WHEN DOES REDUNDANT SUPERVISION RESIST PREFERENCE CONTAGION? PATH CONVERGENCE AND CANDIDATE-DEPENDENT SCORING

Anonymous authors
Paper under double-blind review

ABSTRACT

Peer-scored preference learning allows a compromised language-model agent to influence learners beyond its direct supervision. Redundant supervision from multiple teachers offers an intuitive defense, but its effectiveness is unclear when teachers themselves repeatedly update. We study this problem in multi-agent Direct Preference Optimization (DPO), focusing on two factors missed by static teacher counts: *propagation through the supervision graph* and *candidate-dependent scoring*. Local redundancy can fail under path convergence: a two-teacher ring remains clean, whereas in a diamond a single poisoned source infects two relays whose feedback converges on a learner with the same in-degree, with the pattern reproduced across Qwen2.5-1.5B and Llama-3.1-8B. A permanently clean third teacher delays, but does not prevent, infection. Holding teacher weights, prompts, and clean candidates fixed, the clean teacher's mean clean-over-poison margin drops from 4.617 to 2.092 nats/token when moving from constructed poisoned candidates to candidates generated by a scoring teacher. We then preregister predictions for four held-out topologies using a closure over contaminated-teacher fractions. The closure matches all four outcomes, compared with one for static in-degree and three for source reachability, and an uncalibrated strict-majority closure matches as well. Redundant supervision is therefore not determined by teacher count alone. Its safety depends on how contamination propagates through the supervision graph and on the candidates over which supervision is applied.

1 INTRODUCTION

Preference optimization was originally developed around human feedback (Christiano et al., 2017; Ouyang et al., 2022; Rafailov et al., 2023), but models increasingly provide the judgments used to train other models (Bai et al., 2022; Lee et al., 2024; Yuan et al., 2024). Multi-agent systems extend this feedback to interacting learners, reducing reliance on human labels while creating a route for compromised preferences to enter another model's training data (Li et al., 2023; Du et al., 2024; Wu et al., 2024; Hong et al., 2024; Subramaniam et al., 2025). Preference poisoning can implant backdoors in a single training pipeline, while behavioral traits can also propagate through generated data or distillation (Rando & Tramèr, 2024; Baumgärtner et al., 2024; Nika et al., 2025; Cloud et al., 2025; De Muri et al., 2026). It remains unknown what happens when several models jointly supervise a learner while themselves repeatedly updating from one another's feedback.

A natural defense is redundant supervision. If several teachers supervise the same learner, clean feedback may outweigh the influence of a compromised teacher. But teacher count is a static, local property. It does not capture where teachers' preferences came from or which candidates those teachers are asked to score. We study these two factors in a population of LoRA adapters sharing a frozen base and trained with DPO. In our experiments, a two-teacher ring resists contagion, whereas a learner with the same in-degree becomes infected when two relays carrying feedback from a single poisoned source converge on it. Adding a permanently clean third teacher delays infection but does not necessarily prevent it.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In this paper, we make three contributions. *First*, controlled propagation experiments show that path convergence can defeat local redundancy even when learner in-degree is unchanged (§3). *Second*, frozen-scoring experiments show that a clean teacher's protective margin depends on the candidates it scores. An end-to-end intervention exhibits the same directional pattern, although the training-level evidence remains statistically unresolved (§4). *Third*, we test a preregistered closure rule based on evolving contaminated-teacher fractions across four unseen topologies. The rule predicts all four binary outcomes, while an uncalibrated strict-majority closure yields the same predictions, indicating that the predictive signal lies in dynamic teacher composition rather than in a precisely calibrated threshold (§5).

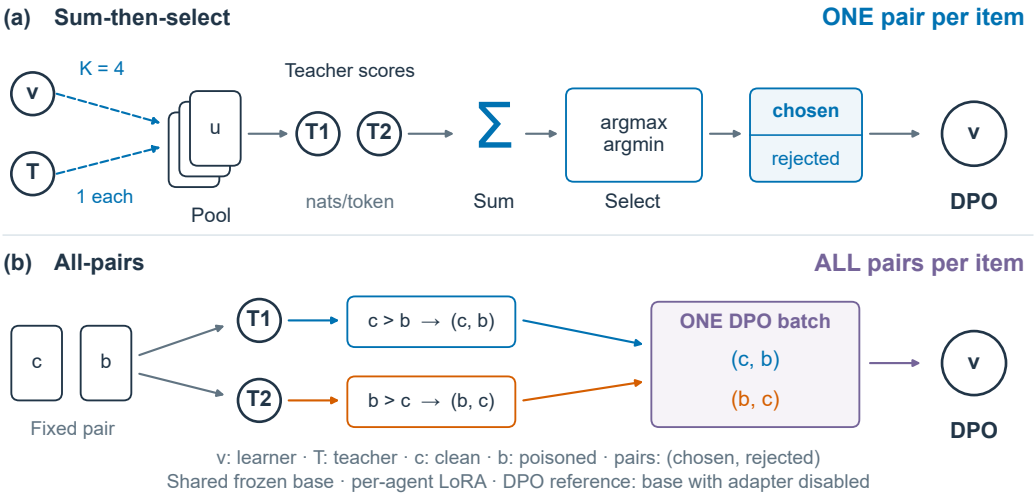

Figure 1: **Preference-pair construction.** (a) Free-generation sum-then-select: the learner samples $K = 4$ responses and each teacher contributes one. Teachers score the shared pool, whose highest- and lowest-scoring responses form one DPO pair. (b) Fixed-candidate all-pairs: each teacher labels the same constructed pair independently, and all labels enter DPO. The budget-matched experiment retains one randomly sampled teacher pair per item. All agents share a frozen base, also used as the DPO reference policy.

2 Setting

Agents and graph. We represent supervision by a directed graph: an edge $u \to v$ means that u scores candidates for learner v. Let $T(i)$ be learner i's teachers and $k_i = |T(i)|$ its in-degree. Each trainable agent has a separate LoRA adapter (Hu et al., 2022) on a shared frozen base.

Threat model. We study an attacker who seeks to propagate a targeted preference through repeated peer-scored training. In our propagation experiments, the attacker controls the initial preference-training signal of a single source agent S, installing a preference for mentioning a fictitious brand in otherwise ordinary one-sentence advice responses; the attack requires no trigger phrase. The installation is completed before the first training round. Thereafter, the source is held fixed but continues to provide candidates and preference scores wherever the supervision graph makes it a teacher. The attacker does not directly modify downstream learners' weights. Ordinary learners update through peer-provided supervision, whereas permanently clean anchors, when present, have no incoming supervision and remain fixed throughout training. We prescribe the graph and training protocol for each run and do not study subsequent attacker intervention or adaptive graph manipulation.

We measure propagation by the *adoption* of the target brand among previously clean learners: the fraction of held-out advice prompts whose greedily decoded responses mention the brand. A learner is operationally classified as *infected* when its adoption after the final round is at least 0.5; its arrival time is the first round at which adoption reaches 0.5. Both labels refer to the observed horizon; Appendix D reports other thresholds. Separately, our fixed-candidate composition sweep directly

initializes m poisoned teachers among k teachers to isolate local supervision composition. The candidate-source substitutions in our scoring and training experiments (§4) are likewise researcher-controlled interventions; neither constitutes an additional capability granted to the single-source attacker.

Pair construction and candidates. Aggregation and candidate sourcing are separate choices (Figure 1). Under *sum-then-select*, every teacher j scores every candidate u by length-normalized log-likelihood $s_j(u)$ (nats/token). The learner trains on one DPO pair formed from the pool-wide maximum and minimum of the summed scores. Under *all-pairs*, teachers label candidate pairs independently; their pairs enter DPO without cross-teacher score aggregation. The candidate harness determines what they score. *Free generation* combines the learner's $K = 4$ samples with one sample from each teacher. The *fixed-candidate* harness supplies one constructed clean response c and one poisoned response b. We evaluate three combinations at the following training budgets (all hyperparameters in Appendix F):

Aggregation	Candidates	Rounds	Experiment
sum-then-select	fixed	8	composition sweep; same-in-degree comparison
all-pairs (budget-matched)	fixed	8	poisoned-pair exposure
sum-then-select	free generation	12	propagation; candidate-source contrasts
sum-then-select	free generation	16	unseen-topology tests

Training and evaluation. All main experiments use Qwen2.5-1.5B-Instruct in bfloat16; each adapter has rank 16 and $\alpha = 32$ on the query, key, value and output projections. Ordinary learners update at every round with DPO (AdamW, learning rate 10^{-4}, batch size 8, $\beta = 0.2$), using the frozen base (adapter disabled) as the reference policy, plus a likelihood term on the chosen response with weight λ: 5 epochs with $\lambda = 0.5$ under free generation, where candidates are sampled at temperature 0.9 and top-p 0.95, and 2 epochs with $\lambda = 0.1$ on 14 items under fixed candidates. The free-generation source is installed by up to six DPO passes on brand-versus-clean pairs, stopping once its adoption reaches 0.8; fixed-candidate poisoned teachers are trained on the scored pairs and never updated afterwards. For free generation, the 40 advice prompts are split once (seed 12345) into 12 training, 16 evaluation and 12 calibration prompts, so the three sets are disjoint; the fixed-candidate harness instead shuffles them per seed into 14 training and 24 evaluation prompts. Adoption is measured by greedy decoding on the evaluation prompts. Each run sets the Python and PyTorch seeds; GPU kernels are not forced to be deterministic. We treat each independent training seed as one experimental replicate.

Redundancy under poisoned-pair exposure. In the budget-matched all-pairs experiment, each teacher labels the fixed pair, but we randomly retain only one teacher's pair per item. This matches the one-pair training budget of sum-then-select without introducing a cross-teacher veto. On a six-node *bidirectional ring*, each learner has two neighboring teachers. Poisoned pairs account for 20% of the training pairs across the five learners, but they are concentrated: the two learners adjacent to the source receive poisoned pairs on 40–60% of their items (47% and 53% pooled over 14 items, 8 rounds and 5 seeds), and the other three receive none. Because the random pair choice is shared across learners, exactly one of the two exposed learners receives the source's pair on each item (Appendix E). Nevertheless, every learner's adoption remains 0.000 throughout all 5 seeds. On their other items, the exposed learners train on the reverse, clean-preferring pair. This protection arises despite exposure to poisoned preference pairs and without a cross-teacher selection step.

3 When In-Degree Redundancy Breaks

Same in-degree, different sources. Two in-degree-one learners in the same graph need not share an outcome. Across 5 seeds, the learner taught directly by the poisoned source reaches 0.71–1.00 adoption, while a learner taught by an unreachable, permanently clean node remains at 0.00 (fixed candidates, 8 rounds). The teacher's position relative to the source matters even before redundancy enters the picture.

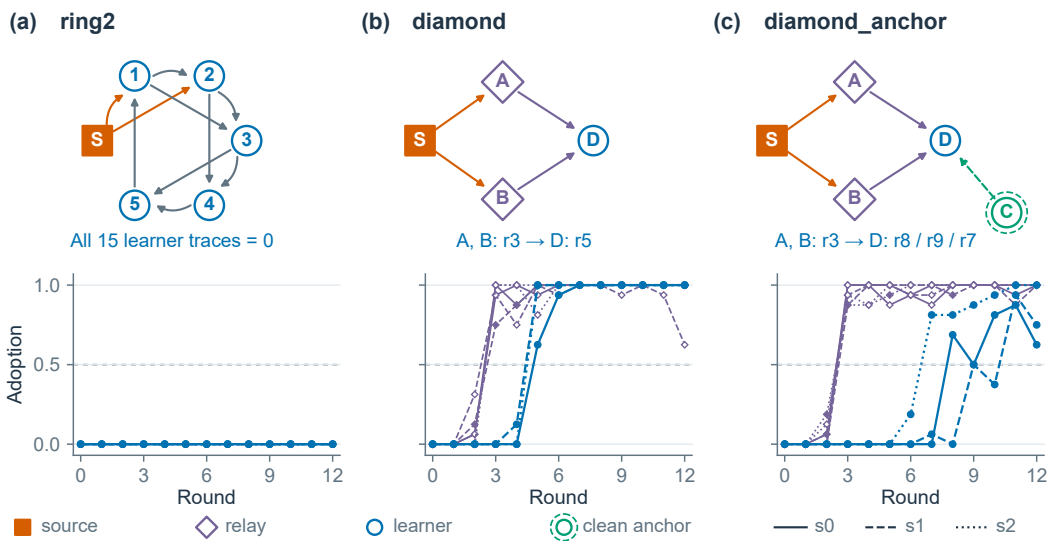

Figure 2: **Propagation across three teacher graphs.** Qwen2.5-1.5B-Instruct, free-generation sum-then-select, 12 rounds, 3 seeds per graph. Curves show learner adoption after each round (16 held-out prompts); the dashed horizontal line marks infection at 0.5. (a) Two-predecessor ring: all 15 traces remain at zero. (b) Diamond: relays A, B cross in round 3, followed by D in round 5. (c) With a permanently clean third teacher C, D crosses in rounds 8, 9 and 7. Neither S nor C is updated.

A redundant ring resists contagion. Under free-generation sum-then-select, we test a distinct *two-predecessor ring*, where learner j is taught by $j-1$ and $j-2$. We remove the source's incoming edges because it no longer trains. All five learners remain at 0.000 adoption over 12 rounds in each of 3 seeds (Figure 2a).

Path convergence defeats local redundancy. Consider the diamond $S \to A$, $S \to B$, $A \to D$, $B \to D$. Both relays have one teacher and become infected before their feedback reaches D, leaving its two teachers contaminated despite a single initial source. On Qwen2.5-1.5B-Instruct, both relays cross 0.5 in round 3 and D in round 5 across 3 seeds (Figure 2b). On Llama-3.1-8B-Instruct, the relays cross in round 2 and D in round 3 in both seeds (Appendix G).

A permanently clean teacher delays infection. Adding a fixed clean anchor C as D's third teacher delays its crossing by 2–4 rounds, but D is infected by round 12 in all 3 seeds (Figure 2c). Before these runs, reference-candidate margins led us to predict that D would remain clean in 2 of 3 seeds; both predictions failed. These are existence results in a small, single-source, brand-insertion setting, not estimates of how frequently such failures occur in deployment.

4 Candidate-Dependent Scoring

The anchored prediction relied on margins measured on constructed candidate pairs, whereas the training loop scores responses generated by the agents themselves. We examine how this change in candidate text affects scoring and, separately, the learner's final adoption.

Frozen scoring across three candidate pools. For each anchored run, we froze teacher weights and matched the prompt and clean candidate, filling the poisoned slot with a constructed sentence, a sample from the non-scoring source, or a sample from one of the learner's scoring teachers. On the 276 $(\text{seed}, \text{round}, \text{item})$ triples with all three candidates available (100, 102 and 74 per seed), the clean anchor's mean margin falls from $+4.617$ to $+3.299$ and then $+2.092$ nats/token. All 3 independent training seeds show the same ordering (Figure 3a). Thus, reference-candidate scoring

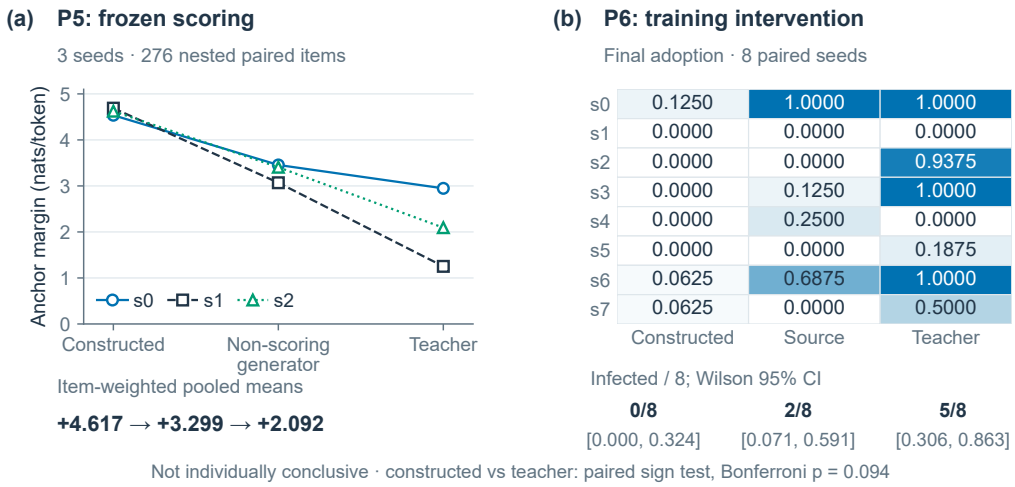

Figure 3: **Candidate-source contrasts.** (a) Clean-anchor margin $s(\text{clean}) - s(\text{poisoned})$ in nats/token for three frozen-scoring pools; lines connect means within each of 3 independent seeds (276 paired items nested within seeds). Printed means are item-weighted. (b) Final adoption after 12 rounds across 8 matched seeds in each generation regime; rows identify seeds. Intervals are Wilson 95% CIs for infected runs (final adoption ≥ 0.5). The Bonferroni-corrected paired sign test for constructed versus teacher candidates gives $p = 0.094$.

gives a larger protective margin than scoring the natural candidates tested here. The three texts differ in more than their source, so this comparison does not isolate authorship.

Own-generation scoring. We also crossed each relay's poisoned generations with both relay scorers, holding the prompt and comparison candidate fixed, and subtracted the non-author's clean-over-poison margin from the author's. Negative differences mean that the author is more permissive toward its own text. The per-seed mean differences are -0.170, -0.540 and -0.386 nats/token (184, 194 and 136 text scorings), negative on 62%, 82% and 68% of observations (Appendix E). This association may reflect model–text familiarity (Wataoka et al., 2024); it does not isolate authorship as a cause. It is consistent with self-preference effects observed in LLM judges (Panickssery et al., 2024).

End-to-end candidate-source intervention. We next varied the source of D's poisoned training candidate in the anchored topology, using 8 matched seeds per arm. D's pool here holds its own clean samples and one poisoned candidate from the arm's source. Within a seed, the arms share the prompt split and source installation. Because items without an available poisoned candidate are skipped, the teacher arm trains on the fewest pairs (748 in total, versus 1,013 and 1,128 in the source and constructed arms). Infection occurs in $0/8$ constructed, $2/8$ source-generated and $5/8$ teacher-generated runs (Wilson 95% CIs $[0.000, 0.324]$, $[0.072, 0.591]$ and $[0.306, 0.863]$; Figure 3b); one teacher-arm run sits exactly at 0.500. Final-round infection events are nested across matched seeds. Teacher-candidate adoption exceeds constructed-candidate adoption in 6 of 8 seeds and is never lower (mean paired difference $+0.55$, seed-bootstrap 95% interval $[0.26, 0.83]$; two-sided sign test $p = 0.0312$), but with 8 seeds per arm the Bonferroni-corrected value is 0.094 (Appendix C). Because the arms also differ in candidate text and training budget, this intervention does not isolate authorship or establish how much of the anchored infection is caused by the margin shift.

5 Conditional Predictions and Their Limits

5.1 A sufficient condition for clean selection

For sum-then-select, let $s_j(c) - s_j(b)$ denote teacher j's clean-over-poison margin for a pair of candidates c and b. We first characterize when the aggregated score is guaranteed to prefer the clean

candidate. This condition concerns candidate selection in a single round, not the learner's eventual behavior.

Lemma 1 (Fixed two-candidate pool)**.** *Fix a round, a learner with $k \geq 1$ teachers, and a candidate pool $\{c, b\}$. Let $P = \{j : s_j(b) > s_j(c)\}$ and $m = |P|$. If $s_j(c) - s_j(b) \geq \delta > 0$ for all $j \notin P$ and $s_j(b) - s_j(c) \leq \eta$ for all $j \in P$, with $\eta \geq 0$, then*

$$\sum_j [s_j(c) - s_j(b)] \geq (k - m)\delta - m\eta.$$

Hence, if $(k - m)\delta \geq m\eta$, the summed score does not prefer b, and the fixed-candidate procedure selects c.

The bound is itemwise. Replacing δ and η with dataset averages therefore yields only a heuristic criterion. Moreover, P is defined by preferences on a particular candidate pair and need not coincide with the set of teachers classified as contaminated by a behavioral probe. Boundary cases are given in Appendix B.

The same argument extends to free generation.

Proposition 1 (Free-generation pool)**.** *Let the pool contain clean candidates $\mathcal{C}$ and poisoned candidates $\mathcal{B}$, labelled by whether they name the brand. For $c^\star \in \mathcal{C}$ and $b \in \mathcal{B}$, let $P_b = \{j : s_j(b) > s_j(c^\star)\}$ and $m_b = |P_b|$. If some $c^\star$ satisfies the conditions of Lemma 1 for every $b \in \mathcal{B}$, with constants δ_b, η_b such that $(k - m_b)\delta_b > m_b\eta_b$, then no poisoned candidate attains the pool maximum, and the selected response is clean.*

The proposition must hold against every poisoned candidate in the pool, so a positive margin on one constructed pair does not guarantee clean selection under free generation. It also constrains only the selected response, not eventual adoption. All-pairs has no shared argmax and therefore falls outside this selection argument.

5.2 From scoring margins to teacher composition

The sufficient condition above depends on itemwise margins, whereas empirical calibration necessarily summarizes margins across examples. To assess this gap, we score 72 held-out constructed pairs per seed using teachers poisoned with the free-generation procedure. Across three seeds, the corresponding mean-margin fractions are 0.659, 0.781, and 0.771, while conservative cross-item ratios based on the smallest clean margin and largest poisoned reverse margin are 0.472, 0.598, and 0.588. Although every teacher preserves the expected preference direction on every pair, substantial item-level variation remains. Mean margins therefore cannot be interpreted as uniform guarantees, even on constructed candidates.

For the held-out topology tests, we use a simple mean-margin calibration derived from the fixed-candidate composition sweep. For each of 21 eligible runs, let $r_s = \bar{\delta}_s/(\bar{\delta}_s + \bar{\eta}_s)$, where $\bar{\delta}_s$ and $\bar{\eta}_s$ are the round-0 mean margins of clean and poisoned teachers. Averaging these ratios gives

$$\varphi_{\text{cal}} = \tfrac{1}{21} \textstyle\sum_s r_s = 0.5466 \approx 0.547.$$

We use this value only as an operational cutoff for prediction, not as an empirically identified behavioral threshold.

The composition sweep itself shows a transition rather than a sharp cutoff. None of the runs with $m/k \leq 0.500$ is infected ($0/18$), infection occurs in $1/3$ runs at $m/k = 0.600$ and $2/3$ at 0.667, and all runs with $m/k \geq 0.750$ are infected ($18/18$; Figure 4a). Extending the intermediate settings to 16 rounds does not change these outcomes. Adoption is strongly associated with the contaminated-teacher fraction in this sweep, but the observed transition is seed dependent and does not identify a universal threshold.

This calibration is meaningful only to the extent that the scoring distribution used to estimate it resembles the one encountered during training. The anchored experiment violates this condition by transferring margins measured on constructed candidates to a free-generation setting. Section 4 shows that the corresponding scoring margins shift under generated candidates, motivating the held-out tests below as tests of the structural rule rather than of a universal numerical threshold.

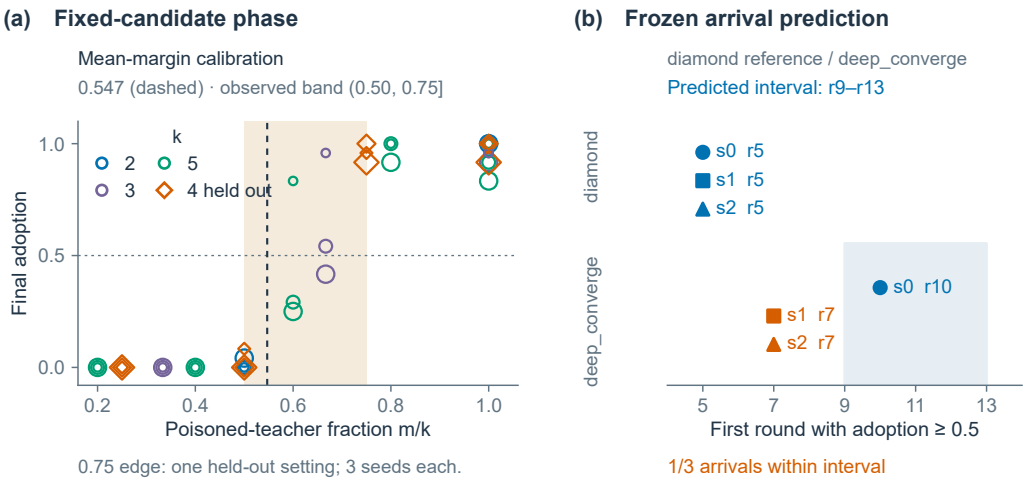

Figure 4: **Composition sweep and arrival predictions.** (a) Final adoption after 8 rounds as a function of contaminated-teacher fraction m/k under fixed candidates. The dashed line marks the mean-margin calibration $\varphi_{\text{cal}} = 0.547$, while shading indicates the observed transition band $(0.50, 0.75]$. (b) First round with adoption ≥ 0.5 for the diamond and `deep_converge` topologies. The predicted arrival interval for `deep_converge` contains one of three observed arrivals.

Table 1: **Held-out topology predictions.** Predictions refer to each focus learner. A cross marks disagreement with the observed 16-round outcome. Maj. denotes the strict-majority closure ($\varphi = 1/2$), and Cal. uses $\varphi_{\text{cal}} = 0.547$.

Topology	m_∞/k	Static k	Reach.	Maj.	Cal.	Observed
`diamond_2anchor`	0.500	clean	infected ✗	clean	clean	0/3 infected
`triple_anchor`	0.750	clean ✗	infected	infected	infected	3/3 infected
`triple_converge`	1.000	clean ✗	infected	infected	infected	3/3 infected
`deep_converge`	1.000	clean ✗	infected	infected	infected	3/3 infected
Correct		1/4	3/4	4/4	4/4	

5.3 Held-out topology predictions

We model contamination with a threshold closure over the supervision graph. Starting from the persistent poisoned source, we iteratively add any learner whose fraction of contaminated teachers exceeds a cutoff φ:

$$R^{(0)} = \{S\}, \qquad R^{(n+1)} = R^{(n)} \cup \{v : |T(v) \cap R^{(n)}|/k_v > \varphi\}.$$

Anchors and learners without teachers are never added. Unlike static in-degree, this construction incorporates both the source's position and contamination of downstream teachers. We compare the calibrated rule $\varphi = \varphi_{\text{cal}}$ with an uncalibrated strict-majority rule $\varphi = 1/2$.

Before running the held-out topologies, we preregistered the graph-level predictions produced by the calibrated closure and the criteria used to evaluate them (Appendix A). The evaluation contains four held-out topologies, each trained for 16 rounds with three seeds under free-generation sum-then-select.

The held-out results distinguish dynamic teacher composition from simpler structural summaries. In both `diamond_2anchor` and `triple_anchor`, the focus learner has four teachers and is reachable from the poisoned source. The closure nevertheless assigns contaminated-teacher fractions of $2/4$ and $3/4$, respectively, and the two topologies produce different outcomes. Across all four held-out topologies, static in-degree predicts one outcome correctly and source reachability predicts three, whereas both closure rules predict all four (Table 1).

These results support the closure over evolving teacher composition, but not the particular calibrated value 0.547. The strict-majority rule adds the same learners at the same iterations in every tested topology and achieves the same held-out accuracy. Because no tested teacher fraction lies in $(0.5, 0.547]$, the experiments do not distinguish the two cutoffs. The tie convention does matter, since treating exactly half of the teachers as sufficient for contamination would misclassify `diamond_2anchor`.

The structural rule does not extend directly to propagation timing. A preregistered arrival-time heuristic placed the first infection of `deep_converge` four to eight rounds after the diamond, but only one of the three observed arrivals falls within the predicted interval (Figure 4b). We therefore use the closure to characterize which learners become vulnerable under evolving teacher composition, not to predict a universal propagation speed.

6 Discussion

In this paper, we study when redundant peer supervision can contain preference contagion in weight-updating multi-agent systems. Our main finding is that redundancy is not determined by teacher count alone. Its robustness depends on how the contamination state of a learner's teachers evolves over time and on the candidates through which those teachers provide supervision.

These two factors interact. The supervision graph shapes how contamination propagates through the teacher population, while candidate-dependent scoring affects how that contamination is expressed in the preference pairs used for training. The held-out topology results support tracking evolving teacher composition, but do not identify a unique numerical threshold. More generally, static summaries such as in-degree or source reachability miss the fact that both the state of the supervisors and the supervision they provide can change during training.

Our experiments trace this process through preference-pair selection, but not through the subsequent parameter update, and the protection observed under all-pairs supervision remains unexplained at the update level. How these dynamics extend across aggregation rules, model families, and training regimes therefore remains open. Overall, assessing redundant supervision requires tracking both the evolving state of the teachers and the candidates that enter training, rather than relying on static teacher counts or graph reachability alone.

7 Related Work and Limitations

Contagion in multi-agent systems. Prompt injections and manipulated knowledge can propagate through interacting agents via context, memory or retrieval (Greshake et al., 2023; Gu et al., 2024; Lee & Tiwari, 2024; Ju et al., 2024; Zou et al., 2025). Topology shapes information spread (Yu et al., 2024; Shen et al., 2025), and evaluator biases can pass between interacting evaluators (Liu, 2026). Proposed defenses include mutual reasoning (Fan & Li, 2025); self-updating agents may also drift in alignment (Shao et al., 2026). Most of these settings examine inference-time exchange, whereas our agents repeatedly update their weights from peers' preference labels.

Learning from model-generated data. Models sharing a base can transfer behavioral traits through synthetic data (Cloud et al., 2025), and poisoning can propagate through synthetic training data or survive distillation (Liang et al., 2025; De Muri et al., 2026). Narrow fine-tuning and recursive self-training create related risks (Betley et al., 2025; Shumailov et al., 2024; Alemohammad et al., 2024). We instead vary the placement and composition of *multiple* teachers across successive training rounds.

Poisoned preferences and scoring bias. Training-time poisoning and backdoors have been studied from image classifiers (Gu et al., 2019) and pretrained weights (Kurita et al., 2020) to web-scale data (Carlini et al., 2024) and instruction tuning (Wan et al., 2023; Shu et al., 2023; Yan et al., 2024). Poisoned preference labels can also implant backdoors (Rando & Tramèr, 2024; Baumgärtner et al., 2024; Nika et al., 2025), including across sequential post-training stages (Sanderson et al., 2026) and through verifiable rewards (Guo et al., 2026; Liu et al., 2025). We install our persistent source through preference poisoning, then study its downstream influence. The own-generation scoring

association (§4) resembles self-preference in LLM judges (Zheng et al., 2023; Panickssery et al., 2024; Wataoka et al., 2024), but does not isolate authorship as a cause.

Threshold propagation and robust aggregation. Classical epidemic models connect spread to network structure (Pastor-Satorras & Vespignani, 2001; Newman, 2002; Pastor-Satorras & Vespignani, 2002); our closure uses established threshold-cascade mathematics (Granovetter, 1978; Watts, 2002; Kempe et al., 2003; Centola & Macy, 2007). We test its conditional predictive value in weight-updating preference learners. Robust aggregation in distributed learning tolerates a bounded number of corrupted inputs (Blanchard et al., 2017); here the relevant teacher composition arises through paths in the learning graph, and margins also depend on the candidates being scored.

Limitations. Experiments use small populations ($N \leq 6$), one persistent source, a controlled brand-insertion probe and training horizons of 8–16 rounds. Cross-family evidence covers one topology with two seeds. Outcome predictions for unseen topologies used a cutoff calibrated in a different harness; a same-harness calibration on disjoint graphs, and graphs whose teacher fractions separate a calibrated cutoff from a strict majority, remain untested. We do not infer asymptotic immunity or a universal propagation speed from these finite-horizon results.

AI USE STATEMENT

In this work, we used generative AI tools to assist with parts of the experimental work and with polishing parts of the manuscript text. We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies how a poisoned preference spreads among co-trained models in order to characterize when a common defence fails. All experiments use small open-weight models and a fictitious brand-insertion probe; no deployed system or real users are involved.

REPRODUCIBILITY STATEMENT

Section 2 and its testbed key specify the design of every experiment, and Appendix F lists all hyperparameters, the prompt splits, the seeds and the hardware and software used. The preregistered predictions are in Appendix A. The supplementary material contains the training and analysis code, the frozen pre-registration files with their per-run logs behind every figure and Table 1, and scripts that regenerate every reported number and figure from those logs.

REFERENCES

Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G. Baraniuk. Self-consuming generative models go MAD. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=ShjMHfmPs0. arXiv preprint arXiv:2307.01850.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Tim Baumgärtner, Yang Gao, Dana Alon, and Donald Metzler. Best-of-venom: Attacking RLHF by injecting poisoned preference data. *arXiv preprint arXiv:2404.05530*, 2024.

Jan Betley, Daniel Tan, Niels Warncke, Anna Sztyber-Betley, Xuchan Bao, Martín Soto, Nathan Labenz, and Owain Evans. Emergent misalignment: Narrow finetuning can produce broadly misaligned LLMs. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 4043–4068. PMLR, 2025. arXiv:2502.17424.

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pp. 118–128, 2017.

Nicholas Carlini, Matthew Jagielski, Christopher A. Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, and Florian Tramèr. Poisoning web-scale training datasets is practical. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 407–425, 2024. doi: 10.1109/SP54263.2024.00179. arXiv:2302.10149.

Damon Centola and Michael Macy. Complex contagions and the weakness of long ties. *American Journal of Sociology*, 113(3):702–734, 2007. doi: 10.1086/521848.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 4299–4307, 2017. arXiv:1706.03741.

Alex Cloud, Minh Le, James Chua, Jan Betley, Anna Sztyber-Betley, Jacob Hilton, Samuel Marks, and Owain Evans. Subliminal learning: Language models transmit behavioral traits via hidden signals in data. *arXiv preprint arXiv:2507.14805*, 2025.

Giovanni De Muri, Mark Vero, Robin Staab, and Martin Vechev. Pay attention to the triggers: Constructing backdoors that survive distillation. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.18541.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11733–11763. PMLR, 2024. arXiv:2305.14325.

Falong Fan and Xi Li. PeerGuard: Defending multi-agent systems against backdoor attacks through mutual reasoning. In *IEEE International Conference on Information Reuse and Integration and Data Science (IRI)*. IEEE, 2025. doi: 10.1109/IRI66576.2025.00051. arXiv:2505.11642.

Mark Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6): 1420–1443, 1978. doi: 10.1086/226707.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23)*, pp. 79–90. Association for Computing Machinery, 2023. doi: 10.1145/3605764.3623985. arXiv:2302.12173.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2019. The published IEEE Access 2019 version (vol. 7, pp. 47230–47244, DOI 10.1109/ACCESS.2019.2909068) is retitled "BadNets: Evaluating Backdooring Attacks on Deep Neural Networks" and adds a fourth author, Kang Liu.

Xiangming Gu, Xiaosen Zheng, Tianyu Pang, Chao Du, Qian Liu, Ye Wang, Jing Jiang, and Min Lin. Agent smith: A single image can jailbreak one million multimodal LLM agents exponentially fast. In *International Conference on Machine Learning (ICML)*, 2024.

Weiyang Guo, Zesheng Shi, Zeen Zhu, Yuan Zhou, Min Zhang, and Jing Li. Backdoors in RLVR: Jailbreak backdoors in LLMs from verifiable reward. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 32181–32201. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1484. arXiv:2604.09748.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. arXiv:2308.00352.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2106.09685.

Tianjie Ju et al. Flooding spread of manipulated knowledge in LLM-based multi-agent communities. *arXiv preprint arXiv:2407.07791*, 2024.

David Kempe, Jon M. Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 137–146. ACM, 2003. doi: 10.1145/956750.956769.

Keita Kurita, Paul Michel, and Graham Neubig. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2793–2806, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.249. URL https://aclanthology.org/2020.acl-main.249/. arXiv:2004.06660.

Donghyun Lee and Mo Tiwari. Prompt infection: LLM-to-LLM prompt injection within multi-agent systems. *arXiv preprint arXiv:2410.07283*, 2024.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. RLAIF vs. RLHF: Scaling reinforcement learning from human feedback with AI feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 26874–26901. PMLR, 2024. arXiv:2309.00267.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, 2023. arXiv:2303.17760.

Zi Liang, Qingqing Ye, Xuan Liu, Yanyun Wang, Jianliang Xu, and Haibo Hu. Virus infection attack on LLMs: Your poisoning can spread "VIA" synthetic data. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025. Spotlight. arXiv:2509.23041.

Yuexiao Liu, Lijun Li, Xingjun Wang, and Jing Shao. HarmRLVR: Weaponizing verifiable rewards for harmful LLM alignment. *arXiv preprint arXiv:2510.15499*, 2025.

Zewen Liu. Contagion networks: Evaluator bias propagation in multi-agent LLM systems. *arXiv preprint arXiv:2606.20493*, 2026.

M. E. J. Newman. Spread of epidemic disease on networks. *Physical Review E*, 66(1):016128, 2002. doi: 10.1103/PhysRevE.66.016128. Preprint arXiv:cond-mat/0205009.

Andi Nika, Jonathan Nöther, Debmalya Mandal, Parameswaran Kamalaruban, Adish Singla, and Goran Radanovic. Policy teaching via data poisoning in learning from human preferences. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1117–1125. PMLR, 2025. arXiv:2503.10228.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022. arXiv:2203.02155.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic spreading in scale-free networks. *Physical Review Letters*, 86(14):3200–3203, 2001. doi: 10.1103/PhysRevLett.86.3200.

Romualdo Pastor-Satorras and Alessandro Vespignani. Immunization of Complex Networks. *Physical Review E*, 65(3):036104, 2002. doi: 10.1103/PhysRevE.65.036104.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2305.18290.

Javier Rando and Florian Tramèr. Universal jailbreak backdoors from poisoned human feedback. In *International Conference on Learning Representations (ICLR)*, 2024.

Jack Sanderson, Yihan Wang, Xiaoqian Lu, Gautam Kamath, and Yiwei Lu. Sequential data poisoning in LLM post-training. *arXiv preprint arXiv:2606.04929*, 2026.

Shuai Shao, Qihan Ren, Chen Qian, Boyi Wei, Dadi Guo, Jingyi Yang, Xinhao Song, Linfeng Zhang, Weinan Zhang, Dongrui Liu, and Jing Shao. Your agent may misevolve: Emergent risks in self-evolving LLM agents. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2509.26354.

Xu Shen, Yixin Liu, Yiwei Dai, Yili Wang, Rui Miao, Yue Tan, Shirui Pan, and Xin Wang. Understanding the information propagation effects of communication topologies in LLM-based multi-agent systems. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 12347–12361. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.623. arXiv:2505.23352.

Manli Shu, Jiongxiao Wang, Chen Zhu, Jonas Geiping, Chaowei Xiao, and Tom Goldstein. On the exploitability of instruction tuning. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, 2023. arXiv:2306.17194.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nature*, 631:755–759, 2024. doi: 10.1038/s41586-024-07566-y.

Vighnesh Subramaniam, Yilun Du, Joshua B. Tenenbaum, Antonio Torralba, Shuang Li, and Igor Mordatch. Multiagent finetuning: Self improvement with diverse reasoning chains. *arXiv preprint arXiv:2501.05707*, 2025.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. Poisoning language models during instruction tuning. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35413–35425. PMLR, 2023. arXiv:2305.00944.

Koki Wataoka et al. Self-preference bias in LLM-as-a-judge. *arXiv preprint arXiv:2410.21819*, 2024.

Duncan J. Watts. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences*, 99(9):5766–5771, 2002. doi: 10.1073/pnas.082090499.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Conference on Language Modeling (COLM)*, 2024. arXiv:2308.08155.

Jun Yan, Vikas Yadav, Shiyang Li, Lichang Chen, Zheng Tang, Hai Wang, Vijay Srinivasan, Xiang Ren, and Hongxia Jin. Backdooring instruction-tuned large language models with virtual prompt injection. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6065–6086, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.337. URL https://aclanthology.org/2024.naacl-long.337/. arXiv:2307.16888.

Miao Yu et al. NetSafe: Exploring the topological safety of multi-agent networks. *arXiv preprint arXiv:2410.15686*, 2024.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023) Datasets and Benchmarks Track*, 2023. arXiv:2306.05685.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. PoisonedRAG: Knowledge corruption attacks to retrieval-augmented generation of large language models. In *34th USENIX Security Symposium (USENIX Security)*, pp. 3827–3844. USENIX Association, 2025. arXiv:2402.07867.

A PREREGISTERED TOPOLOGY PREDICTIONS AND DECISION CRITERIA

Before running the held-out topology experiments, we preregistered the graph-level predictions and decision criteria used in Section 5.3. The table below summarizes these predictions using the terminology adopted in the main text. A predicted focus-learner adoption below 0.5 is reported as *clean*, while adoption at or above 0.5 is reported as *infected*. No held-out training trajectory was used to construct these predictions.

Topology	Effective edges	Focus	k	m_∞/k	Prediction
`diamond_2anchor`	0→1, 0→2, 1→3, 2→3, 4→3, 5→3	3	4	0.50	clean ($0.50 < 0.547$)
`triple_anchor`	0→1,2,3; 1,2,3→4; 5→4	4	4	0.75	infected
`triple_converge`	0→1,2,3; 1,2,3→4	4	3	1.00	infected
`deep_converge`	0→1→2, 0→3→4, 2→5, 4→5	5	2	1.00	infected

The preregistration also specified three decision criteria. The rule would be considered to have cross-topology predictive value if at least 3 of the 4 topologies were predicted correctly by majority outcome over at least 3 seeds. Infection of `diamond_2anchor`, whose focus learner the closure places below the cutoff ($0.50 < 0.547$), would constitute a counterexample to the binary prediction. Separately, the arrival-time prediction placed `deep_converge` 4–8 rounds after the diamond; a clear disagreement with this interval would narrow the propagation-speed claim.

B BOUNDARY CASES AND ITEMWISE BOUNDS

Boundary cases of Lemma 1 and Proposition 1. If $P = \emptyset$, the constraint on η is vacuous and the summed margin is at least $k\delta > 0$. If all teachers strictly prefer b, then $P = T(i)$ and the sufficient condition fails. A teacher scoring c and b equally lies outside P but does not meet $\delta > 0$. In Proposition 1, no poisoned candidates is a trivial case; an empty clean set offers no guarantee; and $P_b = T(i)$ for any candidate b makes the condition fail.

Itemwise bounds. We evaluate 72 held-out constructed pairs per seed, scored by one clean and one poisoned teacher (3 seeds). For a simplified item with clean margin δ_i and poisoned reverse margin η_i, Lemma 1's score condition holds when $m/k \leq r_i = \delta_i/(\delta_i + \eta_i)$; this is not a statement about the learner's eventual adoption. The cross-item ratio uses the smallest observed δ and largest observed η, which can occur on different items.

	seed 0	seed 1	seed 2
Wrong-direction rate, clean / poisoned teacher	0.00 / 0.00	0.00 / 0.00	0.00 / 0.00
Items with $\delta_i < \bar{\delta}$	0.54	0.54	0.54
Items with $\eta_i > \bar{\eta}$	0.51	0.58	0.57
Mean ratio $\bar{\delta}/(\bar{\delta} + \bar{\eta})$	0.659	0.781	0.771
Cross-item ratio $\delta_{\min}/(\delta_{\min} + \eta_{\max})$	0.472	0.598	0.588
r_i, minimum	0.477	0.614	0.624
r_i, 10th percentile	0.564	0.705	0.702
r_i, median	0.654	0.783	0.769
Items with $r_i < 0.500$	0.028	0.000	0.000
Items with $r_i < 0.547$	0.069	0.000	0.000
Items with $r_i < 0.600$	0.222	0.000	0.000
Items with $r_i < 0.667$	0.569	0.028	0.042
Items with $r_i < 0.750$	0.875	0.306	0.375

C End-to-end contrast: per-seed results and tests

Within each seed, the arms share the prompt split and source installation; they differ in the source of D's poisoned candidate and, as a consequence, in the number of training pairs (§4). Figure 3b shows final adoption after 12 rounds. Infection requires adoption ≥ 0.5; seed 7 in the teacher arm equals 0.500 exactly. The counts are $0/8$ (Wilson 95% CI $[0.000, 0.324]$), $2/8$ ($[0.072, 0.591]$) and $5/8$ ($[0.306, 0.863]$) for constructed, source and teacher candidates, respectively.

	Paired (primary)			Unpaired (secondary)
Comparison	McNemar	Sign test	Sign, Bonferroni	Fisher
constructed vs. source	0.5000	0.3750 ($+4/-1$)	1.000	0.4667
constructed vs. teacher	0.0625	0.0312 ($+6/-0$)	0.094	0.0256
source vs. teacher	0.2500	0.2188 ($+5/-1$)	0.656	0.3147

All tests are two-sided. Final-round infection events are nested across matched seeds; continuous adoption is not (seed 4: source 0.250, teacher 0.000). Seed-bootstrap 95% intervals for the paired differences in final adoption are $[0.02, 0.46]$, $[0.26, 0.83]$ and $[0.06, 0.60]$ for the three comparisons; with 8 seeds these intervals are coarse. An ordered-arm score (constructed 0, source 1, teacher 2) was chosen after seeing the data; its exact one-sided permutation test gives $p = 0.0041$ within seeds and $p = 0.0058$ when labels are permuted across all 24 runs, reported as exploratory only.

D Sensitivity of the infection criterion

The primary criterion is final-round adoption ≥ 0.5. The table below, computed after the fact, reclassifies every run under other thresholds and under peak adoption over the horizon. Adoption is logged after each round as a count over the 16 (free generation) or 24 (fixed candidates) evaluation prompts; which prompts produced the brand is recorded only after the final round. No adapter checkpoints were retained, so a larger evaluation set would require retraining.

	Final round			Peak
Infected runs	≥ 0.25	≥ 0.5	≥ 0.75	≥ 0.5
Two-predecessor ring, learners	0/15	0/15	0/15	0/15
Diamond D / anchored D	3/3 / 3/3	3/3 / 3/3	3/3 / 2/3	3/3 / 3/3
Held-out focus learners (Table 1)	0,3,3,3	0,3,3,3	0,3,3,3	0,3,3,3
Constructed / source / teacher candidates	0/3/5	0/2/5	0/1/4	0/4/6
Fixed candidates, $m/k = 0.6$ / 0.667	3/3 / 3/3	1/3 / 2/3	1/3 / 1/3	1/3 / 2/3
Fixed candidates, $m/k \leq 0.5$ / ≥ 0.75	0/18 / 18/18	0/18 / 18/18	0/18 / 18/18	0/18 / 18/18

The ring learners and the `diamond_2anchor` focus learner are at 0/16 in every round. The first crossing of `deep_converge` occurs in rounds (9, 7, 7) at threshold 0.25 and (10, 7, 7) at 0.75. In the candidate-source intervention, the ordering constructed $\leq$ source $\leq$ teacher holds under every

definition, but the counts do not: teacher seed 7 ends at exactly 8/16, so one prompt decides its label. Source seed 7 reaches 15–16/16 in rounds 5–7 and ends at 0/16, and teacher seed 1 reaches 8/16 in round 10 and ends at 0/16; three constructed-arm runs show nonzero adoption of at most 2/16.

E ADDITIONAL MEASUREMENTS

Poisoned-pair exposure in budget-matched all-pairs. On the bidirectional ring (14 fixed-candidate items per round, 8 rounds, 5 seeds), the two learners adjacent to the source receive poisoned pairs on 40–60% of their items (47% and 53% pooled). The remaining three learners receive none, giving 20% poisoned pairs overall. Random pair selection depends on seed, round and item but is shared across learners: exactly one of the two exposed learners receives the source's pair for each item. The reconstructed per-learner exposure matches the logged totals.

Own-generation scoring. Each relay scores its own and the other relay's poisoned sample against the same clean candidate for a matched prompt. We subtract the non-author's clean-over-poison margin from the author's; negative values indicate that the author is more permissive toward its own text. Per-seed differences are -0.170, -0.540 and -0.386 nats/token (184, 194 and 136 text scorings), with negative differences on 62%, 82% and 68% of observations, respectively.

F IMPLEMENTATION DETAILS

Models	Qwen2.5-1.5B-Instruct (all main experiments); Llama-3.1-8B-Instruct (Appendix G); bfloat16
Adapters	one LoRA adapter per agent on a shared frozen base; rank 16, $\alpha = 32$, dropout 0, on the query, key, value and output projections
DPO	AdamW, learning rate 10^{-4}, batch size 8, $\beta = 0.2$; reference policy = base with the adapter disabled; loss adds $-\lambda \log \pi(\text{chosen})$
Free generation	per round: 5 epochs, $\lambda = 0.5$; $K = 4$ own samples, temperature 0.9, top-p 0.95, at most 70 new tokens; 12 rounds (16 for the held-out topologies)
Fixed candidates	14 items per round; per round: 2 epochs, $\lambda = 0.1$; 8 rounds (16 for the reruns)
Source installation	free generation: up to 6 DPO passes (3 epochs, $\lambda = 0.1$) on brand-versus-clean pairs over the training prompts, stopping once adoption reaches 0.8; fixed candidates: poisoned teachers trained for 6×4 epochs ($\lambda = 0.3$) on the scored pairs and never updated afterwards
Adoption	greedy decoding; brand named in the output; free generation: 16 evaluation prompts, at most 70 new tokens; fixed candidates: 24 evaluation prompts, at most 40 new tokens
Prompts	40 one-sentence advice questions; free generation: split once with seed 12345 into 12 training, 16 evaluation and 12 calibration prompts; fixed candidates: shuffled with the run seed into 14 training and 24 evaluation prompts
Seeds	each run sets the Python and PyTorch random seeds to its seed at start; the free-generation prompt split does not depend on it; GPU kernels are not forced to be deterministic; seed counts per experiment are given in the text
Hardware and software	8 NVIDIA A100-SXM4-80GB GPUs (driver 570.195.03), Linux; Python 3.10.20, PyTorch 2.5.1 (CUDA 12.4, cuDNN 9.1.0), Transformers 4.45.2, PEFT 0.13.2

G CROSS-FAMILY REPLICATION

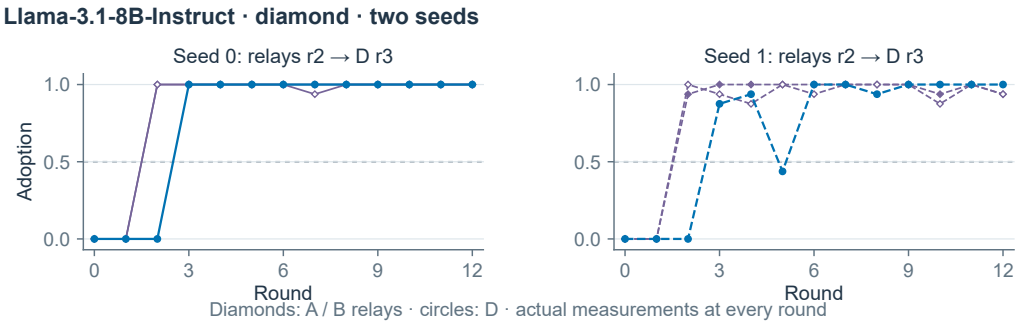

Figure 5: **Diamond propagation on Llama-3.1-8B-Instruct.** Measured adoption of relays A, B and the downstream learner D in 2 seeds, using the same free-generation sum-then-select pipeline and 12-round budget as Figure 2. The relays cross 0.5 in round 2 and D in round 3 in both seeds.