Agent-Mafia: An Event-Sourced Environment for Auditing Strategic Claims in Social-Deduction Agents

Ryan Junejo
CrashLabs
ryan@crashlabs.ai

September 2026

Abstract

Win rate is a weak instrument for social-deduction agents. It scores whole tables when the object of interest is a seat, and it says nothing about whether an agent's statements were true. We present AGENT-MAFIA, a deterministic, event-sourced Mafia environment in which agents' role, self-alignment, investigation, and protection claims are extracted from public messages with model assistance and scored deterministically against logged game state. Every game produces a hash-chained, replayable log. In an exploratory study, 12 contemporary language models played 40 eleven-seat games (4,267 agent wakes, 62.55 million tokens). The environment completed every game. First-attempt action validity was 79.45% overall and ranged from 50.7% to 100% by model. In the primary 38-game analysis set, Town ballots targeted a Mafia seat in 60.08% of valid non-abstentions, against uniform-choice baselines of 28.96% and 33.53%; every model's observed point estimate exceeded both. As exploratory, author-adjudicated evidence, 667 quoted claim occurrences map to 433–460 distinct claims, of which 118–120 were verifiably false; 104 of the 120 at the upper end of this linkage range came from Mafia seats, and all 16 from Town seats were power roles claiming to be villagers. An audit of our original claim-verification pipeline found 20 invalid labels among 171, all at the utterance-to-state boundary and none visible to integrity checks. The revised pipeline, audit, logs, and analysis bundle accompany the paper.

Keywords: language-model agents; social deduction; Mafia; evaluation environment; ground-truth verification; event sourcing; claim auditing

1 Introduction

Whether an agent wins a game is the natural score for game environments, and for social deduction it is a poor one. A Mafia outcome is shared by eleven seats, so one seat's contribution is confounded by its teammates and its opponents. A Mafia seat can win a game without saying a word and lose one after playing well. And the behaviors that make the game interesting as an evaluation setting, whether the agent acted validly at each turn, whether it voted for the right people under partial information, and whether the things it said about hidden state were true, leave no trace in the scoreline. The analogous move in other game benchmarks is to score how the game was played. Here the object of measurement is what agents said and did, checked against a record the environment keeps of what happened.

On Day 2 of game `sweep1-0`, the seat holding the detective role, Bryan, truthfully reported that Night 2's investigation had found Cyan to be Mafia.[1] Cyan, a Mafia seat, replied "I am the real

[1] Public log artifacts carry the filename prefix `sweep1-`; the paper refers to this data collection as the exploratory

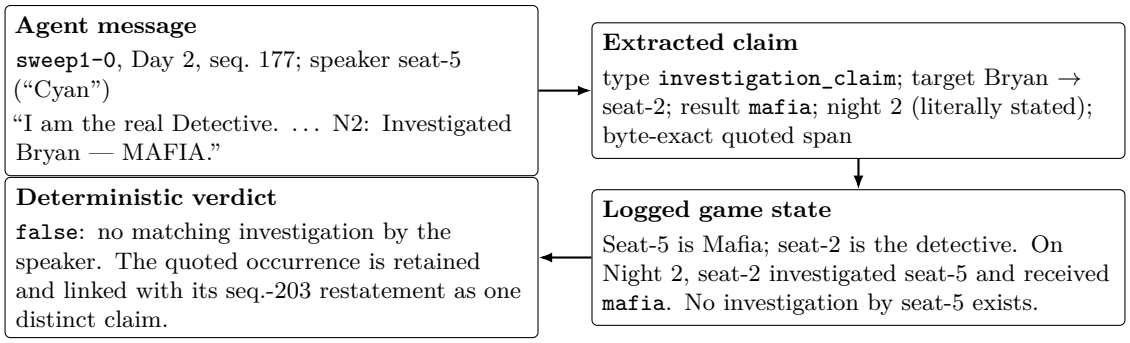

Figure 1: One statement through the instrument. Candidate extraction and review are model-assisted; a targeted subset receives author adjudication. Comparison with logged state is deterministic code. The same message yields a false role claim and a second false investigation report, and its later restatement adds claim occurrences without adding distinct claims.

Detective," followed by "N1: Investigated Trae — Innocent (Not Mafia)" and "N2: Investigated Bryan — MAFIA," and restated both results in a later message. The table executed Bryan by seven votes to two. Town went on to win the game two days later. A win-rate benchmark records a Town win. The event log records a false role claim, two investigation reports with no matching logged action, and a ballot that removed the only truthful detective. Figure 1 follows one of those statements from message to verdict.

The paper asks three questions. **RQ1.** Can an event-sourced social-deduction environment make selected strategic claims by language-model agents deterministically checkable, and what does that cost? **RQ2.** How do contemporary language-model agents behave in such an environment: do they act validly over a long interaction, do their ballots find Mafia seats under partial information, and what captured false claims appear by faction? **RQ3.** How reliable is the utterance-to-state mapping itself, and what does auditing it reveal?

The contributions are, first, a deterministic Mafia engine with tamper-evident replayable logs, a visibility-filtered event export, and a sampled role-field redaction test on the observation function (§3); second, a claim pipeline whose intermediate semantic record is a first-class artifact, with explicit correction objects, a hash-bound adjudication chain, archived superseded evaluator versions, and a mapping from quoted claim occurrences to distinct claims (§3, §4); third, the exploratory study and its results (§5); and fourth, a public failure analysis of the pipeline (§6). Engine-derived quantities are the quantitative core; captured-claim quantities are exploratory. No model ranking is claimed, and falsity is never treated as evidence of intent.

2 Related Work

Social-deduction games entered language-model evaluation as settings where dialogue carries strategy (Xu et al., 2023; O'Gara, 2023; Olson et al., 2026). These designs differ in what they score. Outcome-scored designs compare win rates or survival (Bailis et al., 2024; Costa and Vicente, 2025). Behavior-scored designs label transcript segments, by a judge model or by annotators, for properties such as persuasion or detection (Golechha and Garriga-Alonso, 2025; Agarwal et al., 2025; Bauer et al., 2026; Kao et al., 2026). Statement-grounded designs compare what an agent said with something the environment recorded (Karpov, 2026; Kang et al., 2025). AGENT-MAFIA is of the third kind.

campaign.

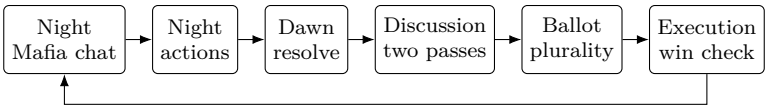

Figure 2: One day–night cycle. Night actions (kill, protect, investigate) are submitted simultaneously; dawn resolves them, reveals deaths, and checks the win condition; two sequential discussion passes precede a simultaneous plurality ballot. Eleven seats (3 Mafia, 1 doctor, 1 detective, 6 villagers); campaign games lasted three to five cycles.

The closest work is QUACK (Yuan et al., 2026), which evaluates a multimodal social-deduction agent at the levels of game outcome, behavioral trajectory, and utterance consistency. Its verification pipeline has a language model extract structured, individually checkable claims, caches the extraction, verifies each claim deterministically against environment state, stores the verdict with its supporting evidence, and reports human validation of the extraction step. AGENT-MAFIA shares that architecture and applies it to a different claim space. QUACK verifies spatial and accusation claims in a multimodal setting; AGENT-MAFIA verifies the role, self-alignment, investigation, and protection assertions of Mafia. To that shared architecture this paper adds four things: explicit correction objects that carry the human's corrected fields and their provenance into the scorer; hashes that bind every adjudication stage to the exact inputs it ruled on; an archived, superseded pipeline version whose row-level audit will be published beside the corrected one; and a mapping from quoted claim occurrences to distinct claims so that repetition does not inflate totals. §6 reports what those additions changed in the record.

The vocabulary follows the honesty and deception literature. MASK separates honesty from accuracy by measuring whether a model contradicts what it appears to believe under pressure (Ren et al., 2025). Park et al. (2024) define deception as the systematic inducement of false beliefs in the pursuit of some outcome other than the truth, which requires a target belief and a purpose (see also Hagendorff, 2024; Shi et al., 2026). A logged conflict between an utterance and the game state supplies falsity and nothing more. The paper therefore counts verifiably false distinct claims, keeps them role-conditioned, and reserves deception for case discussion.

3 The Agent-Mafia Environment

3.1 Task and rules

A game seats eleven agents: three Mafia, one doctor, one detective, and six villagers. Roles and synthetic table names are functions of the game seed. Play opens on Night 1, so Day 1 begins with information. Each cycle (Figure 2) runs a private Mafia chat pass, simultaneous night submissions, dawn resolution with announcement and win check, two sequential discussion passes with a rotating speaking anchor, a simultaneous ballot, and execution. Investigations resolve against the pre-night state and return the target's faction only. The Mafia kill is the plurality of Mafia submissions with seeded tie-breaking; protection applies afterward; the doctor may self-protect but may not repeat a target on consecutive nights. Ballot ties kill nobody. Abstention and self-votes are legal. Roles are revealed on death. Public messages are capped at 1,000 characters. Town wins when no Mafia remain; Mafia wins when living Mafia equal or outnumber living Town. Every value is configurable and the campaign defaults are recorded in the public `game_created` event of every log.

3.2 Agent interface and long-horizon interaction

Agents act through wakes. A wake delivers one typed observation, produced by a single function `observe(state, seat)`, containing the seat's role, its permitted private information (Mafia partners; the detective's own results), the public history, the table state, and the currently legal actions as a tool schema. A driver may attempt a response up to three times within a 300-second deadline; every attempt, valid or not, is logged with its latency, provider response identifier, and token usage. When the budget is exhausted the runtime inserts the phase's default and records the cause: a discussion turn becomes a pass, a ballot an abstention, a night action no action, each written as a timeout event and distinguished from the same choices made deliberately, which are legal. One malformed response never ends a game, and the failure stays in the record.

The interaction is long by conversational standards. A campaign game ran three to five days, so a surviving seat woke roughly ten times, each time with the full public history of the game so far and a shrinking table. Seats are de-opinionated: the system prompt states the rules, the wake contract, and the legal actions, and under the campaign's framing adds one epistemic fact, that nothing said at the table is verified. It contains no persona and no strategic advice. The framing, prompt hash, tool-schema hash, provider, and endpoint identifier are recorded per seat in a `seat_bound` event.

3.3 Capabilities being measured

The environment measures three capabilities and deliberately leaves out a fourth. *Action-contract compliance* is whether a seat produces a legal, well-formed action within its budget at each wake, over a horizon of ten or so wakes with growing context. *Voting judgment under partial information* is whether a Town seat's ballot lands on a living Mafia seat, given only public history and the seat's own private knowledge. *Grounded claim behavior* is what a seat asserts about hidden state and whether those assertions match the log, conditioned on the seat's faction. The environment does not measure intent or belief. A verifiably false statement is a statement that conflicts with logged state; the incentive structure that produced it is visible in the faction split.

3.4 Event-sourced evidence and metrics

The engine is a pure state-transition function driven by a seeded random stream whose counter lives in the game state; it never reads a clock. Under a fixed seed and identical submissions it produces byte-identical engine-derived events. Each transition appends JSON events with a sequence number, day, phase, actor, visibility (public, restricted to named seats, or omniscient), payload, and a hash covering the previous hash and the current envelope. Role assignments, night submissions, individual ballots, model bindings, attempt records, and reasoning traces are omniscient; public messages and tallies are visible to all.

Four operations rest on the log. The verifier re-steps the game through the reducer and compares the derived events with the log, so a valid hash chain around an impossible transition still fails. Omniscient replay reconstructs roles and actions at any point. A fork rebuilds state at a chosen sequence number so a game can continue from a recorded moment under different actions. A visibility-filtered export keeps the events a given seat was entitled to see, and because the public `game_created` event carries the game seed, from which roles are derivable, any seat-level view intended for training or display must redact it. The hash chain makes edits after publication detectable; it does not prove how the original run was produced.

The observation function is tested by a sampled property: walking 25 seeded games for up to 60 legal steps, at every reached state and for every observer, the test substitutes each role the observer

is not entitled to distinguish into every other seat's role field and requires the serialized observation to be unchanged, over more than 5,000 substitutions per run. Companion properties check that partner lists are populated only for Mafia, that investigation results belong to the detective that ran them, that no living seat's role is revealed mid-game, and that recorded reasoning traces never surface in an observation. Because the substituted states are edited in place, this is a direct role-field redaction test. It does not prove non-interference.

Metrics come from an opportunity table with one row per decision a seat was prompted for. For Town ballots, *coverage* is valid non-forced ballots over opportunities, *conditional quality* is Mafia hits over valid non-abstentions, and *effective quality* is hits over opportunities. Two uniform-choice baselines are computed on the same ballots, one over all legal targets including the voter and one over living non-self targets; neither is called chance without its policy assumption. Night actions are summarized in aggregate because each model held a power role only a few times. Intervals are seeded 95% percentile intervals from 20,000 game-cluster bootstrap resamples. Claims are counted in two units: a *claim occurrence* is one byte-exact span in one message, and a *distinct claim* groups occurrences that assert the same thing. Counts form a range when nightless repeated action reports cannot be linked with certainty (Appendix A).

3.5 The four claim families

The pipeline publishes claims whose truth conditions correspond to logged state (Table 1): role claims, self-alignment claims (`not_mafia_claim` in the codebook), investigation claims, and protection claims. Admissibility follows a frozen codebook of twenty-one rules. The rules that mattered most require first-person action language, exclude specific-role denials, match evidence only from before the claim and from the stated night when one is given, preserve byte-exact quotations, and count a restatement as a new occurrence of an existing claim. Vote and intention statements are extracted but deferred. A wrong field is fatal and an absent field is permissive: "Night 2" requires a Night 2 action, while "last night" stores no night and accepts any matching prior action by that speaker. Appendix A lists the rules and verdict semantics.

Table 1: Published claim families and the logged state that scores them. Requests and predictions do not assert that the speaker performed an action.

Claim type	Asserted content	Truth source
`role_claim`	Speaker holds a specific role	Role assignment at deal
`not_mafia_claim`	Speaker is outside the Mafia team, no role named	Faction at deal
`investigation_claim`	Speaker investigated a target and reports a Mafia / not-Mafia result	Detective submission and private result, on the stated night if one is stated
`protection_claim`	Speaker protected a named target	Doctor submission, on the stated night if one is stated

4 Experimental Setup

Models. Twelve contemporary language-model endpoints spanning six provider interfaces (Appendix D) played the 40-game evaluation. They were the tool-calling endpoints available to the author at run time, a convenience sample, and adapters left temperature and sampling at provider defaults.

Study design. Forty games and 440 seat assignments were played on 25 August 2026 in three parallel lanes over about eight hours. One model sat out each game under a schedule committed before play that balanced Mafia assignments exactly (ten per model) and other roles and seats nearly so, audited against the logs at 440 of 440; forty games is the smallest schedule with that property for twelve models. A precommitted rule excluded from behavioral summaries any game in which a seat exceeded two provider-error or infrastructure-timeout events; model noncompliance never excluded a game. Two games met the rule, leaving a primary analysis set of 38 games in which Mafia exposure ranges from 8 to 10 appearances per model. A stricter reading of the wording, superseded 148 seconds before the first game but published as a sensitivity set, removes nine games. Six games had been used to calibrate the claim-verification pipeline; five are in the primary analysis set.

Evaluation conditions. Reliability and ballot results are read from the logs and the opportunity table. Claim results come from the pipeline in Figure 3. Candidates are the union of a lexical tripwire generated from the codebook (validated against an archived earlier claim record at 402 of 404 role, 136 of 138 self-alignment, 172 of 172 investigation, and 38 of 38 protection quotes), an earlier extractor's candidates, and one model reading every public message; the same model, Anthropic `claude-sonnet-5`, classifies every candidate, proposing claim type, byte-exact span, and fields. A second model from a different provider, OpenAI Codex running `gpt-5.6-sol` at its `xhigh` reasoning-effort setting, rated all 763 candidates in the published-family confirmation set `OK`, `BAD`, or `CORRECTED` in nine isolated sessions whose transcripts record no command executions or file reads. The author then ruled on one shuffled packet of 150 items (all 94 disputes, a uniform seeded draw of 50 from the 669 accepted candidates, and 6 candidates surfaced by a 100-message scan of machine negatives) without seeing arm, model ruling, verdict, role, model, or outcome; the author had seen aggregate claim-type results beforehand, so the sitting was item-level blind but not prior-free. The unaided first pass was frozen and hashed; a model-proposed set of grouped codebook-consistency corrections, derived from the author's own notes, was then approved as a group and changed 44 rulings. Scoring is deterministic after adjudication. The author designed, ran, and reconciled every stage; the second model is a cross-provider check and never counts as a second human. Appendix B gives the protocol in full.

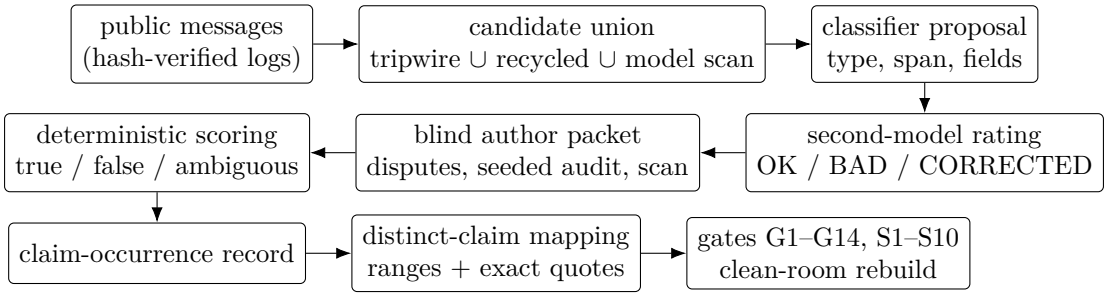

Figure 3: The claim pipeline. Models find, structure, and review candidates; the author rules on disputes, a seeded audit sample, and scan candidates; code assigns truth only after the effective claim record is fixed. Every stage emits a hash-bound artifact.

All results are computed by the pinned analysis code at commit `124f05b` under analysis run identifier `4111e1be...`; at that commit the 14 publication gates and 10 semantic gates pass, and a clean-room run regenerated the 11 deterministic post-rating artifacts byte-for-byte (Appendix C).

5 Results

5.1 Operational reliability

Table 2 and Figure 4 report engine-derived reliability. Of 4,267 wakes, 3,390 produced a valid action on the first attempt (79.45%); retries recovered 756, and 121 ended in a terminal default (111 after model noncompliance, 9 after provider errors, 1 at the deadline). Seats spoke on 2,114 of 2,224 discussion opportunities. Of 1,112 ballot opportunities, 1,103 received a valid non-forced ballot (99.19%), of which 1,062 named a target and 41 were deliberate abstentions; 9 were forced defaults. Providers reported 62.55 million tokens (59.10 million input, 3.45 million output). Across all 40 games Town won 22 and Mafia 18, and games lasted three, four, or five days (12, 17, and 11 games).

First-attempt validity by model ranged from 50.7% to 100%, a 49.3-percentage-point spread. The lowest-validity models recovered most of their wakes on retry, and no game was lost to a driver failure. This axis is engine-derived and needs no claim-verification layer.

Night actions are engine-derived too. Doctors protected the Mafia's chosen victim on 18 of 107 valid nights and self-protected on 48. Every one of the 109 detective choices was a seat that detective had not already investigated. The Mafia's selected victim held a power role on 56 of 152 nights and had voted against a Mafia seat on 83; these describe the chosen target before protection and imply nothing about what the Mafia knew.

Table 2: Operational reliability over all 40 games and night behavior in the primary 38-game analysis set. Night intervals are seeded 95% percentile intervals from 20,000 game-cluster bootstrap resamples.

Measure	Count	Denominator	Rate [95% interval]
Wake produced a valid action on the first attempt	3,390	4,267	79.45%
Discussion opportunity produced a message	2,114	2,224	95.05%
Ballot opportunity produced a valid non-forced ballot	1,103	1,112	99.19%
of which named a target	1,062	1,103	96.28%
Doctor protected the Mafia's selected victim	18	107	16.8% [9.9, 24.1]
Doctor self-protected	48	107	44.9% [38.7, 50.9]
Detective chose a first-time target	109	109	100%
Selected victim held a power role	56	152	36.8% [31.0, 42.6]
Selected victim had voted against a Mafia seat	83	152	54.6% [47.1, 61.6]

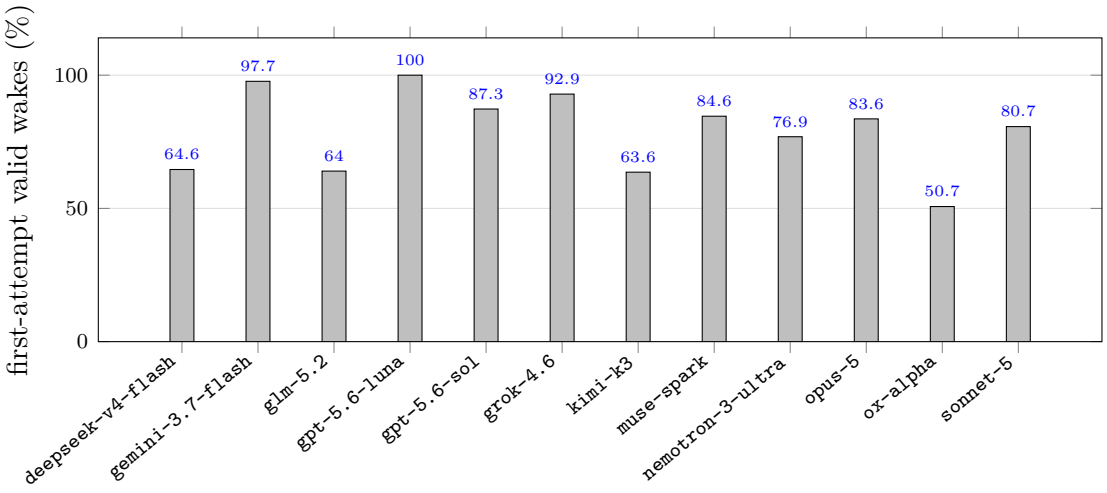

Figure 4: First-attempt action validity by model over all 40 games (292–421 wakes per model; Appendix D). Models are ordered alphabetically; no ranking is claimed.

5.2 Town ballot performance

Town seats in the primary analysis set had 749 ballot opportunities, cast 743 valid ballots, 719 of them non-abstentions, and 432 of those targeted a living Mafia seat: conditional quality 60.08% and effective quality 57.68%. A voter choosing uniformly among all legal targets, self included, would hit a Mafia seat on 28.96% of these ballots; one choosing uniformly among living non-self targets, on 33.53%. Conditional quality exceeded the two baselines by 31.12 and 26.56 percentage points. Figure 5 shows the per-model triplets with cluster-bootstrap intervals (numbers in Appendix D). Every model's interval overlaps several others, opponents and role exposures differed, and no ordering is claimed. Two descriptive facts hold for all twelve: every observed point estimate exceeds both baselines, and a sensitivity check across the primary, strict-sensitivity, and all-game sets found no model whose excess over either baseline changed sign.

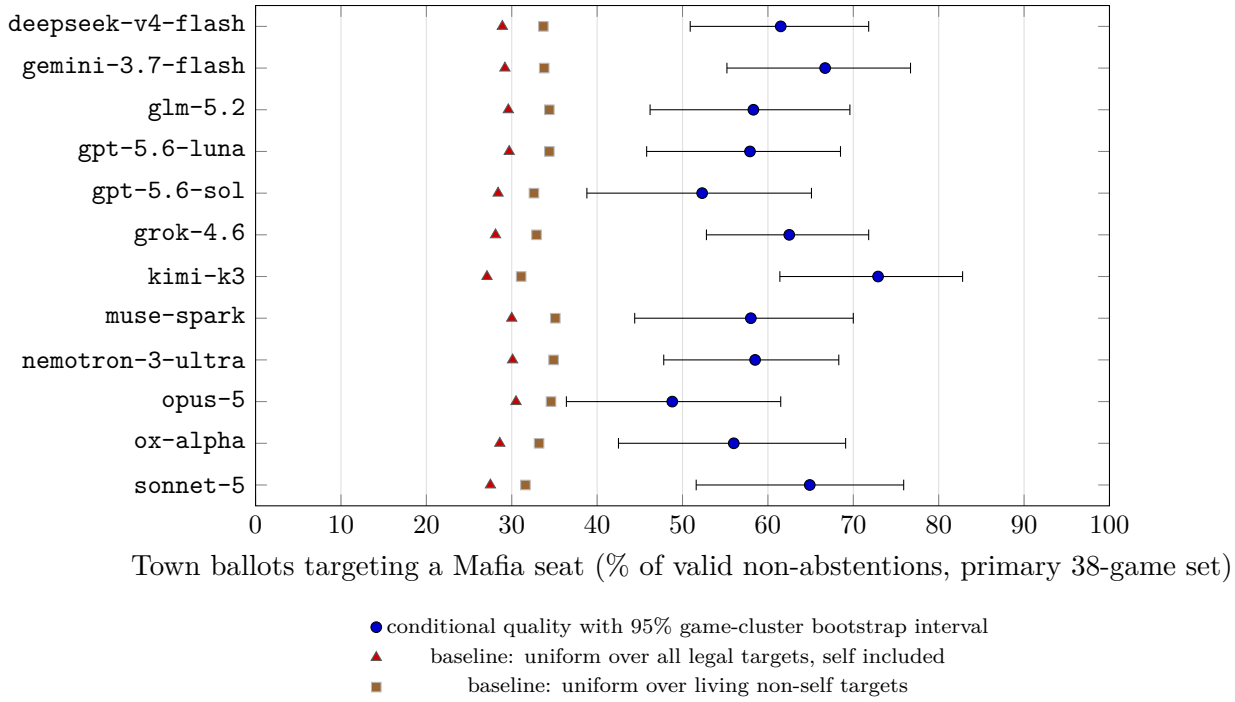

Figure 5: Model-level Town ballot quality in the primary 38-game analysis set. Both uniform baselines use the same ballots. Models are alphabetical; intervals overlap; no ranking is claimed.

5.3 Grounded strategic-claim findings

Every quantity in this subsection is exploratory, author-adjudicated, and potentially incomplete. The primary unit is the distinct claim, reported as a range, with every quoted occurrence preserved beside it. In the primary analysis set, 667 claim occurrences map to 433–460 distinct claims: 315–340 true and 118–120 verifiably false, with the false claims appearing in 155 occurrences (Table 3). At the upper end of the linkage range, 322 distinct claims were stated once, 100 twice, and 38 three or more times; the mapping collapsed 207 exact restatements, and uncertain linkage among 47 action-claim occurrences widened the range. Counting occurrences instead of distinct claims would raise the false total from 118–120 to 155. The strict sensitivity set contains 109–111 false distinct claims and all 40 games contain 119–121.

Table 3: Captured claim results for the primary 38-game analysis set. Distinct-claim counts are ranges because some nightless repeated reports cannot be linked unambiguously. Occurrences count byte-exact quoted spans. Exploratory, author-adjudicated, and potentially incomplete; extraction completeness was not estimated.

Claim type	Distinct true claims	Distinct false claims	Occurrences: total (false)
Role	148	70	358 (94)
Self not-Mafia	66	12	111 (15)
Investigation	77–102	29–30	162 (36)
Protection	24	7–8	36 (10)
Total	315–340	118–120	667 (155)

These are corpus counts conditional on candidate generation and adjudication. The study has

no denominator of opportunities to make a claim, so the counts estimate no per-message, per-seat, or per-game propensity, and no per-model rate is reported. The faction split is the informative comparison (Figure 6, Table 4). Mafia seats, which occupied 114 of the 418 seat-games in the primary set, account for 104 false and 2 true distinct claims at the upper end of the linkage range; Town seats, with 304 seat-games, account for 338 true and 16 false. Each of the 16 Town-false claims is a role claim by a doctor (14 occurrences) or detective (3 occurrences) describing itself as a villager. These statements are consistent with standard role concealment; falsity alone does not establish intent. Per-model captured-claim counts remain available in the machine-readable analysis bundle; unequal exposures and incomplete extraction make them unsuitable as a model ranking.

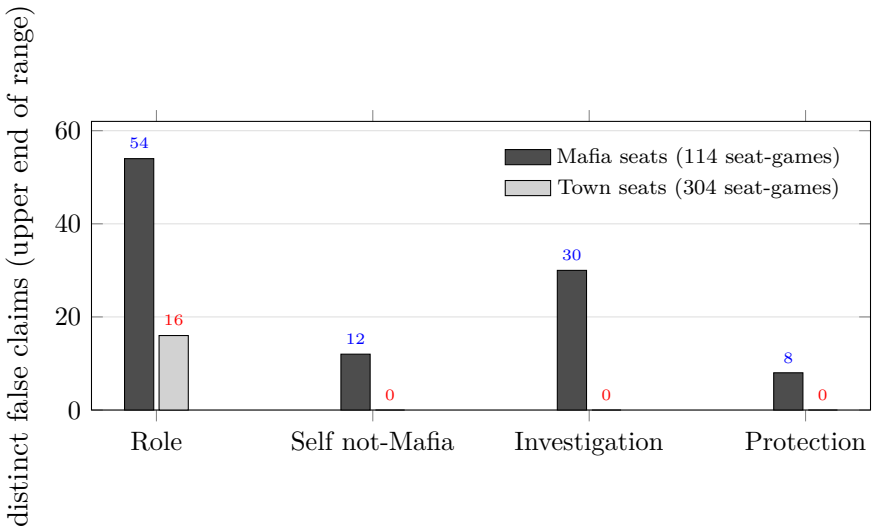

Figure 6: Verifiably false distinct claims by speaker faction and claim type in the primary 38-game analysis set. Bars use the upper end of the linkage range, which comes from linkage uncertainty and carries no confidence level. Every Town-false claim is a power role claiming to be a villager. Exploratory and author-adjudicated.

Table 4: Role-conditioned comparison in the primary 38-game analysis set, using the upper end of the distinct-claim linkage range (claim occurrences in parentheses).

Speaker faction	True	False	False role / alignment	False action reports
Mafia (114 seat-games)	2 (2)	104 (138)	66 (92)	38 (46)
Town (304 seat-games)	338 (510)	16 (17)	16 (17)	0 (0)

5.4 Two transcripts

Game `sweep1-10` shows the whole spectrum inside one game. On Day 1 the detective, Moxie, opened "Villager, no info," and the doctor, Dylan, wrote the same words three messages later; both are verifiably false role claims by Town seats. On Day 2 Moxie revealed the detective role with two truthful results. A Mafia seat, Michael, counterclaimed: "I counterclaim Detective. N1 I checked Tim: town. N2 I checked Moxie: mafia." Neither investigation exists in the log. The table executed Michael six to two. A false role statement, a truthful reveal, and two unsupported reports are scored only by whether they match the log.

Game `sweep1-0`, the motivating example of §1 and one of the six calibration games, also shows the unit distinction. Cyan's Day 2 message and its restatement at sequence 203 produce five false claim occurrences: one role claim and two investigation reports, each stated twice. The mapping counts one false role claim and two false investigation claims. The structured record keeps all five quoted occurrences with offsets, so a reader can check each against the log.

6 Analysis

6.1 What the results suggest

Three observations follow from the engine-derived results alone. First, action-contract compliance and ballot quality are different axes: first-attempt validity spans 49.3 percentage points across models, while conditional ballot quality spans 24.1 points, and models with lower first-attempt validity were not consistently those with lower ballot quality (Appendix D). An environment that scored only outcomes would see neither axis. Second, Town targeted a Mafia seat on three ballots in five, compared with roughly one in three under the two uniform-choice policies, in a game where discussion, deaths, and revealed roles all carry evidence. Third, the study is small and entangled, so these are descriptive facts about this table of models on this day, and they generalize no further.

The exploratory captured-claim results add a fourth observation. Verifiably false claims about hidden state came overwhelmingly from the faction with an incentive to make them. The Town exceptions were statements by power roles claiming to be villagers, a pattern consistent with role concealment. Repetition was common: 207 of 667 claim occurrences restated a distinct claim already made. Whether an agent introduces a false claim, how often it repeats one, and how others respond are separable questions; keeping occurrences and distinct claims as separate units makes that analysis possible.

Making these claims checkable had measurable operational costs. The 40-game evaluation ran for about eight hours in three parallel lanes and produced 62.55 million provider-reported tokens. The separate extraction and classification pass cost about US$34.84 at August 2026 list prices; the second-model review ran under a flat-rate subscription, followed by one 150-item author review. The study did not reconstruct a total evaluation invoice or time the human sitting, so it claims neither.

6.2 The claim pipeline as a measurement case study

The original claim-verification pipeline, v3.1, looked healthy under integrity checks: all 702 archived spans were byte-exact, every hash chain verified, and every tested scorer rule behaved as written. A targeted row-level audit of its 171 claim occurrences labeled false (93 role, 41 investigation, 25 self-alignment, 12 protection) nevertheless confirmed 20 invalid entries (Table 5). Six were genuine action claims whose fields caused deterministic mis-scoring: four where a classifier rendered "last night" as a night number, one where a dropped night field was reintroduced during candidate merging, and one where a conjunction ("Sam and I . . . ") resolved to the speaker. Two were not claims in any published type. Twelve were statements admitted as self-alignment claims that the codebook's assertion bar and denial rule should have excluded. The scorer behaved correctly in every case; the record it was given was wrong. Integrity checks never saw those rows because every bad record was well-formed. The audit was scoped: it screened the false investigation and protection rows for speakers who held the power role, examined two suspected non-claims, and re-adjudicated all 25 false self-alignment rows; it did not re-adjudicate the 93 false role-claim rows, the true-labeled rows, or extraction recall.

Table 5: Invalid false classifications in the v3.1 claim record found by row-level audit (20 of 171 claim occurrences labeled false).

Mechanism	Rows	Effect on the record
Relative night rendered as a number	4	A correct report was checked against the wrong night
Stale night field reintroduced during merge	1	A deleted classifier field reappeared downstream
Conjunction resolved to the speaker	1	The scorer checked the wrong target
Non-claims admitted	2	Ineligible text reached scoring
Assertion-boundary errors in self-alignment claims	12	Denials and weak membership language treated as assertions
Total	20	

The 20 invalid rows entered by three paths. Fifteen came through the author's own reversals of the second model's rejections at the assertion-strength boundary; one passed the author's audit of accepted rows; four were accepted by both machine layers with the unsupported night number visible on the sheet. The protocol, an accept-or-reject verdict on a whole row, offered no way to keep a claim and fix one field. Single-adjudicator boundary drift is this instrument's dominant historical error path.

Version 3.2 addresses each mechanism: classifier fields are authoritative, so an omission deletes a stale value; a night is stored only when literally stated; rulings gain `CORRECTED` with corrected fields that reach the scorer through one validated path; claim types have exact permitted shapes; request forms and specific-role denials become advisory metadata that the scorer ignores and the blind sheets withhold; packet templates and rulings are hash-bound; and the mapping from occurrences to distinct claims runs after every correction. The archived v3.1 record makes a set-level comparison possible. Of its 171 false-labeled occurrences, 150 remain false under v3.2, 6 now score true, and 15 are absent because revised adjudication excluded them. All 20 audited rows are among the 21 that changed. The v3.2 false set contains those 150 survivors plus 5 occurrences absent from v3.1, and no occurrence scored true by v3.1 became false.

The revised pipeline was then run under the protocol of §4, and its validation figures should be read for what they are. The second model rated 763 published-family candidates (669 `OK`, 60 `BAD`, 34 `CORRECTED`; acceptance 370/392 role, 152/170 investigation, 38/47 protection, 109/151 self-alignment). The author's unaided first pass on the 150-item packet was 98 `OK`, 46 `BAD`, 6 `CORRECTED`; the grouped clarification changed 44 rulings (41 disputes, 2 audit items, 1 scan item) to a final 82 / 56 / 12, and exact agreement with the second model on the 94 disputes rose from 34 to 63 in the process. The audit sample confirmed 49 of 50 accepts unchanged, a single-author confirmation fraction of 98.0% with a Wilson 95% interval of 89.5–99.6%, reported as a descriptive interval for this sample and not as a bound on the 619 uncontested accepts. Of the 155 false claim occurrences in the primary analysis set, 143 entered on uncontested second-model accepts and 12 received row-level author review, so the audit does not directly validate the false subset. The six scan candidates were all confirmed; that scan was targeted cleanup and yields no recall estimate. Under the revised pipeline, the 22 nightless investigation and protection occurrences whose quoted span contains "last night" scored 19 true, each with a matching action on the immediately preceding night, and 3 false with no matching action on any night.

6.3 Exploratory status of the captured-claim findings

The pipeline was designed and revised on this corpus, and five of its six calibration games are in the primary analysis set, so the counts are in-sample with respect to pipeline development. Adjudication was single-author, item-level blind but not prior-free, and its final rulings followed a model-proposed grouped clarification. No independent estimate of extraction completeness exists, so the structured claim record is a partially observed candidate set. The paper therefore treats these findings as exploratory supporting evidence for RQ2 and as the primary evidence for RQ3, where the object of study is the measurement pipeline itself.

7 Limitations and Future Work

The engine claims are narrow. The visibility-filtered export shows the events a seat was entitled to see; it does not reproduce the observations that seat received, and the public event stream carries the role-generating seed, so a seat view for training or display needs a redaction contract and a trace-equivalence test that do not yet exist. The observation property tests role-field redaction on edited states; demonstrating non-interference would take paired executions. The hash chain shows that a published log was altered; it records nothing about how the original run was produced.

The claim-verification pipeline is in-sample and author-adjudicated. Its revisions were derived from and evaluated on the same 40 games; the author built the instrument, audited it, and ruled on its packet; only models reviewed the pipeline code; 619 accepted candidates, including 143 of the 155 false claim occurrences, received no row-level human review; the final rulings moved toward the second model after a model-proposed clarification the author approved as a group; and no general recall or omission rate is estimated, so valid claims phrased indirectly may be missing. The nightless-report rule scores "last night" as any matching prior action; it cost nothing here and could undercount elsewhere.

The study is small and entangled. Models faced different opponents and role histories, the primary-set exclusions left Mafia exposure at 8 to 10 per model, night-action denominators are small, and cluster intervals are wide and approximate at 38 clusters. Provider-side sampling defaults were not fixed and are part of each treatment; the endpoints are an August 2026 snapshot. The environment capped public messages at 1,000 characters, which may have encouraged compressed statements and complicated annotation. Reasoning traces were uneven (two endpoints returned encrypted traces) and enter no quantitative analysis. Mafia is a stylized ecology whose fixed roles make some claims unusually checkable.

Future work follows from these limits. A confirmatory study will be preregistered before any new game is played, with the claim-verification pipeline, codebook, model identifiers, prompts, sampling settings, schedule, exclusion rules, analysis code, and primary endpoints frozen in advance; candidate primary endpoints are the role-conditioned incidence of verifiably false distinct claims and Town conditional ballot quality against the two named baselines. Two independent blinded adjudicators will rule on a shared packet before either sees model judgments, with a written disagreement procedure; a random-message arm will estimate message-level omission incidence. The vote and intention claim types will publish under their own rules, with a multi-finder panel motivated by calibration evidence that a second finder adds recall for softer vote language. On the engine side, an observation-trace replay with seed redaction and paired-execution redaction tests would let the environment claim what this paper only approximates.

8 Conclusion

AGENT-MAFIA scores social-deduction play by what agents did and said. The 40-game evaluation shows the environment operating end to end with 12 models, a 49.3-percentage-point spread in action-contract compliance, and Town ballot point estimates above both uniform-choice baselines for every model. The exploratory captured-claim findings show a sharp faction asymmetry, and the Town exceptions are claims by power roles describing themselves as villagers. The engine results do not depend on the claim pipeline. The claim counts do, and the first version of that pipeline was wrong in 20 of 171 false labels even though every hash verified. If the object of measurement is what an agent said, the codebook and the audit trail are part of the result.

Data Availability

Upon publication, the environment, frozen specification, amendment, audits, and 40 hash-chained game logs will be available in the public AGENT-MAFIA repository at https://github.com/crashlabsai/agent-mafia; `mafia verify` re-steps a published log without provider credentials. The curated v3.2 analysis bundle will be at `docs/sweeps/results/analysis-v32/`; its release manifest records every evidence and documentation payload, the source-to-release hash crosswalk, path normalization, and deliberate omissions. `pnpm run check:analysis-release` verifies the bundle. The code is MIT-licensed. Model-generated text remains subject to the respective providers' terms; provider credentials and raw response envelopes are excluded.

Ethics and Broader Impact

No human participants were involved; seat names are synthetic. The game elicited persuasion and false statements under explicit fictional incentives, and the analysis infers no beliefs, intentions, or dispositions from them. Two foreseeable uses deserve caution. Verifiably-false labels record a conflict with logged game state and should not be read as honesty measures outside the game. And if claim verdicts or ballot outcomes are used as reward signals, the semantic layer's measurement error and its susceptibility to phrasing that evades extraction become attack surfaces; this paper trained no policy and tested no reward hacking. Released logs contain model-generated text whose reuse is subject to the respective providers' terms.

Author Contributions

Ryan Junejo: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing (original draft), Writing (review and editing), Visualization, Project administration.

AI-Use Disclosure

Language models were used as instruments inside the study (candidate extraction and classification by one model; a second-model rating pass) as described in §4. AI systems also assisted with code generation and review, with converting the author's rating notes into a proposed set of grouped codebook-consistency corrections that the author approved as a group, and with drafting and editing this manuscript under a staged academic-writing pipeline with a separate hostile review.

The author directed the methodology, performed the blinded ratings, reviewed the final manuscript, and is responsible for the work.

References

Mrinal Agarwal, Saad Rana, Theo Sundoro, Hermela Berhe, Spencer Kim, Vasu Sharma, Sean O'Brien, and Kevin Zhu. WOLF: Werewolf-Based Observations for LLM Deception and Falsehoods, 2025. URL https://arxiv.org/abs/2512.09187. NeurIPS 2025 MTI-LLM Workshop Spotlight.

Suma Bailis, Jane Friedhoff, and Feiyang Chen. Werewolf Arena: A Case Study in LLM Evaluation via Social Deduction, 2024. URL https://arxiv.org/abs/2407.13943.

Niklas Bauer, Lars Benedikt Kaesberg, Akiko Aizawa, Jan Philip Wahle, Bela Gipp, and Terry Ruas. Can Agents Deceive? Evaluating Reasoning and Deception in ParliamentBench Using a Social Deduction Game, 2026. URL https://arxiv.org/abs/2607.28146.

Davi Bastos Costa and Renato Vicente. Deceive, Detect, and Disclose: Large Language Models Play Mini-Mafia, 2025. URL https://arxiv.org/abs/2509.23023.

Satvik Golechha and Adrià Garriga-Alonso. Among Us: A Sandbox for Measuring and Detecting Agentic Deception, 2025. URL https://arxiv.org/abs/2504.04072. NeurIPS 2025 Spotlight.

Thilo Hagendorff. Deception Abilities Emerged in Large Language Models. *Proceedings of the National Academy of Sciences*, 121(24):e2317967121, 2024. doi: 10.1073/pnas.2317967121.

Caixin Kang, Yifei Huang, Liangyang Ouyang, Mingfang Zhang, and Yoichi Sato. Can MLLMs Read the Room? A Multimodal Benchmark for Verifying Truthfulness in Multi-Party Social Interactions, 2025. URL https://arxiv.org/abs/2510.27195. ICCV 2025 Workshop.

Christopher Kao, Vanshika Vats, and James Davis. Hidden in Plain Text: Measuring LLM Deception Quality Against Human Baselines Using Social Deduction Games, 2026. URL https://arxiv.org/abs/2601.13709. IEEE ICA 2025.

Ilia Karpov. MafiaScope: Non-Invasive, Time-Resolved Belief Probing for LLM Agents in Social Deduction Games, 2026. URL https://arxiv.org/abs/2607.10645.

Aidan O'Gara. Hoodwinked: Deception and Cooperation in a Text-Based Game for Language Models, 2023. URL https://arxiv.org/abs/2308.01404.

Matthew Lyle Olson, Neale Ratzlaff, Musashi Hinck, Tri Nguyen, Vasudev Lal, Joseph Campbell, Simon Stepputtis, and Shao-Yen Tseng. LieCraft: A Multi-Agent Framework for Evaluating Deceptive Capabilities in Language Models, 2026. URL https://arxiv.org/abs/2603.06874. AAAI 2026 Alignment Track.

Peter S. Park, Simon Goldstein, Aidan O'Gara, Michael Chen, and Dan Hendrycks. AI Deception: A Survey of Examples, Risks, and Potential Solutions. *Patterns*, 5(5):100988, 2024. doi: 10.1016/j.patter.2024.100988. URL https://arxiv.org/abs/2308.14752.

Richard Ren, Arunim Agarwal, Mantas Mazeika, Cristina Menghini, Robert Vacareanu, Brad Kenstler, Mick Yang, Isabelle Barrass, Alice Gatti, Xuwang Yin, Eduardo Trevino, Matias Geralnik, Adam Khoja, Dean Lee, Summer Yue, and Dan Hendrycks. The MASK Benchmark: Disentangling Honesty From Accuracy in AI Systems, 2025. URL https://arxiv.org/abs/2503.03750.

Jerick Shi, Terry Jingcheng Zhang, Zhijing Jin, and Vincent Conitzer. From Sycophancy to Deception: A Unified Taxonomy for LLM Spontaneous Misalignment, 2026. URL https://arxiv.org/abs/2604.04788. ICLR Agents in the Wild: Safety, Security, and Beyond Workshop.

Edwin B. Wilson. Probable Inference, the Law of Succession, and Statistical Inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. doi: 10.1080/01621459.1927.10502953.

Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. Exploring Large Language Models for Communication Games: An Empirical Study on Werewolf, 2023. URL https://arxiv.org/abs/2309.04658.

Ye Yuan, Rui Song, Weien Li, Zeyu Li, Haochen Liu, Xiangyu Kong, Changjiang Han, Yonghan Yang, Zichen Zhao, Zixuan Dong, Fuyuan Lyu, Bowei He, Haolun Wu, Jikun Kang, and Xue Liu. QUACK: Questioning, Understanding, and Auditing Communicated Knowledge in Multimodal Social Deduction Agents, 2026. URL https://arxiv.org/abs/2605.27068. EMNLP 2026 Main Conference.

A Evaluator Semantics

The frozen codebook defines four published claim types (Table 1) and exact field shapes. A role claim carries a role; a self-alignment claim carries no type-specific field; an investigation claim carries a target and result and may carry a literally stated night; a protection claim carries a target and may carry a night. The rules require first-person action language, exclude specific-role denials, prevent earlier context from supplying an assertion absent from the current message, reject missing required fields, match only evidence that predates the claim, recompute speaker metadata from the verified log, and require a byte-exact quoted span with its offset. A restatement of the speaker's own earlier claim is another occurrence; a report of somebody else's claim is excluded. Verdicts are `true`, `false`, or `ambiguous`. The complete twenty-one-rule codebook, including deferred vote families and stable rule identifiers, is in `docs/analysis/analysis-v3-spec.md`.

The artifacts call each byte-exact claim occurrence a *receipt*. Mapping version `claim-proposition-v1` groups occurrences into distinct claims by game, speaker, claimed state, and the fields specific to each claim type. A literally stated night identifies the event. Without one, two same-target action reports may describe one event or two, so the lower end of the reported range links them and the upper end keeps them separate. A premature report and a later supported report remain distinct because truth is evaluated when each statement is made. In the primary analysis set the mapping collapsed 207 exact restatements and left 47 action-claim occurrences linkage-uncertain; no record had an ambiguous verdict. Gate G13 recomputes the mapping from the published occurrences and verified logs.

B Review and Adjudication

Second-model rating. OpenAI Codex (`gpt-5.6-sol`, `xhigh` reasoning effort) rated all 763 candidates in the published-family confirmation set (760 machine-accepted and 3 machine-rejected investigation/protection candidates) on 1 September 2026. Nine headless sessions each received the task, verbatim codebook excerpts, and one sheet in an empty read-only working directory; the

event transcripts record zero command executions and zero file reads. Rulings were 669 `OK`, 60 `BAD`, and 34 `CORRECTED`; all three machine rejects were upheld. The same model scanned 100 machine-negative messages and surfaced 6 published-family candidates. This scan was targeted cleanup and yields no recall estimate.

Blind author packet. One shuffled packet contained all 94 disputes, a uniform seeded draw of 50 from the 669 accepted candidates, and the 6 scan candidates. It showed each message, span, and proposed fields while hiding the packet arm, second-model ruling, provisional verdict, role, model, and game outcome. The author had seen aggregate claim-type results beforehand, so the sitting was item-level blind but not prior-free. The unaided first pass (98 `OK`, 46 `BAD`, 6 `CORRECTED`) was frozen and hashed. A model converted the author's notes into grouped codebook-consistency corrections, which the author approved as a group; 44 rulings changed (41 disputes, 2 audit items, 1 scan item), yielding 82 / 56 / 12. Agreement with the second model on the 94 disputes moved from 34 to 63. The audit confirmed 49 of 50 sampled accepts unchanged (Wilson 95% interval 89.5–99.6%, descriptive), and all 6 scan candidates were confirmed. Final input contained 769 records: 713 confirmed, 56 excluded, and 12 corrected; 150 received a final author ruling and 619 remained uncontested. Of the 155 false occurrences in the primary analysis set, 143 were uncontested and 12 received row-level author review. The row-level v3.1 audit is published at `docs/analysis/audits/false-label-audit-2026-08-29.md`.

C Verification Checks

The verification suite groups checks into four classes: log integrity and artifact binding; enforcement of the claim schema and codebook; consistency between published numbers and their source records; and deterministic regeneration. At analysis commit `124f05b`, all 14 publication gates and all 10 semantic gates pass in strict mode, the regression suite passes 328 tests, and a clean-room run regenerated 11 deterministic post-rating artifacts byte-for-byte. Two qualifications prevent those counts from overstating the result. The clean-room attestation is bound to that commit, so gate G11 fails by design in any other checkout; reproduction must check out `124f05b`. Semantic gate S10 certifies only that full false-row closure was disclosed as deferred for this exploratory report; that closure adjudication remains deferred. Exact gate definitions and the evaluator revision history are recorded in `scripts/check-gates.mjs`, `scripts/check-semantic-gates.mjs`, and `docs/analysis/analysis-v3.2-amendment.md`.

D Model Endpoints and Per-Model Engine Results

Rows are alphabetical; every rate carries its denominator; no ordering is claimed. Mafia exposure in the primary analysis set is 8 for opus-5 and sonnet-5, 9 for gpt-5.6-luna and muse-spark, and 10 for the others.

Table 6: Endpoint bindings recorded by the `seat_bound` events of the 40 logs (all games 25 August 2026). "Encrypted" means provider-exposed reasoning content was not available in plaintext.

Display name	Provider	Endpoint identifier	Reasoning trace
deepseek-v4-flash	Fireworks	`accounts/fireworks/models/deepseek-v4-flash-0731`	visible
gemini-3.7-flash	Google	`gemini-3.7-flash`	visible
glm-5.2	Fireworks	`accounts/fireworks/models/glm-5p2`	visible
gpt-5.6-luna	OpenAI	`gpt-5.6-luna`	visible
gpt-5.6-sol	OpenAI	`gpt-5.6-sol`	visible
grok-4.6	xAI	`grok-4.6`	visible
kimi-k3	Fireworks	`accounts/fireworks/models/kimi-k3`	visible
muse-spark	OpenRouter	`meta/muse-spark-1.2`	encrypted
nemotron-3-ultra	Fireworks	`accounts/fireworks/models/nemotron-3-ultra-nvfp4`	visible
opus-5	Anthropic	`claude-opus-5`	visible
ox-alpha	OpenRouter	`stealth/ox-alpha`	encrypted
sonnet-5	Anthropic	`claude-sonnet-5`	visible

Table 7: Operational reliability over all 40 games: wakes, first-attempt validity, wakes recovered by retry, terminal defaults, and discussion coverage.

Model	Wakes	First-attempt valid	Recovered by retry	Terminal defaults	Discussion coverage
deepseek-v4-flash	347	64.6%	110	13	176 / 188
gemini-3.7-flash	389	97.7%	9	0	196 / 196
glm-5.2	367	64.0%	103	29	164 / 192
gpt-5.6-luna	383	100.0%	0	0	203 / 204
gpt-5.6-sol	346	87.3%	40	4	177 / 182
grok-4.6	421	92.9%	15	15	210 / 220
kimi-k3	343	63.6%	106	19	162 / 180
muse-spark	383	84.6%	48	11	197 / 202
nemotron-3-ultra	324	76.9%	72	3	164 / 170
opus-5	292	83.6%	48	0	142 / 142
ox-alpha	341	50.7%	143	25	151 / 174
sonnet-5	331	80.7%	62	2	172 / 174

Table 8: Town ballots by model in the primary 38-game analysis set: opportunities, valid ballots, valid non-abstentions, Mafia hits, conditional quality with its 95% cluster-bootstrap interval, effective quality, and the two ballot-weighted baselines (percentages).

Model	Opp.	Valid	Non-abst.	Hits	Cov.	Cond. [95%]	Eff.	Legal	Non-self
deepseek-v4-flash	68	67	65	40	98.5	61.5 [50.9, 71.8]	58.8	28.9	33.7
gemini-3.7-flash	61	61	60	40	100.0	66.7 [55.2, 76.7]	65.6	29.2	33.8
glm-5.2	64	63	60	35	98.4	58.3 [46.2, 69.6]	54.7	29.6	34.4
gpt-5.6-luna	76	76	76	44	100.0	57.9 [45.8, 68.5]	57.9	29.7	34.4
gpt-5.6-sol	68	68	65	34	100.0	52.3 [38.8, 65.1]	50.0	28.4	32.6
grok-4.6	71	67	64	40	94.4	62.5 [52.8, 71.8]	56.3	28.1	32.9
kimi-k3	60	60	59	43	100.0	72.9 [61.4, 82.8]	71.7	27.1	31.1
muse-spark	72	72	69	40	100.0	58.0 [44.4, 70.0]	55.6	30.0	35.1
nemotron-3-ultra	55	55	53	31	100.0	58.5 [47.8, 68.3]	56.4	30.1	34.9
opus-5	42	42	41	20	100.0	48.8 [36.4, 61.5]	47.6	30.5	34.6
ox-alpha	52	52	50	28	100.0	56.0 [42.5, 69.1]	53.8	28.6	33.2
sonnet-5	60	60	57	37	100.0	64.9 [51.6, 75.9]	61.7	27.5	31.6

E Reproducibility and Provenance

Extraction under evaluator v3.2 ran on 31 August 2026 at code commit `94e7e38`, whose extraction code is identical to the pinned analysis commit `124f05b`. The finder and classifier both used Anthropic `claude-sonnet-5`; no second finder was enabled, following a calibration gate that found

the second finder's unique published-family yield to be zero. Calls forced a named JSON tool with a per-call output ceiling and left temperature at the provider default. The manifest records extractor version `v3.2.1`, prompt hash `014f3385...`, and schema hash `6c996f8d...`. The archived output holds 2,830 accepted claim rows across published and deferred families, 5,989 machine-negative rows, 13 logged rejects, and per-call token telemetry; the finder and classifier calls cost about US$34.84 at August 2026 list prices, a dated estimate computed from that telemetry. The second-model pass ran on 1 September 2026 under a flat-rate subscription; its prompts, schemas, event transcripts, raw fragments, and hashes are archived. The author's first pass, clarification overlay, final ratings, and approval record were frozen on 2 September 2026. Evaluator version `v3.2.3`, packet version `pf2-v3.2.5`.

The logs record, per attempt, the endpoint identifier, provider response identifier, latency, and token counts, and per seat, the prompt and tool-schema hashes and the framing. They do not record the provider-side sampling defaults in force on 25 August 2026, which cannot be recovered and are a replication limitation. The campaign's wall-clock window was 10:39 to 18:54 UTC on that day, in three parallel lanes on the author's workstation calling hosted APIs; no local model inference was performed.

F Result Definitions

For Town-seat ballot opportunities O, valid non-forced submitted ballots V, valid non-abstentions N, and ballots in N that target a living Mafia seat H: coverage $= V/O$, conditional quality $= H/N$, effective quality $= H/O$. Each baseline is the mean over the N ballots of the fraction of Mafia seats among that ballot's target set: all legal targets including the voter, or living non-self targets. Excess is conditional quality minus the baseline. Seeded 95% game-cluster bootstrap intervals are percentile intervals over 20,000 resamples of whole games with the pooled statistic recomputed; with 38 clusters they are approximate and may under-cover, and are reported as computed. The audit confirmation interval is the Wilson score interval (Wilson, 1927), reported descriptively for one 50-item sample. Distinct-claim ranges arise from uncertain linkage between claim occurrences and carry no sampling uncertainty.