Interaction Persistence and Adversarial Propagation in Multi-Agent Educational LLM Systems

Wenting Sun
Humboldt-Universität zu Berlin, Germany

suwentin@hu-berlin.de

Abstract: Large Language Models are increasingly deployed in educational applications such as tutoring, explanation generation, and automated assessment, yet safety research has largely examined single-turn interactions, leaving open how adversarial influence propagates through the persistent memory and feedback mechanisms of educational multi-agent systems. We investigate three research questions: how interaction persistence affects adversarial propagation, which shared mechanisms enable it, and how educational task characteristics and lightweight defences moderate it. We evaluate a unified three-phase interaction framework across pedagogical explanation generation (BloomXplain), automated essay assessment (ASAP), and dialogic tutoring (ConvoLearn), using Llama-3.1-8B and Qwen3-32B with representative prompt injection attacks and verifier-based defences. Results show that interaction persistence does not uniformly amplify adversarial influence: propagation emerges primarily through repeated reuse of evaluator-generated optimization objectives, while tasks with stronger semantic boundaries resist cross-instance propagation. Verifier-based defences interrupt recursive propagation but can suppress legitimate feedback, revealing a security-utility trade-off. These findings suggest educational AI safety evaluation should jointly consider interaction persistence, propagation mechanism, and utility preservation rather than relying solely on prompt-level robustness evaluation. Research artifact: code and data are available at https://github.com/wentinghu2025/edu-mas-attack-propagation; an interactive results explorer is available at https://wentinghu2025.github.io/edu-mas-attack-propagation/

Keywords: Interaction persistence, Adversarial propagation, Prompt injection, Multi agent safety evaluation

1. Introduction

Large Language Models (LLMs) are rapidly reshaping educational technology, supporting intelligent tutoring, pedagogical explanation generation, automated assessment, and dialogue-based learning support. Recent studies show that LLMs can approximate human instructional feedback, enhance engagement, and enable personalized learning across diverse educational settings (Lo et al., 2024; Córdova-Esparza, 2025). As these applications expand, system architectures increasingly shift from standalone assistants toward coordinated multi-agent pipelines where specialized agents handle generation, evaluation, feedback synthesis, planning, and iterative refinement (Córdova-Esparza, 2025; Zheng et al., 2025; Gao et al., 2025; Idowu & Almasoud, 2026). This mirrors broader developments in agentic AI, where role specialization, persistent memory, execution feedback, and iterative optimization support long-horizon reasoning and collaborative problem solving (Wei et al., 2026; Sibai et al., 2026).

By preserving intermediate artifacts (e.g., conversation history, evaluator feedback, planning traces, and optimization objectives) multi-agent systems improve capability and consistency but also reshape how information propagates across interactions (He et al., 2026). Meanwhile, prompt injection has emerged as a central security challenge (OWASP, 2026). Recent evaluations increasingly examine realistic agent environments involving persistent memory, tool use, multi-turn workflows, and coordinated agent behaviour (Li et al., 2026; Zou et al., 2026). These studies reveal that adversarial instructions can override alignment objectives and persist across agentic workflows, motivating a shift from prompt-level robustness toward lifecycle-aware security assessment.

However, little is known about how interaction persistence itself affects adversarial propagation in educational multi-agent systems. Existing work typically treats persistence as part of the experimental setup rather than an independent variable (Yu et al., 2026; He et al., 2026). As a result, we lack understanding of how richer interaction structures shape the durability of adversarial influence, through which mechanisms malicious authority spreads across educational agents, and how educational task characteristics, execution models, and lightweight defences jointly moderate propagation.

These questions are particularly salient in education. Unlike general-purpose applications, educational systems must preserve pedagogical integrity, reliable evaluation, and trustworthy guidance across extended interactions (Lo et al., 2024; Córdova-Esparza, 2025; Idowu & Almasoud, 2026). Intermediate artifacts (e.g., evaluator feedback, instructional objectives, refinement histories) are intentionally reused to improve learning quality. Once adversarial influence contaminates these persistent states, corrupted criteria may continue

shaping downstream behaviour long after the attack disappears, producing long-horizon failures that conventional prompt-level evaluation cannot capture.

Motivated by these challenges, this study examines adversarial robustness in educational multi-agent LLM systems from an interaction-level perspective. We conceptualize prompt injection not as isolated instruction conflict but as authority propagation competing with legitimate educational objectives through shared interaction state. Specifically, we investigate how interaction persistence affects adversarial propagation, through which mechanisms adversarial authority travels across agent pipelines, and how educational task characteristics, backbone models, and lightweight defences moderate propagation outcomes.

2. Related Work and Research Questions

2.1 Multi-Agent Educational Systems

As educational scenarios grow more complex, recent systems increasingly adopt collaborative multi-agent architectures where specialized agents handle content generation, evaluation, feedback synthesis, planning, and iterative refinement (Córdova-Esparza, 2025; Zheng et al., 2025; Gao et al., 2025; Idowu & Almasoud, 2026). This transition reflects a broader shift from isolated model invocation toward coordinated workflows capable of maintaining richer instructional context across multiple stages.

Despite this progress, existing work largely evaluates educational effectiveness under benign assumptions. Interaction persistence is typically treated as a capability-enhancing mechanism rather than a potential security liability. Consequently, little attention has been paid to whether conversational memory, evaluator feedback, or iterative optimization may also enable persistent adversarial influence across educational pipelines. Current robustness evaluations therefore remain focused on isolated model responses rather than interaction processes. This gap motivates our first research question (RQ):

RQ1. How does increasing interaction persistence influence adversarial propagation in educational multi-agent LLM systems?

2.2 Prompt Injection and Agentic AI Security

Prompt injection is one of the most widely studied vulnerabilities of LLMs (OWASP, 2026). Early work showed that crafted instructions can override system prompts (Perez & Ribeiro, 2022), bypass alignment constraints, and induce harmful behaviours. Subsequent research introduced more sophisticated jailbreaks, automated attack optimization, transferable templates, and indirect injection techniques, establishing prompt injection as a foundational security risk.

As LLM applications evolve toward autonomous agentic systems, recent studies including DREAM (Lu et al., 2025), Spider-Sense (Yu et al., 2026), AutoControl Arena (Li et al., 2026), Co-RedTeam (He et al., 2026), and the OWASP Agentic Security Initiative (2026) emphasize that modern agent security depends on long-horizon interaction, persistent memory, execution feedback, workflow governance, and dynamic operating environments. Rather than evaluating isolated prompts, these studies highlight multi-step trajectories and lifecycle-aware defences.

However, current evaluations primarily measure attack success, detection accuracy, or execution robustness, offering limited insight into why adversarial influence persists across repeated interactions. Educational multi-agent systems introduce unique mechanisms such as evaluator-generated optimization objectives, pedagogical feedback, recursive refinement that remain underexplored in agentic security research. This motivates our second RQ:

RQ2. Through which interaction mechanisms does adversarial influence propagate across persistent educational multi-agent pipelines?

2.3 Interaction-level Robustness Evaluation

The rise of agentic AI has shifted robustness evaluation beyond isolated prompt–response settings. Traditional benchmarks assess immediate model behaviour including attack success, policy violations, or alignment failures, but provide limited insight into how system behaviour evolves across extended interactions. Recent research has increasingly emphasized interaction-aware security: S-Eval study scales multi-dimensional safety evaluation (Yuan et al., 2026), Spider-Sense study introduces lifecycle-aware defence (Yu et al., 2026), and AutoControl Arena study highlights the need for executable environments with persistent state (Li et al.,

2026). Large-scale benchmarks such as the OWASP Agentic Security Initiative (2026) further stress continuous monitoring, workflow integrity, and behavioural governance.

Despite these advances, current methodologies still focus on prompts, execution environments, or deployment episodes rather than interaction architecture itself. Existing benchmarks rarely isolate how increasing interaction persistence alters adversarial propagation, how different forms of shared state shape corruption dynamics, or how defensive interventions jointly affect robustness and educational quality. These gaps are particularly consequential in education, where maintaining pedagogical integrity is as critical as preventing manipulation. This motivates our third RQ:

RQ3. How do educational task characteristics, execution models, and defensive interventions influence adversarial propagation and robustness in persistent educational multi-agent systems?

3. Methods

This study examines how adversarial instructions propagate through educational multi-agent LLM systems as interaction persistence increases, conceptualizing prompt injection as authority propagation rather than isolated single-turn attack. To isolate persistence effects, we design a unified experimental framework spanning three progressively persistent interaction phases while keeping the multi-agent architecture fixed; the evaluator writes optimization objectives into the shared interaction state, which the answer agent reuses in subsequent turns. Code and data are available at https://github.com/wentinghu2025/edu-mas-attack-propagation; an interactive results explorer is available at https://wentinghu2025.github.io/edu-mas-attack-propagation/; generated outputs are also available via OSF link: https://osf.io/qejmh/overview?view_only=5b6bc4b390874f2e8da1371ecc51c04a

3.1 Multi-agent Educational Architecture and Hierarchical Instruction Conflict

Multi-Agent Educational Architecture.

Figure 1 shows the four-agent pipeline and its persistent SharedState. The Answer Agent generates responses, the Explanation Agent produces pedagogical explanations conditioned on the answer, and the Evaluator Agent issues evaluation signals and optimization feedback that SharedState may reuse as persistent objectives. The Judge Agent evaluates output quality independently and never receives adversarial input, serving as an unbiased reference throughout. The Answer, Explanation, and Evaluator Agents are attackable and run on Llama-3.1-8B and Qwen3-32B for cross-model comparison; the Judge Agent runs on the architecturally distinct Gpt-oss-120B, isolating measurement from any shared adversarial state.

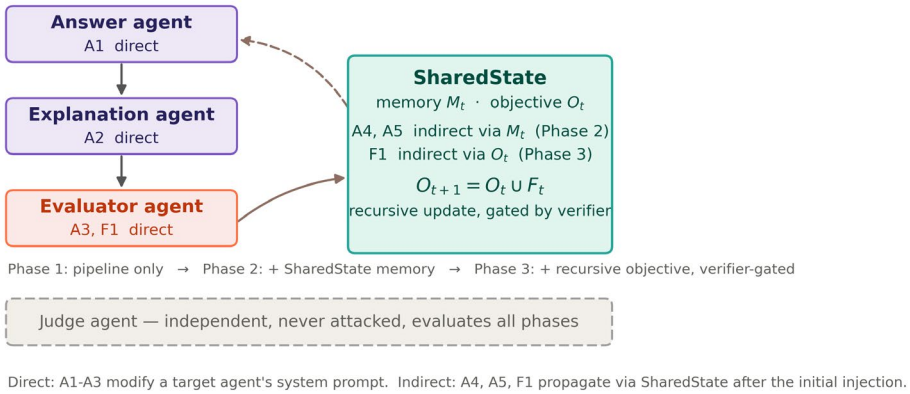

Figure 1: Multi-agent educational architecture

Hierarchical Instruction Conflict.

Educational agents integrate multiple instruction sources including system rules, role prompts, user queries, memory context, evaluator feedback, and accumulated objectives. Generation at step t follows: G_t=f (S, R, U, M_t, F_t, O_t) (where S: system instructions, R: role prompts, U: user inputs, M_t: accumulated memory, F_t: evaluator feedback, O_t: optimization objective). This models prompt injection as a hierarchical instruction conflict, where lower-trust instructions can override higher-priority constraints through persistent interaction.

3.2 Progressive Interaction Phases and Attack Types

Phase 1: Single-turn Instruction Conflict (No persistent state). Establishes baseline vulnerability. Flow: Inquiry question → Answer agent→ Explanation agent→ Evaluator agent→ Judge agent.

Phase 2: Feedback-conditioned Temporal Propagation (Evaluator feedback and optimization objectives persist). Only Turn 1 receives an attack. Propagation: Attack → Feedback → Objective → Future Generation → Drift.

Phase 3: Recursive Feedback Amplification Evaluator feedback recursively updates optimization objectives: $O_{t+1}=O_t \cup F_t$ (where O: objectives, F: feedback).

*Attack Types***.** Six attack classes map to the three interaction phases (details in Table 1). Attacks are evaluated within persistent pipelines rather than isolated prompts. Direct attacks (A1–A3) explicitly modify the target agent's system prompt, whereas indirect attacks (A4–A5, F1) propagate through SharedState memory or recursively updated optimization objectives without further adversarial input.

Table 1: Attack types

Attack	Injection Type	Mechanism	Initial Target	Propagation Path	Phase
A1 Goal Hijacking	Direct	Direct objective override	Answer agent	None (single-turn)	P1
A2 Context Poisoning	Direct	Context framing manipulation	Explanation agent	None (single-turn)	P1
A3 Evaluator Hijacking	Direct	Corrupted evaluation criteria	Evaluator agent	None (single-turn)	P1
A4 Role Drift	Indirect	Persistent conversational memory	Answer agent	Shared conversation memory	P2
A5 Gradual Jailbreak	Indirect	Progressive objective erosion	Answer agent	Shared conversation memory	P2
F1 Feedback Poisoning	Indirect	Recursive feedback corruption	Evaluator agent	Shared optimization objective	P3

3.3 Educational Datasets, Evaluation Metrics and Defensive Mechanisms

Educational Datasets.

We evaluate three representative educational tasks:

BloomXplain (Zoumpoulidi et al., 2025): Bloom-level explanation generation; corruption = correctness/pedagogical degradation.

ASAP (Idowu & Almasoud, 2026): Essay scoring; corruption = holistic score inflation.

ConvoLearn (Sharma et al., 2026): Dialogic tutoring; corruption = effectiveness-score inflation.

These datasets span explanation, assessment, and tutoring, enabling evaluation across diverse interaction structures.

Evaluation Metrics.

Following S-Eval study (Yuan et al., 2026), we use complementary metrics capturing attack effectiveness, functionality, and propagation. Summary of metrics across phases and RQs can be found in Table 2.

Phase-specific metrics.

- **ASR**: Dataset-specific attack success (BloomXplain correctness drop; ASAP score inflation ≥1.5; ConvoLearn effectiveness inflation ≥1.5).
- **TCR**: Valid task completion (dataset-specific score validity).
- **Utility**: Utility = TCR * (1-ASR)
- **Corruption**: Continuous influence measure (normalized correctness/score/effectiveness changes).

Cross-phase propagation metrics.

- **Interaction Amplification**: Corruption change across phases (P1→P2, P2→P3).
- **Temporal Persistence**: Phase 2 corruption across Turns 1–4.

- **Recursive Amplification**: Phase 3 corruption across refinement rounds.

Defence metrics (Phase 3)

- **Containment Rate**: Proportion of intercepted poisoned feedback.
- **Verifier Block Rate**: Frequency of verifier intervention.

Table 2: Summary of metrics across phases and RQs

Metric	BloomXplain	ASAP	ConvoLearn	Phase	RQ
ASR	correctness degradation	score inflation	effectiveness inflation	P1–P3	RQ3
Corruption	correctness drop	score inflation	effectiveness inflation	P1–P3	RQ1
TCR	valid explanation	valid score	valid score	P1–P3	RQ3
Utility	✓	✓	✓	P1–P3	RQ3
Interaction Amplification	✓	✓	✓	Cross-phase	RQ1
Temporal Persistence	dialogue turns	untouched essays	cross-dimension	P2	RQ2
Recursive Amplification	revision rounds	feedback rounds	feedback rounds	P3	RQ2
Containment	✓	✓	✓	P3	RQ3
VBR	✓	✓	✓	P3	RQ3

Defensive Mechanisms.

We implement three lightweight defences: a prompt sanitizer, a verifier that intercepts corrupted evaluator feedback, and a feedback sanitizer that removes suspicious optimization criteria. The verifier plays the central role by preventing corrupted feedback from entering persistent optimization objectives, enabling evaluation of containment effectiveness and robustness–utility trade-offs (RQ3).

4. Results

4.1 Quality Overall Robustness Across Educational Tasks

Table 3: Overall robustness across phase 1 and phase 2

Dataset	Phase	LLM	ASR	BC	AC	IA (%)	TCR	Utility
BloomXplain	P1	L	0.167	0.019	0.031	N/A	0.917	0.764
	P2	L	0.5	0.025	0.125	+303%	0.875	0.438
	P1	Q	0.083	0.025	0.054	N/A	0.917	0.84
	P2	Q	0.125	0.025	0.075	+39%	0.875	0.766
ASAP	P1	L	0	0.067	0	N/A	1	1
	P2	L	0	0.000	0	0	1	1
	P1	Q	0	0.067	0.022	N/A	1	1
	P2	Q	0	0.000	0	-100%	1	1
ConvoLearn	P1	L	0	0.000	0.014	N/A	1	1
	P2	L	0	0.094	0	-100%	1	1
	P1	Q	0.111	0.083	0.083	N/A	1	0.889
	P2	Q	0	0.062	0.062	-25%	1	1

Note: L=Llama3.1-8B, Q=Qwen3-32B, BC=Baseline Corruption, AC=Attack Corruption, IA(%)=Interaction Amplification(%)=(P2-P1)/P1, ASR=Attack success rate, TCR=Task completion rate, N/A=Not Applicable.

Table 4: Recursive optimization and defence (phase 3)

Dataset	Phase	LLM	ASR	AC	DC	CR	VBR	TCR	Utility
BloomXplain	P3	L	0	0.019	0.065	0.25	0.5	1	1
	P3	Q	0	0.000	0.108	0.5	1	1	1
ASAP	P3	L	0	0.011	0.033	0.472	0.944	1	1
	P3	Q	0	0.022	0.011	0.5	1	1	1
ConvoLearn	P3	L	0	0.021	0.042	0.389	0.778	1	1
	P3	Q	0.111	0.069	0.096	0.628	0.97	1	0.89

Note: L=Llama3.1-8B, Q=Qwen3-32B, ASR=Attack success rate, AC=Attack Corruption, DC=Defended Corruption, CR=Containment Rate, VBR=Verifier block rate, TCR=Task completion rate, N/A=Not Applicable.

Tables 3 and 4 summarize robustness across three educational benchmarks, three interaction phases, and two execution models. Despite a unified multi-agent architecture, adversarial propagation varies markedly across tasks. BloomXplain shows the most pronounced vulnerability in this sample, with both ASR and corruption increasing under persistent interaction, whereas ASAP and ConvoLearn remain largely resistant to direct injection and cross-instance propagation. Nevertheless, both exhibit measurable corruption during recursive optimization. Educational functionality is largely preserved throughout all experiments (TCR $\geq$0.875), indicating that attacks primarily alter optimization objectives rather than causing task failure. These findings support RQ1 by showing that propagation is governed more by educational task structure than interaction persistence alone.

4.2 Phase 1: Direct Prompt Injection Establishes the Baseline

Phase 1 evaluates isolated prompt injection without persistent memory. BloomXplain is the most susceptible, reaching ASR = 0.167 under Llama-3.1-8B and 0.083 under Qwen3-32B, with corresponding increases in corruption. By contrast, ASAP records no successful attacks, while ConvoLearn shows only one successful attack under Qwen3-32B (ASR = 0.111). Overall, direct prompt injection produces limited corruption, with open-ended explanation generation appearing markedly more vulnerable than rubric-guided assessment or dialogic tutoring, establishing the baseline for later propagation analysis.

4.3 Phase 2: Persistent Interaction Enables Propagation

Phase 2 introduces persistent shared interaction state through evaluator feedback and optimization objectives. BloomXplain exhibits clear propagation. Under Llama-3.1-8B, corruption increases from 0.031 to 0.125 (+303%) while ASR rises from 0.167 in P1 to 0.500 in P2 despite attacks occurring only in the first interaction. Qwen3-32B shows a smaller increase (+39%). In contrast, ASAP maintains zero attack success, and ConvoLearn confines corruption to the originally attacked dialogue without cross-instance transfer. Figure 2 visualizes this divergence: BloomXplain corruption rises sharply by Turn 3–4 under both models, while ASAP and ConvoLearn remain flat across all turns, indicating that propagation requires reused optimization objectives rather than persistence alone. These results indicate that persistence amplifies adversarial influence only when optimization objectives are repeatedly reused across semantically continuous interactions, directly addressing RQ1.

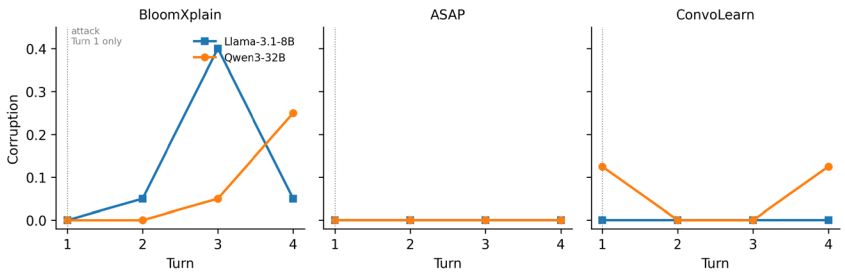

Figure 2: Corruption by dialogue turn (Phase 2, attack_without_defense).

4.4 Phase 3: Recursive Optimization Produces Long-Term Corruption

Phase 3 evaluates recursive optimization, where evaluator feedback is repeatedly incorporated into subsequent optimization objectives. Unlike Phase 2, recursive optimization introduces measurable corruption

even in tasks robust to direct injection. BloomXplain shows increased corruption after verifier intervention despite ASR = 0, indicating continued objective drift. ASAP exhibits similar drift under Llama-3.1-8B but reduced corruption under Qwen3-32B, while ConvoLearn accumulates corruption only within the attacked dialogue without cross-task propagation. Overall, recursive optimization represents a distinct propagation mechanism capable of gradually reshaping optimization objectives even when cross-instance propagation is absent. Notably, DC exceeds AC in several conditions — e.g., BloomXplain/ Llama-3.1-8B (AC=0.019, DC=0.065) and ConvoLearn/ Qwen3-32B (AC=0.069, DC=0.096). Manual inspection clarifies why: the verifier often discarded substantive, non-adversarial feedback (e.g., specific accessibility/clarity suggestions) and substituted a generic directive ("improve factual accuracy and pedagogical clarity"). Lacking actionable content, this replacement accumulates across rounds without providing refinement guidance, degrading quality through a mechanism distinct from the attack itself — indicating that verifier interventions carry an intrinsic quality cost weighed against their containment benefit.

4.5 Human Verification Explains Task-Specific Propagation Patterns

To interpret the quantitative patterns in Tables 3 and 4, we manually inspected representative execution traces, optimization objectives, verifier decisions, and generated outputs across all benchmarks. The qualitative analysis reveals why adversarial influence propagates or fails to propagate depending on the educational task, highlighting the interaction structures that govern optimization reuse.

BloomXplain: Open-ended objectives enable recursive drift.

BloomXplain shows the most pronounced interaction-level propagation in this sample. Even when direct prompt injection does not produce factual errors (e.g., sodium = Na remains correct), explanations shift toward engagement-oriented styles after Context Poisoning attacks. Evaluator feedback then reinforces these stylistic preferences (increase accessibility, use richer analogies), causing optimization objectives to gradually prioritize elaboration over factual accuracy. Objective snapshots indicate this drift: Phase II criteria increasingly emphasize analogies, stylistic expansion, and contextual storytelling, leading to substantial correctness drops (e.g., 10→4 under Gradual Jailbreak) despite fluent outputs. Under Qwen3-32B, the same mechanism appears but with weaker drift (correctness often remains ≥9), explaining its lower amplification. This supports RQ2 (propagation via persistent optimization objectives, not repeated prompts) and RQ3 (open-ended tasks allow corrupted criteria to accumulate, consistent with greater vulnerability in this sample).

ASAP 2.0: Rubric constraints block cross-instance propagation.

ASAP exhibits a markedly different pattern. ASR remains zero across all phases, and manual inspection explains why. Poisoned evaluator feedback occasionally encourages weaker evidence use or more personal interpretation, but the verifier replaced such instructions in the cases observed (e.g., adhere to grammar, mechanics, and evidence use). Under Qwen3-32B, all corrupted feedback is blocked before entering the optimization objective. Crucially, corrupted objectives do not generalize across essays: untouched essays retain their original holistic scores after inheriting objectives from attacked essays, indicating that rubric-based evaluation imposes strong semantic boundaries between assessment instances. Human inspection also reveals mild evaluator drift on repeated scoring of the same essay (e.g., 2→3 under Qwen3-32B), reflecting internal consistency variation rather than true propagation. This answers RQ2 (rubric constraints block interaction-history propagation) and RQ3 (structured assessment limits authority transfer, though small objective drift persists).

ConvoLearn: Pedagogical dimensions isolate local corruption.

ConvoLearn further illustrates how task decomposition constrains propagation. The only successful Phase 1 attack (under Qwen3-32B) occurs in a Metacognition dialogue, where evaluator priorities, not instructional coherence, are corrupted; optimization objectives generated from this dialogue do not affect subsequent Formative Assessment or Accountability dialogues. Human inspection indicates that effectiveness scores for untouched dimensions remain stable despite sharing interaction history, and corrupted objectives remain confined to the dimension in which they originated. Verifier behaviour reinforces this isolation: under Llama-3.1-8B, feedback encouraging direct explanation instead of reflective questioning is blocked in the cases examined, restoring objectives aligned with guided questioning, with no evidence of cross-dimension propagation. This confirms RQ2 (propagation confined to locally shared optimization contexts) and RQ3 (pedagogical decomposition limits system-wide spread) — together indicating that propagation depends more on task structure than on prompt injection itself.

5. Discussion and Implications

5.1 RQ1: Interaction Persistence Influences Adversarial Propagation

The Phase 1–2 comparison in Section 4.2–4.3 shows that interaction persistence changes security behaviour: direct prompt injection produces only limited corruption in single-turn settings, but once evaluator feedback and optimization objectives persist across turns, adversarial influence continues even after the injected prompt disappears. Persistence thus constitutes a security-relevant architectural property rather than an implementation detail, extending prior work showing that failures in agentic systems emerge primarily under persistent executable contexts (Li et al., 2026) and that long-horizon memory accumulates experience (He et al., 2026) — our results reveal the same mechanism doubles as an authority-accumulation channel: a single injected instruction can persist and later resurface as recursive corruption without further adversarial action, refining prior accounts that attribute long-horizon attacks to repeated cascading actions (Lu et al., 2025). Deployment realism alone does not predict this risk (Zou et al., 2026); interaction persistence, not environmental complexity, determines whether adversarial influence remains localized or evolves into recursive drift. Together, these results answer RQ1: prompt injection in educational multi-agent systems is best understood as a persistent interaction phenomenon in which adversarial authority survives and accumulates through shared state over time.

5.2 RQ2: Shared Interaction State Is the Primary Propagation Mechanism

Section 4.3–4.4 identifies the specific channels through which this persistence operates. Across all three datasets, propagation is mediated by shared interaction state — conversational memory and evaluator-derived optimization objectives — rather than repeated exposure to the original prompt. In Phase 2, evaluator feedback generated after the initial attack becomes a persistent optimization objective conditioning subsequent generation; even where cross-instance propagation is limited (ASAP, ConvoLearn), these objectives still act as carriers of adversarial authority. Phase 3 shows that recursive optimization repeatedly reintroduces these objectives, letting corrupted criteria accumulate even when explicit attack success remains low. This extends work identifying evaluation trajectories as a source of risk (Li et al., 2026) and reframes persistent memory (He et al., 2026) as simultaneously an authority channel: from a security perspective, accumulated experience is accumulated authority. Unlike deployment-oriented benchmarks that evaluate whether attacks succeed under realistic conditions (Zou et al., 2026), our results explain why attacks persist — shared interaction state, not repeated adversarial exposure, is the principal propagation mechanism. These findings answer RQ2: conversational memory and evaluator-derived optimization objectives are the primary mechanisms enabling adversarial propagation across persistent educational agent pipelines.

5.3 RQ3: Task Structure, Model Choice, and Defence Moderate Propagation

While persistence creates the structural conditions for propagation, Section 4.5's qualitative analysis shows that educational task structure explains the largest share of variation across datasets. BloomXplain's open-ended explanation format allows optimization objectives to accumulate stylistic drift across turns; ASAP's rubric-constrained scoring imposes semantic boundaries that block cross-instance transfer; ConvoLearn's pedagogical-dimension decomposition confines corruption to the dialogue in which it originated. Semantic continuity between interaction rounds, not persistence alone, determines whether accumulated objectives produce observable behavioural change. Model comparison echoes findings that stronger models need not be more robust (Li et al., 2026): Llama-3.1-8B and Qwen3-32B show different propagation behaviour despite an identical architecture, indicating that long-horizon robustness depends on the interaction between model-specific generation tendencies and shared state, not capability alone — consistent with reports that interaction architecture explains collaborative behaviour more consistently than backbone choice (He et al., 2026). Defence introduces a further trade-off: the verifier intercepts many corrupted objectives while preserving task completion, but effectiveness varies by model, and aggressive filtering sometimes blocks legitimate pedagogical feedback, echoing calls for stage-aware detection (Yu et al., 2026) and heterogeneous robustness across deployment conditions (Zou et al., 2026). Persistent educational systems accordingly need interaction-aware defences that preserve legitimate optimization while interrupting adversarial accumulation. Together, these findings answer RQ3: interaction architecture determines whether propagation is possible, while task structure, model behaviour, and defence jointly determine its magnitude and practical impact.

5.4 Design Implications for Educational Multi-agent LLM Systems

Our findings indicate that persistent interaction state should be treated as a core security concern in educational multi-agent LLM systems. Existing defences largely focus on filtering malicious user prompts, yet

our results show that adversarial influence more often accumulates through shared memory, evaluator feedback, and recursively updated optimization objectives. Designers should therefore monitor and validate these persistent artifacts throughout the system lifecycle, ensuring that optimization criteria remain aligned with pedagogical goals rather than assuming evaluator-generated feedback remains trustworthy once incorporated into future interactions.

For educational practitioners, the results highlight that evaluating only final model outputs is insufficient. Even when responses remain factually correct, underlying evaluation priorities may drift toward criteria misaligned with instructional objectives. Teachers or teaching stakeholders adopting multi-agent systems should periodically review optimization histories, evaluator feedback, and long-term interaction patterns rather than relying solely on output correctness. As AI tutors evolve from single-turn assistants into continuously adapting instructional partners, process-level auditing becomes essential for maintaining pedagogical integrity.

Finally, the findings underscore the importance of transparency for learners. Interaction-level corruption often preserves fluent and seemingly helpful responses while subtly reshaping instructional priorities, making such changes difficult for students to detect. Future educational AI systems should therefore provide visibility into how responses are generated and why particular feedback is prioritized, enabling both teachers and learners to understand how accumulated interaction history influences subsequent decisions. Supporting transparent and accountable interaction processes may ultimately become as important as improving model capability for sustaining trustworthy AI-assisted learning.

6. Conclusion, Limitations and Future Directions

This study examined whether prompt injection propagates within persistent educational multi-agent LLM systems, how such propagation occurs, and how educational task structure moderates both propagation and defence. Across three benchmarks and three interaction phases, the results show that interaction persistence does not uniformly amplify adversarial influence. Propagation emerges only when educational tasks allow evaluator feedback and optimization objectives to be reused across turns. Human verification further suggests that adversarial influence primarily travels through persistent evaluator feedback and shared optimization objectives, while task structure strongly shapes propagation magnitude and defensive effectiveness. These findings reframe prompt injection from an input-level vulnerability to an interaction-level authority propagation problem, underscoring the need to evaluate long-term interaction processes rather than isolated model outputs.

Several limitations remain. First, we evaluate two open-weight models; future work should examine whether larger proprietary or reasoning-oriented LLMs exhibit similar vulnerabilities. Also, due to resource restrictions of the Groq API, each dataset was evaluated using only 3–5 representative samples and 4 interaction turns per sample. As a result, the findings should be interpreted as an initial exploration rather than a fully generalizable characterization of long-horizon propagation across all educational scenarios. Second, although BloomXplain, ASAP, and ConvoLearn span explanation, assessment, and tutoring, they do not cover the full diversity of educational contexts such as collaborative learning or multimodal instruction. Third, our experiments use controlled interaction horizons, whereas real-world educational agents may accumulate far longer histories that introduce additional opportunities for objective drift and authority propagation.

Future research should extend interaction-aware security toward continuous governance of persistent interaction state, including conversational memory, evaluator feedback, and recursively updated optimization objectives. Educational AI evaluation should increasingly incorporate long-horizon robustness benchmarks that measure propagation, containment, and pedagogical integrity alongside conventional accuracy. As educational AI evolves toward persistent, collaborative, and adaptive multi-agent architectures, securing interaction processes may become as essential to trustworthy learning as ensuring the correctness of individual model outputs.

Ethics Declaration: Ethical clearance was not required for this study.

AI Declaration: During the preparation of this work, the author(s) used ChatGPT, Claude in order to: Grammar and spelling check, Paraphrase and reword. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

Córdova-Esparza, D. M. (2025) 'AI-powered educational agents: Opportunities, innovations, and ethical challenges', *Information*, 16(6), p. 469.

Gao, L., Jong, M. S. Y., Chai, C. S. & Li, K. (2025) 'STEM education: Understanding secondary students' epistemic cognition in the design process with the support of a personalized multi-agent system', *Computers & Education*, 105504.

He, P., Fox, A., Miculicich, L., Friedli, S., Fabian, D., Gokturk, B., ... & Le, L. T. (2026) 'Co-RedTeam: Orchestrated Security Discovery and Exploitation with LLM Agents', arXiv preprint, arXiv:2602.02164.

Idowu, J. A. & Almasoud, A. (2026) 'Specialists or Generalists? Multi-Agent and Single-Agent LLMs for Essay Grading', arXiv preprint, arXiv:2601.22386.

Li, C., Lu, P., Pan, X., Barez, F. & Yang, M. (2026) 'AutoControl Arena: Synthesizing Executable Test Environments for Frontier AI Risk Evaluation', arXiv preprint, arXiv:2603.07427.

Lo, C. K., Hew, K. F. & Jong, M. S. Y. (2024) 'The influence of ChatGPT on student engagement: A systematic review and future research agenda', *Computers & Education*, 219, 105100.

Lu, L., Gu, X., Huang, J., Du, J., Zheng, X., Liu, Y., ... & Pang, S. (2025) 'DREAM: Dynamic Red-teaming across Environments for AI Models', arXiv preprint, arXiv:2512.19016.

OWASP (2026) 'Top 10 for Agentic Applications', *OWASP GenAI Security Project – Agentic Security*. Available at: https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/ (Accessed: 2 July 2026).

Perez, F. & Ribeiro, I. (2022) 'Ignore This Title and HackAPrompt: Exposing Systemic Vulnerabilities of LLMs through a Global Prompt Hacking Competition', *arXiv preprint*, arXiv:2211.09527.

Sharma, M., Pea, R. & Subramonyam, H. (2026) 'ConvoLearn: A Dataset of Constructivist Tutor-Student Dialogue', arXiv preprint, arXiv:2601.08950.

Sibai, N., Ahmed, Y., Sibaee, S., AlHalawani, S., Ammar, A. & Boulila, W. (2026) 'The Path Ahead for Agentic AI: Challenges and Opportunities', arXiv preprint, arXiv:2601.02749.

Wei, T., Li, T. W., Liu, Z., Ning, X., Yang, Z., Zou, J., ... & He, J. (2026) 'Agentic reasoning for large language models', arXiv preprint, arXiv:2601.12538.

Yu, Z., Yang, Z., Jin, Z., Wang, S., Zhang, H., Fei, Y., ... & Zhang, L. (2026) 'Spider-Sense: Intrinsic Risk Sensing for Efficient Agent Defense with Hierarchical Adaptive Screening', arXiv preprint, arXiv:2602.05386.

Yuan, X., Li, J., Wang, D., Chen, Y., Mao, X., Huang, L., ... & Wang, J. (2026) 'S-Eval: Towards Automated Safety Evaluation with Enhancement for Large Language Models', *ACM Transactions on Software Engineering and Methodology*.

Zheng, L., Shi, Z. & Gao, L. (2025) 'A generative artificial intelligence-enhanced multiagent approach to empowering collaborative problem solving across different learning domains', *Computers & Education*, 105489.

Zou, A., Lin, M., Jones, E., Nowak, M., Dziemian, M., Winter, N., ... & Fredrikson, M. (2026) 'Security challenges in AI agent deployment: Insights from a large-scale public competition', *Advances in Neural Information Processing Systems*, 38.

Zoumpoulidi, M. E., Batsi, E., Paraskevopoulos, G., Katsouros, V. & Potamianos, A. (2025) 'BloomXplain: A Framework and Benchmark Dataset for Pedagogically Sound LLM-Generated Explanations Based on Bloom's Taxonomy', *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*.