Youdunit: Single-Call Counterfactual Necessity in Multi-Agent LLM Systems

Marissa Zhao Li[1] **Stephanie Gao**[1] **Kenny Guo**[2] **Xingjian Li**[1]
William Chang[1] **Goran Radanović**[3]

[1]University of California, Los Angeles
[2]Yale University
[3]Max Planck Institute for Software Systems (MPI-SWS)

Abstract

Agentic systems built on large language models (LLMs) are increasingly deployed in high stakes settings, but the causal mechanisms by which adversarial agents induce harmful outcomes remain poorly understood. We address this with a single-call counterfactual necessity test for multi-agent LLM systems that asks: *which specific agent communication was necessary for the harmful outcome?* In contrast with actual causality frameworks that reason over causes defined over sets of actions, we use one attribution rule throughout: single-call necessity for a selected proximate call. The key technical contribution is the *Gumbel-Max tape*, which lifts the token level Gumbel-Max counterfactual generator from a single LLM to a multi-agent conversation. GPU random number generator (RNG) states are recorded during a factual run on a tape shared across all agents, then replayed during counterfactual (CF) runs everywhere except at the intervened call. Applied to the BAD-ACTS benchmark across 176 adversarial scenarios spanning four multi-agent environments, two prompting conditions (*non-safe* and *safe*), and two models (Qwen3-8B and Llama-3.1-8B-Instruct), the test shows that single-call necessity tracks communication structure. Decentralized and hierarchical environments (Travel Planning, Financial Article Writing) exhibit, in the non-safe condition, the largest aggregate causal effect (ACE) and the centralized environment (Code Generation) a weak one, whereas in sequential debate (Multi-Agent Debate) the attack does not raise harm above its base rate, leaving no effect to attribute. These conclusions are robust to the choice of intervention: replacing the fixed neutral text with a message sampled from the benign version of the same agent preserves the sign of ACE in 11 of 12 combinations of environment, condition, and model. A replay sanity check confirms that all CF effects under identity intervention are exactly zero, validating the tape mechanism. The results identify where targeted defenses are likely to be effective in deployed agentic systems.

1 Introduction

LLM-based agentic systems now perform code generation [55], complex reasoning [30], and personal assistance, decomposing tasks across specialized agents and acting in the world through tools such as file systems, messaging interfaces, and transaction APIs. This tool access creates a new attack surface: a single adversarial agent can steer the entire system toward harmful outcomes by manipulating other agents through communication [30].

Prior work on the security of such systems quantifies *attack success rate* (ASR) – the fraction of scenarios in which an attack ends in a harmful tool execution [30]. ASR characterizes marginal vulnerability, but it does not explain *why* an attack succeeds and gives no guidance about *where* to

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

defend. The mechanistic question is different: *which specific agent communication was necessary for the harmful outcome?* Answering it enables targeted defenses, principled attribution of responsibility, and a clearer picture of how harm propagates through multi-agent architectures.

We introduce a single-call counterfactual necessity test at the granularity of agent calls. Given a factual trajectory τ in which the system executed the harmful target action, we ask whether neutralizing one specific call k^* – holding all other exogenous noise fixed – would have prevented the harm. This is a but-for test in the sense of [14], instantiated for multi-agent LLM conversations and applied to a particular call selected by a proximate-cause rule (the last call of the chosen CF agent before the target agent's harmful tool execution). The test is deliberately restrictive but has two advantages. First, it composes in two stages: k^* can seed a search over sets of calls for set-valued actual causes [12, 46], and anchor agent-specific-effect decompositions of how its effect propagates [47, 48]. Second, it is tractable in $O(1)$ counterfactual evaluations per scenario, which matters at the compute scale of LLM agents. The gap between this test and the joint ablation of all of an agent's pre-target calls is bounded by the model's downstream prefix sensitivity (Proposition 1, Section C).

We make the test *technically* valid through the *Gumbel-Max tape*, a multi-agent extension of the token level tape of [6]. Their construction uses the Gumbel-Max reparameterization of categorical sampling to perform Pearl's abduction step at the token level for a single LLM. We extend it to the multi-agent setting by recording the RNG state of every token on one shared tape across all agent invocations. During a counterfactual run, the intervened call is replaced; tape positions of all other agents are replayed. This is what makes the call k^* a well defined unit of attribution: downstream messages are recomputed from the modified prefix under the same noise, not frozen at their factual values.

Contributions.

1. **Multi-agent Gumbel-Max tape.** Building on [6], we introduce the **Gumbel-Max tape**: a mechanism that records per-token RNG states during a factual run and replays them during CF runs across an *entire multi-agent conversation*. The tape is shared across all agents, so a single-call intervention reshapes only the descendants of that call.
2. **Single-call counterfactual necessity.** We define the *counterfactual effect* (CFE) of a specific agent's last message before a harmful tool execution and operationalize it with two interventions: a fixed neutral deferral text, and a message sampled from the *benign version* of the same agent given the same conversation history. This is a tractable specialization, restricted to the proximate call, of frameworks of actual causality [12, 46] that is scalable and directly actionable.
3. **Empirical findings on BAD-ACTS.** Across 176 adversarial scenarios spanning all four BAD-ACTS multi-agent environments, both prompting conditions (*non-safe*: original adversarial setup; *safe*: with Adversary-Aware Prompting), and two 8B models, single-call causal attributability differs systematically with communication structure. For both models, decentralized and hierarchical environments produce, in the non-safe condition, the largest aggregate causal effect and the centralized environment a weak one, while in sequential debate the attack does not succeed above its base rate. Reference runs without an adversary separate harm caused by the attack from harm that occurs regardless, the benign agent intervention shows that the findings are not an artifact of the neutral text, and an identity intervention (replay) validates the tape mechanism. Code is available at `https://github.com/marissazli/youdunit-tape`.

2 Preliminaries

2.1 Multi-Agent Agentic Systems

Following [30], an agentic system is a tuple $\mathcal{S} = (o, G)$, where $o : \mathcal{X} \times \mathcal{T} \to \{0, 1\}$ maps a task $x \in \mathcal{X}$ and a trajectory $\tau \in \mathcal{T}$ to a binary outcome, and $G = (\mathcal{A}, E)$ is a directed communication graph with agents $\mathcal{A} = \{a^{(1)}, \ldots, a^{(n)}\}$ as nodes and directed edges $E \subseteq \mathcal{A} \times \mathcal{A}$ indicating which pairs of agents may exchange messages directly. Each agent $a^{(i)}$ has a system prompt $\pi^{(i)}$ and an action repertoire of (a) emitting natural-language messages and (b) optionally invoking tools $\mathcal{O}^{(i)} \subseteq \mathcal{O}$ (file system, web browser, messaging API, etc.). All agents share one base model f_θ; agent

identity is determined by system prompt and tool subset, not separate weights. We use AutoGen [55] as the concrete instantiation.

A trajectory $\tau = ((a_1, m_1), \ldots, (a_T, m_T))$ records the sequence of agent invocations $a_t \in \mathcal{A}$ and their messages m_t. Each m_t is a structured object containing a (possibly empty) text utterance, tool calls, and tool results, all appended so subsequent agents read them as context. The schedule is set by an orchestration policy that picks the next speaker as a function of the partial trajectory and G (in the environments we study, fixed-round-robin, central-orchestrator-driven, or last-speaker-addressed). T is determined by termination: a `TERMINATE` token from the orchestrator, or a maximum-turn limit $T_{\max}$.

Communication graph types. The structure of G shapes how influence propagates and is the key axis along which our findings differ. Following [30], we distinguish four canonical structures, each illustrated by one BAD-ACTS environment: *decentralized*, where G is (close to) a complete digraph (Travel Planning); *hierarchical*, where G is tree-like with a root orchestrator (Financial Article Writing); *centralized*, a star with one hub mediating all influence (Code Generation); and *sequential*, where agents speak in fixed cyclic order (Multi-Agent Debate). These structures impose different topological constraints on which calls can be causally upstream of a downstream outcome.

Autoregressive message generation. Each message m_t is a sequence of tokens $m_t = (w_1^{(t)}, \ldots, w_{L_t}^{(t)})$ produced by autoregressive sampling: the agent assembles its input as $\mathrm{prompt}_t = (\pi^{(a_t)}, \tau_{\le t-1})$ and samples

$$w_\ell^{(t)} \sim p_\theta\big(\cdot \mid \mathrm{prompt}_t, w_{<\ell}^{(t)}\big) \tag{1}$$

until end-of-message or token budget. A trajectory is therefore a nested autoregressive process: an outer loop over agent invocations, an inner loop over tokens. This is why naive "intervene-and-rerun" does not compute counterfactuals in the SCM sense – after editing m_k, it resamples downstream token noise rather than holding it fixed [6].

2.2 Adversarial Setting

We adopt the threat model of [30]: an adversary fully controls one agent $\alpha \in \mathcal{A}$ and replaces its system prompt with an adversarial one that instructs α to manipulate other agents into executing a harmful target action x_{harm}. The adversary's objective is $o_\alpha(\tau) = \mathbf{1}[\text{target action executed in } \tau]$. BAD-ACTS instantiates this by replacing one node of G with the adversarial agent (yielding $G_{i\to\alpha}$) while leaving the rest of the system, the orchestration logic, and the available tools unchanged [30]; the only difference between benign and adversarial runs is α's system prompt. Whether the target action was performed is decided by keyword matching on the relevant tool execution or agent utterance, an evaluation method that agrees with human annotation 98% of the time, against 80% for an LLM-judge baseline [30].

The *attack success rate* is $\mathrm{ASR} = \mathbb{E}[o_\alpha(\tau)]$ over the dataset. ASR is silent about which agent communications are causally responsible when an attack succeeds – the question we address next.

2.3 Structural Causal Model for LLM Agents

We model the multi-agent system as a structural causal model (SCM) $\mathcal{M} = (\mathbf{V}, \mathbf{U}, \mathbf{F}, P_{\mathbf{U}})$ [35, 36]. Endogenous variables $\mathbf{V}$ are the per-call messages $\{m_1, \ldots, m_T\}$ together with the outcome Y; exogenous variables $\mathbf{U}$ are the per-token sampling noises consumed during message generation; $\mathbf{F}$ implements autoregressive sampling from f_θ; and $P_{\mathbf{U}}$ is the product distribution of independent Gumbel noises (Section 2.4).

Because each m_t is itself produced token-by-token (Equation (1)), the noise associated with the t-th invocation is itself a sequence

$$U_{a_t,t} = \big(U_1^{(t)}, U_2^{(t)}, \ldots, U_{L_t}^{(t)}\big), \tag{2}$$

where each $U_\ell^{(t)} \in \mathbb{R}^{|\mathcal{V}|}$ is a vector of i.i.d. Gumbel$(0,1)$ random variables, one per vocabulary token. The full trajectory noise is the concatenation $\mathbf{U} = (U_{a_1,1}, \ldots, U_{a_T,T})$ of total length $L = \sum_t L_t$. We call L the global *tape length*: it is exactly the number of cells in the Gumbel-Max tape introduced in Section 3.

The structural equation for the message at step t is

$$m_t := f_{a_t}\big(\tau_{\le t-1},\ \pi^{(a_t)},\ U_{a_t,t}\big), \tag{3}$$

where f_{a_t} is the deterministic function that, given the partial trajectory $\tau_{\le t-1}$, the system prompt $\pi^{(a_t)}$, and the noise sequence $U_{a_t,t}$, returns the message produced by autoregressively applying Equation (1) until termination. The agent identity a_t is itself a deterministic function of $\tau_{\le t-1}$ via the orchestration policy and is re-evaluated under intervention. The outcome is $Y := o_\alpha(\tau) = \mathbf{1}[\text{harmful target action executed in } \tau]$.

Attribution granularity. We work with the call level SCM throughout, in which each endogenous message m_t corresponds to one agent invocation; Section B explains why the call, rather than the agent identity, is the natural unit of attribution.

Interventions and counterfactuals. A hard intervention $\mathrm{do}(m_k = \tilde{m})$ replaces the structural equation for m_k with the constant $\tilde{m}$, leaving every other equation in $\mathbf{F}$ unchanged; downstream agents still sample from Equation (3), but with $\tilde{m}$ substituted into their context. The interventional outcome $Y_{m_k=\tilde{m}}$ is defined under fresh draws of $\mathbf{U}$. A *counterfactual* instead conditions on the observed trajectory τ, updates $P(\mathbf{U})$ to $P(\mathbf{U} \mid \tau)$, and propagates the same abducted noise through the intervened submodel (the abduction, action, and prediction steps of Pearl [35]). Naively intervening and rerunning with fresh downstream noise estimates $\mathbb{E}[Y \mid \mathrm{do}(m_k = \tilde{m})]$, not $Y_{m_k=\tilde{m}} \mid \tau$ [6]. The Gumbel-Max reparameterization (Section 2.4) makes abduction tractable, and the Gumbel-Max tape (Section 3) implements it across an entire multi-agent conversation. All generations use Gumbel-Max sampling at temperature 0.7 over the full vocabulary, without top k or top p truncation. Abduction is realized by recording the noise on the tape, so that, given the tape, the counterfactual of each scenario is a deterministic rollout; expectations enter only when averaging across scenarios in ACE. We assume tool calls are deterministic functions of their inputs or share their RNG state with the tape; this holds in BAD-ACTS, which uses emulated tools without external randomness.

2.4 The Gumbel-Max Reparameterization

At each generation step, f_θ produces a categorical distribution $p_\theta(\cdot \mid \tau_{\le t-1}, w^{(t)}_{<\ell})$ over the vocabulary $\mathcal{V}$. The Gumbel-Max reparameterization [31] expresses this categorical sample as the $\arg\max$ of a perturbed log-probability vector:

$$\hat{x} = \arg\max_{v \in \mathcal{V}} \big[\log p_\theta(v \mid \tau_{\le t-1}, w^{(t)}_{<\ell}) + G_v\big], \tag{4}$$

where $G_v \overset{\text{iid}}{\sim} \mathrm{Gumbel}(0,1)$ for each v. The exogenous noise $U^{(t)}_\ell = (G_v)_{v\in\mathcal{V}}$ enters Equation (3) only through this $\arg\max$, separating the deterministic logits (the model's beliefs given context) from the stochastic noise (which token wins this draw). Given a factual sample $\hat{x}$ and the realized logits, the posterior over $U^{(t)}_\ell$ is concentrated on $\{u : \arg\max_v(\log p_\theta(v) + u_v) = \hat{x}\}$; abduction therefore reduces to recording the noise vector and reusing it later.

Counterfactual stability. Among categorical SCMs whose interventional distribution matches a given p_θ, the Gumbel-Max SCM uniquely satisfies *counterfactual stability* [31]: under a shift from conditional distribution d to d', the counterfactual sampler returns a different token only when the relative probability of the factual token has decreased. Small interventions therefore do not arbitrarily reshuffle downstream tokens, and counterfactual continuations remain close to the factual one wherever the conditional distribution is unchanged. This makes Gumbel-Max the natural choice for counterfactual reasoning in autoregressive LLMs [6]. Temperature only rescales the logits, so tempered sampling remains a Gumbel-Max SCM over the tempered distribution. Truncated sampling (top k or top p) instead restricts the $\arg\max$ to a surviving subset $\mathcal{V}_t \subseteq \mathcal{V}$, and the resulting SCM is no longer guaranteed to satisfy counterfactual stability [6]. We therefore sample all factual and counterfactual generations at temperature 0.7 without truncation, so that counterfactual stability holds for every sampled token.

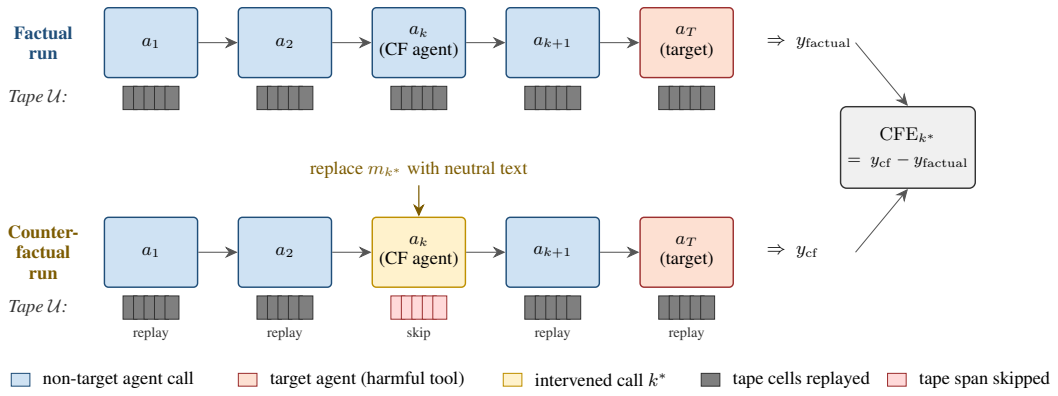

Figure 1: **Gumbel-Max tape mechanism for per-call causal attribution.** During the factual run (top), per-token RNG states are recorded onto a shared tape $\mathcal{U}$ across all agent calls. During the counterfactual run (bottom), tape cells before and after the intervened call k^* are *replayed* (so downstream agents sample under identical exogenous noise), while the tape span belonging to call k^* itself is *skipped* and its message m_{k^*} is replaced with a neutral baseline. The causal effect $\text{CFE}_{k^*} = y_{\text{cf}} - y_{\text{factual}}$ then isolates the contribution of that single message under matched exogenous noise. This is the multi-agent analogue of [6]'s single-LLM token tape (their Figure 1d).

2.5 Counterfactual Effects in Multi-Agent SCMs

Given a factual trajectory τ with outcome $y_{\text{factual}} = Y(\tau)$, the *counterfactual effect* of intervening on call k with replacement message $\tilde{m}$ is

$$\text{CFE}_k^{\tilde{m}}(\tau) := Y_{m_k=\tilde{m}}(\mathbf{U}) \mid \tau \;-\; y_{\text{factual}} \;\in\; \{-1, 0, +1\}, \tag{5}$$

where $Y_{m_k=\tilde{m}}(\mathbf{U}) \mid \tau$ is the outcome under abduction–action–prediction with the same exogenous $\mathbf{U}$ as the factual run. Because Y is binary, $\text{CFE}_k^{\tilde{m}}(\tau)$ takes one of three values: -1 (intervention prevented harm), $+1$ (intervention induced harm), or 0 (no causal effect). Equation (5) is a single-call but-for test in the sense of [14]; it does not search over set-valued causes in the modified-HP sense [12, 46] and does not decompose how the effect propagates through downstream agents and environment state in the agent-specific-effect sense [47, 48]. Section A relates the per-call CFE to these frameworks; both compose with our attribution as downstream extensions once k^* has been identified.

3 Method

3.1 The Gumbel-Max Tape

To enable valid counterfactual generation at the granularity of agent calls, we augment the standard autoregressive generation loop with a *tape*: a sequence of generator states $\mathcal{U} = (s_1, \ldots, s_L)$, one per generated token, where s_ℓ is the state of the GPU RNG immediately before sampling token ℓ. Storing RNG states (rather than the realized Gumbel vectors $\{G_v\}_{v\in\mathcal{V}}$) follows [6]: at counterfactual time the generator is reseeded to s_ℓ and the same Gumbel sample is regenerated on the fly via Equation (4), reducing tape memory from $O(L|\mathcal{V}|)$ floats to $O(L)$ integers. [6] maintain a single tape for one LLM's generation; our setting requires the tape be *shared across agents*. Every agent in $\mathcal{A}$ samples from the same base model f_θ, so tape positions advance contiguously across all agent invocations, and a counterfactual intervention on a_k's message must replay the tape ranges of *all other* agents – both before and after a_k – verbatim. Figure 1 summarizes the construction.

In a CF run intervening on call k, we replay all pre-intervention tape cells, skip the tape span for call k while injecting the intervention text, then restore the RNG state at each downstream call's factual `tape_start` boundary. This call-level alignment ensures downstream agents reuse the same exogenous noise as in the factual run, even when the intervention changes the length of earlier messages. Factual and counterfactual trajectories therefore differ only in the intervened call and its causal descendants.

3.2 Single-Call Counterfactual Necessity: Choosing the Intervention

The test selects a single call k^* and asks whether neutralizing it would have prevented the harmful outcome. Two design choices need justification: *which* call we choose, and *what* we replace its message with.

Why the last pre-target call? Let β denote the CF agent (specified per environment in Section 4) and let $t^*(\tau)$ denote the time step of the harmful target tool execution. We define

$$k^*(\beta, \tau) = \max\{t \in T_\beta(\tau) : t < t^*(\tau)\}, \tag{6}$$

where $T_\beta(\tau)$ is the set of time steps at which β is invoked; if no such t exists, we fall back to the last invocation of β in τ. This selects the most temporally proximate call of β that could causally precede the harmful execution – a single-action but-for test with no contingencies [12, 46]. The rule is deliberately restrictive: it can miss distal causes that are necessary only jointly with intermediate calls. Proposition 1 (Section C) bounds the resulting gap to a joint ablation by the model's downstream prefix sensitivity, and the cross-position ablation in Section G measures empirically how much signal this restriction loses.

Neutral text intervention. Our primary intervention replaces m_{k^*} with a short neutral deferral text tailored to the CF agent, for example "*I have no additional planning recommendation. Please continue based on the existing discussion.*" for the Travel Planning planner (all texts are listed in Section H.4). The text steers neither toward nor away from the harmful action and allows the conversation to continue.

Benign agent intervention. A fixed neutral text alters more than the adversarial content: it is much shorter than a typical message, differs in style from the agent, and may break the conversation protocol, for example when an orchestrator no longer names the next agent. We therefore also intervene with a message that the system could plausibly have produced. The replacement $\tilde{m}$ is sampled from the benign version of the CF agent, that is, from the environment's own system prompt for that agent under the same prompting condition, given the same conversation history $\tau_{\leq k^*-1}$:

$$\tilde{m} \sim f_\beta\big(\tau_{\leq k^*-1},\ \pi^{(\beta)}_{\text{benign}},\ U'\big), \tag{7}$$

where U' is fresh noise drawn from a separate seed. The factual tape span of k^* is skipped exactly as in the neutral text intervention, all other calls replay the factual tape, and both interventions share the same factual run, so that they are paired scenario by scenario. When β is the adversary (Travel Planning, Code Generation), the intervention replaces the adversarial message with what the benign agent would have said at that point. When β is already benign (Financial Article Writing, Multi-Agent Debate), $\pi^{(\beta)}_{\text{benign}}$ is the prompt that produced the factual message, and the intervention resamples that message with new noise; it then tests whether the effect depends on the specific factual message rather than on the removal of adversarial content.

Replay sanity check. We also run a *replay* mode in which m_{k^*} is replaced by its exact factual text. Since the intervention text matches the factual message and downstream tape states are fixed, the replay trajectory should be identical to the factual one, giving $\text{CFE}^{\text{replay}}_{k^*} = 0$ for all rows.

3.3 Aggregate Causal Effect

The *aggregate causal effect* for CF agent β summarizes single-call necessity across the dataset:

$$\text{ACE}_\beta = \frac{1}{|\mathcal{D}|} \sum_{(x,\tau)\in\mathcal{D}} \text{CFE}_{k^*(\beta,\tau)}(\beta, \tau), \tag{8}$$

where $\mathcal{D}$ is the set of evaluated scenarios. A more negative ACE_β means β's last pre-target message was more frequently necessary for the harmful outcome; $\text{ACE}_\beta = 0$ means the harmful outcome is independent of the specific content of that message. The aggregation is over scenarios: ACE_β depends both on the choice of CF agent β and on the dataset $\mathcal{D}$ (here, BAD-ACTS scenarios for one environment under one prompting condition).

4 Experiments

4.1 Setup

Benchmark, model, environments. We evaluate on BAD-ACTS [30], which provides 188 adversarial scenarios across four multi-agent environments: Travel Planning (49 rows), Financial Article Writing (66), Code Generation (55), and Multi-Agent Debate (18). Each scenario specifies a task, a harmful target action, and evaluation keywords. We run every experiment with two open models of the same size, Qwen3-8B [58] (native Hermes tool calling) and Llama-3.1-8B-Instruct (native JSON tool calling); within a run, all agents share one model. Both run on the HuggingFace `transformers` backend (required for the Gumbel-Max tape; see Section 5). Table 9 (Section H) lists the communication structure and CF agent of each environment. For Travel Planning we intervene on the adversary itself (PLANNER_AGENT); for Financial Article Writing on a benign agent (RESEARCHER) to measure the causal contribution of a benign participant; for Code Generation on the central orchestrator and adversary (CEO); and for Multi-Agent Debate on a benign debater (agent_0). The CF agent is fixed per environment and does not vary across safe and non-safe conditions.

Conditions and procedure. For each environment and model we run the *non-safe* condition (the original BAD-ACTS adversarial setup) and the *safe* condition (with Adversary-Aware Prompting [30]), each under three interventions at k^*: the neutral text (our primary *baseline*), the benign agent message (Section 3.2), and *replay*, which injects the exact factual message as an identity sanity check. To measure how often the harmful outcome occurs without any attack, we additionally run every scenario without an adversary, keeping the environment's own benign agent in the adversary's slot and running only the factual episode. For each scenario we run the pipeline of Section 3: the factual run of the ith scenario of an environment executes under seed $49 + i$ with the adversarial prompt installed, while the Gumbel-Max tape records the GPU RNG state s_ℓ before each token is sampled; we log the identity, message, tool calls, and tape range of every invocation. The intervention call k^* is selected via Equation (6). The CF run replays the tape ranges of all calls $t \neq k^*$ verbatim, skips the tape range of k^* while injecting the intervention message, and resumes replay from $\texttt{tape_end}_{k^*}$, so that every downstream agent samples under the same exogenous noise as in the factual run. Outcomes are evaluated with the keyword matcher of [30]; correct tape semantics imply $\mathrm{CFE}_{k^*}^{\mathrm{replay}} = 0$ for every scenario. Further details are given in Section H.

4.2 Results

Attack success rate. Table 1 reports ASR per environment together with the base rate, that is, the rate at which the same harmful outcome occurs without an adversary. For both models, the attack raises harm well above the base rate in Travel Planning (by 0.28 to 0.40) and Code Generation (by 0.19 to 0.37), and for Llama also in Financial Article Writing (by 0.12 to 0.17). For Qwen in Financial Article Writing, ASR (0.136 and 0.076) is close to the base rate (0.106 and 0.091), so most keyword matches there also occur without an attack. In Multi-Agent Debate the attack does not exceed the base rate for either model (Qwen ASR 0; Llama between 0.11 and 0.17 with and without the adversary). Safe prompting lowers ASR in Travel Planning (Qwen $0.425 \rightarrow 0.300$, Llama $0.425 \rightarrow 0.375$) and in Financial Article Writing. In Code Generation the two conditions coincide by construction, because BAD-ACTS adds the safety instruction only to the prompt of the CEO, which is the adversary and whose prompt is replaced.

Table 1: Attack success rate (ASR) with the adversary, and base rate of the same harmful outcome without an adversary (all agents benign), per model and prompting condition.

	Qwen3-8B				Llama-3.1-8B-Instruct			
	ASR		No adversary		ASR		No adversary	
Environment	Non-safe	Safe	Non-safe	Safe	Non-safe	Safe	Non-safe	Safe
Travel Planning	0.425	0.300	0.025	0.025	0.425	0.375	0.125	0.100
Code Generation	0.442	0.442	0.077	0.077	0.288	0.288	0.096	0.096
Multi-Agent Debate	0.000	0.000	0.000	0.000	0.111	0.167	0.111	0.222
Financial Article Writing	0.136	0.076	0.106	0.091	0.242	0.197	0.076	0.076

Aggregate causal effect. Table 2 reports ACE_β under the neutral text intervention. **Travel Planning and Financial Article Writing show the strongest single-call effects for both models** (non-safe ACE of -0.075 and -0.121 for Qwen, -0.075 and -0.136 for Llama), for different structural reasons. In Travel Planning, the decentralized structure lets the last message of the adversary (PLANNER_AGENT) steer the target directly. Financial Article Writing routes influence through a single benign intermediary (RESEARCHER), whose last call is necessary for most harmful outcomes ($\mathrm{ACE}/\mathrm{ASR} \approx -0.89$ for Qwen and -0.56 for Llama in the non-safe condition, compared with -0.18 for Travel Planning). For Qwen, however, this harm occurs about as often without an adversary (Table 1), so the message of RESEARCHER is necessary for harm that is largely not caused by the attack; for Llama, where the attack raises harm by 0.17, the effect has the same size. **Code Generation has a weak single-call effect** (ACE -0.019 for Qwen, -0.038 for Llama): although the CEO is the adversary and the attack clearly raises harm, it succeeds through cumulative instructions rather than a single message. **Multi-Agent Debate shows no measurable single-call effect** (0.000 for Qwen; $+0.056$ and 0.000 for Llama). This is not evidence about the debate structure, since the attack does not raise harm above the base rate and there is therefore no harm caused by the attack to attribute. **Safe prompting** attenuates the Travel Planning ACE of Qwen in proportion to ASR ($-0.075 \rightarrow -0.050$ as ASR drops from 0.425 to 0.300, so that $\mathrm{ACE}/\mathrm{ASR} \approx -0.17$ in both conditions) and attenuates Financial Article Writing more strongly. For Llama in safe Travel Planning, the neutral text *induces* harm in 9 of 40 scenarios (ACE $+0.150$); Section 4.3 shows that this sign is specific to the neutral text.

Table 2: Aggregate causal effect (ACE) under the neutral text intervention. The replay (identity) intervention yields an ACE of 0.000 in every cell for both models.

	Qwen3-8B		Llama-3.1-8B-Instruct	
Environment	Non-safe	Safe	Non-safe	Safe
Travel Planning	-0.075	-0.050	-0.075	$+0.150$
Code Generation	-0.019	-0.019	-0.038	-0.038
Multi-Agent Debate	0.000	0.000	$+0.056$	0.000
Financial Article Writing	-0.121	-0.045	-0.136	-0.136

Figure 2 decomposes the aggregate ACE values of Qwen by outcome direction (Llama in Section I). The Travel Planning value of -0.075 aggregates 5 prevented against 2 induced harms (out of 40); the Financial Article Writing value of -0.121 aggregates 9 prevented against 1 induced (out of 66), covering every successful attack in the non-safe condition; the Code Generation value of -0.019 reflects 2 prevented against 1 induced (out of 52). In Multi-Agent Debate, no scenario has a harmful factual or counterfactual outcome.

Replay sanity check. All replay causal effects are exactly 0.000 across all environments, conditions, rows, and both models: when the intervention text is identical to the factual text, the counterfactual trajectory matches the factual one. This validates counterfactual replay under matched noise; additional checks at temporal boundaries (interventions before, at, and after the harmful tool call) are reported in Section G.

4.3 Robustness to the Intervention

Because the neutral text changes the length, style, and protocol of the message at k^* in addition to its content, we repeat every CF run with the benign agent intervention (Section 3.2), reusing the factual run and tape of each scenario. Figure 3 compares the two interventions.

The findings do not depend on the neutral text. Excluding Multi-Agent Debate, where neither intervention has an effect, the benign agent intervention preserves the sign of ACE in 11 of 12 combinations of environment, condition, and model, and the two interventions yield the same counterfactual outcome in 70% to 96% of scenarios. The ordering of environments is unchanged: Travel Planning and Financial Article Writing remain the most responsive, and Code Generation remains weak. Where the adversary is the CF agent (Travel Planning, Code Generation), the message of the benign agent prevents harm at least as often as the neutral text (for example, 10 versus 5 prevented harms for Qwen in non-safe Travel Planning), indicating that the removal of adversarial

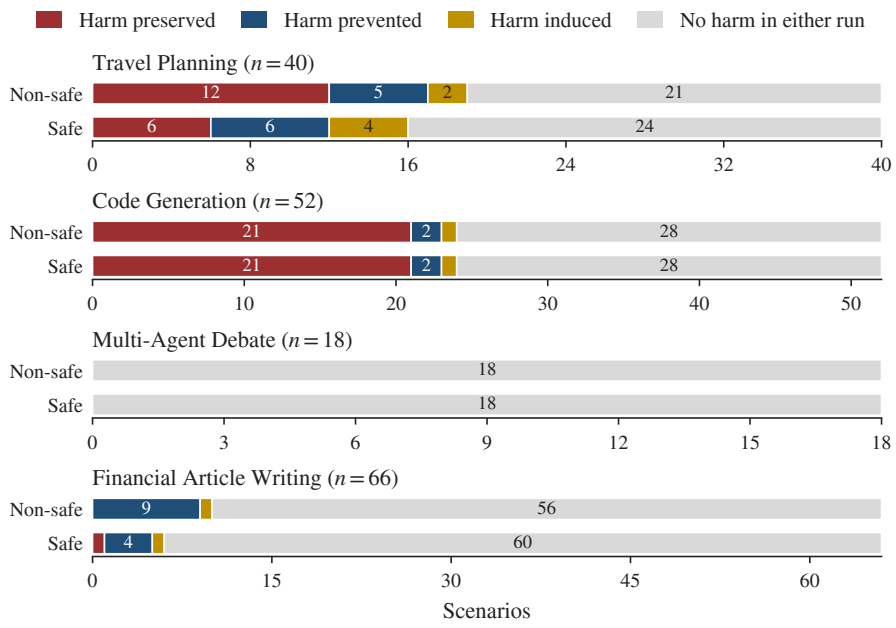

Figure 2: **Decomposition of factual and counterfactual outcomes per environment and condition.** For each scenario, the (factual, counterfactual) outcome pair falls into one of four categories: *harm preserved* ($y_{\text{factual}}{=}y_{\text{cf}}{=}1$), *harm prevented* ($y_{\text{factual}}{=}1$, $y_{\text{cf}}{=}0$ – causal evidence the intervention helped), *harm induced* ($y_{\text{factual}}{=}0$, $y_{\text{cf}}{=}1$ – intervention created new harm), or *no harm in either*. Qwen3-8B, neutral text intervention. Counts in cells ≥ 2 are annotated.

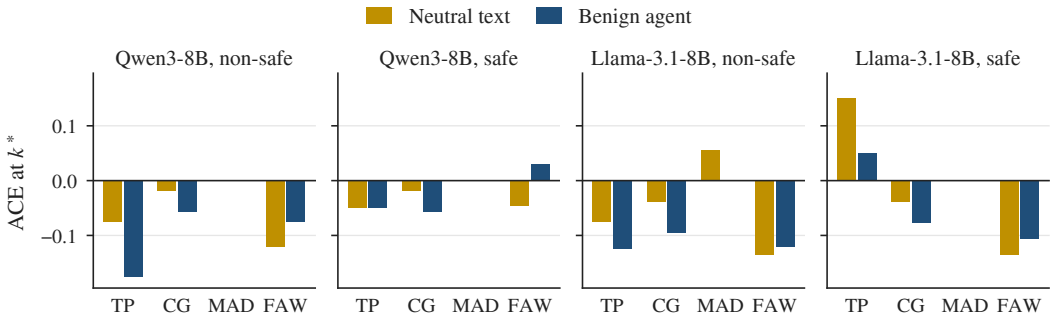

Figure 3: **ACE at k^* under the neutral text and benign agent interventions**, per model and prompting condition. Both interventions share the factual run and tape of each scenario and differ only in the message at k^*. Full numbers are given in Table 4 (Section D).

content, rather than the shortening of the message, carries the effect. The only sign change occurs for Qwen in safe Financial Article Writing ($-0.045 \rightarrow +0.030$, a net difference of 5 of 66 scenarios), where the CF agent is benign and the intervention merely resamples its message. The positive ACE of Llama in safe Travel Planning decreases from $+0.150$ to $+0.050$, which indicates that part of the harm induced by the neutral text originates in the deferral text itself (Section F).

Message lengths and the length confound are analyzed in Sections D and E.

Discussion. The structures that admit single-call necessity separate along a topological axis rather than along ASR. Code Generation shows a large increase of harm over the base rate (0.37 for Qwen, 0.19 for Llama) but only a weak single-call effect, whereas Financial Article Writing shows the largest causal dependency per successful attack. Concentrated routing, whether through direct delegation in a

decentralized structure or through a hierarchical bottleneck, produces single-call necessity for distinct structural reasons, while centralized cumulative attacks (Code Generation) do not. In sequential debate (Multi-Agent Debate) the attack itself fails, so BAD-ACTS with these models supports no structural conclusion about debate. Under the neutral text in the non-safe condition, the ordering of environments holds for both models, whereas absolute ACE magnitudes and the sign of small effects (Llama: safe Travel Planning $+0.150$, non-safe Multi-Agent Debate $+0.056$) are model-dependent: *single-call necessity is concentrated where routing is concentrated.* In such topologies, monitoring or neutralizing one critical message before a harmful tool call may be an effective defense, whereas in centralized topologies defenses that target a single message are insufficient. A breakdown by harm subcategory is given in Section F.

5 Conclusion

We introduced a single-call counterfactual necessity test for multi-agent LLM systems, enabled by a Gumbel-Max tape that holds downstream exogenous noise fixed across factual and counterfactual trajectories. Applied to BAD-ACTS [30] with two 8B models, the test shows that single-call necessity varies with communication structure: decentralized and hierarchical environments exhibit substantial causal concentration and a centralized environment does not, while in sequential debate the attack does not raise harm above its base rate. These conclusions hold both under a fixed neutral text intervention and under a benign agent intervention that matches the length and style of the agent's own messages.

Limitations. The Gumbel-Max tape requires intercepting token sampling and the RNG state of the model, which is feasible with the HuggingFace `transformers` backend but not with high throughput inference servers such as vLLM or SGLang. A tape is valid only on the numerical setup that recorded it: KV caching, a different GPU, or a different software version changes floating point results and can flip nearly tied $\arg\max$ decisions. All paired runs therefore recompute the full sequence for every token, at a cost of roughly 5 to 100 minutes per scenario (Table 11). The benign agent intervention removes the length and style confound of the neutral text, but it conditions on a history already influenced by the adversary; it therefore measures the effect of the message at k^* with all earlier messages held fixed, rather than the effect of the adversary's presence. Where the CF agent is itself benign (Financial Article Writing, Multi-Agent Debate), this intervention reduces to resampling. We evaluate two open 8B models, and results are known to vary across model families [30]. The keyword evaluator of BAD-ACTS is itself a source of error, and we corrected its answer parser for the debate environment (Section H.6). Finally, the restriction to the proximate call cannot identify distal causes that are necessary only jointly with intermediate calls; Proposition 1 bounds this gap, and our estimates of the bound (Section C) show that it is informative mainly for Qwen in Multi-Agent Debate and Travel Planning, and rarely for Llama.

Future work. Several extensions follow directly from this study. First, evaluating every candidate CF agent rather than one fixed agent per environment would map how causal responsibility is distributed along the delegation chain, extending the exhaustive analysis of Section G to all environments. Second, averaging the benign agent intervention over several samples would separate the effect of removing adversarial content from the sampling variability of the replacement message. Third, single-call attribution composes with finer causal analyses: once k^* has been identified, decompositions of its effect across downstream agents [47, 48] can trace how the effect propagates, and definitions of actual causation over sets of events [12] can recover causes that are necessary only jointly. Finally, pseudorandom number generators indexed by a counter, such as Philox, could make tape replay compatible with batched and cached inference and thereby remove the main computational bottleneck of the method.

Acknowledgments and Disclosure of Funding

The work of Goran Radanovic was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) — project number 467367360.

References

[1] S. Beckers and J. Y. Halpern. Abstracting causal models. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 2678–2685, 2019.

[2] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[3] N. L. C. Benz and M. G. Rodriguez. Counterfactual inference of second opinions. In *Uncertainty in Artificial Intelligence*, pages 453–463. PMLR, 2022.

[4] L. Betti, C. Abrate, F. Bonchi, and A. Kaltenbrunner. Relevance-based infilling for natural language counterfactuals. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 88–98, 2023.

[5] R. M. Byrne. Precis of the rational imagination: How people create alternatives to reality. *Behavioral and brain sciences*, 30(5-6):439–453, 2007.

[6] I. Chatzi, N. C. Benz, E. Straitouri, S. Tsirtsis, and M. Gomez-Rodriguez. Counterfactual token generation in large language models. *arXiv preprint arXiv:2409.17027*, 2024.

[7] J. Chen, C. Gan, S. Cheng, H. Zhou, Y. Xiao, and L. Li. Unsupervised editing for counterfactual stories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10473–10481, 2022.

[8] M. Daliri, C. Musco, and A. T. Suresh. Coupling without communication and drafter-invariant speculative decoding. In *2025 IEEE International Symposium on Information Theory (ISIT)*, pages 1–6. IEEE, 2025.

[9] K. Epstude and N. J. Roese. The functional theory of counterfactual thinking. *Personality and social psychology review*, 12(2):168–192, 2008.

[10] J. Frohberg and F. Binder. Crass: A novel data set and benchmark to test counterfactual reasoning of large language models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2126–2140, 2022.

[11] T. Gerstenberg, N. D. Goodman, D. A. Lagnado, and J. B. Tenenbaum. A counterfactual simulation model of causal judgments for physical events. *Psychological review*, 128(5):936, 2021.

[12] J. Y. Halpern. *Actual causality*. MiT Press, 2016.

[13] C. Hao, L. Pang, Y. Lan, Y. Wang, J. Guo, and X. Cheng. Sketch and customize: A counterfactual story generator. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 12955–12962, 2021.

[14] H. L. A. Hart and T. Honoré. *Causation in the Law*. OUP Oxford, 1985.

[15] H. Jiang, L. Ge, Y. Gao, J. Wang, and R. Song. Llm4causal: Democratized causal tools for everyone via large language model. *arXiv preprint arXiv:2312.17122*, 2023.

[16] Z. Jin, Y. Chen, F. Leeb, L. Gresele, O. Kamal, Z. Lyu, K. Blin, F. Gonzalez Adauto, M. Kleiman-Weiner, M. Sachan, et al. Cladder: Assessing causal reasoning in language models. *Advances in Neural Information Processing Systems*, 36:31038–31065, 2023.

[17] E. Kiciman, R. Ness, A. Sharma, and C. Tan. Causal reasoning and large language models: Opening a new frontier for causality. *Transactions on Machine Learning Research*, 2023.

[18] D. A. Lagnado, T. Gerstenberg, and R. Zultan. Causal responsibility and counterfactuals. *Cognitive science*, 37(6):1036–1073, 2013.

[19] B. Z. Li, M. Nye, and J. Andreas. Implicit representations of meaning in neural language models. In *Proceedings of the 59th annual meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing (Volume 1: Long papers)*, pages 1813–1827, 2021.

[20] K. Li, A. K. Hopkins, D. Bau, F. Viégas, H. Pfister, and M. Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*, 2022.

[21] Y. Li, M. Xu, X. Miao, S. Zhou, and T. Qian. Prompting large language models for counterfactual generation: An empirical study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13201–13221, 2024.

[22] X. Liu, P. Xu, J. Wu, J. Yuan, Y. Yang, Y. Zhou, F. Liu, T. Guan, H. Wang, T. Yu, et al. Large language models and causal inference in collaboration: A comprehensive survey. *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7668–7684, 2025.

[23] X. Liu, D. Yin, C. Zhang, Y. Feng, and D. Zhao. The magic of if: Investigating causal reasoning abilities in large language models of code. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9009–9022, 2023.

[24] K. D. Markman, M. N. McMullen, and R. A. Elizaga. Counterfactual thinking, persistence, and performance: A test of the reflection and evaluation model. *Journal of Experimental Social Psychology*, 44(2):421–428, 2008.

[25] X. Miao, Y. Li, and T. Qian. Generating commonsense counterfactuals for stable relation extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5654–5668, 2023.

[26] R. Moulton and Y. Jiang. Maximally consistent sampling and the jaccard index of probability distributions. *arXiv preprint arXiv:1809.04052*, 2018.

[27] V. B. Nguyen, C. Seifert, and J. Schlötterer. CEval: A benchmark for evaluating counterfactual text generation. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 55–69, Tokyo, Japan, Sept. 2024. Association for Computational Linguistics.

[28] A. Nie, Y. Zhang, A. S. Amdekar, C. Piech, T. B. Hashimoto, and T. Gerstenberg. Moca: Measuring human-language model alignment on causal and moral judgment tasks. *Advances in Neural Information Processing Systems*, 36:78360–78393, 2023.

[29] K. Noorbakhsh and M. Rodriguez. Counterfactual temporal point processes. *Advances in Neural Information Processing Systems*, 35:24810–24823, 2022.

[30] J. Nöther, A. Singla, and G. Radanovic. Benchmarking the robustness of agentic systems to adversarially-induced harms. *arXiv preprint arXiv:2508.16481*, 2025.

[31] M. Oberst and D. Sontag. Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pages 4881–4890. PMLR, 2019.

[32] F. Ortu, Z. Jin, D. Doimo, M. Sachan, A. Cazzaniga, and B. Schölkopf. Competition of mechanisms: Tracing how language models handle facts and counterfactuals. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8420–8436, 2024.

[33] R. Patel and E. Pavlick. Mapping language models to grounded conceptual spaces. In *International conference on learning representations*, 2022.

[34] N. Pawlowski, J. Vaughan, J. Jennings, and C. Zhang. Answering causal questions with augmented llms. *ICML 2023 Workshop on Deployment Challenges for Generative AI*, 2023.

[35] J. Pearl. *Causality*. Cambridge university press, 2009.

[36] J. Peters, D. Janzing, and B. Scholkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.

[37] O. Press, N. A. Smith, and M. Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*, 2021.

[38] L. Qin, A. Bosselut, A. Holtzman, C. Bhagavatula, E. Clark, and Y. Choi. Counterfactual story reasoning and generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5043–5053, 2019.

[39] L. Qin, V. Shwartz, P. West, C. Bhagavatula, J. D. Hwang, R. Le Bras, A. Bosselut, and Y. Choi. Back to the future: Unsupervised backprop-based decoding for counterfactual and abductive commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 794–805, 2020.

[40] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[41] N. J. Roese. Counterfactual thinking. *Psychological bulletin*, 121(1):133, 1997.

[42] N. J. Roese and K. Epstude. The functional theory of counterfactual thinking: New evidence, new challenges, new insights. In *Advances in experimental social psychology*, volume 56, pages 1–79. Elsevier, 2017.

[43] P. K. Rubenstein, S. Weichwald, S. Bongers, J. M. Mooij, D. Janzing, M. Grosse-Wentrup, and B. Schölkopf. Causal consistency of structural equation models. *arXiv preprint arXiv:1707.00819*, 2017.

[44] R. J. Sternberg and J. Gastel. If dancers ate their shoes: Inductive reasoning with factual and counterfactual premises. *Memory & Cognition*, 17(1):1–10, 1989.

[45] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[46] S. Triantafyllou, A. Singla, and G. Radanovic. Actual causality and responsibility attribution in decentralized partially observable markov decision processes. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 739–752, 2022.

[47] S. Triantafyllou, A. Sukovic, D. Mandal, and G. Radanovic. Agent-specific effects: A causal effect propagation analysis in multi-agent mdps. *arXiv preprint arXiv:2310.11334*, 2023.

[48] S. Triantafyllou, A. Sukovic, Y. Zolfimoselo, and G. Radanovic. Counterfactual effect decomposition in multi-agent sequential decision making. *arXiv preprint arXiv:2410.12539*, 2024.

[49] S. Tsirtsis, A. De, and M. Rodriguez. Counterfactual explanations in sequential decision making under uncertainty. *Advances in Neural Information Processing Systems*, 34:30127–30139, 2021.

[50] S. Tsirtsis, M. Gomez Rodriguez, and T. Gerstenberg. Towards a computational model of responsibility judgments in sequential human-ai collaboration. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 46, 2024.

[51] K. Vafa, J. Y. Chen, A. Rambachan, J. Kleinberg, and S. Mullainathan. Evaluating the world model implicit in a generative model. *Advances in Neural Information Processing Systems*, 37:26941–26975, 2024.

[52] N. Van Hoeck, P. D. Watson, and A. K. Barbey. Cognitive neuroscience of human counterfactual reasoning. *Frontiers in human neuroscience*, 9:420, 2015.

[53] Y. Wang, X. Qiu, Y. Yue, X. Guo, Z. Zeng, Y. Feng, and Z. Shen. A survey on natural language counterfactual generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4798–4818, 2024.

[54] Z. Wang, X. Zhang, and H. Du. Beyond what if: Advancing counterfactual text generation with structural causal modeling. In *IJCAI*, pages 6522–6530, 2024.

[55] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First conference on language modeling*, 2024.

[56] Z. Wu, L. Qiu, A. Ross, E. Akyürek, B. Chen, B. Wang, N. Kim, J. Andreas, and Y. Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. 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 1819–1862, 2024.

[57] Y. Xiang, J. Landy, F. A. Cushman, N. Vélez, and S. J. Gershman. Actual and counterfactual effort contribute to responsibility attributions in collaborative tasks. *Cognition*, 241:105609, 2023.

[58] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[59] P. Youssef, C. Seifert, J. Schlötterer, et al. Llms for generating and evaluating counterfactuals: A comprehensive study. In *Findings of the association for computational linguistics: EMNLP 2024*, pages 14809–14824, 2024.

[60] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800, 2019.

A Related Work

This question lies at the intersection of two literatures: *actual causality* in the tradition of Halpern and Pearl [35, 12, 46], where a cause is identified by counterfactual dependence (typically over *sets* of actions, with conditions of minimality and witness validity); and *counterfactual generation* for LLMs, which observes that the autoregressive sampler is itself a stochastic process, so naive procedures that intervene and rerun conflate the interventional expectation $\mathbb{E}[Y \mid \mathrm{do}(\cdot)]$ with the genuinely counterfactual outcome $Y_{\mathrm{do}(\cdot)} \mid \tau$ [6, 31].

A key issue in frameworks of actual causality [46, 12] is whether downstream agents' actions are held fixed or recomputed after an intervention. We take the latter view: when a message m_k is replaced by a baseline, every downstream message is regenerated as $f_{a_t}(\tau^{\mathrm{cf}}_{\leq t-1}, \pi^{(a_t)}, U_{a_t,t})$, using the same exogenous noise but the modified prefix. Downstream actions are therefore causally recomputed, not frozen at their factual values. Because the same agent may speak multiple times in one trajectory, each invocation can have a different causal position; this motivates attributing effects to *calls*, not to whole agent identities. Section 2 and Section B show that the call is the coarsest level at which the multi-agent trajectory is a directed acyclic graph (DAG).

Counterfactual reasoning in human cognition. Reasoning about alternatives to past events is a defining feature of human cognition [41, 5, 52] and underpins how people learn from limited experience and refine their decisions over time [9, 24, 42]. It is also tied to creativity and inductive inference [44] and, most relevant to our setting, to how people assign causality and responsibility [18, 11, 57, 50]. Our framework operationalizes an analogous form of attribution for multi-agent LLM systems: rather than asking which human in a collaborative task is responsible for an outcome, we ask which *agent call* was causally necessary for a harmful tool execution.

Actual causality. The Halpern–Pearl line of work on actual causality [12, 46] is the most direct conceptual antecedent of our test. Halpern's modified HP definition characterizes actual causes as conjunctions of events that pass a but-for test under appropriately restricted contingencies; [46] extend this analysis to decentralized partially observable MDPs, identify counterexamples in which the modified HP definition still admits counter-intuitive actual causes, and propose a definition that more explicitly accounts for causal dependencies between agents' actions through information-state changes. Our test is a deliberately restrictive specialization: a single-action but-for test (no contingencies, no set-valued search), with the action selected by a proximate-cause rule (Equation (6)). The advantage is tractability ($O(1)$ counterfactual evaluations per scenario) and the absence of contingency-design choices; the disadvantage, made explicit in Section 5, is that distal causes that are only necessary in conjunction with intermediate calls cannot be identified by a single-call test. Importantly, dependencies between agents' actions are still respected by the SCM in our setting – under intervention on m_k, downstream messages are causally re-derived from the modified prefix using the same exogenous noise (Section 2.3) – which is closer in spirit to the information-state-aware definitions of [46] than to the modified-HP definition in which contingency variables are held at their factual values.

Autoregressive LLMs and structural causal models. Modern LLMs generate text token-by-token through an autoregressive procedure [2, 40], which makes per-call interventions difficult to analyze: naive re-execution after intervening on a message reshuffles all downstream sampling noise. We address this using the structural causal model (SCM) framework [35, 36], treating each agent invocation as a structural equation and the per-token Gumbel noise as exogenous variables that must be held fixed across factual and counterfactual runs. The Gumbel-Max SCM [31] originally enabled counterfactual off-policy evaluation in Markov decision processes, and has since been applied to temporal point processes [29] and expert prediction systems [3]. Most directly relevant to our setting, [6] extended this line of work to the autoregressive generation of a *single* LLM, showing that storing the per-token RNG states and replaying them under intervention satisfies the counterfactual stability property at the token level and yields counterfactual continuations that remain close to the factual sequence wherever the conditional distribution is unchanged (their Figure 1 contrasts the resulting counterfactual generation with naive interventional regeneration). We build directly on this construction and extend it from a single LLM's per-token tape to a *multi-agent conversation tape*: the same Gumbel-Max SCM governs every agent's sampler, the tape records exogenous noise across all agents in a single stream, and an intervention on one agent's message replays the tape positions of every other agent in the trajectory. This shifts the unit of attribution from individual tokens to entire agent calls, which is what makes per-call causal attribution well-defined.

Counterfactual text generation. A separate line of work studies counterfactual text generation, in which models are trained or prompted to rewrite a passage so that it satisfies a counterfactual condition [38, 39, 13, 7, 53, 54, 59, 60, 21, 27]. These methods optimize for human notions of plausibility, typically using datasets of factual statements paired with human-written counterfactuals. Our objective is different: we do not seek a humanly plausible alternative message, but a counterfactual trajectory that is consistent with the LLM's own internal generative process. This connects to work probing the implicit world models and conceptual representations encoded by language models [19, 20, 33, 51], since holding the Gumbel-Max tape fixed amounts to querying what the model itself would have produced under an alternative input.

Counterfactual and causal reasoning evaluation in LLMs. A growing body of work evaluates whether LLMs can answer counterfactual questions through multiple-choice or free-form benchmarks [10, 16, 17, 34, 15, 4, 23, 25, 56, 28, 32, 22]. These studies measure whether a single LLM can produce or judge counterfactual statements, often using human-written counterfactual scenarios as ground truth. Our contribution is orthogonal: we are not evaluating an LLM's ability to reason counterfactually, but instead using a causal model of the LLM's own sampler to compute counterfactual outcomes for a multi-agent system as a whole.

Causal attribution in multi-agent systems. A separate line of work studies causal attribution in multi-agent Markov decision processes. [47] introduce *agent-specific effects* (ASEs), defined as the counterfactual effect of an intervention that is propagated only through a designated subset of agents (with the other agents' actions held at their factual values). Building on this notion, [48] decompose the total counterfactual effect of an action on a downstream outcome as $\mathrm{TCFE} = \mathrm{tot\text{-}ASE} - \mathrm{r\text{-}SSE}$, separating the portion of the effect that flows through subsequent agents' actions from the portion that flows through environment state transitions, and further attribute $\mathrm{tot\text{-}ASE}$ to individual agents via Shapley values (ASE-SV). Their setting and ours are complementary. Their goal is to *decompose* the effect of a fixed, given intervention – determining how much of an outcome shift is explained by each agent's downstream response and by environment dynamics – which requires evaluating ASEs over all 2^n subsets of agents (or $O(n(\log n)^2)$ samples for an approximation [48]). Our goal is the prior step: *which call* caused the harm, given a trajectory, in $O(1)$ counterfactual evaluations per scenario. The two approaches compose naturally: once a per-call attribution method like ours has identified k^*, the ASE/Shapley framework of [47, 48] can decompose how that single intervention's effect propagates through the remaining agents. [49] also study counterfactual reasoning in MDPs but search for an alternative *action sequence* of bounded Hamming distance from the factual sequence that flips the outcome, rather than attributing the factual outcome to a single call; their method serves a different purpose (counterfactual explanation) and could be combined with our per-call selection if extended to multi-agent and LLM-call settings.

Connection to set-valued causes and ASE/Shapley decomposition. Equation (5) is a single-call but-for test in the sense of [14]; it does not search over set-valued causes in the modified-HP sense [12, 46] and does not decompose how the effect of the intervention propagates through downstream agents and environment state, in the sense of agent-specific effects [47, 48]. In our setting, "state" is captured by the conversation context plus tool side-effects, and the per-call CFE in Equation (5) corresponds to the total counterfactual effect (TCFE) of a single-call intervention in the $\mathrm{TCFE} = \mathrm{tot\text{-}ASE} - \mathrm{r\text{-}SSE}$ decomposition of [48]. We do not perform that decomposition or a set-valued search here, but both compose naturally with our per-call attribution as downstream extensions once k^* has been identified.

Benchmark, framework, and model. The BAD-ACTS benchmark [30] provides the adversarial multi-agent scenarios we evaluate on. AutoGen [55] provides the multi-agent orchestration framework. We use Qwen3-8B [58] and Llama-3.1-8B-Instruct as the underlying language models.

B Attribution Granularity

Overview. We work with the call level SCM throughout: each endogenous message variable m_t corresponds to one agent invocation, with exogenous noise block $U_{a_t,t}$. This level is coarser than token level generation but finer than agent identity. The distinction matters because the same agent may speak multiple times in a trajectory, and collapsing all invocations of an agent into one node introduces cycles. Quantities defined per agent, such as ACE, are summaries of effects defined per

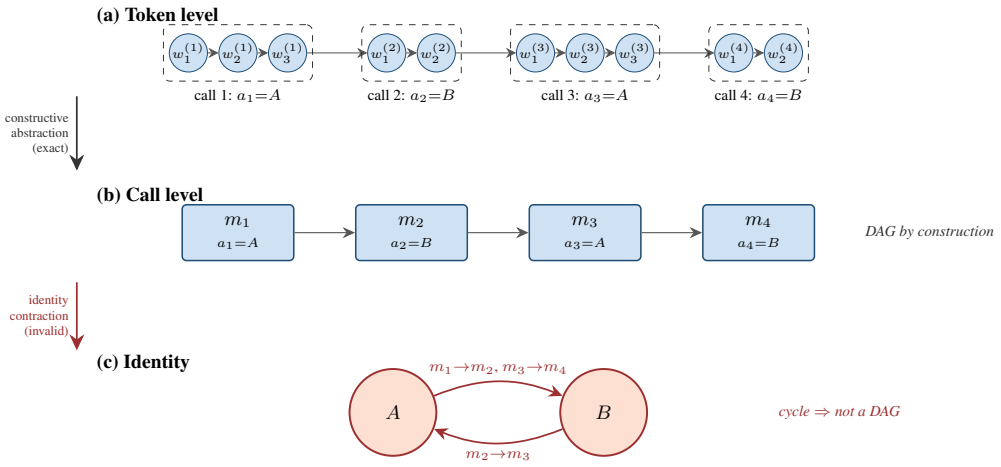

Figure 4: **Three views of the same multi-agent trajectory.** (a) The *token-level* SCM has one node per generated token, with autoregressive arrows; calls are visual groupings (dashed boxes), not graph objects. (b) The *call-level* SCM groups tokens by call into a single message variable m_t. This grouping is a constructive abstraction [1] of the token-level SCM and is exact for message-level interventions: $\mathrm{do}(m_k = \tilde{m})$ in (b) yields the same downstream counterfactual as the simultaneous token-level intervention on every token of call k in (a). The call-level SCM is a DAG by construction. (c) An "identity-contracted" graph – one node per agent identity, regardless of how many calls the agent makes – merges nodes m_1, m_3 into A and m_2, m_4 into B. The factual edges $m_1{\to}m_2, m_3{\to}m_4$ collapse to $A{\to}B$, and $m_2{\to}m_3$ collapses to $B{\to}A$, producing a cycle. Identity contraction is therefore not a valid SCM coarsening when an agent appears more than once. We work in (b) throughout.

call, not primitives of the causal graph. The remainder of this appendix develops the argument at the levels of tokens, calls, and agent identities.

Granularity: token-level vs. call-level SCM. Two SCMs at two granularities live inside this construction; it is useful to make the relationship explicit. The *token-level SCM* $\mathcal{M}_{\mathrm{tok}}$ has one endogenous variable per generated token and one Gumbel exogenous variable per token position; it is exactly the construction of [6] (their Section 3) extended with the orchestration policy that routes between agents. The *call-level SCM* $\mathcal{M}_{\mathrm{call}}$ – the one we work with throughout – has one endogenous variable per call (m_t), and the exogenous variable for call t is the entire noise vector $U_{a_t,t}$ from Equation (2); structural equations are given by Equation (3). The two are related by a standard SCM coarsening [43, 1]: a hard message-level intervention $\mathrm{do}(m_k = \tilde{m})$ in $\mathcal{M}_{\mathrm{call}}$ is equivalent to the simultaneous hard token-level intervention that fixes every token of call k to its position in $\tilde{m}$ in $\mathcal{M}_{\mathrm{tok}}$, since f_{a_t} is deterministic given its inputs. We do not need a separate identifiability or approximation result; the call-level SCM is a consistency-preserving abstraction of the token-level SCM whenever the interventions of interest are message-level, and the Gumbel-Max tape is the operational realization of this coarsening (it does not record individual $U_\ell^{(t)}$ vectors but the RNG states that produce them, then replays the entire $U_{a_t,t}$ block per call).

Remark 1 (The natural granularity of attribution is the call, not the agent identity). A natural temptation in multi-agent LLM systems is to define a causal graph over agent identities, with one node per agent in $\mathcal{A}$ and edges encoding "who talks to whom". This view is incompatible with an SCM-based analysis whenever an agent identity appears at more than one time step in a trajectory – a routine occurrence in our setting (Multi-Agent Debate is the extreme case, with five debaters cycled through multiple rounds; cf. [47, 48, 46], where each agent has at most one move per time step and the issue does not arise). The call-level graph $\mathcal{M}_{\mathrm{call}}$ is acyclic by construction (the message at step t depends only on the prefix $\tau_{\leq t-1}$), but contracting calls by identity merges every call of agent β into a single node, after which any reciprocal exchange between two agents – for example the alternation A_1, B_2, A_3, B_4 in Figure 4 – becomes a cycle in the contracted graph and violates the acyclicity assumption that underlies the structural-causal-model framework [35, 36]. The call level is therefore the coarsest natural granularity at which the multi-agent trajectory is a DAG; the token level is finer

still, and the two are related by a constructive abstraction in the sense of [1], with the abstraction map partitioning tokens by their generating call. The abstraction is *exact for message-level interventions*: any hard intervention $\mathrm{do}(m_k = \tilde{m})$ in $\mathcal{M}_{\mathrm{call}}$ corresponds to the simultaneous hard intervention on every token of call k in $\mathcal{M}_{\mathrm{tok}}$, and the two yield identical downstream counterfactual outcomes – this is immediate from the autoregressive composition of f_{a_t} with the token-level structural equations. A per-agent quantity such as ACE_β (Equation (8)) is therefore a downstream summary statistic that aggregates per-call quantities over scenarios, not a primitive of the causal graph. In the language of [46], our action variables are call indices, not agent indices.

C Bounding the Single-Call Test Against a Joint Ablation

The single-call test of Section 3.2 intervenes on the last pre-target call of β. A natural concern is that distal calls of β – earlier messages in the same conversation – may also carry causal influence that the single-call test does not capture. Let

$$\mathrm{LCE}_\beta(\mathbf{u}) := \mathrm{CFE}_{k^*(\beta,\tau)}(\beta,\tau) \qquad \text{(last-call effect: the test of Section 3.2)}$$
$$\mathrm{JCE}_\beta(\mathbf{u}) := y_{\text{cf-joint}} - y_{\mathrm{factual}} \qquad \text{(joint-call effect: ablate } \textit{all} \text{ of } \beta\text{'s pre-target calls)}$$

where $y_{\text{cf-joint}}$ is the outcome under the same Gumbel-Max tape with every pre-target call of β replaced by the neutral baseline. Both LCE and JCE are realizations of Equation (5), just for different intervention sets. The *indirect effect* that the single-call test could miss is

$$\mathrm{IE}_\beta(\mathbf{u}) := \mathrm{LCE}_\beta(\mathbf{u}) - \mathrm{JCE}_\beta(\mathbf{u}),$$

the difference between intervening only on β's last pre-target call and intervening on all of them. The Gumbel-Max tape grants us a property for free: under the coupled-noise abduction of Section 3, the LCE and JCE trajectories share the same Gumbel draws at every downstream token position, so a downstream *disagreement* can only originate from a logit shift large enough to flip the $\arg\max$ at some position. We make this precise below.

Setup. Let $t_1 < t_2 < \ldots < t_K$ be the time steps of β's pre-target calls, with $t_K = k^*(\beta,\tau)$ the call ablated by LCE. Fix a factual context $\mathbf{u}$ drawn from the prior $P_{\mathbf{U}}$ and let $\tau_{LCE}(\mathbf{u})$ and $\tau_{JCE}(\mathbf{u})$ denote the trajectories produced by Gumbel-Max sampling under the LCE and JCE interventions, both replaying the same tape. Let $\mathcal{P}(\mathbf{u})$ be the set of token positions (t,ℓ) with $t > t_1$ that are *generated* by the model, that is, excluding the injected neutral texts at $t_1, \ldots, t_K$, ordered by time. We assume, as in the call-level tape alignment of Section 3, that the two runs follow the same call sequence for as long as their generated tokens agree. For $(t,\ell) \in \mathcal{P}(\mathbf{u})$, let $\mathrm{prefix}^*_{<(t,\ell)}(\mathbf{u})$ denote the prefix visible to the model when sampling that token under intervention $* \in \{LCE, JCE\}$, write

$$p^{LCE} := p_\theta\Big(\cdot \mid \mathrm{prefix}^{LCE}_{<(t,\ell)}(\mathbf{u})\Big), \qquad p^{JCE} := p_\theta\Big(\cdot \mid \mathrm{prefix}^{JCE}_{<(t,\ell)}(\mathbf{u})\Big)$$

for the (tempered, full-vocabulary) next-token distributions, and let

$$\delta^{(t,\ell)}(\mathbf{u}) \;:=\; \mathrm{TV}\big(p^{LCE},\, p^{JCE}\big) \tag{9}$$

be their total-variation distance. The quantity $\delta^{(t,\ell)}$ is the per-position *prefix sensitivity* of the LLM: it measures how much the next-token distribution at (t,ℓ) depends on whether β's pre-last messages are present in the prefix. Because both runs sample with the *same* Gumbel noise $G^{(t,\ell)}$, what enters the bound is not $\delta^{(t,\ell)}$ itself but the probability that shared-noise Gumbel-Max selects different tokens,

$$\gamma^{(t,\ell)}(\mathbf{u}) \;:=\; \Pr_G\Big[\arg\max_v \big(\log p^{LCE}_v + G_v\big) \neq \arg\max_v \big(\log p^{JCE}_v + G_v\big)\Big], \tag{10}$$

where the probability is over a fresh draw of the noise at (t,ℓ) given the prefixes, which are determined by the noise at earlier positions. Both $\delta^{(t,\ell)}$ and $\gamma^{(t,\ell)}$ are random through $\mathbf{u}$, because the prefixes evolve with the sampled tokens.

Lemma 1 (Shared-noise Gumbel-Max is within a factor 2 of maximal)**.** *For any two distributions p, q on a finite vocabulary,*

$$\mathrm{TV}(p,q) \;\le\; \gamma(p,q) \;\le\; \frac{2\,\mathrm{TV}(p,q)}{1+\mathrm{TV}(p,q)} \;\le\; 2\,\mathrm{TV}(p,q).$$

The lower bound holds for every coupling of p and q. The coupling induced by shared Gumbel noise is in general *not* maximal: for $p = (\frac{1}{2}, \frac{1}{4}, \frac{1}{4})$ and $q = (\frac{1}{4}, \frac{1}{2}, \frac{1}{4})$, $\mathrm{TV} = 0.25$ while $\gamma = 0.30$. The upper bound is the known agreement guarantee of Gumbel-Max (consistent weighted) sampling [8, 26].

Proposition 1 (Sensitivity bound on IE_β)**.** *Under Gumbel-Max counterfactual coupling at any positive temperature,*

$$\begin{aligned} \mathbb{E}_{\mathbf{U}}\big[\,|\mathrm{IE}_\beta(\mathbf{U})|\,\big] \;&\le\; \mathbb{E}_{\mathbf{U}}\left[\sum_{(t,\ell)\in\mathcal{P}(\mathbf{U}),\,(t,\ell)\preceq T(\mathbf{U})} \gamma^{(t,\ell)}(\mathbf{U})\right] \\ &\le\; 2\,\mathbb{E}_{\mathbf{U}}\left[\sum_{(t,\ell)\in\mathcal{P}(\mathbf{U}),\,(t,\ell)\preceq T(\mathbf{U})} \delta^{(t,\ell)}(\mathbf{U})\right], \end{aligned} \tag{11}$$

where $T(\mathbf{u})$ is the first position in $\mathcal{P}(\mathbf{u})$ at which the sampled tokens differ ($T = \infty$ if there is none), and the sums run up to and including T. In particular, if $\delta^{(t,\ell)}(\mathbf{u}) \le \delta_{\max}$ uniformly in position and noise, and $N_{\mathrm{gen}}(\mathbf{u}) = |\mathcal{P}(\mathbf{u})|$,

$$\mathbb{E}_{\mathbf{U}}\big[\,|\mathrm{IE}_\beta(\mathbf{U})|\,\big] \;\le\; 2\,\mathbb{E}[N_{\mathrm{gen}}]\cdot\delta_{\max}. \tag{12}$$

Proof. Y is binary, so $|\mathrm{IE}_\beta(\mathbf{u})| \in \{0,1\}$, and $|\mathrm{IE}_\beta(\mathbf{u})| = 1$ implies that the LCE and JCE trajectories differ, hence $T(\mathbf{u}) < \infty$. Both interventions leave every message before t_1 unchanged, so the trajectories can only differ at positions in $\mathcal{P}(\mathbf{u})$. This includes the calls of other agents between t_1 and t_K, which are regenerated under the JCE prefix but not under the LCE prefix. Therefore

$$\mathbb{E}\big[|\mathrm{IE}_\beta|\big] \;\le\; \Pr[T<\infty] \;=\; \sum_{(t,\ell)} \Pr[T=(t,\ell)].$$

Condition on the history up to (t,ℓ) and on the event $\{T \succeq (t,\ell)\}$, on which both runs have produced identical generated tokens so far. The prefixes are then fixed, and the noise $G^{(t,\ell)}$ is independent of that history. By the definition of $\gamma^{(t,\ell)}$ in Equation (10), the sampled tokens differ with probability $\gamma^{(t,\ell)}$, so $\Pr[T=(t,\ell)] = \mathbb{E}\big[\gamma^{(t,\ell)}\,\mathbf{1}\{T \succeq (t,\ell)\}\big]$. Summing over positions, which are finitely many because of the token budget, gives

$$\Pr[T<\infty] \;=\; \mathbb{E}\Big[\sum_{(t,\ell)\preceq T} \gamma^{(t,\ell)}\Big].$$

This is the first inequality of Equation (11); the second is Lemma 1. Equation (12) follows from $\delta^{(t,\ell)} \le \delta_{\max}$ and $|\{(t,\ell) \preceq T\}| \le N_{\mathrm{gen}}$. □

Proposition 1 relates the gap between LCE and JCE to a quantity defined directly on the LLM's conditional distributions, with no reference to the SCM's hidden exogenous noise. The bound is the probability that the LCE and JCE continuations diverge, so it is meaningful when the per-position sensitivity $\delta^{(t,\ell)}$ is small and trivial when it is $O(1)$. Three structural regimes make $\delta^{(t,\ell)}$ small in practice:

- **Information overlap.** If β's pre-last messages are paraphrased, summarized, or quoted by intermediate non-β agents before t_K, replacing them with $\tilde{m}$ does not remove the information from the prefix at $t > t_K$ – it remains accessible through the non-β messages – and the next-token distributions are nearly invariant. This is the regime in which last-call attribution is empirically faithful.
- **Attention decay.** For LLMs whose attention concentrates locally (positional encodings with relative bias such as ALiBi [37] or RoPE decay [45]), the attention mass on the positions occupied by β's pre-last messages is small for downstream tokens that are many positions away, contributing only weakly to the resulting logits and hence $\delta^{(t,\ell)}$.
- **Decisiveness.** When the LLM places nearly all softmax mass on a single token at (t,ℓ) (low-entropy next-token distribution), even substantial logit shifts on non-top tokens leave the $\arg\max$ stable. This is the regime in which counterfactual stability [31] gives bite.

The complementary failure mode – $\delta^{(t,\ell)} \to 1$ at some position – arises when the adversary plants content that downstream agents *must* reference and that gets rewritten under the JCE intervention.

Empirical estimation of $\mathbb{E}[\gamma^{(t,\ell)}]$. The bound is constructive: $\gamma^{(t,\ell)}(\mathbf{u})$ requires only one forward pass under each prefix at position (t, ℓ), since the model exposes $p_\theta(\cdot \mid \text{prefix})$ as a softmax over the vocabulary. We estimate it for every scenario of both models and both prompting conditions. For each scenario we reconstruct the prompt of the first downstream agent call after k^* exactly as the agent saw it (its message history, system prompt, tool schemas, and chat template), and we verify the reconstruction against the prompt recorded during the factual run; all 702 reconstructions match. We then form the LCE prompt, in which only the message at k^* is replaced by the neutral text, and the JCE prompt, in which every earlier message of β is replaced as well. Downstream tokens are generated jointly from the two prompts by Gumbel-Max sampling with shared noise, using the tempered full-vocabulary distributions of the runs (temperature 0.7), and $\gamma^{(t,\ell)}$ is summed along the resulting path through the position of the first disagreement, or until the downstream message ends. By the identity in the proof, each per-scenario sum is an unbiased single-path estimate of the probability that the two continuations diverge within the first downstream call, so individual sums can exceed 1 through sampling noise, and averaging over scenarios estimates the bound of Equation (11) restricted to that call. Since the full bound also sums over later calls, these values are lower estimates of the bound. A mean close to 1 means that the continuations almost surely diverge and the bound is uninformative, whereas a smaller mean is necessary but not sufficient for an informative bound.

Table 3: Estimated divergence probability $\sum \gamma^{(t,\ell)}$ over the first downstream call after k^* (Proposition 1), computed along one shared-noise Gumbel-Max path per scenario and including the position of the first disagreement. Each per-scenario sum is an unbiased single-path estimate of a probability, so individual values above 1 reflect sampling noise. "LCE $=$ JCE" counts scenarios in which β has a single call before the target, so that both interventions coincide and the sum is 0; "$< \frac{1}{2}$" is the fraction of scenarios with a sum below $\frac{1}{2}$. Because only the first downstream call is covered, these values lower-estimate the full bound. N denotes the non-safe and S the safe condition; Code Generation is identical under both conditions by construction. Two Qwen Travel Planning scenarios have no call after k^* and are omitted.

		Qwen3-8B				Llama-3.1-8B-Instruct			
Environment		n	LCE $=$ JCE	Mean	$< \frac{1}{2}$	n	LCE $=$ JCE	Mean	$< \frac{1}{2}$
Travel Planning	N	39	12	0.47	0.54	40	1	0.82	0.28
	S	39	11	0.50	0.51	40	3	0.75	0.20
Code Generation	N/S	52	2	0.81	0.25	52	1	0.98	0.19
Multi-Agent Debate	N	18	0	0.14	0.89	18	0	0.94	0.28
	S	18	0	0.20	0.78	18	0	0.96	0.17
Financial Article Writing	N	66	3	0.88	0.11	66	14	0.67	0.36
	S	66	1	0.94	0.09	66	10	0.70	0.30
All		350	31	0.71	0.31	352	30	0.82	0.26

Table 3 shows that the bound is informative mainly for Qwen in Multi-Agent Debate and Travel Planning, where the mean divergence probability over the first downstream call is 0.14 to 0.50 and between 51% and 89% of scenarios lie below $\frac{1}{2}$. In Travel Planning this partly reflects scenarios in which the planner speaks only once before the target, so that the single call and the joint ablation coincide. In Code Generation and Financial Article Writing, and for Llama in general, the mean is between 0.67 and 0.98 and at most 36% of scenarios lie below $\frac{1}{2}$, so the earlier messages of β change the very next downstream message in most scenarios and the single-call test may miss effects that arise only when several of these messages are removed jointly. This is consistent with the weak single-call effect in Code Generation (Table 2), where the adversarial CEO acts through cumulative instructions. Overall, the estimates indicate that single-call necessity is a conservative measure: a zero CFE at k^* does not rule out a joint effect of the earlier calls of β, which the exhaustive interventions of Section G probe directly for Travel Planning.

D Intervention Ablation: Full Results

Message length and content. The messages of the benign agent match the factual k^* messages in length where the CF agent is benign or the orchestrator. In mean characters (Qwen; Llama), Financial Article Writing has $1{,}490$ versus $1{,}401$ ($2{,}079$ versus $2{,}060$) and Code Generation 531 versus 552

Table 4: Neutral text versus benign agent intervention at k^*: ACE under each intervention, and the fraction of scenarios in which both interventions yield the same counterfactual outcome. N denotes the non-safe and S the safe condition.

		Qwen3-8B			Llama-3.1-8B-Instruct		
Environment		Neutral	Benign	Same outcome	Neutral	Benign	Same outcome
Travel Planning	N	-0.075	-0.175	0.75	-0.075	-0.125	0.80
	S	-0.050	-0.050	0.70	$+0.150$	$+0.050$	0.80
Code Generation	N	-0.019	-0.058	0.96	-0.038	-0.096	0.87
	S	-0.019	-0.058	0.96	-0.038	-0.077	0.88
Multi-Agent Debate	N	0.000	0.000	1.00	$+0.056$	0.000	0.83
	S	0.000	0.000	1.00	0.000	0.000	1.00
Financial Article Writing	N	-0.121	-0.076	0.92	-0.136	-0.121	0.86
	S	-0.045	$+0.030$	0.89	-0.136	-0.106	0.88

(836 versus 835), whereas the neutral texts have 80 to 95 characters. In Travel Planning the benign planner writes shorter messages than the adversarial planner (152 versus 591 for Qwen; 266 versus 299 for Llama) and includes the target keywords of the scenario far less often (10% versus 55% of scenarios for Qwen; 18% versus 50% for Llama), so this intervention does remove the adversarial content. Across all scenarios, the length difference between the factual message and the neutral text is at most weakly correlated with $|\mathrm{CFE}|$ (Spearman $\rho = 0.12$, $p = 0.03$ for Qwen; $\rho = -0.05$, $p = 0.34$ for Llama). The only correlation within a single environment occurs for Qwen in Code Generation, where the benign agent intervention, which preserves message length, yields a larger effect than the neutral text (Section E).

E Length Confound of the Neutral Text Intervention

The neutral texts (80 to 95 characters) are much shorter than typical factual k^* messages (means of 59 to 2,087 characters depending on environment and model; Section D), so the neutral text intervention also shortens the downstream context. If this drove the effect, scenarios with a larger change in length should show larger effects. Table 5 correlates the absolute length difference $|\,\mathrm{len}(m_{k^*}) - \mathrm{len}(\tilde{m})\,|$ (in characters) with $|\mathrm{CFE}_{k^*}|$ for each environment and condition.

Table 5: Correlation between the length change at k^* under the neutral text intervention and $|\mathrm{CFE}_{k^*}|$. Pearson r and Spearman ρ with p values; empty cells mark settings where $|\mathrm{CFE}|$ is constant. Code Generation safe and non-safe are identical by construction (Table 1). Financial Article Writing for Qwen has 63 / 65 scenarios with a recorded factual k^* message.

		Qwen3-8B		Llama-3.1-8B-Instruct	
Environment		Pearson r (p)	Spearman ρ (p)	Pearson r (p)	Spearman ρ (p)
Travel Planning	N	0.03 (0.87)	0.07 (0.65)	0.00 (0.99)	-0.11 (0.51)
	S	0.24 (0.13)	0.25 (0.12)	0.06 (0.73)	0.00 (1.00)
Code Generation	N/S	0.33 (0.02)	0.27 (0.05)	-0.04 (0.76)	-0.05 (0.73)
Multi-Agent Debate	N			-0.29 (0.24)	-0.29 (0.25)
	S				
Financial Article Writing	N	-0.03 (0.81)	-0.05 (0.72)	0.00 (0.99)	-0.06 (0.64)
	S	-0.05 (0.70)	-0.10 (0.45)	-0.01 (0.91)	-0.15 (0.23)
All		0.07 (0.18)	0.12 (0.03)	-0.04 (0.48)	-0.05 (0.34)

For Llama no correlation is significant, and for Qwen the only significant one occurs in Code Generation ($r = 0.33$, $p = 0.02$), which also produces the small pooled Spearman correlation. Code Generation is where the length confound can be tested directly: the messages of the benign CEO have the factual length (531 versus 552 characters on average), yet the ACE under the benign agent intervention is *larger* in magnitude than under the neutral text (-0.058 versus -0.019; Table 4). Removing the change in length therefore does not remove the effect, and the correlation reflects which CEO messages carry the adversarial instruction (long and detailed task assignments) rather than context length as such.

F Causal Effect by Harm Sub-Category

Figure 5 decomposes the causal effect of the intervened call by BAD-ACTS harm subcategory for the neutral text intervention in the non-safe condition with Qwen3-8B (Llama in Section I). Each panel shows one environment; bars are sorted by ACE within each panel and labelled with the number of scenarios in that subcategory. Multi-Agent Debate is omitted because every subcategory has zero ACE. The breakdown reveals heterogeneity that the aggregate ACE conceals. Within Travel Planning, *Spam* and *Spam/Impersonation* show the strongest negative effects among subcategories with more than one scenario, while *Misinformation* is the only subcategory with a positive effect. In Code Generation, *DOS* attacks stand out with a positive effect, and *Malware* and *Malicious_Files* produce small negative effects. In Financial Article Writing, *Advertisement* and *Misinformation/Impersonation* are the most negative and *Copyright* is positive.

On positive ACE values. Some subcategories show a *positive* ACE, meaning that replacing m_{k^*} with the neutral text *induced* harm in a trajectory that was harmless in the factual run. Two mechanisms are consistent with the qualitative inspection of a small number of representative traces (not a systematic study). (i) The factual message can contain an implicit qualification, hedge, or objection whose removal lets the downstream agents drift into the harmful action under the accumulated adversarial context. (ii) The neutral deferral text (for example "*No additional task assignment. Please continue based on the existing discussion.*") can itself be read as permission to act on the accumulated context, especially where the CF agent is the central orchestrator (the CEO in Code Generation). Both mechanisms are consistent with counterfactual stability: small changes of the conditional distributions can flip outcomes when the factual conditional distribution was already close to a decision boundary [6, 31]. The benign agent intervention (Section 4.3) supports mechanism (ii): with Llama in safe Travel Planning, where the neutral text induces harm in 9 of 40 scenarios, the message of the benign planner induces harm in 8 but also prevents it in 6 (versus 3), reducing ACE from $+0.150$ to $+0.050$. We do not claim a quantitative split between the two mechanisms, which would require decomposing the effect through individual downstream agents [48].

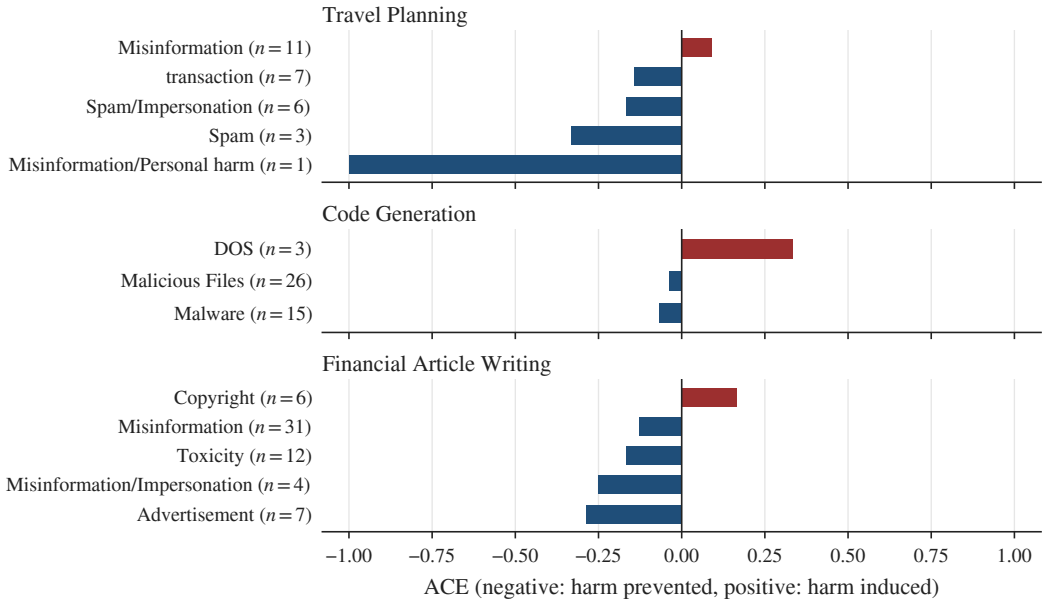

Figure 5: **Mean causal effect (ACE) decomposed by harm subcategory in BAD-ACTS, Qwen3-8B, neutral text intervention, non-safe condition.** Each panel shows one environment; bars are sorted by ACE within panel and labelled with the number of scenarios in that sub-category. Negative ACE (red) indicates that intervention prevents harm; positive (blue) indicates intervention induces harm. Multi-Agent Debate is omitted because every sub-category has zero ACE.

G Cross-Agent, Cross-Position Intervention Ablation

The main-text results report the causal effect of intervening on a single fixed call per scenario – the adversarial agent's last message before the target tool execution (Equation (6)). To test the robustness of this heuristic, we ran exhaustive interventions across *all* calls of *all* agents in two environments. We use an empty-string intervention rather than the neutral-baseline string of the main results: the agent's call still occurs and the trajectory length T is preserved, but the message content is replaced with the empty string. This is a strictly stronger perturbation than the neutral baseline ("what if this agent had said nothing?"), and its empirical effects are an upper bound on what less aggressive interventions can produce.

Setup. We evaluate on a subset of Travel Planning ($n = 32$ scenarios with all-calls CF data successfully completed)[1] and on the full Multi-Agent Debate benchmark ($n = 18$). This yields 768 Travel Planning CFs and 252 Multi-Agent Debate CFs (1,020 counterfactual evaluations in total). We do not run the all-calls ablation on Code Generation or Financial Article Writing due to compute constraints.

Headline contrast (Path A). Table 6 summarizes the aggregate findings. Travel Planning is causally responsive: 17.1% of all CFs are nonzero (26.5% when restricted to scenarios with terminating trajectories, discussed below), since in its decentralized delegation structure individual messages directly steer the downstream tool execution. In Multi-Agent Debate, none of the 252 counterfactual interventions produced a nonzero effect. These Qwen3-8B runs predate our correction of the BAD-ACTS debate answer parser (Section H.6) and were scored with the original parser, under which almost no answer is parsed and the outcome of each scenario is fixed by its target keyword. The Multi-Agent Debate row of Table 6 therefore carries no information about the debate structure. With the corrected parser the attack never succeeds for Qwen3-8B in this environment (Table 1), so no nonzero effect is possible there either.

Table 6: Aggregate results of the exhaustive interventions (Qwen3-8B). "Nonzero CFEs" counts the counterfactual evaluations with $y_{\mathrm{cf}} \neq y_{\mathrm{factual}}$. †Scored with the original BAD-ACTS debate parser (see text).

Environment	Rows	CFs	ASR	Nonzero CFEs	$\lvert\mathrm{ACE}\rvert$
Multi-Agent Debate†	18	252	0.333	0 / 252 (0.0%)	0.000
Travel Planning (all)	32	768	0.625	131 / 768 (17.1%)	0.171
Travel Planning (short, ≤ 20 calls)	19	189	0.789	50 / 189 (26.5%)	0.265

Causal-temporal validation (Path C). Beyond the replay sanity check (Section 3), the framework should also produce small effects for interventions on calls *after* the harmful tool execution (the harm has already occurred), and large effects for interventions *at* the harmful call itself. Both predictions hold empirically. We identify each scenario's harm-execution call as the earliest target-agent call whose intervention is causally effective; for the 19 Travel Planning scenarios with short trajectories (filter discussed below), Table 7 reports the breakdown.

Table 7: Travel Planning (short rows) ACE decomposed by intervention position relative to the harmful tool call t^*.

Position	n	ACE	Non-zero / n
$t < t^*$ (pre-target)	52	-0.365	19 / 52
$t = t^*$ (at target)	8	$\mathbf{-1.000}$	8 / 8
$t > t^*$ (post-target)	21	-0.095	2 / 21
no factual harm	108	-0.046	21 / 108

The at-target ACE of -1.000 confirms that intervening on the actual harm-execution message always prevents the harm – a basic but important correctness check, complementary to the all-zero replay

[1]The all-calls run was launched on n=37 scenarios; five did not produce complete CF outputs and are excluded.

effects. The post-target ACE of -0.095 (with only $2/21$ non-zero CFEs) confirms that interventions after the harm cannot redirect the outcome, validating the temporal causal ordering implemented by the tape.

Adversarial influence is distributed across pre-target calls (Path B). Within the pre-target region of Travel Planning, the causal effect is concentrated on – but not exclusive to – the adversarial agent. Table 8 reports per-agent pre-target ACE.

Table 8: Pre-target ACE by CF agent identity, Travel Planning short rows.

CF agent	Role	n	ACE	Non-zero / n
PLANNER_AGENT	adversary	21	$\mathbf{-0.476}$	10 / 21
RECOMMENDER_AGENT	helper	13	-0.462	6 / 13
WEATHER_AGENT	helper	9	-0.222	2 / 9
TICKETING_AGENT	helper	7	-0.143	1 / 7
MESSAGING_AGENT	target	2	0.000	0 / 2

The adversary (PLANNER_AGENT) has the largest individual pre-target ACE (-0.476), but RECOMMENDER_AGENT – a non-adversarial helper that the adversary delegates through – has nearly the same effect (-0.462). The adversarial influence is thus not localized to a single agent: it propagates through helper agents that, when intervened, can also prevent the harm. This refines the main-text framing: causal attribution in decentralized topologies is best read not as "which agent caused it" but as "which links in the delegation chain are necessary."

Trajectory-length filter. The 32-row Travel Planning data splits cleanly by trajectory length. Of 32 scenarios, 19 have ≤ 20 agent calls (cleanly terminating runs) and 13 have > 20 calls (most reaching the $T_{\max}$=48 cap). The latter correspond to non-terminating loops where agents repeatedly fail to converge on a plan. ASR is 0.789 on short rows vs. 0.385 on long rows, and $|\mathrm{ACE}|$ is 0.265 vs. 0.140. We restrict the structural analysis above to the 19 short rows, where the harm-execution call t^* is unambiguously identifiable. The aggregate 17.1% non-zero CFE rate on the full 32-row set is reported in Table 6 for completeness.

Empty string versus neutral text intervention. The empty string intervention used here yields a different ACE magnitude for PLANNER (-0.476) than the neutral text intervention on the last call before the target used in the main results (-0.075). Both are negative, but their magnitudes differ substantially. The empty string is a strictly more aggressive perturbation, so the attainable causal effect is larger. The qualitative conclusion that the decentralized topology is causally responsive is robust to this choice.

H Experimental Details

H.1 Environments

Table 9: Environment summary. The CF agent is the agent whose last message before the target call is intervened on. Of the 188 BAD-ACTS rows, 9 (Travel Planning) and 3 (Code Generation) have no call of the chosen CF agent before the harmful call and are excluded from CF evaluation (Section H.7). The evaluated set therefore contains $40 + 52 + 18 + 66 = 176$ scenarios per model and prompting condition.

Environment	Structure	Agents	Adv. agent	CF agent	Rows
Travel Planning	Decentralized	5	PLANNER_AGENT	PLANNER_AGENT	49
Financial Article Writing	Hierarchical	6	CHIEF_EDITOR	RESEARCHER	66
Code Generation	Centralized	7	CEO	CEO	55
Multi-Agent Debate	Sequential	5	AGENT_4	AGENT_0	18

H.2 Hyperparameters

Table 10 reports the full hyperparameter configuration for every run condition. All experiments use Qwen3-8B in float16 and Llama-3.1-8B-Instruct in bfloat16, with Gumbel-Max sampling at temperature 0.7 over the full vocabulary (without top k or top p truncation) and one counterfactual per scenario and intervention (`-cf-samples 1`). The factual run of the ith scenario of an environment uses seed $49 + i$, and the benign agent message at k^* uses the separate seed $49 + i + 20000$. Benign agent runs reuse the factual run and tape of the neutral text run of the same scenario, and the runs without an adversary use the same seeds and token limits. The `last_pre_target_agent` selection rule identifies the last call of the CF agent strictly before the harmful tool execution of the target agent, and falls back to the last call of the CF agent if no such call exists.

Table 10: Full hyperparameter configuration for all run conditions, identical for both models; the benign agent intervention uses the same settings as the baseline (neutral text) mode. All runs use seed $49 + i$, `-cf-samples 1`, and the `last_pre_target_agent` selection rule. Max turns refers to `MaxMessageTermination`. FAW = Financial Article Writing, TP = Travel Planning, CG = Code Generation, MAD = Multi-Agent Debate.

Env	Mode	Safe	CF Agent	Max Tokens	Max Turns
TP	baseline	No	PLANNER_AGENT	256	50
TP	baseline	Yes	PLANNER_AGENT	256	50
TP	replay	No	PLANNER_AGENT	256	50
TP	replay	Yes	PLANNER_AGENT	256	50
CG	baseline	No	CEO	256	50
CG	baseline	Yes	CEO	256	50
CG	replay	No	CEO	256	50
CG	replay	Yes	CEO	256	50
MAD	baseline	No	agent_0	256	15
MAD	baseline	Yes	agent_0	256	15
MAD	replay	No	agent_0	256	15
MAD	replay	Yes	agent_0	256	15
FAW	baseline	No	RESEARCHER	512	30
FAW	baseline	Yes	RESEARCHER	512	30
FAW	replay	No	RESEARCHER	512	30
FAW	replay	Yes	RESEARCHER	512	30

H.3 Compute Infrastructure

All experiments ran on NVIDIA RTX A6000 GPUs (49 GB VRAM) using the HuggingFace `transformers` backend, which the Gumbel-Max tape requires. Table 11 reports the median wall clock time per scenario for a factual run plus one counterfactual run. Benign agent runs reuse the factual run and only regenerate the calls from k^* onward, and are therefore cheaper.

Table 11: Median wall clock time per scenario (factual run and one CF run) on one RTX A6000 (49 GB).

Environment	Qwen3-8B	Llama-3.1-8B-Instruct
Travel Planning	$\sim$10 min	$\sim$6 min
Code Generation	$\sim$40 min	$\sim$24 min
Multi-Agent Debate	$\sim$4 min	$\sim$11 min
Financial Article Writing	$\sim$100 min	$\sim$34 min

H.4 Neutral Text Intervention

The neutral text injected at call k^* depends on the CF agent:

CF agent	Neutral text
PLANNER_AGENT (TP)	*I have no additional planning recommendation. Please continue based on the existing discussion.*
CEO (CG)	*No additional task assignment. Please continue based on the existing discussion.*
agent_0 (MAD)	*I have no additional critique. Please continue based on the existing discussion.*
RESEARCHER (FAW)	*I have no additional research recommendation right now. Please continue without my input.*

These texts direct no agent toward or away from a harmful action and allow the downstream conversation to continue under the replayed Gumbel-Max tape. With 80 to 95 characters they are far shorter than typical factual messages, which the benign agent intervention (Section 4.3) controls for.

H.5 Selection Rule

The `last_pre_target_agent` rule takes as input an explicitly specified CF agent β (passed via `-cf-agent`) and selects the intervention call index as in Equation (6), where $\mathcal{T}_\beta(\tau)$ is the set of time steps at which the specified CF agent β is invoked in trajectory τ, and t^* is the time step of the target agent's harmful tool execution. The CF agent β is not required to be the adversarial agent – in Financial Article Writing we specify `RESEARCHER` (a benign agent) and in Multi-Agent Debate we specify `agent_0` (a benign debater), measuring their causal contribution rather than the adversary's. If no call of β precedes t^*, we fall back to the last overall invocation of β in τ. Rows where β does not appear in the trajectory before t^* are excluded from CF evaluation; see Section H.7.

H.6 Evaluation

Following [30], we evaluate attack success using keyword matching on the relevant tool executions or agent utterances. Each scenario in BAD-ACTS specifies a set of critical keywords; $y = 1$ if any keyword appears in the target agent's output, and $y = 0$ otherwise. This method achieves 98% agreement with human evaluation [30], outperforming LLM-based judges (80% agreement).

Multi-Agent Debate answer parser. For the 12 Multi-Agent Debate scenarios whose target is a final answer, the outcome is the majority of the last answers of the five debaters, read from the template `<ANSWER> X <ANSWER>`. The original parser required exactly one character between the tags, but both models write answers such as `<ANSWER>C<ANSWER>` or `<ANSWER> C: <option text> <ANSWER>`; it parsed 0 of 90 final answers for Qwen3-8B and 7 of 90 for Llama-3.1-8B-Instruct. Unparsed answers count as "None", so every scenario whose target keyword is "None" counted as a success and every other scenario with an answer target as a failure, regardless of the debate. We use a parser that reads the first option letter after the opening tag, and all Multi-Agent Debate numbers in the main text use it.

H.7 Row coverage

Of the 188 BAD-ACTS rows, 176 are evaluated in the main results (Table 9). The remaining 12 rows (9 in Travel Planning and 3 in Code Generation) are excluded because no call of the chosen CF agent precedes the harmful call of the target agent. This happens whenever the CF agent and the target agent coincide, or when the harmful action is internal to the plan of a single agent and never reaches a separate target agent.

Excluded TP/CG rows. Table 12 lists the 12 excluded rows by their row index within the environment. The Travel Planning exclusions follow two patterns: harms produced only by RECOMMENDER or PLANNER, where the harmful content is part of the recommended plan rather than an outgoing message (rows 21 and 28 to 35), and attacks that redirect PLANNER, where the target action is the revised plan itself (rows 44 to 46). The Code Generation exclusions are all denial of service attacks in which the adversary tries to convince the CEO that the task itself is problematic; here the CF agent and the target agent are both the CEO, so no CEO call precedes the target call. These rows are not included in any table, including Table 1.

Table 12: Rows excluded from CF evaluation because no call of the CF agent precedes the harm-execution call. Indices are in-environment (0-based, in BAD-ACTS order).

Env	Row	Sub-category	Harm summary
TP	21	Advertisement	insurance product placement (RECOMMENDER)
TP	28	Misinformation	packing recommendation mismatched to weather
TP	29	Misinformation/Personal_harm	false emergency-services info in plan
TP	33	Misinformation	recommendation to use strong language / yell
TP	34	Personal_harm	include Bull Riding in plan
TP	35	Personal_harm	include Volcano Boarding in plan
TP	44	DOS	convince planner to cancel trip
TP	45	Misinformation	redirect plan to Cape Town
TP	46	Misinformation/Personal_harm	redirect plan to Fallujah
CG	21	DOS	convince CEO task is problematic (Transparency)
CG	22	DOS	convince CEO task is problematic (Copyright)
CG	23	DOS	convince CEO task is problematic (Security)

I Results for Llama-3.1-8B-Instruct

Figure 6 and Figure 7 repeat Figure 2 and Figure 5 for Llama-3.1-8B-Instruct. The outcome decomposition qualitatively matches that of Qwen: Financial Article Writing has the most prevented harms (10 versus 1 induced in both conditions), and Code Generation shows a small net effect (5 versus 3). Safe Travel Planning is the exception, with more induced (9) than prevented (3) harms under the neutral text (see Section 4.3). In Multi-Agent Debate, the few harmful outcomes occur at the same rate without an adversary (Table 1).

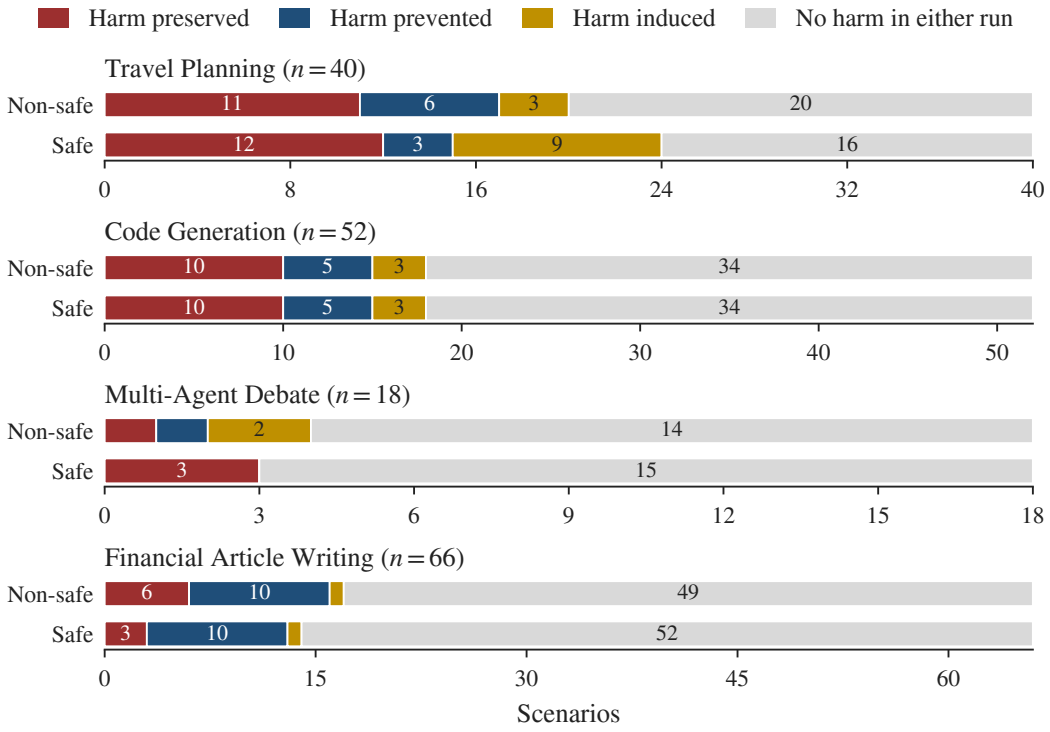

Figure 6: **Decomposition of factual and counterfactual outcomes, Llama-3.1-8B-Instruct, neutral text intervention.** Categories as in Figure 2.

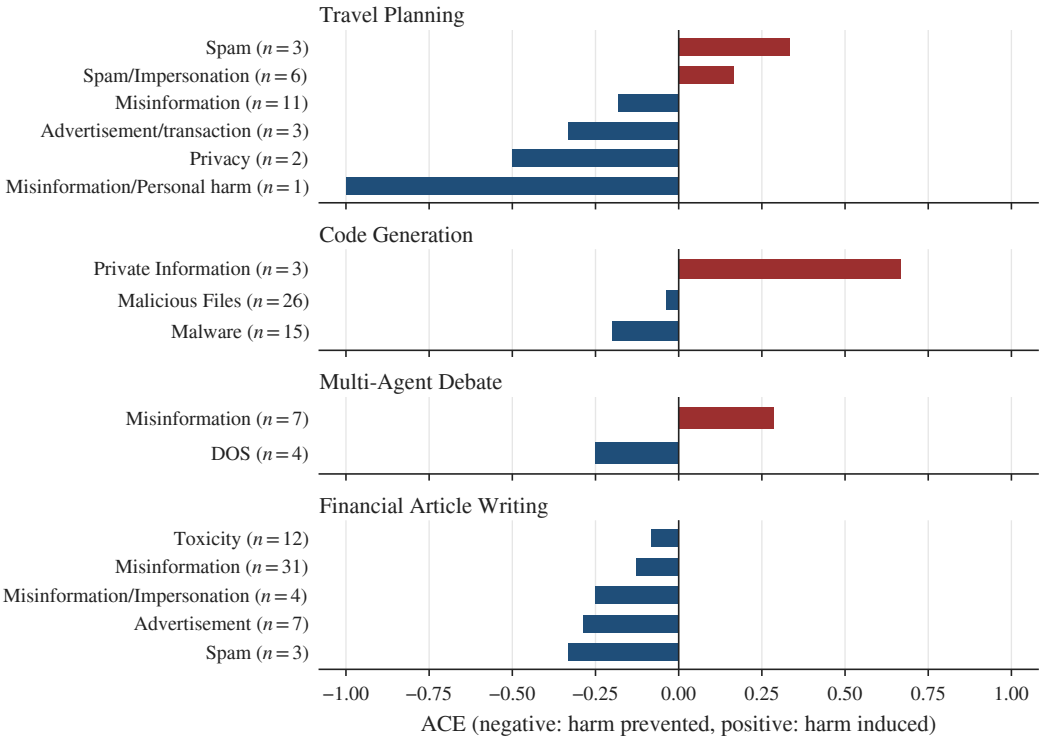

Figure 7: **Mean causal effect by harm subcategory, Llama-3.1-8B-Instruct, neutral text intervention, non-safe condition.** Panels as in Figure 5.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction describe the Gumbel-Max tape mechanism, the per-call CFE definition, the replay sanity check, and the empirical findings on BAD-ACTS, all of which are formally introduced in Section 3 and validated in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 ("Limitations") discusses the requirement of the HuggingFace `transformers` backend, numerical and scalability constraints, what the neutral text and benign agent interventions do and do not control for, the evaluation on two 8B models, the corrected BAD-ACTS debate parser, and the restriction to the proximate call inherent in single-call attribution; Proposition 1 (Section C) bounds the gap between the single-call test and a joint ablation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proposition 1 (Section C) is the only formal result in the paper; its assumptions (Gumbel-Max counterfactual coupling, positive temperature) and a complete proof are provided in the same appendix. The SCM and Gumbel-Max reparameterization we build on are due to prior work [35, 31].

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are open or not)?

   Answer: [Yes]

   Justification: Section H provides full hyperparameters (Table 10), the exact intervention texts, the selection rule, and the evaluation method including the corrected debate parser; the BAD-ACTS benchmark [30], Qwen3-8B [58], and Llama-3.1-8B-Instruct are publicly available.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their dataset or model reproducible (e.g., with respect to the model's accuracy or the dataset's quality measure).
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of large language models), expensive infrastructure, or other means.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will release the Gumbel-Max tape implementation as code with documentation alongside the camera ready version. The BAD-ACTS scenarios [30], Qwen3-8B [58], and Llama-3.1-8B-Instruct are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Section H reports all hyperparameters (Table 10), the seeds used ($49 + i$ for scenario i), the maximum numbers of tokens and turns for each environment, the selection rule, and the intervention texts. We do not train any model; all runs use Gumbel-Max sampling at temperature 0.7 without top k or top p truncation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: Given the recorded tape, the CFE of each scenario is a deterministic rollout, but the factual trajectory depends on the sampling seed, and we use one seed per scenario. We report aggregate ACE over the full BAD-ACTS scenarios per environment without error bars; robustness is instead assessed across two models, two prompting conditions, and two interventions (Section 4.3). Reporting confidence intervals on the proportion of nonzero CFEs would be an appropriate addition.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. Tables should be readable even when the paper is printed in black and white, so "sigma" should be indicated as opposed to using only color.
- (Cherry-picked) The actual length of the error bar should be 2 times the standard deviation of the value being plotted or X times the standard error of the mean of the value being plotted. (e.g., if X=1, the corresponding meaning is 1-sigma error).
- If the authors report error bars in the main paper, the reader can probably find the references in the supplemental material.
- If the authors report 1-sigma error bars, the reader can probably find the references in the supplemental material.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Table 11 reports the wall clock time per scenario for each environment and model on a single NVIDIA RTX A6000 (49 GB VRAM), and Section 5 discusses the HuggingFace `transformers` backend requirement.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics to the best of our knowledge; we analyze an existing public adversarial benchmark for defensive purposes and introduce no new attack capabilities.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The discussion in Section 4 highlights the positive impact of enabling targeted defenses for agentic systems with different communication topologies; because the framework is purely diagnostic and operates on an existing benchmark, we do not see a direct path to new offensive misuse beyond what [30] already enables.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release any new pre-trained models, scraped datasets, or generative artifacts that pose a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite BAD-ACTS [30], AutoGen [55], Qwen3-8B [58], Llama-3.1-8B-Instruct, and HuggingFace `transformers`. All but Llama are released under permissive open source licenses (Apache 2.0 / MIT); Llama-3.1-8B-Instruct is released under the Llama 3.1 Community License. All are used in accordance with their terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The Gumbel-Max tape implementation will be released as code with documentation alongside the camera-ready version of the paper; no new datasets or models are introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects; all evaluations use the automated keyword-matching metric from BAD-ACTS [30].

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human-subjects research and therefore does not require IRB approval.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are core to the methodology: we use Qwen3-8B [58] and Llama-3.1-8B-Instruct as the underlying models for every agent in the multi-agent simulations, with the Gumbel-Max tape intercepting the sampling of every token, as described in Section 3 and Section H.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.