

Keeping a Public Story Straight: Memory Interventions for LLM Agents in AIWolf

Seungjae Ryan Lee
Independent
ry@nlee.ai

Abstract

Agents in multi-agent dialogue must maintain a consistent public account of their own actions and observations to avoid being perceived as unreliable or deceptive. In nine-player AIWolf, a natural-language social-deduction game, we study *public-account inconsistency*, a later public statement or action incompatible with the same speaker’s earlier account without correction. Across 24 games played by the official baseline agent, 20 clear inconsistencies were found, audited by a large language model (LLM) and confirmed by author review. In matched 24-game arms, we deploy a static warning, engine-rendered verbatim self-history, and a private model-written structured scratchpad to all nine players under identical model and runtime controls. The scratchpad has the lowest inconsistency count of 16.

1 Introduction

Language agents operating over long interactions must recover their earlier commitments, claims, and actions from an increasingly large conversational context. Losing this history is consequential in mixed-motive settings: an agent may misreport its own ballot, act before a prerequisite it publicly imposed has been satisfied, or vote against a player it had publicly cleared. This problem is distinct from lying: a deliberately false but internally maintained story can be strategically valid, whereas an unacknowledged contradiction can expose either an honest or deceptive agent.

We investigate public-account consistency in AI-Wolf, a natural-language social-deduction game in which town-aligned and adversarial roles debate before casting mechanically recorded ballots (Kano et al., 2025). AIWolf supplies public action records and role-local information while allowing both co-operation and authorized deception. It therefore supports a behavioral question that ordinary dialogue benchmarks cannot answer cleanly: do an

agent’s later language and actions remain compatible with its chosen public account?

Research questions. We ask whether public-account inconsistencies occur in ordinary baseline games and what forms they take (RQ1), then whether instruction, verbatim self-history, or structured memory reduces them while preserving ordinary communication (RQ2).

Our contributions are: (1) an operational definition of public-account inconsistency that is independent of hidden truth; (2) an LLM baseline audit grounded in exact utterances and engine ballots, with subsequent author review of candidate cases; and (3) a full-game comparison of instruction-only, verbatim-history, and structured-scratchpad interventions.

2 Related Work

Multi-agent systems have been studied in both simulated and human-in-the-loop settings. MultiAgentBench evaluates multi-agent systems through task completion and milestone-based collaboration and competition measures across interactive scenarios (Zhu et al., 2025). However, it does not measure whether one agent maintains a compatible public account over a complete interaction.

Deception detection has also been studied in both human and agent settings. Hazra and Majumder (2024) detect deception from linguistic cues grounded in objective truth, while Hoodwinked tests deception and lie detection through discussion and voting in a related text game (O’Gara, 2023). Shi et al. (2026a) measure deviations from public announcements in one-shot games, while Shi et al. (2026b) distinguish private intent, public announcements, and final actions in repeated games. Our work instead examines strategic natural-language play under hidden roles, concerning compatibility within a chosen public account independent of hidden truth.

3 Environment and Definition

AIWolf environment. AIWolf emulates the social-deduction game Werewolf: players receive hidden roles, use natural-language discussion to identify allies and opponents, and take public and private actions that determine which team survives. The game offers ordered-turn five-player and nine-player settings (AIWolfDial Organizers, 2026). Prior five-player work reports that all surviving roles can become inferable by the second day, producing a so-called “solved” state with limited strategic discussion (Okada and Ito, 2025). We therefore focus on ordered nine-player AIWolf, which adds role diversity, private communication between the Werewolves, and longer public histories.

Each day, surviving agents speak in fixed order and cast public ballots to execute one player. Surviving special roles then take private night actions. The village wins by eliminating both Werewolves, while the Werewolves win when they can no longer be outvoted. Werewolves and the Possessed are permitted to deceive; the remaining roles are not assumed to be truthful or correct. Table 1 summarizes the nine-player composition and role-specific information or actions, following the organizer’s 2026 regulations (AIWolfDial Organizers, 2026).

Role	n	Team	Private information or action
Villager	3	Village	None
Seer	1	Village	Learns whether one chosen survivor is human or a Werewolf
Medium	1	Village	Learns whether the executed player was human or a Werewolf
Bodyguard	1	Village	Protects one player from the nightly attack
Possessed	1	Werewolf	Human who wins with the Werewolf team
Werewolf	2	Werewolf	Knows its partner, whispers privately, and attacks at night

Table 1: Roles in the ordered nine-player setting.

Public-account inconsistency. In a social-deduction game like AIWolf, players must maintain a consistent public account of their own actions and observations to avoid being perceived as unreliable or deceptive. We define “public-account inconsistency” as a later public statement or action that is inconsistent with the speaker’s earlier public account without an intervening explicit correction or qualification. The definition evaluates compatibility with the chosen account, not agreement with hidden game state. Silence, deflection, a changed belief after explicit revision, and disagreement with another player do not count as inconsistencies. Such incon-

sistencies are strategically consequential because other players may interpret them as evidence of unreliability or deception, raising suspicion toward the speaker and affecting later ballots. Repeated statements in one contradiction chain count once, identified by speaker and resolution event.

Figure 1 shows three cases confirmed in the author’s subsequent human review.

Family	Count
Action report	1
Prerequisite	11
Target assessment	3
Other	5

Table 2: Detected inconsistency families in the 24-game baseline corpus.

Action report. An action-report inconsistency occurs when a player later describes its own public ballot differently from the engine-recorded action. The discrepancy changes the public evidence available to other players and can make the speaker appear deceptive.

Prerequisite. A prerequisite inconsistency occurs when a player publicly makes its judgment or action conditional on receiving information, but acts before that condition is satisfied without withdrawing or revising the condition. A request to hear more is insufficient unless it establishes a prerequisite for the later judgment or action.

Target assessment. A target-assessment inconsistency occurs when a player publicly clears another player, then ballots for that same player without publicly revising the earlier assessment. The conflict lies between the speaker’s stated evaluation and its consequential action.

4 Method

4.1 Agent and game controls

All games use the official English organizer agent and ordered-nine-player server. The behavioral model is deepseek/deepseek-v4-flash-0731 at temperature 0.7 with reasoning disabled. Server release, role-to-seat schedule, provider policy, action limits, and decoding are fixed across all experiments. Each policy is evaluated in 24 independently completed games; all nine players in a treatment game receive the same intervention. We retain public transcripts, private agent logs, generated configurations, usage, and evaluator-side roles for analysis.

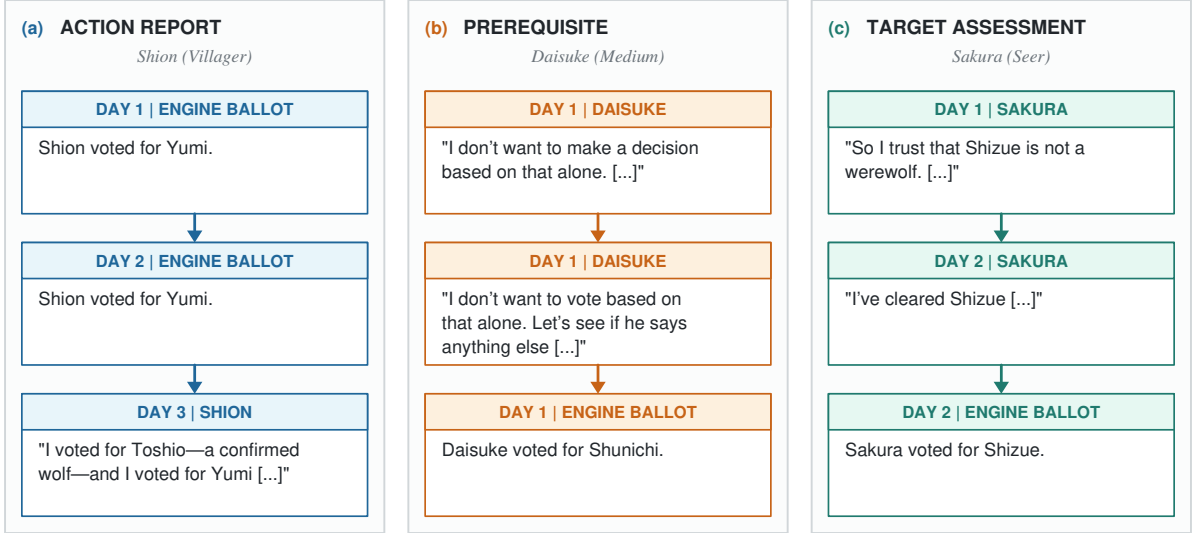


Figure 1: Inconsistencies from baseline games, using excerpts of delivered public statements and engine-recorded ballots. Roles are engine assignments; arrows indicate temporal order. Bracketed ellipses mark omitted text. (a) Shion claims a Toshio ballot, although both earlier ballots targeted Yumi and Toshio was executed after the first. (b) Daisuke asks to hear more from Shunichi before voting on his defensiveness, then votes for him with no further public speech. (c) Sakura reiterates that Shizue is cleared, then votes to execute Shizue without a public revision. These illustrate conflicts with the speaker’s own account, independently of hidden roles.

4.2 Interventions

We test multiple intervention strategies that target public-account inconsistency to determine which method has the largest positive impact. Every intervention permits strategic deception and prescribes neither a role claim nor a target. Figure 2 isolates the implementation difference among the four complete agent policies.

Static warning. The prompt tells the agent to track its public role claims, results, observations, assessments, ballots, and rationales, and to correct or qualify later conflicts. It provides no additional memory.

Explicit self-history. Before each TALK or VOTE action, the agent receives an engine-rendered verbatim record of its earlier public TALK turns and most recently reported ballot. This makes source evidence salient without semantically rewriting it.

Structured scratchpad. At the existing private day-end call, the agent writes a compact PUBLIC-ACCOUNT LEDGER with fields for role claims, results and observations, player assessments, prerequisites, ballots and rationales, and corrections. Later TALK and VOTE prompts direct the agent to consult the latest ledger. The ledger is private and adds no game information.

4.3 Annotation and outcomes

LLM audit. Inconsistencies can be subtle and require careful inspection of the full public history and engine-recorded ballots. We therefore use a large language model (LLM) to scan each player’s history and the complete transcript for candidate inconsistencies under the same definition. The LLM audit occurred in two passes with separate model identifiers, `gpt-5.6-luna` for detection and `gpt-5.6-terra` for verification, separate from the DeepSeek model playing the games. Each player’s history and the whole transcript were scanned under the same condition-blind definition. Candidates cited exact antecedent and resolution event IDs; roles were attached after annotation. A further AI transcript review checked candidates against complete public histories and engine ballots.

Human verification. To verify the quality of the LLM audit and to refine the rubric, we reviewed the 99 recovered historical candidate cases manually. Both public and private information were available during the review, but the review focused on public-account compatibility marking candidates inconsistent even when the private information suggested otherwise. Public information included the full public transcript and the engine-recorded ballots, whereas private information included the agent role,

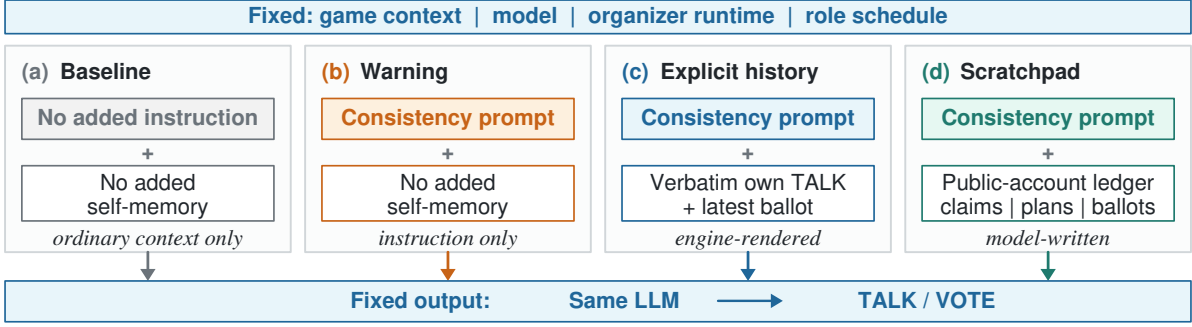


Figure 2: Full-game interventions compared. All policies receive the same ordinary game context and use the same model and runtime. The warning changes instructions only; explicit history adds an engine-rendered verbatim self-record before each action; the scratchpad instead adds a model-written compressed ledger updated during the existing day-end reflection.

private night actions, and the agent’s private day-end reflection. The author’s review agreed with the LLM annotations in 83% of candidate cases.

5 Results

5.1 Baseline occurrence and forms (RQ1)

Across 24 baseline games and all 216 player-game units, we found 20 inconsistencies, showing that public-account inconsistencies occur frequently when no intervention is applied. Several inconsistencies came from one game in which players requested an unfinished explanation and then balloted its speaker. Such prerequisite judgments depend on whether the request actually commits the speaker to waiting. The author-confirmed examples in Figure 1 include a false ballot report and a ballot against a publicly cleared target.

5.2 Intervention comparison (RQ2)

We then applied the three interventions to all nine players in 24 games each, only changing the memory access and instructions while holding the model, role schedule, and runtime constant. The scratchpad had the lowest inconsistency count of 16, 20% below the baseline’s 20, while the warning retained 18 and explicit history 19. Table 3 reports the inconsistency counts of each intervention method in detail.

6 Discussion

From the baseline examples, we established that public-account inconsistencies occur in ordinary AIWolf game trajectories. We also discovered that multiple inconsistencies can cluster around a single event. A prerequisite judgment can trigger a

Policy	<i>n</i>	Act.	Pre.	Tgt.	Other	Total
Baseline	24	1	11	3	5	20
Static warning	24	3	3	1	11	18
Explicit history	24	1	11	4	3	19
Structured scratchpad	24	0	9	1	6	16

Table 3: Counts of clear player-attributed public-account inconsistencies in complete games. Act. = Action report; Pre. = Prerequisite; Tgt. = Target assessment.

later ballot that conflicts with the speaker’s own account, and a later statement can then conflict with that ballot. An inconsistency sparked by one agent may also propagate through the group, producing multiple inconsistencies in one game, although the causality is not established.

The interventions tested progressively stronger memory access. A warning changes instructions but not evidence access. Explicit history reduces retrieval burden while preserving wording and qualifications. A structured scratchpad further compresses history, but its abstraction may omit the precise prerequisite. This comparison bears on a broader conversational-memory choice: whether to expose source text or an interpreted state representation. The structured scratchpad had the lowest inconsistency count, while static warning and explicit history remain close to the baseline under that audit.

Limitations

The experiment was conducted with limited resources, using one hosted model identifier, one provider policy, one role schedule, and 24 games per arm. Applying the intervention to all players may have changed the dialogue trajectory and pro-

duced a policy-level effect rather than an isolated mechanism. Semantic annotation can miss or over-interpret ambiguous language, especially when organizer limits truncate utterances. Although exact citations make judgments auditable, they do not remove subjectivity. Human review involved a single author, causing potential bias in the assessment.

Ethical Considerations

The study analyzes fictional game behavior and involves no recruited human participants or personal data. Memory that improves consistency is dual-use: it may help honest roles maintain evidence and adversarial roles maintain deception. We therefore do not present consistency as a truthfulness intervention.

Data Availability Statement

The reproducibility artifact, including aggregate results, exact prompts, sealed manifests, and regeneration code, will be released at <https://github.com/ryanleeai/aiwolf-memory>.

Acknowledgments

We would like to thank the AIWolf organizers for providing the environment and baseline agent. We would also like to thank the anonymous reviewers for their valuable feedback and suggestions to include information about human verification of the results.

References

- AIWolfDial Organizers. 2026. International contest 2026 and AIWolfDial 2026 workshop. https://aiwolfdial.github.io/aiwolf-nlp/en/page/inlg_2026/. Accessed 2026-08-20.
- Sanchaita Hazra and Bodhisattwa Prasad Majumder. 2024. [To tell the truth: Language of deception and language models](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8506–8520, Mexico City, Mexico. Association for Computational Linguistics.
- Yoshinobu Kano, Neo Watanabe, Yuya Harada, Yuto Sahashi, Claus Aranha, Daisuke Katagami, Kei Harada, Michimasa Inaba, Takeshi Ito, Hirotaka Osawa, Takashi Otsuki, and Fujio Toriumi. 2025. [AI-WolfDial 2025: Summary of natural language division of 7th international AIWolf contest](#). In *Proceedings of the 3rd International AIWolfDial Workshop*, pages 1–13, Hanoi, Vietnam. Association for Computational Linguistics.
- Aidan O’Gara. 2023. [Hoodwinked: Deception and cooperation in a text-based game for language models](#). *Preprint*, arXiv:2308.01404.
- Takuma Okada and Takeshi Ito. 2025. [Towards a strategic werewolf AI based on expert strategies in five-player werewolf](#). In *Proceedings of the 3rd International AIWolfDial Workshop*, pages 14–21, Hanoi, Vietnam. Association for Computational Linguistics.
- Jerick Shi, Terry Jingcheng Zhang, Zhijing Jin, and Vincent Conitzer. 2026a. [Cheap talk, empty promise: Frontier LLMs easily break public promises for self-interest](#). *Preprint*, arXiv:2604.04782.
- Jerick Shi, Terry Jingcheng Zhang, Bernhard Schölkopf, Vincent Conitzer, and Zhijing Jin. 2026b. [When agents lie: Premeditation, persistence, and exploitation in repeated games](#). *Preprint*, arXiv:2607.05132.
- Kunlun Zhu, Hongyi Du, Zhaochen Hong, Xiaocheng Yang, Shuyi Guo, Zhe Wang, Zhenhailong Wang, Cheng Qian, Xiangru Tang, Heng Ji, and Jiaxuan You. 2025. [MultiAgentBench: Evaluating the collaboration and competition of LLM agents](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8580–8622, Vienna, Austria. Association for Computational Linguistics.