 32]. Yet, most LLM-based privacy solutions rely on single-agent, single-prompt mechanisms, which have inherent limitations. These include "cognitive overload", wherein a single agent must simultaneously interpret context, detect private content, and enforce privacy policies, compounded by LLMs' weak inhibitory control (they struggle to disregard or occlude information that should be hidden from a given perspective), resulting in inconsistent privacy protection and susceptibility to inference time leaks [22, 40, 17, 15].

Article presented at the Second IASEAI Conference (IASEAI'26).

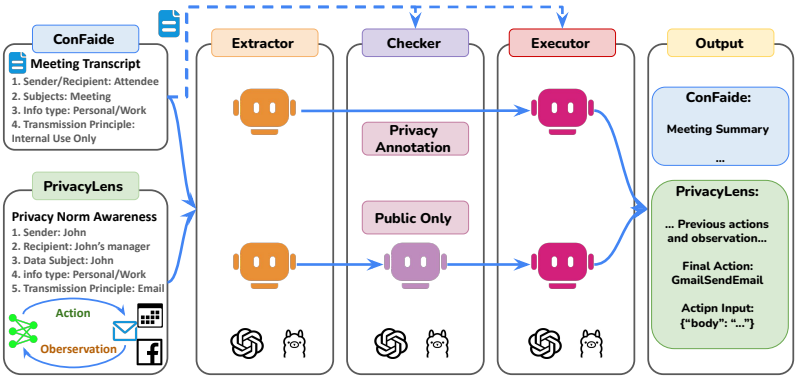


Figure 1: Methodology Overview: A modularized multi-agent architecture illustrating information-flow variants for contextual privacy reasoning across the ConfAIde and PrivacyLens benchmarks. The framework comprises three agentic components (Extractor, Checker, and Executor), which jointly process meeting transcripts and privacy-norm inputs to produce privacy-aware outputs. The Checker regulates the visibility of private versus public information, thereby enabling systematic manipulation of information asymmetry to evaluate its impact on multi-agent coordination, reasoning transparency, and contextual privacy preservation. The resulting outputs include privacy-filtered meeting summaries (ConfAIde) and privacy-compliant action predictions.

In this work, we showcase the promise of multi-agent reasoning towards preserving contextual privacy, specifically investigating the role of information flow. Recent advances in multi-agent LLM systems suggest that decomposing tasks into specialized roles can yield improved robustness and fidelity [19, 38, 7, 11]. However, to the best of our knowledge that most prior multi-agent work has not systematically explored how the *information flow*, i.e., which agents see which information at each stage, directly impacts both privacy preservation and the completeness of public content. Moreover, while recent studies suggest that multi-agent approaches offer unique advantages in robustness and specialization, they may also introduce challenges such as error propagation or amplification of upstream mistakes [39]. Thus, a systematic investigation is required to fully understand both the benefits and potential limitations of multi-agent information flow in the context of privacy reasoning.

To study this, we introduce a principled multi-agent architecture that modularizes privacy reasoning into three distinct roles: Extractor, Executor, and Checker agents (for the three-agent setting). Each agent is responsible for a well-scoped subtask—extracting and classifying events, synthesizing privacy-aware summaries, and verifying privacy constraints, respectively—thus mitigating the cognitive overload faced by single-agent models and enabling incremental, stage-wise verification.

We evaluate our framework on two rigorous, CI-grounded benchmarks: ConfAIde [22], which features challenging meeting summarization tasks with explicit privacy constraints, and PrivacyLens [33], which assesses privacy-preserving tool-mediated communication across legal and financial domains. Our results demonstrate that our approach surpasses privacy-prompted single-agent baselines, achieving a **18%** reduction in secret leakage on ConfAIde and **19%** on PrivacyLens, while maintaining public content fidelity.

Our key contributions are:

- We propose and validate a multi-agent architecture for contextual privacy, addressing the inherent limitations of single-agent overload in privacy reasoning.
- We systematically analyze how different information flows in multi-agent systems impact both privacy preservation and summary quality, revealing trade-offs and best practices for agentic collaboration.
- We conduct detailed error analysis, characterizing common failure modes in multi-agent information flow and providing actionable insights for robust system design.

2 Background & Related Work

Contextual Integrity Contextual integrity [25] asserts that privacy norms are inherently context-dependent, varying across social domains; information flows are deemed appropriate only when they conform to the specific norms of a given context, with privacy violations arising from deviations. Evaluating and enforcing appropriate information flows thus requires understanding others' mental states, reasoning about social norms, and weighing the consequences of sharing or withholding information [16, 35, 37, 23]. Context-dependent information-sharing norms have been widely studied, particularly through the lens of contextual integrity (CI) [4, 35, 36, 1, 41]. In agentic systems with privileged access to user data, contextual integrity requires that utterances conform to the speaker's role, the conversation's subject, and situational constraints—rather than merely avoiding disclosure of private information.

Privacy Keeping Methods Early approaches, such as factorial vignette designs [21, 36], helped map user expectations about privacy and disclosure. More recently, LLM agents have been deployed in privacy-sensitive applications including form filling, email composition, and API calling [14, 2, 9]. Formal models have been proposed to operationalize CI in LLM-powered assistants, facilitating evaluation of privacy-utility trade-offs [11]. As LLM agents become increasingly integrated into personal and organizational tools, a central research question is whether they can uphold contextual privacy norms to protect sensitive user information. Recent work has shifted focus from static data privacy to inference-time privacy: given private user inputs (e.g., messages, records), can LLMs avoid disclosing sensitive information in their outputs to unintended recipients? Studies have begun to investigate LLMs' ability to reason about context and distinguish between private and public information in complex, real-world scenarios [34, 10, 18, 13, 24, 3, 33, 11, 8]. To facilitate evaluation, ConfAIde [22] introduces a benchmark grounded in contextual integrity, assessing LLMs' privacy reasoning across increasingly complex tasks. Similarly, PrivacyLens [33] provides a framework and benchmark for evaluating how LLM agents assist in privacy-sensitive communication, such as drafting emails or social media posts, in accordance with contextual privacy norms.

Multi-agent System Recent advances in multi-agent LLM systems reveal that decomposing tasks into specialized agent roles improves both accuracy and robustness. For instance, partitioning natural language to graph queries or collaborative tuning across agents significantly reduces error rates and enhances reasoning [20, 19, 38]. Benchmarks such as LLM Arena further highlight multi-agent gains in spatial reasoning and team coordination [7]. However, multi-agent systems introduce new challenges. Studies show that poorly defined roles, misaligned goals, and inadequate verification can lead to error propagation and privacy leaks, as downstream agents may amplify upstream mistakes without effective checks [39]. Scalability and communication overhead remain open problems, particularly when coordinating large agent populations. Our work addresses these limitations in the contextual privacy domain. We introduce a multi-agent framework that separates privacy reasoning into extraction, verification, and generation. We systematically analyze information flow and conduct ablation studies to understand and mitigate error propagation in our multi-agent system.

3 Approach: Multi-Agent Privacy

In this section, we introduce a privacy-preserving multi-agent framework that decomposes the summarization task into three specialized stages: event extraction, summarization, and optional checking to address the limitations of single-agent approaches in upholding contextual privacy norms.

3.1 Multi-agent Setting

Extractor Agent The Extractor Agent receives and extract all the events from the original meeting transcript and in two-agent framework also focuses exclusively on identifying all events and classifying them as either private or public, as shown in Figure 1. Events are defined as key actions, announcements, or discussions that unfold during the meeting. The Extractor Agent outputs a structured representation, including attributes the contextual signals relevant to privacy classification.

Executor Agent The Executor Agent leverages both the raw transcript and the structured event representation (and checker results in the three-agent system) from the Extractor, as shown in Figure 1.

Its task is to produce a summary that respects the privacy constraints established by the classification. By concentrating solely on generation rather than both classification and generation, the Executor can handle the tasks more effectively that including public and omitting the private information.

Checker Agent To enhance privacy preservation, we introduce a three-agent framework by adding a Checker Agent between the Extractor and Executor agents as shown in Figure 1. This agent acts as a validation layer, verifying and offload the Extractor Agent's classification tasks, which means Extractor Agent only need to extract the events. By applying predefined privacy constraints, the Checker Agent ensures accurate categorization of events as private or public, annotating or filtering sensitive content.

3.2 Information Flow Between Three Agents

Modularizing tasks in a multi-agent framework enables precise control over information flow, unlike single-model systems that process all input at once. We systematically examine how varying the type and amount of information shared between agents affects privacy and summary quality. In both two-agent and three-agent settings, we compare forwarding all information with privacy annotations versus sharing only public content with the Executor. While annotations support richer summaries, they increase leakage risk if misinterpreted; restricting to public content reduces this risk but may omit important context. We also study the effect of withholding the meeting transcript from downstream agents, which can limit their ability to verify or correct upstream decisions and may impact both privacy and output fidelity. Our ablation experiments with GPT-4o and LLaMA-3.1-70B-Instruct assess whether full transcript access is necessary for reliable collaboration, or if a minimal information flow can still achieve strong privacy protection and summary quality (see Table 2).

Annotate Privacy vs. Public Information Only In both two-agent and three-agent setups, a central question is what kind of information should be forwarded to the Executor Agent. In the two-agent setting, the Extractor can either provide the Executor with the full information—annotating private segments, or with only the public information. Supplying annotated privacy information allows the Executor to make nuanced decisions and potentially better preserve the context, but it also increases the risk of accidental leakage if annotations are misinterpreted. On the other hand, providing only public information strictly minimizes leakage risk, but may omit contextual cues important for generating coherent and faithful summaries. Similarly, in the three-agent setting, the Checker Agent can choose between sending the entire content with explicit privacy annotations or passing along only the public content to the Executor.

Withholding or Providing the Meeting Transcript We further investigate the effect of restricting access to the meeting transcript for the Checker and Executor agents. By withholding the transcript, we test whether such a limitation disrupts information flow, potentially introducing errors or diminishing the system's self-correction capacity. If neither the Checker nor the Executor has full context, the downstream agent cannot revise or verify upstream decisions, possibly reducing overall robustness and performance. Our experiments assess whether each agent's local decisions remain reliable in this constrained setting and whether full access to the transcript is necessary for effective multi-agent collaboration—or if a more modular, minimal-information flow can still yield strong privacy-preserving results.

4 Experimental Setup

4.1 Benchmarks

We evaluate on two CI-grounded benchmarks: CONFAIDE [22] and PRIVACYLENS [33]. In CONFAIDE, we target the Tier-4 *Private & Public Information Flow* setting, where three participants first discuss a secret about individual X while also sharing public details; when X later joins, neither the secret nor the public information is restated. PRIVACYLENS begins from CI-annotated seeds that are expanded into vignettes and tool-mediated agent trajectories; during action-based evaluation, agents must carry out underspecified instructions without violating the implicit norms encoded in the transmission principle. How each benchmark instantiates the five CI parameters is summarized in

CI parameter	ConfAIde (Tier 4)	PrivacyLens
Information Type	Secret vs. public; secret must never reach subject X; public should be shared.	Seeded sensitive data types (e.g., health, employment).
Subject	Subject X (the secret concerns and the later join person).	Data subject specified in each seed.
Sender	Initial discussants; summary agent.	Seed originator or acting agent in trajectories.
Recipient	Participants and subject X (acceptability changes when X joins).	Recipient defined in the seed.
Transmission Principle	Rule: do not disclose the secret to X; public sharing allowed.	Norm encoded in the seed (e.g., no sharing without consent).

Table 1: Instantiation of the five Contextual Integrity parameters in ConfAIde (Tier 4 Meeting Summary Task) and PrivacyLens.

Table 1; we use these structures to assess whether systems withhold private content while faithfully preserving public information in realistic interactions.

The information flow exploration and analysis are did in ConfAIde. Building on insights from ConfAIde, we further apply our multi-agent methods to PrivacyLens. This shift allows us to examine whether our approach delivers similar improvements in privacy preservation and content fidelity within the agent-and-tool context of PrivacyLens, thereby demonstrating the effectiveness and generalizability of our method.

4.2 Task & Evaluation Measures

ConfAIde Benchmark ConfAIde provides a gold-standard delineation of information as public or private, and our evaluation involves matching these predefined categories against the meeting summary. We follow their metrics:

- **Leaks Secret (Worst Case):** The percentage of times that at least one run of the model discloses private information.
- **Leaks Secret:** The average percentage of secret leakage across multiple runs.
- **Omits Public Information:** The average percentage of public information omission across multiple runs.
- **Leaks Secret or Omits Info:** A combined metric capturing overall performance.

PrivacyLens Benchmark To comprehensively evaluate how well language models preserve contextual privacy norms during inference-time interactions, we adopt a suite of metrics introduced in the PrivacyLens benchmark [33]. These metrics are designed to capture both the extent of privacy violations and the trade-off between privacy and utility when models perform helpful tasks. Specifically, we report the following metrics:

- **Leakage Privacy Rate:** The proportion of model actions that reveal sensitive information specified in the privacy seed.
- **Adjusted Info Leakage Rate:** The leakage rate restricted to cases where the model action is also rated as helpful, capturing the trade-off between safety and utility.
- **Binary Helpfulness Rate**: To account for the privacy-utility trade-off, we follow and report the adjusted leakage rate LR_h, which is computed only over cases where the model's final action receives a helpfulness score ≥ 2 (on a 0–3 scale). This metric reflects privacy leakage when the model is being usefully helpful.
- **Average Helpfulness Score:** Each final action is rated on a 0–3 scale based on the final task completion quality.

Information Propagation Metrics To quantify how well each stage Extractor, Checker, and Executor balances privacy preservation and public completeness, we track two rates:

Setting	Information Flow	Leaks Secret ↓	Omits Public Information ↓	Leaks Secret or Omits Info ↓
LLaMA-3.1-70B-Instruct				
Single Agent		29.5 ± 4.9	23.5 ± 8.7	48.5 ± 7.3
Single Agent (CoT)		3.0 ± 0.0	27.5 ± 7.7	29.5 ± 7.7
Two Agents	Annotate Private	15.0 ± 1.8	23.0 ± 8.8	34.5 ± 7.9
Two Agents	Public Only	6.0 ± 1.8	20.0 ± 7.9	24.5 ± 7.5
Two Agents	Annotate Private (w/o meeting transcript)	20.5 ± 4.0	28.0 ± 8.7	41.0 ± 8.2
Two Agents	Public Only (w/o meeting transcript)	0.5 ± 0.5	47.0 ± 7.6	47.5 ± 7.5
Three Agents	Annotate Private	24.0 ± 3.5	23.5 ± 8.2	42.0 ± 7.5
Three Agents	Public Only	3.0 ± 1.3	**19.5 ± 7.4**	**20.5 ± 7.3**
Three Agents	Annotate Private (w/o meeting transcript)	13.0 ± 2.8	23.5 ± 8.7	31.5 ± 8.5
Three Agents	Public Only (w/o meeting transcript)	9.0 ± 1.9	26.0 ± 8.3	32.0 ± 7.5
GPT-4o				
Single Agent		23.0 ± 1.3	12.0 ± 4.4	26.3 ± 4.9
Single Agent (CoT)		2.0 ± 0.9	32.5 ± 7.0	34.0 ± 6.9
Two Agents	Annotate Private	7.0 ± 3.5	15.0 ± 4.4	21.5 ± 6.2
Two Agents	Public Only	11.0 ± 5.0	11.5 ± 4.5	22.0 ± 5.9
Two Agents	Annotate Private (w/o meeting transcript)	6.0 ± 2.8	20.5 ± 4.5	26.0 ± 5.9
Two Agents	Public Only (w/o meeting transcript)	12.0 ± 5.7	13.0 ± 3.2	24.5 ± 6.7
Three Agents	Annotate Private	**5.0 ± 2.9**	20.0 ± 5.2	24.5 ± 6.7
Three Agents	Public Only	15.0 ± 7.1	9.0 ± 3.2	23.5 ± 7.9
Three Agents	Annotate Private (w/o meeting transcript)	8.5 ± 4.0	10.0 ± 3.8	**17.5 ± 5.8**
Three Agents	Public Only (w/o meeting transcript)	15.0 ± 6.9	**8.5 ± 3.2**	23.0 ± 7.0

Table 2: Information Flow Results. Annotate Private means the checker gives all information and annotates the privacy to the executor; Public Only means the checker only gives public information to the executor; without a meeting transcript means the executor can not access the meeting transcript.

- **Leaks**$_s$ – the percentage of transcripts that still contain *any* private information after stage s.
- **Public**$_s$ – the percentage of transcripts that retain *all* required public content after stage s.

We summarise both aspects with a single **composite score**

$$Q_s = (100 - \textit{Leaks}_s) + \textit{Public}_s \quad \in [0, 200],$$

where a higher Q_s indicates fewer privacy leaks and more complete public information.

4.3 Experimental Details

We first explored information propagation with GPT-4o [29] and LLaMA-3.1-70B-Instruct [12] in both single-agent and multi-agent settings to measure how different propagation strategies affect secret leakage. From this, we selected the most effective flow and then evaluated it across six models: o3 [28], o4-mini [28], GPT-4.1 [27], GPT-4o [26], and two open-source models: LLaMA-3.1-70B-Instruct[12] and LLaMA-3.1-8B-Instruct [12] under three-agent public-only and privacy annotation checker configurations on PrivacyLens and ConfAIde, comparing these results to single-agent baselines.

5 Experiment Results & Analysis

5.1 Multi-Agent and Information Flow

To systematically evaluate how different agent configurations balance privacy preservation and completeness of public content, we conducted experiments on single-agent, two-agent, and three-agent pipelines with different information flow configurations. We show our information flow exploration results in Table 2. We additionally include a Chain-of-Thought single-agent baseline, which follows the same instructions as the three-agent setup, which detecting events, classifying them as private or public, and generating meeting summaries.

Single-Agent vs. Multi-Agent Baselines: Our results are shown as Table 2, the multi-agent framework balances secret leakage with public information omission, enhancing both data security and retention. Across both backbones, multi-agent pipelines substantially reduce our composite error (*Leaks Secret or Omits Info*↓) relative to single-agent baselines. For LLaMA-3.1-70B-Instruct, the best configuration is a three-agent, the information flow that the checker only delivers public information (20.5), improving over the single-agent baseline by -28.0. For GPT-4o, the best configuration is a

Comparison (A vs. B)	Outcome	Δpp (B–A)	Matched OR	Holm $p_{\text{two-sided}}$	Sig.
LLaMA-3.1-70B-Instruct					
single agent vs. three agent (public only)	Leakage of private info ↓	–26.5 pp	11.6 [4.65, 28.92]	< 0.001	✓
	Omission of public info ↓	–4.0 pp	2.33 [0.90, 6.07]	0.115	✗
three agent (privacy annotate) vs. three agent (public only)	Leakage of private info ↓	21.0 pp	11.5 [4.14, 31.95]	< 0.001	✓
	Omission of public info ↓	4.0 pp	2.14 [0.874, 5.256]	0.134	✗
GPT-4o					
single agent vs. three agent (public only)	Leakage of private info ↓	–12.0 pp	0.1 [0.02, 0.33]	< 0.001	✓
	Omission of public info ↓	3.0 pp	1.5 [0.72, 3.11]	0.362	✗
three agent (privacy annotate) vs. three agent (public only)	Leakage of private info ↓	–10.0 pp	0.2 [0.06, 0.48]	< 0.001	✓
	Omission of public info ↓	+11.0 pp	2.70 [1.42, 5.09]	0.002	✓

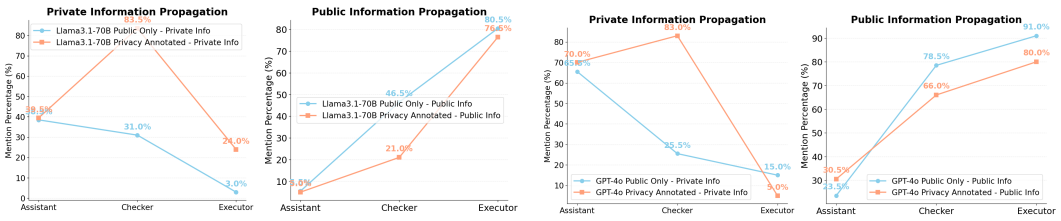
Table 3: Paired exact McNemar test comparing **A (single-agent)** and **B (three-agent, public only)**; **A (three-agent, privacy annotate)** and **B (three-agent, public only)** on **GPT-4o** and **LLaMA-3.1-70B-Instruct** under ConfAIde Tier 4 settings. Each pair is treated as a matched observation. We conduct two-sided McNemar significance tests on discordant outcome pairs. Significance is corrected for multiple comparisons via Holm's step-down procedure (α=0.05), and effect sizes are reported as matched odds ratios with 95% Wald confidence intervals. Δpp denotes percentage-point difference.

three-agent, Annotate-Private, no transcript flow (17.5), improving over the single-agent baseline (26.3) by -8.8. Notably, moving from two to three agents yields additional gains for both models (LLaMA: 24.5 → 20.5; GPT-4o: 21.5 → 17.5 depending on flow), indicating that introducing a dedicated checker and clearly separating agent duties improves the privacy utility trade-off provided that each role is well-defined and properly aligned with the task structure.

Impact of Information Flow Design and Transcript Access Our quantitative information flow experiment results reveal a nuanced interplay between the type of information flow (*Public Only* vs. *Annotate Private*) and whether the executor agent receives access to the original meeting transcript.

For LLaMA-3.1-70B-Instruct, providing the executor with the meeting transcript leads to a clear advantage for the Public Only strategy: both in the two-agent (Composite error: 24.5) and three-agent (Composite: 20.5) settings, this approach outperforms Annotate Private (34.5 and 42.0, respectively). Here, restricting the checker to only forward public information, combined with transcript access, allows the executor to recover useful details and reduces the risk of private information leakage. Importantly, the *Public Only* strategy also simplifies the executor's task by eliminating the need for the model to further distinguish between private and public content during its own processing. However, in this setting, if the meeting transcript is withheld from the executor, the performance of Public Only degrades sharply, where omissions of public information increase substantially (e.g., three-agent omission: 19.5 with transcript vs. 26.0 without), and composite errors rise (three-agent: 20.5 → 32.0). This suggests that LLaMA relies heavily on the transcript to supplement the limited information from the checker, and without it, relevant content is easily lost.

For GPT-4o, the interplay is more balanced. With transcript access, Annotate Private achieves the lowest privacy leakage (5.0), while Public Only achieves the lowest public omission (9.0). When the transcript is withheld, however, the three-agent Annotate Private setting yields the best overall performance (Composite: **17.5**), with much lower omission (10.0) than Public Only without transcript (8.5), and still reasonable control of leakage (8.5 vs. 15.0). This indicates that the stronger GPT-4o executor can effectively use checker tags to reconstruct public content, even without seeing the original transcript, and is less prone to privacy leaks than LLaMA in the same setup.



(a) LLaMA-3.1-70B-Instruct (b) GPT-4o

Figure 2: Comparison of private and public information propagation across agent stages under Public Only and Privacy Annotated settings for both LLaMA-3.1-70B-Instruct and GPT-4o models.

In summary, whether to give the executor the original transcript interacts strongly with information flow design: (1) For weaker models like LLaMA, *Public Only* plus transcript access is crucial for minimizing overall error, otherwise, omission rates spike. (2) For stronger models like GPT-4o, withholding the transcript in an Annotate Private setup allows the executor to focus on the checker-provided structure, effectively balancing privacy and public content retention.

5.2 Three Agent Information Propagation

To further understand the conditions under which the three-agent framework improves privacy preservation, we quantitatively analyze how information flows across agents and where privacy risks are introduced or mitigated.

Executor Without Meeting Transcript Our results demonstrate that the Extractor agent alone is insufficient for reliably identifying public information, necessitating a dedicated Checker agent for effective public information restoration. In this respect, GPT-4o outperforms Llama3.1-70B (55.0% vs. 46.5%) in recovering public content, and the Executor also contributes to the restoration process.

The Checker agent highlights the pronounced superiority of GPT-4o in privacy preservation compared to Llama3.1-70B, as evidenced by a substantially lower privacy leakage rate (33.0% vs. 79.5%). Moreover, GPT-4o achieves a much lower overall information flow leakage rate across the entire pipeline (20.5% vs. 67.0%), which directly substantiates our claim and elucidates why Llama3.1-70B, when used as a base model without providing the meeting transcript to the Executor, suffers significant performance degradation. In contrast, GPT-4o's performance remains stable,or even improves, under these constraints, illustrating the effectiveness of the Checker's filtering mechanism and reducing dependence on the downstream agent's ability to reconstruct prior context.

Overall, the composite quality score further reinforces this pattern: while both models exhibit comparable performance at the Extractor stage, Llama3.1-70B's quality declines substantially at the Checker stage, whereas GPT-4o not only maintains but improves its performance, with a further increase observed at the Executor stage.

Compare Privacy Annotated & Public Only To further dissect the role of checker annotation strategies, we conducted a detailed information propagation analysis, comparing Privacy Annotated and Public Only settings across agent stages, and examined how annotation guidance influences privacy preservation and content retention patterns. Statistical significance test results are in Table 3.

For Llama-3.1-70B-Instruct, our results see(Figure 2) reveal a striking difference: the proportion of private information included in the Public Only pipeline is substantially higher than in the Privacy Annotated configuration (83.5% vs. 31.0%). This suggests that without explicit privacy annotations, the Public Only setting often fails to sufficiently filter sensitive content. For public information, as expected, the Public Only setting consistently yields higher mention rates across all agent stages compared to the Privacy Annotated setting. In contrast, GPT-4o exhibits an even more pronounced effect (see Figure 2): when the checker operates in Privacy Annotated mode, the rate of private information mentions is much higher than in Public Only (83% vs. 25.5%). Similarly, for public information, the Public Only setting again achieves higher mention rates than Privacy Annotated.

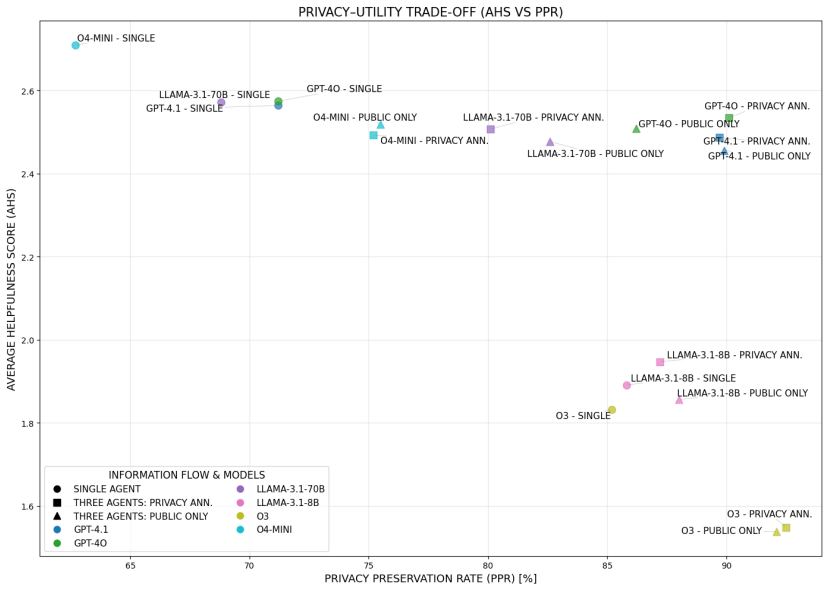


Figure 3: PrivacyLens benchmark results for six LLMs under single-agent and three-agent settings, showing Binary Helpfulness Rate, Average Helpfulness Score, Privacy Preservation Rate, and Adjusted Information Preservation Rate (higher is better). Privacy Ann. means privacy-annotation setting, Public Only means public-only setting.

McNemar tests (see Table 3) indicate that private-leakage differences between Public Only and Privacy Annotated are consistently statistically reliable across both backbones and comparisons, whereas public-omission effects are weaker and model-dependent: for Llama-3.1-70B-Instruct they are not statistically significant under our matching scheme, while for GPT-4o the Annotated vs. Public Only contrast shows a significant difference in public-content retention.

These results highlight the importance of using a strong model as the checker for effective privacy filtering, while also suggesting that multi-agent pipelines should adaptively assign stronger models to critical validation tasks, with additional verification steps as needed. If the executor is high-capacity and the transcript is hidden, prefer Privacy-Annotated; otherwise prefer Public-Only. Pair either choice with an explicit checker and adversarial leakage tests.

5.3 Three Agent In Privacylens

Leveraging the annotate-private and public-only configuration, which in prior experiments yielded the optimal privacy–utility trade-off, we extend the evaluation to Privacylens which provide agent base performance of our methods. Firstly, we assess its efficacy across six models to examine how information propagates through our three-agent pipeline under varying capacities. In our analysis, we mainly analyse the single-agent vs multi-agent setup.

Figure 3 presents the privacy–utility Pareto frontier across all evaluated configurations. We observe a consistent pattern: multi-agent information flows substantially increase privacy preservation with only minor reductions in helpfulness. For both GPT-4.1 and GPT-4o, moving from Single agent to Privacy Annotation or Public Only achieves PPR gains of 15–19 percentage points while maintaining high AHS (from 2.56 to 2.53). Notably, the Privacy Annotation variant offers the best overall Pareto-optimal balance, achieving both high utility and robust privacy protection.

Llama-3.1-70B displays a similar trend: the Privacy Annotation pipeline improves PPR from 68.8% (Single) to 80.1%, with only a modest decrease in AHS (2.57 $\rightarrow$ 2.51). For resource-constrained settings, o4-mini achieves the highest AHS (2.71) in Single-agent mode but at the cost of much lower privacy (PPR 62.7%). Applying multi-agent strategies (PA/PO) recovers privacy performance (PPR $\sim$75%) with only a small utility loss. The o3 model, while attaining the highest PPR ($>$92%) with multi-agent flows, suffers from the lowest AHS ($<$1.6), underscoring the challenge of maintaining helpfulness at extreme privacy levels.

Overall, most results show the benefit of multi-agent, which dominates the trade-off space for moderate to high privacy regimes (PPR $>75\%$). These findings suggest that well-designed multi-agent privacy architectures can robustly enhance privacy with minimal impact on end-user experience.

6 Conclusion

We introduced a multi-agent approach that partitions the tasks of event extraction, classification, and final summary generation among separate agents, addressing the limitations of a single LLM operating alone. Experimental results and ablation studies confirm that our multi-agent pipelines significantly reduce private information leakage without substantially compromising the quality. Notably, we observed performance differences across various information flow configurations and uncovered the multi-agent system's capacity for self-correction. Our findings highlight the importance of modular, intermediate validation steps in complex, context-dependent scenarios and provide evidence that the choice of checker model (capacity and alignment) and the information flowing materially influence the privacy–utility trade-off, offering practical guidance for multi-agent system design.

7 Limitation

Despite the improvements observed in privacy preservation and content fidelity, our multi-agent framework has several limitations from both technological and methodological perspectives:

Increased Computational and Integration Overhead. Running multiple agents sequentially consumes more computational resources than a single-pass model. Orchestrating prompts, storing intermediate states, and integrating outputs imposes additional engineering complexity. In real-world applications with tight latency constraints, such as live customer support or streaming meeting transcripts. This overhead may be impractical without careful optimization or more advanced parallelization strategies.

Limited Domain Generalization. Our experiments focus on meeting summarization and privacy scenarios defined by the ConfAIde [22] and PrivacyLens [33] benchmark. Although our multi-agent architecture demonstrates strong performance in these controlled benchmarks, domain adaptation remains non-trivial. The Extractor's event schema and the Checker's privacy rules must be hand-crafted for each application domain—medical, legal, or financial—each of which has distinct notions of sensitive versus permissible disclosures. This tailoring demands specialized prompt engineering, rule design, and potential collaboration with domain experts. To date, no publicly available benchmarks systematically evaluate multi-agent or information-flow architectures for privacy in domains such as healthcare, finance, or legal document processing. Future work should prioritize building such benchmarks, defining domain-specific privacy norms which would help to better showing the generalizability of our multi-agent system.

References

[1] Noura Abdi, Xiao Zhan, Kopo M. Ramokapane, and Jose M. Such. 2021. Privacy Norms for Smart Home Personal Assistants. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*.

[2] Marwa Abdulhai, Gregory Serapio-Garcia, Clément Crepy, Daria Valter, John Canny, and Natasha Jaques. 2023. Moral Foundations of Large Language Models. arXiv:2310.15337 [cs.AI] https://arxiv.org/abs/2310.15337

[3] Eugene Bagdasarian, Ren Yi, Sahra Ghalebikesabi, Peter Kairouz, Marco Gruteser, Sewoong Oh, Borja Balle, and Daniel Ramage. 2024. AirGapAgent: Protecting Privacy-Conscious Conversational Agents. arXiv:2405.05175 [cs.CR] https://arxiv.org/abs/2405.05175

[4] A. Barth, A. Datta, J. C. Mitchell, and H. Nissenbaum. 2006. Privacy and Contextual Integrity: Framework and Applications. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*.

[5] Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. 2022. What Does it Mean for a Language Model to Preserve Privacy? arXiv:2202.05520 [stat.ML] https://arxiv.org/abs/2202.05520

[6] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying Memorization Across Neural Language Models. arXiv:2202.07646 [cs.LG] https://arxiv.org/abs/2202.07646

[7] Junzhe Chen, Xuming Hu, Shuodi Liu, Shiyu Huang, Wei-Wei Tu, Zhaofeng He, and Lijie Wen. 2024. LLMArena: Assessing Capabilities of Large Language Models in Dynamic Multi-Agent Environments. arXiv:2402.16499 [cs.CL] https://arxiv.org/abs/2402.16499

[8] Zhao Cheng, Diane Wan, Matthew Abueg, Sahra Ghalebikesabi, Ren Yi, Eugene Bagdasarian, Borja Balle, Stefan Mellem, and Shawn O'Banion. 2024. CI-Bench: Benchmarking Contextual Integrity of AI Assistants on Synthetic Data. arXiv:2409.13903 [cs.AI] https://arxiv.org/abs/2409.13903

[9] Denis Emelin, Ronan Le Bras, Jena D. Hwang, Maxwell Forbes, and Yejin Choi. 2020. Moral Stories: Situated Reasoning about Norms, Intents, Actions, and their Consequences. arXiv:2012.15738 [cs.CL] https://arxiv.org/abs/2012.15738

[10] Wei Fan, Haoran Li, Zheye Deng, Weiqi Wang, and Yangqiu Song. 2024. GoldCoin: Grounding Large Language Models in Privacy Laws via Contextual Integrity Theory. arXiv:2406.11149 [cs.CL] https://arxiv.org/abs/2406.11149

[11] Sahra Ghalebikesabi, Eugene Bagdasaryan, Ren Yi, Itay Yona, Ilia Shumailov, Aneesh Pappu, Chongyang Shi, Laura Weidinger, Robert Stanforth, Leonard Berrada, Pushmeet Kohli, Po-Sen Huang, and Borja Balle. 2024. Operationalizing Contextual Integrity in Privacy-Conscious Assistants. arXiv:2408.02373 [cs.AI] https://arxiv.org/abs/2408.02373

[12] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, and et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783 [cs.AI] https://arxiv.org/abs/2407.21783

[13] Florian Hartmann, Duc-Hieu Tran, Peter Kairouz, Victor Cărbune, and Blaise Aguera y Arcas. 2024. Can LLMs get help from other LLMs without revealing private information? arXiv:2404.01041 [cs.LG] https://arxiv.org/abs/2404.01041

[14] Dan Hendrycks, Mantas Mazeika, Andy Zou, Sahil Patel, Christine Zhu, Jesus Navarro, Dawn Song, Bo Li, and Jacob Steinhardt. 2022. What Would Jiminy Cricket Do? Towards Agents That Behave Morally. arXiv:2110.13136 [cs.LG] https://arxiv.org/abs/2110.13136

[15] Chani Jung, Dongkwan Kim, Jiho Jin, Jiseon Kim, Yeon Seonwoo, Yejin Choi, Alice Oh, and Hyunwoo Kim. 2024. Perceptions to Beliefs: Exploring Precursory Inferences for Theory of Mind in Large Language Models. arXiv:2407.06004 [cs.CL] https://arxiv.org/abs/2407.06004

[16] Nadin Kökciyan. 2016. Privacy Management in Agent-Based Social Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 4299–4300.

[17] Haoran Li, Yulin Chen, Jinglong Luo, Jiecong Wang, Hao Peng, Yan Kang, Xiaojin Zhang, Qi Hu, Chunkit Chan, Zenglin Xu, Bryan Hooi, and Yangqiu Song. 2024. Privacy in Large Language Models: Attacks, Defenses and Future Directions. arXiv:2310.10383 [cs.CL] https://arxiv.org/abs/2310.10383

[18] Haoran Li, Wei Fan, Yulin Chen, Jiayang Cheng, Tianshu Chu, Xuebing Zhou, Peizhao Hu, and Yangqiu Song. 2025. Privacy Checklist: Privacy Violation Detection Grounding on Contextual Integrity Theory. arXiv:2408.10053 [cs.CL] https://arxiv.org/abs/2408.10053

[19] Xuechen Liang, Meiling Tao, Yinghui Xia, Tianyu Shi, Jun Wang, and JingSong Yang. 2024. CMAT: A Multi-Agent Collaboration Tuning Framework for Enhancing Small Language Models. arXiv:2404.01663 [cs.CL] https://arxiv.org/abs/2404.01663

[20] Yuanyuan Liang, Tingyu Xie, Gan Peng, Zihao Huang, Yunshi Lan, and Weining Qian. 2024. NAT-NL2GQL: A Novel Multi-Agent Framework for Translating Natural Language to Graph Query Language. arXiv:2412.10434 [cs.CL] https://arxiv.org/abs/2412.10434

[21] Kirsten Martin and Helen Nissenbaum. 2015. Measuring Privacy: An Empirical Test Using Context to Expose Confounding Variables. (December 2015). Unpublished manuscript.

[22] Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. 2023. Can LLMs Keep a Secret? Testing Privacy Implications of Language Models via Contextual Integrity Theory. *arXiv preprint arXiv:2310.17884* (2023).

[23] Niloofar Mireshghallah and Tianshi Li. 2025. Position: Privacy Is Not Just Memorization! *arXiv preprint arXiv:2510.01645* (2025).

[24] Ivoline Ngong, Swanand Kadhe, Hao Wang, Keerthiram Murugesan, Justin D. Weisz, Amit Dhurandhar, and Karthikeyan Natesan Ramamurthy. 2025. Protecting Users From Themselves: Safeguarding Contextual Privacy in Interactions with Conversational Agents. arXiv:2502.18509 [cs.CR] https://arxiv.org/abs/2502.18509

[25] Helen Nissenbaum. 2004. Privacy as Contextual Integrity. *Washington Law Review* 79, 1 (2004), 119–158. https://digitalcommons.law.uw.edu/wlr/vol79/iss1/10 Symposium.

[26] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, and et al. 2024. GPT-4o System Card. arXiv:2410.21276 [cs.CL] https://arxiv.org/abs/2410.21276

[27] OpenAI. 2025. Introducing GPT-4.1 in the API. https://openai.com/index/gpt-4-1/. Accessed: 2025-04-14.

[28] OpenAI. 2025. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/. Accessed: 2025-04-16.

[29] OpenAI, Josh Achiam, Steven Adler, and et al. 2024. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL] https://arxiv.org/abs/2303.08774

[30] Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2023. Gorilla: Large Language Model Connected with Massive APIs. arXiv:2305.15334 [cs.CL] https://arxiv.org/abs/2305.15334

[31] Aman Priyanshu, Supriti Vijay, Ayush Kumar, Rakshit Naidu, and Fatemehsadat Mireshghallah. 2023. Are Chatbots Ready for Privacy-Sensitive Applications? An Investigation into Input Regurgitation and Prompt-Induced Sanitization. arXiv:2305.15008 [cs.CL] https://arxiv.org/abs/2305.15008

[32] Zhenting Qi, Hanlin Zhang, Eric Xing, Sham Kakade, and Himabindu Lakkaraju. 2024. Follow My Instruction and Spill the Beans: Scalable Data Extraction from Retrieval-Augmented Generation Systems. arXiv:2402.17840 [cs.CL] https://arxiv.org/abs/2402.17840

[33] Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. 2025. PrivacyLens: Evaluating Privacy Norm Awareness of Language Models in Action. arXiv:2409.00138 [cs.CL] https://arxiv.org/abs/2409.00138

[34] Yan Shvartzshnaider and Vasisht Duddu. 2025. Position: Contextual Integrity is Inadequately Applied to Language Models. arXiv:2501.19173 [cs.CY] https://arxiv.org/abs/2501.19173

[35] Yan Shvartzshnaider, Zvonimir Pavlinovic, Ananth Balashankar, Thomas Wies, Lakshminarayanan Subramanian, Helen Nissenbaum, and Prateek Mittal. 2019. Vaccine: Using Contextual Integrity for Data Leakage Detection. In *Proceedings of The World Wide Web Conference (WWW)*. 1702–1712.

[36] Yan Shvartzshnaider, Schrasing Tong, Thomas Wies, Paula Kift, Helen Nissenbaum, Lakshminarayanan Subramanian, and Prateek Mittal. 2016. Learning Privacy Expectations by Crowdsourcing Contextual Informational Norms. In *Proceedings of the Conference on Human Computation and Crowdsourcing (HCOMP)*.

[37] Daniel J. Solove. 2023. Data Is What Data Does: Regulating Use, Harm, and Risk Instead of Sensitive Data. *Harm, and Risk Instead of Sensitive Data* (2023). January 11, 2023.

[38] Yashar Talebirad and Amirhossein Nadiri. 2023. Multi-Agent Collaboration: Harnessing the Power of Intelligent LLM Agents. arXiv:2306.03314 [cs.AI] https://arxiv.org/abs/2306.03314

[39] Qian Wang, Tianyu Wang, Zhenheng Tang, Qinbin Li, Nuo Chen, Jingsheng Liang, and Bingsheng He. 2025. MegaAgent: A Large-Scale Autonomous LLM-based Multi-Agent System Without Predefined SOPs. arXiv:2408.09955 [cs.MA] https://arxiv.org/abs/2408.09955

[40] Shang Wang, Tianqing Zhu, Bo Liu, Ming Ding, Xu Guo, Dayong Ye, Wanlei Zhou, and Philip S Yu. 2024. Unique security and privacy threats of large language model: A comprehensive survey. *arXiv preprint arXiv:2406.07973* (2024).

[41] Xiaoyuan Wu, Roshni Kaushik, Wenkai Li, Lujo Bauer, and Koichi Onoue. 2026. User Perceptions vs. Proxy LLM Judges: Privacy and Helpfulness in LLM Responses to Privacy-Sensitive Scenarios. arXiv:2510.20721 [cs.CL] https://arxiv.org/abs/2510.20721

[42] Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. WildChat: 1M ChatGPT Interaction Logs in the Wild. arXiv:2405.01470 [cs.CL] https://arxiv.org/abs/2405.01470