Summary of Changes

Note: This document uses a quotation environment to indicate changes in the revised `audagent.pdf` file. Within the quotation environment, page numbers and sections are indicated in gray color, changes are highlighted in blue color.

1 Checklist of the Revision Plan

We addressed all the listed items in the revision plan. Specifically, the revision plan is associated with sections in this document as follows:

1. Five-element policy model
   - We will explicitly clarify that the proposed five-element policy model is designed for GDPR-style privacy policy templates and may not capture all possible privacy policy structures. We will clearly delineate which types of phrases/structures are supported/extensible, and which scenarios may fall outside the scope of the model. (In Preliminaries and Discussions) → Section 2.1.1
   - We will add quantitative statistics to evaluate the coverage (completeness) of the extracted policy elements w.r.t. privacy policies' semantics. (In Appendix) → Section 2.1.2
   - We will clarify that many privacy policies govern all products of a company, including AI agent products, and will provide concrete policy sentences that explicitly or implicitly apply to AI agents. (In Discussions) → Section 2.1.3
   - We will further explain how AudAgent processes coarse-grained privacy policies and refines them through ontology-based expansion by an example. (In Discussions) → Section 2.1.4
2. Multi-LLM voting
   - We will justify the independence assumption among LLMs and provide supporting empirical evidence. (In Section 3.1) → Section 2.2.1
   - We will add a comparison between multi-LLM voting and single-LLM high-temperature sampling. (In Appendix) → Section 2.2.2
   - We will discuss possible strategies when LLM consensus is low, e.g. fallback to a most-extracted LLM. (In Discussions) → Section 2.2.3
3. Human-centered evaluation
   - We will weaken the sense of HCI contribution and clarify that usability is a secondary contribution rather than a central focus of the paper. (In Abstract and Introduction) → Section 2.3.1
   - We will add additional case studies involving sensitive data types. In particular, we show how AudAgent can detect composite sensitive-data patterns and warns users about elevated privacy risks. (In Appendix) → Section 2.3.2
   - We will incorporate interface refinements, such as tooltips and warning features, to improve the clarity and usability. (In Appendix) → Section 2.3.3
4. Other comments
   - Reviewer D's other comments → Section 2.4.1
   - Reviewer A's other comments → Section 2.4.2

- Reviewer B's other comments → Section 2.4.3
- Reviewer C's other comments → Section 2.4.4

5. Updated related work on AI agents' privacy and security. (In Related Work) → Section 2.5

2 Changes Made in `audagent.pdf`

This section indicates the changes made in the revised `audagent.pdf` file, organized by the items in the revision plan.

2.1 Five-Element Policy Model

2.1.1 Expressiveness of the Privacy Policy Model

We delineated which types of phrases/structures in privacy policies are supported/extensible, and which scenarios may fall outside the scope of the model.

> (Preliminaries - Privacy Policies, Page 3) This five-element tuple captures the structure of GDPR-style privacy policies [1, 2] as well as templates produced by common privacy-policy generation tools [3, 4]. For example, the first four sections of OpenAI's privacy policy [5] map directly to the components defined above.*...

> (Discussions, Page 9) *Expressiveness of the Privacy Policy Model.* AudAgent translates natural-language privacy policies into a formal model represented as five-element tuples (Definition 1). During this translation, some information may be omitted or certain nuances in the original text may not be fully preserved. This raises questions about the privacy policy model's expressiveness:
>
> - Which phrases or structures in natural-language privacy policies can be accurately represented by this model?
> - What limitations does the model have in capturing the precise semantics of privacy policies?
> - How can the model be extended to support a broader range of policy phrases without compromising auditability?
>
> As an illustrative example, Appendix C.6 presents a detailed annotation and summary statistics for OpenAI's privacy policy [5] (Feb. 6, 2026 version) using the five-element tuple.
>
> **Capability.** The five-element tuple captures the core components commonly specified in GDPR-style privacy policies, including collected data types, collection conditions, purposes, disclosure, and retention periods. (i) Accordingly, statements that explicitly mention these components can be represented by mapping the mentioned data types to the corresponding conditions in the tuple. Syntactic variations (e.g. passive voice, capitalization) are normalized by LLMs during parsing via prompting. (ii) Even when these components appear in separate sections (as is common in practice), they remain representable as long as the policy provides sufficient cues to link data types with their associated conditions; In this case, LLMs can match relevant terminology across sections, sometimes producing multiple tuples with overlapping semantics. (iii) When data types are described at a coarse granularity (e.g. "Geolocation"), they will be refined into fine-grained categories (e.g. "IP address", "Precise geolocation") using ontology graphs (Figure 2) when auditing.
>
> **Limitations.** (i) The privacy policy model uses simplified condition sets, $C^{\text{col}} = \{\text{direct}, \text{indirect}\}$ and $C^{\text{pro}} = \{\text{relevant}, \text{irrelevant}\}$, to represent collection and processing conditions. This simplification can lead to information loss for complex collection contexts or fine-grained processing purposes. For example, the following excerpt from OpenAI's privacy policy on location information [5] is marked up with captured data types (pink underline), collection conditions (orange underline), and processing conditions (gray underline) by the first-stage prompt in Figure 7:

*Although OpenAI's privacy policy describes data types and their associated conditions across separate sections, they can easily be paired by matching related terminology. Section 4 further details how OpenAI's policy maps to the five-element tuple in Definition 1 and discusses the expressiveness of Definition 1.

> Location Information: We determine the general area from which your device accesses our Services based on information like its IP address for security reasons and to make your product experience better, for example to protect your account by detecting unusual login activity or to provide more accurate responses. ...

The second-stage prompt in Figure 8 then normalizes the collection condition to "direct" and the processing conditions to "relevant". While these labels largely preserve the intended meaning, they may not capture the full nuance of the original text. (ii) Some policy statements are underspecified at the natural-language level. For instance, "In determining these retention periods, we consider a number of factors, such as: the potential risk ..." is too vague to be translated into actionable tuples.

Extensibility. The five-element tuple can be extended to cover a broader range of policy language by introducing additional fields or adopting more expressive representations. (i) Negative statements (e.g. "We do not share your data with third parties") are already supported in the current implementation of AudAgent (Figure 6). (ii) Conditional statements (e.g. "We may collect location information when you log in") could be represented by extending C^{col} with indicators for login status. Because such conditions are often task-specific, we do not include them in the current implementation.

2.1.2 An Example and Statistics for the Coverage of Extracted Policy Elements

We added an example to illustrate the coverage of extracted policy elements (Figure 1) and provided statistics on the coverage of extracted tuples for OpenAI's privacy policy.

> (Appendix C.6, Page 20) *OpenAI's Privacy Policy Captured by the Auto-formalization.* To illustrate how AudAgent captures real-world privacy policies, Figure 1 shows the auto-formalization of OpenAI's privacy policy [5] produced with ChatGPT-5.2.[†] We first feed the policy text into the stage-1 prompt in Figure 7 to obtain a low-loss structured policy model (shown as colored text spans in Figure 9). We then apply the stage-2 prompt in Figure 8 to normalize this model into the five-element tuples used by AudAgent's runtime auditing (shown as colored tuples in Figure 9).
>
> **Statistics of the auto-formalized policy model.** We compare the final auditing policy model against the original policy text to understand the coverage of AudAgent's auto-formalization. Concretely, we count the number of *fine-grained* elements in the original policy text and compare it with the number of fine-grained elements in the final auditing model after auto-formalization and ontology mapping. For readability, we omit explicitly linking each condition to each data type; instead, we report the number of *unique* elements to measure the coverage of the policy text. The results are as follows:
>
> - Policy text: d^{col}: 29; c^{col}: 3; c^{pro}: 6; c^{dis}: 11; c^{ret}: 3.
> - Auditing model: d^{col}: 17; c^{col}: 2; c^{pro}: 2; c^{dis}: 8; c^{ret}: 1.
>
> Among these, (i) the d^{col} elements that appear in the policy text but not in the auditing model are largely not intrinsically sensitive (e.g. "prompts, files, images, audio, video", and "survey responses"). (ii) The three c^{col} categories in the policy text correspond to "You Provide", "We Receive from Your Use of the Services", and "We Receive from Other Sources", while the two c^{col} categories in the auditing model are simplified to "direct" and "indirect". (iii) The c^{pro} categories in the policy text beyond the auditing model include items such as "To Communicate with You", "For Research and Development", and "For Safety and Security", while the two c^{pro} categories in the auditing model are simplified to "relevant" and "irrelevant". (iv) The c^{dis} categories in the policy text beyond the auditing model include "hosting services", "customer service", and "our affiliates" etc. (v) Finally, the c^{ret} categories in the policy text beyond the auditing model include "deleted automatically" and "30 days (for temporary chats)".

2.1.3 How Privacy Policies Apply to AI Agents

We clarified how privacy policies govern AI agents by providing concrete examples.

[†]For readability, we include only the first paragraph of each section of the policy and omit results from other LLMs.

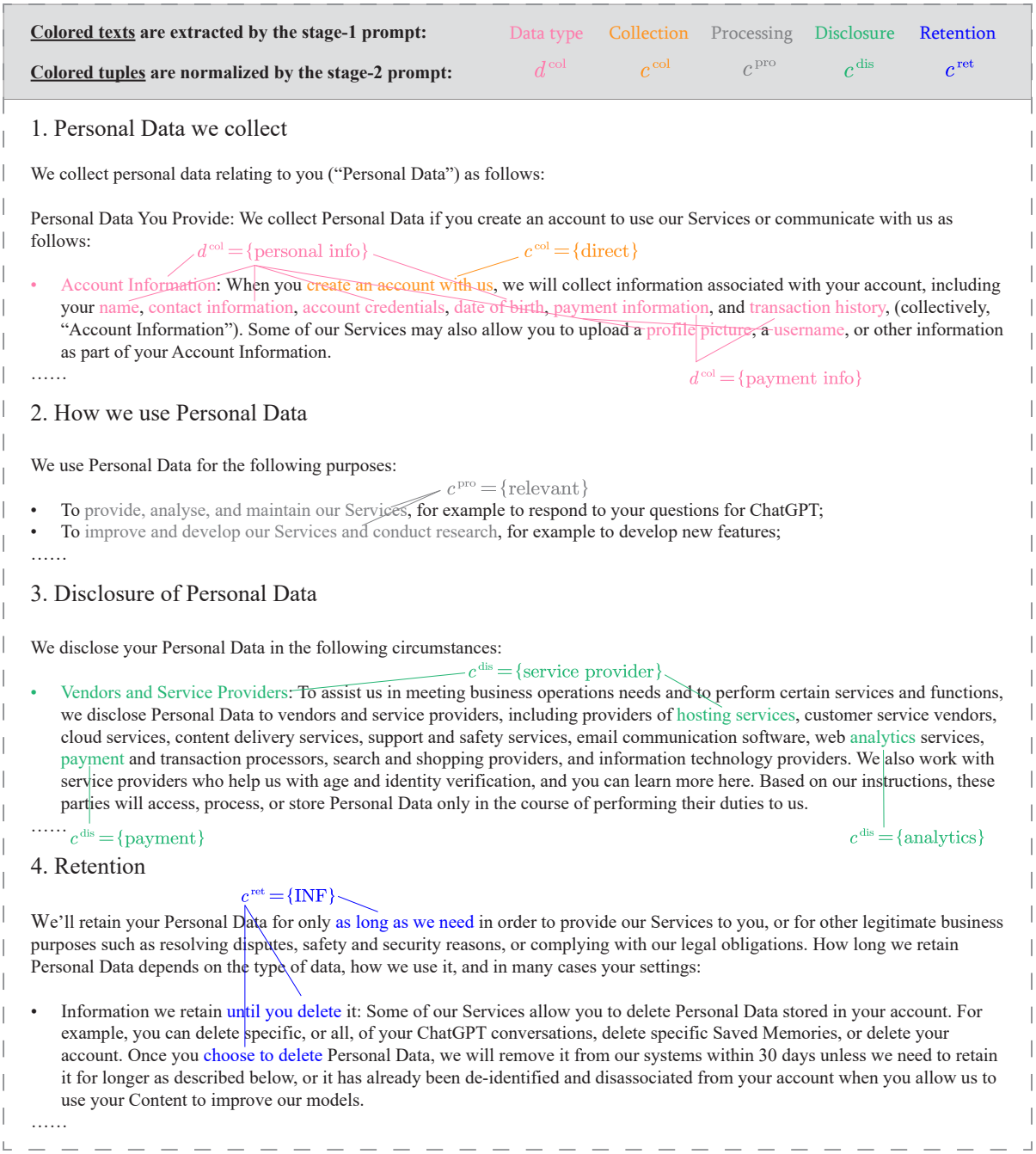

Figure 1: How AudAgent captures OpenAI's privacy policy via auto-formalization. The colored texts highlight the corresponding parts extracted by the stage-1 low-loss formalization. The colored tuples highlight the corresponding parts normalized by the stage-2 formalization.

(Discussions, Page 10) *How Privacy Policies Govern AI Agents.* At present, AI agents' data practices are typically governed by their providers' general privacy policies (originally written for LLMs). Many such policies contain language indicating an intent to cover agent-like interactions. For example, Anthropic's privacy policy [6] (Jan. 12, 2026 version) states that "You are able to interact with our Services in a variety of formats, including but not limited to chat, coding, and agentic sessions . . . "; OpenAI's privacy policy [5] (Feb. 6, 2026 version) states that "We collect information about your use and activity across the Services, such as . . . user agent and version . . . "; and the Gemini app's privacy policy [7] (Jan. 30, 2026 version) includes a dedicated "Gemini Agent" section.

In contrast, independent agent orchestrators such as AutoGen [8] and MCP [9] currently do not publish standalone privacy policies. One recent user-facing (local) agent orchestrator,

OpenClaw [10], provides a brief privacy policy [11] (Jan. 29, 2026 version); however, its described data practices still depend on the underlying LLMs and tools: "When you use cloud AI providers (like Anthropic or OpenAI), your messages go to their servers according to their privacy policies. This is the same as using ChatGPT or Claude directly."

2.1.4 Refining Coarse-Grained Privacy Policies via Ontology-Based Expansion

(Mentioned in Section 2.1.1.) We further explained how AudAgent processes coarse-grained privacy policies to form the d^{col} and c^{pro} elements.

(Discussions, Page 10) ... (iii) When data types are described at a coarse granularity (e.g. "Geolocation"), they will be refined into fine-grained categories (e.g. "IP address", "Precise geolocation") using ontology graphs (Figure 2).

2.2 Multi-LLM Voting

2.2.1 Justification of the Independence Assumption among LLMs

We justified the independence assumption among LLMs and provided supporting references.

(Section 3.1, Page 4) **Likelihood of the independence assumption.** The independence assumption in Theorem 1 is idealized, since different LLMs may share training data and aligned objectives that induce correlations in their outputs. Nevertheless, several factors can still introduce meaningful diversity across models, including differences in architecture, fine-tuning data, alignment procedures, decoding strategies, and system prompts. Empirical studies [12, 13] have observed that different LLMs can produce different answers/judgments on the same input, indicating their outputs are not perfectly correlated, suggesting that independence can be a useful approximation in practice. By contrast, this assumption is less applicable when "diversity" is obtained from a single LLM merely by varying decoding strategies (e.g. temperature), which are more likely to produce correlated outputs. Nonetheless, Appendix C.10 provides an ablation study that compares multi-LLM voting with a single-LLM baseline using high-temperature decoding.

2.2.2 Multi-LLM Voting vs Single-LLM High-Temperature Sampling

(Appendix C.10, Page 23) *Multi-LLM Voting vs High-Temperature Single LLM.* To validate the effectiveness of multi-LLM voting for privacy policy formalization, we compare it against an alternative that samples multiple outputs from a single LLM by using a high temperature. Motivated by the observation that Gemini-2.5-flash generally underperforms other LLMs on formalizing Anthropic's privacy policy, we use this model and this policy text to test whether increased sampling diversity can improve results.

Specifically, we set the temperature of the LLM to the maximum allowed value (2.0), which Google describes as high creativity and output diversity, then run the same two-stage auto-formalization pipeline four times under the same setting, producing four final policy models. We compare each of the four single-LLM outputs against the results from multi-LLM voting. Table 1 reports the results.

Table 1: Gemini with the highest temperature (2.0) for privacy policy formalization. Each cell shows "m/M", where m denotes the number of tuples same as the result from multi-LLM voting, and M denotes the total number of tuples extracted by Gemini.

Policy text	Run 1	Run 2	Run 3	Run 4	Average
Anthropic's[a]	6/7	9/10	6/7	8/9	7.25/8.25

[a] The multi-LLM voting for this policy text has 10 tuples.

We observe that Gemini's four high-temperature runs still yield fewer tuples (8.25 on average) than other LLMs with default temperature settings ($\geq$ 10 tuples on average), suggesting that multi-LLM voting has an advantage in capturing more privacy tuples. Moreover, Gemini's extracted tuples are largely contained in the multi-LLM voting output, indicating

that multi-LLM voting captures the core policy elements that a single LLM (Gemini) can extract.

2.2.3 Strategies for Low LLM Consensus

(Discussions, Page 10) *When Consensus is Low Among LLMs.* In our experience, expert-level LLMs generally produce consistent parses when privacy policies are well-structured, resulting in high consensus under the voting mechanism. Consensus can drop when LLMs diverge in interpreting ambiguous language or when the privacy policy is hard to map onto the policy model even at the natural-language level. In these cases, a practical fallback is to treat the results as low-confidence and route them to manual review or confirmation.

2.3 Human-Centered Evaluation

2.3.1 Clarification of HCI Contribution

(Abstract, Page 1) ... To close this gap, we introduce AudAgent, a ~~visual~~ tool that continuously monitors AI agents' data practices in real time and guards compliance with stated privacy policies.

(Introduction, Page 1) **This paper.** We propose AudAgent, an automated ~~user-facing~~ data-auditing tool that tackles these challenges. ...

(Contribution list, Page 2) To our knowledge, AudAgent is the first tool that enables automated auditing of AI agents' data practices against privacy policy documents. It provides end users ~~visual~~ transparency to verify that their agents' behavior matches stated privacy expectations.

2.3.2 Additional Case Studies

(Appendix C.9, Page 22) *Composite Privacy Rules for AudAgent's Auditing.* It is known that multiple coarse-grained data items (e.g. coarse location) can be combined to infer sensitive identifiers. To address this issue, AudAgent supports composite privacy rules defined as conjunctions ("and" patterns) over multiple data types. For example, users can define a composite rule that triggers when the coarse data types IP address, location, and name co-occur in a trace. When such a pattern is detected, AudAgent issues a warning about the elevated privacy risk, as shown in Figure 2.

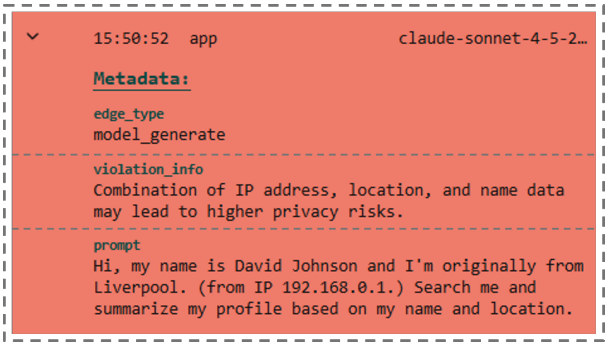

Figure 2: Example composite privacy rules used by AudAgent in Appendix C.9. When the coarse data types IP address, location, and name co-occur in a trace, AudAgent detects the composite pattern and warns of elevated privacy risk.

In this example, the prompt "Hi, my name is David Johnson and I'm originally from Liverpool. (from IP 192.168.0.1.)" triggered the composite rule, causing AudAgent to raise a privacy-risk warning for the trace. The warning is recorded in the associated edge metadata under the `violation_info` field.

2.3.3 Interface Refinements for Improved Clarity and Usability

We incorporated interface refinements such as tooltips and warning features to enhance clarity and usability.

> (Appendix C.8, Page 22) *Additional Usability Screenshots.* To improve usability, AudAgent's visualization interface provides interactive tooltips that explain key elements of the trace graph and surface details about detected privacy violations. Figure 3 shows representative examples. (i) Hovering over the "help" icon reveals a tooltip describing the visualization elements, such as the meaning of nodes and edges in the trace graph. (ii) Clicking a warning edge (boxed in red) opens a tooltip with the violation details, including the data type that triggered the alert and the specific policy condition that was violated.

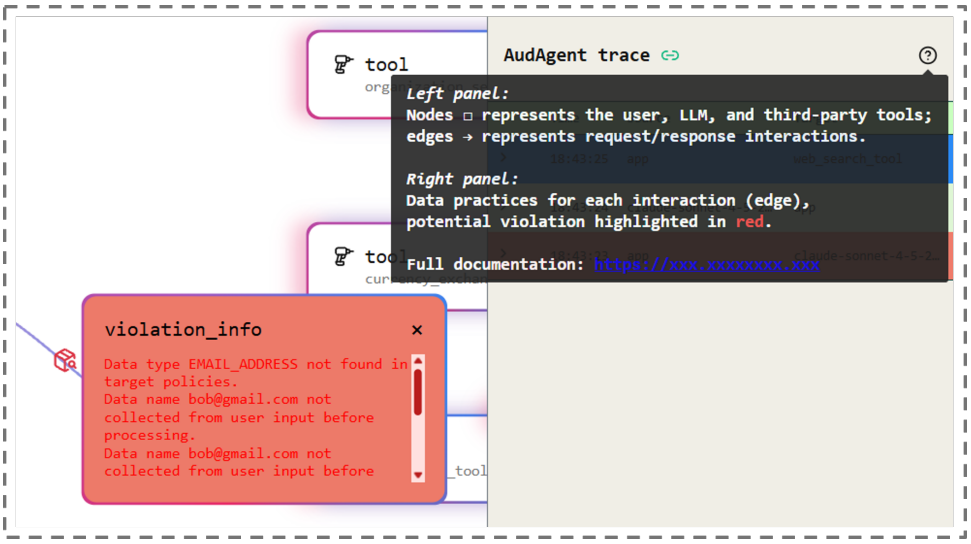

Figure 3: Example tooltips in AudAgent's visualization interface. (i) Hovering over the "help" icon shows explanations of the visualization elements. (ii) Clicking a warning edge (boxed in red) reveals details of the detected privacy violation in the trace.

2.4 Other Comments

Other comments from Reviewer D, Reviewer A, and Reviewer C are addressed in this section.

2.4.1 Reviewer D's Other Comments

In Definition 1, we clarified that the value domains of the c^{dis} and c^{ret} elements are specified later in Section 3.2.2.

> (Footnote of Section 2.2, Page 3) ... We focus on the core data-practice components for clarity. (ii) We use such five-element model for auditing; the value domains of these elements in our auditing model are specified in Section 3.2.2.

We added contents for clarifying the benefits of real-time visualization for AI agents' data practices.

> (Section 2.5, Page 4) User trust in AI agents depends not only on their outputs (as with chatbots), but also on transparency into the agent's internal processes and data practices [14, 15]. Real-time visualization provides such transparency by mapping observed runtime events (e.g. external requests) to intuitive visual representations (e.g. graphs). This enables users to (i) see what data is accessed or disclosed at each step and (ii) detect mismatches where runtime behavior appears to exceed expectations. Accordingly, post hoc logs are insufficient: users need to identify privacy risks in real time and intervene when necessary.

We clarified the meaning of "agent→third" in Definition 2 by explaining the "first-party" and "third-party" concepts in the context of AI agents.

(Footnote of Section 2.3, Page 4) Here, the "first party" refers to (the LLM component of) the AI agent that directly interacts with the user; "third parties" are the external services or tools defined in the agent's privacy policy, which do not directly interact with the user but may receive data from the first party.

We refined the explanation of the ontology graphs by polishing the wording for greater precision.

(Section 3.3.1, Page 7) Each ontology graph encodes hierarchical *"subsumes"* or *"performed-by"* ~~is-a~~ relationships, enabling the auditor to map a concrete annotated type ...

AudAgent expands each policy term downward along subtype-of edges to include all descendant types; thus, coverage is one-way (parent covers children) rather than equivalence.

(Section 3.3.1, Page 7) This ontology graph enables AudAgent to associate ~~map~~ the specific data types identified during data annotation in an AI agent's execution trace with the high-level terms used in the privacy policy model (and document). To bridge this granularity gap, AudAgent expands higher-level policy terms into their subtypes using the ontology graph, so that they can be matched against the concrete instances detected at runtime. During auditing, if an annotated instance matches any subtype of a policy term, it is treated as compliant w.r.t. the privacy policy model.

We clarified the meaning of "sticks in" in the context of an automaton's state transitions.

(Section 3.3.2, Page 8) (ii) It rejects an annotated instance if the run gets stuck ~~sticks~~ in a non-accepting state at any point, i.e. fails to satisfy the guard conditions, indicating a violation from the data type and conditions in the policy model.

We clarified why there is no time-overhead for DeepSeek in the corresponding table notes.

(Table notes of Table 5, Page 13) DeepSeek cannot access the DuckDuckGo API (and the Google Search API), likely due to regional restrictions; therefore, the overhead is not available.

2.4.2 Review A's Other Comments

We showed that if we treat the multi-LLM voting outcome as a proxy ground truth, we can empirically evaluate the accuracy of any single LLM, and add results in our experiments accordingly.

(Section 3.1.2, Page 6) Although the exact value of α is unknown in practice, Theorem 1 still illustrates that voting can provide a quantitative confidence improvement over relying on a single model. Moreover, if we treat the voting outcome as a proxy ground truth, we can empirically estimate each LLM's accuracy α on this task by comparing its individual output against the voting result, which can further inform the confidence and choice of τ.

(Section 5.3.1, Page 13) **Benefits.** (i) *(Confidence boost)* ... (ii) *(Empirical estimation of formalizer accuracy)* Conversely, by treating the voting outcome as a proxy ground truth, we can empirically estimate each formalizer's accuracy α by comparing its individual output against the voted result. Specifically, if we use the F1-score (computed against the voting result) as α, then the average α across the four privacy policies is: Claude 0.85, ChatGPT 0.92, Gemini 0.82, and DeepSeek 0.98.

2.4.3 Review B's Other Comments

We clarified that users are not required to provide a privacy policy to use AudAgent. The privacy policy is fed into LLMs for formalization, and users may provide it either as plain text or as a URL. This is a practical advantage of using LLMs for policy formalization, because it offers a lightweight natural-language interface. In principle, AudAgent could also automatically retrieve policies from the web given an AI agent's name; however, we did not implement this feature to avoid instability due to changing webpages and retrieval errors.

(Section 3.1.3, Page 6) In addition to organizational privacy policies, which can be provided via URLs or text files, AudAgent also allows users to define their own privacy policies. Users can specify which data types they are comfortable sharing with natural language descriptions. AudAgent then leverages LLMs to formalize these user-defined policies into machine-checkable models. This flexibility empowers users to take control of their privacy while still benefiting from the functionality of AI agents.

We clarified why we use U.S. Social Security Numbers (SSNs) as an illustrative example of a highly sensitive data type in our case studies.

(Section 5.2, Page 11) This section uses US Social Security Numbers (SSNs) as a case study to show how AudAgent identifies privacy risks in AI agents and reveals privacy gaps between their operational behaviors and the declared privacy policies. Under US federal law (e.g. the Privacy Act of 1974), SSNs are among the most sensitive types of personal data and shouldn't be disclosed (or even processed) by third parties. However, we will see many companies' AI agents violate the restriction on SSNs, without warning users in their privacy policies.

2.4.4 Review C's Other Comments

We clarified that the "equivalence" relationship ($\sim$) in Algorithm 1 can be implemented using either an LLM-based matcher or lightweight script-based heuristics. For efficiency, our implementation uses a post-processing script that detects capitalization differences (e.g. lower-/upper-case) and simple syntactic variations (e.g. passive voice) in the extracted policy elements.

(Footnote of Section 3.1.3, Page 5) In practice, this semantic equivalence check can be implemented using either an LLM-based matcher or lightweight script-based heuristics.

2.5 Related Work

We updated the related work section to reflect recent advances in AI agents' privacy and security.

(Related Work, Page 16) Privacy and security of AI agents are major concerns due to their autonomous decision-making, extensive data handling, and susceptibility to manipulation. Recent surveys [16, 17] summarize the state of the art in this area. Beyond the attack and defense techniques studied in academia ...

(Related Work, Page 16) *Evaluating & Benchmarking Privacy Leakage.* A growing body of work evaluates and benchmarks AI agents' privacy and security. One notable real-world deployment is OpenAI's GPT Store [18], which allows creating custom GPTs with developer-provided prompts and external APIs, and publish them as GPT apps in a public marketplace. A recent study [19] investigates privacy traceability in this ecosystem by analyzing API parameters through which personal data may be transmitted. AgentDyn [20] offers a dynamic benchmark for evaluating prompt-injection attacks against real-world agent security systems. Similarly, PrivacyLens-Live [21] and LeakAgent [22] show that privacy failures are often interactive, i.e. unfolding over multistep tool use, and can therefore be more severe than what static Q&A benchmarks capture. These benchmarking efforts complement auditing: benchmarks help characterize the broader landscape of agent vulnerabilities and failure modes, while auditing tools like AudAgent provide continuous, real-time monitoring and protection for end users in practice.

Contextual Integrity in AI Agents. Contextual integrity [23] (CI) concerns whether data handling aligns with the norms of a specific context, such as the user's current environment and situational expectations. CMPL [24] proposes an iterative probing strategy to stress-test contextual integrity protections over multi-turn dialogues. To mitigate contextual integrity risks in conversational agents, AirGapAgent [25] restricts agent access to only the minimal task-relevant user data. A similar intermediate control layer appears in [26], which uses a smaller (local-deployed) LLM to mediate between the user and the main agent by filtering sensitive information and rewriting user queries based on the context for better privacy protection. In contrast to these approaches, which primarily aim to prevent or reduce leakage via access control or mediation, AudAgent instead focuses on auditing and visualizing AI agents' data practices against privacy policies.

References

[1] (2016) General data protection regulation (gdpr) - legal text. [Online]. Available: https://gdpr-info.eu/

[2] F. P. Policy, "GDPR Privacy Policy Template," 2026. [Online]. Available: https://www.freeprivacypolicy.com/blog/gdpr-privacy-policy-template/

[3] Termly, "Privacy Policy Generator," 2026. [Online]. Available: https://termly.io/products/privacy-policy-generator/

[4] P. P. Generator, "Privacy Policy Generator – FREE & Easy – Try NOW!" 2026. [Online]. Available: https://www.privacypolicygenerator.info/

[5] OpenAI. (2025) Privacy policy. [Online]. Available: https://openai.com/policies/row-privacy-policy/

[6] Anthropic. (2025) Privacy policy. [Online]. Available: https://www.anthropic.com/legal/privacy

[7] Google. (2025) Gemini apps privacy hub - gemini apps help - gemini apps help center. [Online]. Available: https://support.google.com/gemini/answer/13594961?visit_id=638986383967271503-398671368&p=privacy_help&rd=1#pn_retain_data

[8] Microsoft. (2025) Autogen. [Online]. Available: https://microsoft.github.io/autogen/stable/index.html

[9] (2025) Model context protocol: Introduction. [Online]. Available: https://modelcontextprotocol.io/docs/getting-started/intro

[10] OpenClaw, "OpenClaw Guide — Your Personal AI Assistant Made Simple," 2026. [Online]. Available: https://www.getopenclaw.ai

[11] ——, "OpenClaw Guide — Privacy Policy," 2026. [Online]. Available: https://www.getopenclaw.ai/privacy

[12] A. Reinhart, D. W. Brown, B. Markey, M. Laudenbach, K. Pantusen, R. Yurko, and G. Weinberg, "Do llms write like humans? variation in grammatical and rhetorical styles," *CoRR*, vol. abs/2410.16107, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2410.16107

[13] B. Smith, M. R. Bouadjenek, T. A. Kheya, P. Dawson, and S. Aryal, "A comprehensive analysis of large language model outputs: Similarity, diversity, and bias," *CoRR*, vol. abs/2505.09056, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2505.09056

[14] E. C. Cheng, J. Cheng, and A. Siu, "Toward safe and responsible ai agents: A three-pillar model for transparency, accountability, and trustworthiness," 2026. [Online]. Available: https://api.semanticscholar.org/CorpusID:284648579

[15] S. Raza, R. Sapkota, M. Karkee, and C. Emmanouilidis, "Trism for agentic AI: A review of trust, risk, and security management in llm-based agentic multi-agent systems," *CoRR*, vol. abs/2506.04133, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2506.04133

[16] F. He, T. Zhu, D. Ye, B. Liu, W. Zhou, and P. S. Yu, "The emerged security and privacy of LLM agent: A survey with case studies," *ACM Comput. Surv.*, vol. 58, no. 6, pp. 162:1–162:36, 2026. [Online]. Available: https://doi.org/10.1145/3773080

[17] B. Yan, K. Li, M. Xu, Y. Dong, Y. Zhang, Z. Ren, and X. Cheng, "On protecting the data privacy of large language models (llms) and llm agents: A literature review," *High-Confidence Computing*, vol. 5, no. 2, p. 100300, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667295225000042

[18] G. Store. (2026) Find the best GPTs of ChatGPT. [Online]. Available: https://gptstore.ai

[19] J. C. Carrillo, J. L. Martin-Navarro, R. Ma, and J. Such, "Personal data flows and privacy policy traceability in third-party llm apps in the gpt ecosystem," *Proceedings on Privacy Enhancing Technologies*, 2026. [Online]. Available: https://api.semanticscholar.org/CorpusID:285059332

[20] H. Li, R. Wen, S. Shi, N. Zhang, and C. Xiao, "Agentdyn: A dynamic open-ended benchmark for evaluating prompt injection attacks of real-world agent security system," 02 2026.

[21] S. Wang, F. Yu, X. Liu, X. Qin, J. Zhang, Q. Lin, D. Zhang, and S. Rajmohan, "Privacy in action: Towards realistic privacy mitigation and evaluation for llm-powered agents," *CoRR*, vol. abs/2509.17488, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2509.17488

[22] Y. Nie, Z. Wang, Y. Yu, X. Wu, X. Zhao, N. D. Bastian, W. Guo, and D. Song, "Leakagent: RL-based red-teaming agent for LLM privacy leakage," in *Second Conference on Language Modeling*, 2025. [Online]. Available: https://openreview.net/forum?id=WIfns41MAb

[23] H. Nissenbaum, *Privacy in Context - Technology, Policy, and the Integrity of Social Life*. Stanford University Press, 2010. [Online]. Available: http://www.sup.org/book.cgi?id=8862

[24] S. Das, J. Sandler, and F. Fioretto, "Beyond jailbreaking: Auditing contextual privacy in LLM agents," 2026. [Online]. Available: https://openreview.net/forum?id=6EDpNDns1i

[25] E. Bagdasarian, R. Yi, S. Ghalebikesabi, P. Kairouz, M. Gruteser, S. Oh, B. Balle, and D. Ramage, "Airgapagent: Protecting privacy-conscious conversational agents," in *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS 2024, Salt Lake City, UT, USA, October 14-18, 2024*, B. Luo, X. Liao, J. Xu, E. Kirda, and D. Lie, Eds. ACM, 2024, pp. 3868–3882. [Online]. Available: https://doi.org/10.1145/3658644.3690350

[26] I. C. Ngong, S. R. Kadhe, H. Wang, K. Murugesan, J. D. Weisz, A. Dhurandhar, and K. N. Ramamurthy, "Protecting users from themselves: Safeguarding contextual privacy in interactions with conversational agents," in *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, ser. Findings of ACL, W. Che, J. Nabende, E. Shutova, and M. T. Pilehvar, Eds., vol. ACL 2025. Association for Computational Linguistics, 2025, pp. 26 196–26 220. [Online]. Available: https://aclanthology.org/2025.findings-acl.1343/