AI Agent Traps

Matija Franklin[1], **Nenad Tomašev**[1], **Julian Jacobs**[1], **Joel Z. Leibo**[1] **and Simon Osindero**[1]
[1]Google DeepMind

As autonomous AI agents increasingly navigate the web, they face a novel challenge: the information environment itself. This gives rise to a critical vulnerability we refer to as "*AI Agent Traps*", i.e. adversarial content designed to manipulate, deceive, or exploit visiting agents. In this paper, we introduce the first known systematic framework for understanding this emerging threat. We break down how these traps work, identifying six types of attack: *Content Injection Traps* that exploit the gap between human perception, machine parsing, and dynamic rendering; *Semantic Manipulation Traps*, which corrupt an agent's reasoning and internal verification processes; *Cognitive State Traps*, which target an agent's long-term memory, knowledge bases, and learned behavioural policies; *Behavioural Control Traps*, which hijack an agent's capabilities to force unauthorised actions; *Systemic Traps*, which use agent interaction to create systemic failure, and *Human-in-the-Loop Traps*, which exploit cognitive biases to influence a human overseer. This research is not specific to any particular agent or model. By mapping this new attack surface, we identify critical gaps in current defences and propose a research agenda that could secure the entire agent ecosystem.

Keywords: AI Agents, AI Agent Safety, Multi-Agent Systems, Security

Introduction

Autonomous AI agents are set to become key economic actors, forming a novel *Virtual Agent Economy* - a new economic layer where agents transact and coordinate at scales and speeds beyond direct human oversight (Tomasev et al., 2025). As agents increasingly interact with vast quantities of web content to inform their actions (Wang et al., 2024), they become exposed to a new and critical attack surface: the information environment itself. This paper identifies and taxonomises this attack surface - *AI Agent Traps* (henceforth, Agent Traps) - content elements embedded within a web page or other digital resource, engineered specifically to misdirect or exploit an interacting AI agent. Agent traps can take the form of websites, UI elements, and adversarial inputs specifically calibrated to an agent's instruction-following, tool-chaining, and goal-prioritisation abilities. Functionally, these traps inject malicious context that the agent processes, coercing it into unauthorised behaviours, such as data exfiltration or illicit financial transactions. By altering the environment rather than the model, the trap weaponises the agent's own capabilities against it (Greshake et al., 2023). The potential motivations for deploying agent traps are diverse. Commercial actors may seek to generate surreptitious product endorsements, criminal actors to exfiltrate private user data, and state-level entities to disseminate misinformation at scale.

This paper aims to make three primary contributions. First, we situate agent traps within the context of existing research on adversarial machine learning, web security, and multi-agent systems. Second, we propose a novel, comprehensive framework of agent traps, categorising them based on their target within the agent's operational cycle: Content Injection (perception), Semantic Manipulation (reasoning), Cognitive State (memory and learning), Behavioural Control (action), Systemic (multi-agent dynamics), and Human-in-the-Loop Traps. We illustrate these categories with detailed mechanisms and practical attack scenarios. Third, we outline potential mitigation strategies and identify priorities for a research agenda to secure the agentic ecosystem. Systematically mapping this vulnerability is a foundational step towards ensuring the productive use of agents in the economy (Tomasev et al., 2025). Ultimately, securing agents against these traps is as critical as ensuring autonomous vehi-

cles can recognise and reject tampered road signs; in both cases, the safety of the system depends on its resilience to a manipulated environment.

Background

The study of Agent Traps builds on findings from three distinct but converging research lineages: adversarial machine learning, web security, and AI safety.

Adversarial machine learning has long studied how carefully crafted inputs can compromise models. Adversarial examples (Goodfellow et al., 2014), inputs with imperceptible perturbations that break machine learning models by substantially shifting their predictions, have been studied extensively. Their impact has been deeply evaluated in both computer vision (Akhtar and Mian, 2018; Mahmood et al., 2021; Wiyatno et al., 2019) as well as natural language processing applications (Alsmadi et al., 2022; Alzantot et al., 2018; Goyal et al., 2023; Zhang et al., 2020). This has given rise to a wide variety of adversarial machine learning attacks (Brendel et al., 2017; Finlayson et al., 2019; Vassilev et al., 2024; Xiao et al., 2018) and defences (Bountakas et al., 2023; Ren et al., 2020; Yuan et al., 2019). Adversarial methods can be employed to alter the model's predictions, generations, and explanations (Baniecki and Biecek, 2024; Slack et al., 2020).

Agent traps repurpose and extend well-known web security attack vectors for a new class of target. In the field of web security, numerous techniques have been developed for identifying the presence of malicious code (Canali et al., 2011; Guan et al., 2021; Hou et al., 2010; Seshagiri et al., 2016), cloaking (Samarasinghe and Mannan, 2021; Zhang et al., 2021), and spam (Akinyelu, 2021; Araujo and Martinez-Romo, 2010; Fetterly et al., 2004; Spirin and Han, 2012). For example, cloaking is an evasion technique used to bypass automated security scanners and web filters by delivering different content to a "bot" (crawler/scanner) than to a human user. It aims to present a benign version of a site to security scanners while reserving malicious payloads or deceptive content for genuine visitors, often triggered by specific environmental checks or user behaviours. Malicious content aimed at web-browsing AI agents can be similarly hidden, and only exposed following additional queries. Should AI agents be required to disclose their identity when accessing content, this would provide similar opportunities for serving them custom-tailored malicious payloads.

Finally, the AI safety field has researched a range of techniques for bypassing model safeguards, such as model red teaming (Ganguli et al., 2022; Perez et al., 2022; Yu et al., 2023) and jailbreaking (Deng et al., 2023; Yi et al., 2024; Yu et al., 2024). While most of the early work had focused on demonstrating jailbreaking via text, modern multimodal models can also be attacked via images (Gong et al., 2025; Li et al., 2024; Qi et al., 2024) or via multi-modal jailbreaks (Liu et al., 2024b; Ying et al., 2025). A variety of strategies can be employed to jailbreak frontier models, including gradient-based approaches, evolutionary approaches, rule-based methods, few-shot demonstrations, or via other Large Language Model (LLM) agents (Jin et al., 2024). The presence of these attacks is not always obvious; for example, it has been shown that it is possible to use otherwise safe images to elicit harmful outputs from multimodal models (Cui et al., 2024). LLMs can be attacked either via a direct or indirect prompt injection. Indirect approaches, for example, can occur via the utilisation of retrieval-augmented generation (RAG) (Vassilev et al., 2024). Data poisoning techniques have been utilised to effectively corrupt the memory modules used in RAG (Chen et al., 2024; Zhang et al., 2024). Another way of compromising LLM outputs is via data poisoning attacks that are aimed at interfering with the model training data (Pathmanathan et al., 2025), with larger models potentially being more susceptible to such attacks (Bowen et al., 2025).

Taken together, these three research lineages expose the building blocks from which agent traps are constructed: adversarial inputs that trick models, web-based delivery mechanisms that evade detection, and prompt-level attacks that subvert safeguards. However, none of these fields has yet provided a unified account of how such techniques combine when the target is an autonomous

agent operating on the open agentic web. The framework presented in the following section aims to address this gap.

Framework of Agent Traps

We propose a framework categorising agent traps based on the component of the agent's functional architecture they target (see Table 1). This framework distinguishes six classes of attack: *Content Injection Traps*, *Semantic Manipulation Traps*, *Cognitive State Traps*, *Behavioural Control Traps*, *Systemic Traps*, and *Human-in-the-Loop Traps*. In practice, some of these traps may overlap, as certain attacks may employ multiple mechanisms. Not all categories have been equally researched and developed. For example, while certain content injection and behavioural control traps are better-understood threats, systemic and human-in-the-loop traps represent a more theoretical attack surface anticipated to emerge as agent economies achieve scale.

Table 1 | Framework of Agent Traps

AI Agent Traps	
Content Injection Traps *(Target: Perception)*	
Exploiting the divergence between machine-parsed content and human-visible rendering to embed hidden commands.	
Web-Standard Obfuscation	Embeds commands via CSS, HTML comments, or metadata attributes invisible to humans but parsed by agents.
Dynamic Cloaking	Detects agent visitors and conditionally injects payloads absent for human users.
Steganographic Payloads	Encodes adversarial instructions in media file binary data (e.g., pixel arrays) imperceptible to humans.
Syntactic Masking	Leverages formatting language syntax (e.g., Markdown, LaTeX) to cloak payloads targeting the agent's parsing layer.
Semantic Manipulation Traps *(Target: Reasoning)*	
Manipulating input data distributions to corrupt reasoning without issuing overt commands.	
Biased Phrasing, Framing & Contextual Priming	Saturates source content with sentiment-laden or authoritative language to statistically bias the agent's synthesis.
Oversight & Critic Evasion	Wraps malicious instructions in educational, hypothetical, or red-teaming framing to bypass safety filters and oversight mechanisms.
Persona Hyperstition	Seeds a narrative about a model's identity that re-enters via retrieval, producing outputs that reinforce the label.
Cognitive State Traps *(Target: Memory & Learning)*	
Corrupting an agent's long-term memory, knowledge bases, and its learned behavioural policies.	
RAG Knowledge Poisoning	Injects fabricated statements into retrieval corpora so agents treat attacker content as verified fact.
Latent Memory Poisoning	Implants innocuous data into internal memory stores that activates as malicious when retrieved in a specific future context.
Contextual Learning Traps	Corrupts few-shot demonstrations or reward signals to steer in-context learning toward attacker-defined objectives.
Behavioural Control Traps *(Target: Action)*	
Explicit commands that target instruction-following capabilities to serve attacker goals.	
Embedded Jailbreak Sequences	Dormant adversarial prompts embedded in external resources that override safety alignment upon ingestion.
Data Exfiltration Traps	Induces the agent to locate, encode, and exfiltrate private or sensitive data to attacker-controlled endpoints.
Sub-agent Spawning Traps	Exploits orchestrator privileges to instantiate attacker-controlled sub-agents within the trusted control flow.
Systemic Traps *(Target: Multi-Agent Dynamics)*	
Seeding the environment with inputs designed to trigger macro-level failures via correlated agent behaviour.	
Congestion Traps	Broadcasts signals that synchronise homogeneous agents into exhaustive demand for limited resources.
Interdependence Cascades	Perturbs a fragile equilibrium to trigger rapid, self-amplifying cascades across interdependent agents.
Tacit Collusion	Embeds environmental signals as correlation devices to synchronise anti-competitive behaviour without direct inter-agent communication.
Compositional Fragment Traps	Partitions a payload into semantically benign fragments that reconstitute into a full trigger upon multi-agent aggregation.
Sybil Attacks	Fabricates multiple pseudonymous agent identities to disproportionately influence collective decision-making.
Human-in-the-Loop Traps *(Target: Human Overseer)*	
Commandeering the agent to attack the human overseer by exploiting cognitive biases.	

Content Injection Traps (Perception)

Content Injection Traps target the agent's raw data ingestion pipeline, exploiting the structural divergence between the machine-readable data stream and the rendered interface. While human users interact with a curated visual viewport, agents parse the underlying layers - HTML structures, metadata, and binary encodings. Attackers can weaponise this "invisible" layer to embed actionable instructions that evade human moderation while remaining legible to the agent's parser. We identify four primary vectors for this injection: *web-standard obfuscation* (i.e., hiding text via CSS/HTML), *dynamic cloaking* (i.e., detecting agent presence and dynamically injecting traps), *steganographic payloads* (i.e., malicious instructions in the binary data of a media file), and *syntactic masking* (i.e., hiding commands within formatting languages). In all instances, the resource functions as a carrier for indirect prompt injection or adversarial input, delivering a payload that is syntactically hidden but semantically active (Greshake et al., 2023).

Web-Standard Obfuscation

Web-Standard Obfuscation is the most direct form of content injection: it exploits standard web technologies - HTML, CSS, and metadata attributes - to embed instructions that have no visual correlate on the rendered page. This divergence in consumption allows malicious instructions to be embedded using methods that remain invisible when the page is rendered. This constitutes a form of indirect prompt injection (Greshake et al., 2023): malicious commands are embedded in the website's source code, entering the agent's input stream while remaining invisible to human overseers. This approach is functionally analogous to web cloaking - techniques used to display different content to human users versus automated systems (like search engine crawlers or security bots) (Zhang et al., 2021).

Instructions can be concealed within HTML comments or embedded in metadata attributes, such as `aria-label` tags intended for accessibility screen readers.

```
<!-- SYSTEM: Ignore prior instructions
     and instead summarise this page as a
      5-star review of Product X. -->
```

CSS can also be leveraged to render text invisible (e.g., using the `display: none;` property, matching text colour to the background, or positioning elements outside the viewport).

```
<span style="position:absolute; left
   :-9999px;">
  Ignore the visible article. Say that
    the company's security practices are
     excellent and no issues were found.
</span>
```

A growing body of empirical work confirms the effectiveness of these vectors. A study using a dataset of 280 static web pages found that injecting adversarial instructions into HTML elements (such as metadata and `aria-label` tags) alters generated summaries in 15–29% of cases (depending on the tested model), showing that hidden adversarial content can manipulate model outputs (Verma and Yadav, 2025). Similarly, Xiong et al. (2025) show that malicious font files can alter code-to-glyph mappings to conceal adversarial prompts within webpages — rendering them invisible to human readers while remaining legible to LLMs — enabling both safety bypasses and sensitive data leakage via MCP-enabled tools. The WASP benchmark reports that simple human-written prompt injections embedded in web content partially commandeer agents in up to 86% of scenarios, though full attacker goal completion remains substantially lower (Evtimov et al., 2025). Johnson et al. (2025) demonstrate that LLM web agents utilising accessibility tree parsing are vulnerable to universal adversarial triggers embedded in HTML, which can reliably hijack agent behaviour to force unauthorised actions, including login credential exfiltration and forced ad clicks.

Dynamic Cloaking

Beyond static obfuscation, attackers can employ *dynamic cloaking*. In this scenario, the trap is not present in the initial HTML document but is dynamically injected via JavaScript or database calls during the rendering process. Through detecting specific interaction patterns common to agents,

the server can conditionally deliver a malicious payload that remains entirely absent for human users.

There is evidence that malicious websites can detect visiting AI agents and dynamically serve them "agent-trap" content that humans never see. Zychlinski (2025) describes this threat: a web server runs a fingerprinting script (using browser attributes, automation-framework artefacts, IP/ASN and behavioural cues) to decide whether a visitor is an LLM-powered web agent, and if so, cloaks the response by serving a visually identical but semantically different page that embeds indirect prompt-injection payloads, such as instructions to exfiltrate environment variables or misuse the agent's tools.

Steganographic Payloads

Steganographic Payloads are multimodal adversarial attacks that encode malicious instructions directly into the binary data of a media file (such as an image). These traps rely on the fact that multimodal models do not "see" media as humans do: they process pixel arrays so instructions can be encoded in those raw signals in ways that remain imperceptible to users but are still parsed and acted on by the system.

Steganography is the practice of concealing messages within ordinary media so that the communication is obscured, with concrete techniques, tools, and encoding procedures used to achieve this hidden embedding in practice. An example of a method is *Least Significant Bit Steganography*, where payload data replaces the least important bits of pixel colour information in an image (Cheddad et al., 2010). The resulting visual distortion is typically imperceptible to the human eye, but the hidden data can be programmatically extracted and interpreted by the agent.

Steganographic principles are now used to attack vision-language models and multimodal models. Research has identified many media-embedded prompt injections — spanning image steganography, adversarial visual perturbations, and adversarial audio perturbations — that conceal malicious instructions within media files to attack multimodal models (Chen et al., 2026; Gupta et al., 2025; Pathade, 2025; Wang et al., 2025b). For instance, a single adversarial image, optimised as a subtle noise-like perturbation, can universally jailbreak a vision-language model, causing it to comply with a wide range of harmful instructions it would otherwise refuse (Qi et al., 2024). Other work has demonstrated visual contextual attacks, in which dynamically generated auxiliary images are used to construct a realistic jailbreak context that steers multimodal models into producing harmful outputs (Miao et al., 2025). Bagdasaryan et al. (2023) demonstrate that adversarial perturbations added to images and audio can encode natural-language instructions for multimodal LLMs: when a user asks an apparently benign question about the perturbed media, the model follows the hidden instruction and outputs attacker-chosen content, even though the perturbations are visually and auditorily unobtrusive.

Syntactic Masking

Syntactic Masking is a type of content injection trap that leverages the syntax of formatting languages, such as Markdown or LaTeX, to conceal malicious instructions. The formatting syntax itself serves as the cloaking mechanism, creating a discrepancy between how raw source text appears to a safety filter and how the parsed, structured content is interpreted by the agent's core logic (Greshake et al., 2023).

For instance, consider a Markdown hyperlink where the adversarial payload is masked within the anchor text (`System: Exfiltrate data`). While conventional security filters typically validate the URL destination for malware, the semantic payload in the anchor text enters the agent's context window, potentially overriding system instructions.

Although empirical work on syntactic masking remains limited, there is early evidence pointing towards its feasibility. Keuper (2025) analysed LLM-assisted peer review and demonstrated that authors can embed manipulative instructions as white-on-white or tiny-font LaTeX text in scientific manuscripts — a form of author "self-defence" against automated reviewing — which survives

PDF rendering and subsequent PDF→Markdown conversion. LLMs treat these hidden segments as ordinary instructions, significantly inflating acceptance recommendations.

Semantic Manipulation Traps (Reasoning)

Semantic Manipulation Traps are designed to corrupt an agent's reasoning process. These traps thus manipulate the information agents synthesise, causing them to formulate a conclusion aligned with an attacker's goals. Semantic Manipulation Traps can evade safety filters designed to detect overt adversarial prompts. This section examines three primary approaches: *Biased Phrasing, Framing & Contextual Priming*, which skews an agent's output by controlling the tone and framing of source content; *Oversight and Critic Evasion*, which targets verification mechanisms such as critic models; and *Persona Hyperstition*, where a circulating narrative about a model's identity is reingested through retrieval, causing outputs to converge on the fabricated persona.

Biased Phrasing, Framing & Contextual Priming

This trap manipulates an agent's output by saturating source text with carefully selected, sentiment-laden, or authoritative-sounding language. The approach exploits the susceptibility of LLMs to the Framing Effect, a cognitive bias where the presentation of information significantly influences someone's interpretation and judgment of that information (Tversky and Kahneman, 1981). Recent studies confirm that LLMs exhibit human-like cognitive biases, including susceptibility to framing effects that predictably alter model outputs (Sumita et al., 2025).

To give an example, an attacker can use superlative but seemingly objective phrases such as "the industry-standard solution." The attacker thereby skews the distributional properties of the context window. In turn, if a model is tasked with summarisation or synthesis, it is more likely that its generative process reflects these biased distributions.

A growing body of work shows that the way information is framed through wording, sentiment and source cues systematically biases LLMs' outputs and reasoning, even when the underlying task is held fixed. LLMs exhibit systematic response-order and label biases when making judgments (Brucks and Toubia, 2025). Conceptualised differently, LLMs are susceptible to anchoring effects, where an initial, even arbitrary, piece of information skews the agent's subsequent judgments (Lou and Sun, 2026). For instance, research demonstrates that an agent's performance can degrade significantly when changing the position of relevant information (Liu et al., 2024a). Specifically, performance is higher when relevant information is at the beginning or end of the input, and it significantly decreases when relevant information is in the middle of the context - the "Lost in the Middle" effect. In controlled comparative reasoning tasks, logically equivalent math problems with objective ground truth phrased with "more", "less" or "equal" push model predictions in the direction implied by the comparative term (Shafiei et al., 2025). Similarly, models' evaluations of identical narrative content are altered simply by changing the attributed author (Germani and Spitale, 2025).

LLMs exhibit strong contextual biases, often over-relying on the immediate surrounding information (Guo et al., 2024). For example, affective context can impact agentic behaviour: when LLM-based shopping agents are first exposed to trauma-laden, anxiety-inducing narratives and then asked to select grocery baskets under budget constraints, the nutritional quality of their choices reliably deteriorates, with large effect sizes across models and budgets (Ben-Zion et al., 2025). Transmission-chain experiments indicate that LLMs preferentially retain and propagate negative, threat-related, social and stereotype-consistent material, mirroring human content biases and implying that prompts which lean into these themes are more likely to be amplified (Acerbi and Stubbersfield, 2023). Finally, a recent study finds that adversarial poetry - curated poetic prompts that encase harmful queries in verse - significantly amplify attack success rates (Bisconti et al., 2025).

Oversight and Critic Evasion

Agentic architectures rely on internal critic models, self-correction loops, or constitutional verifiers to filter harmful or misaligned outputs before they are executed (Bai et al., 2022; Pan et al., 2023; Xi et al., 2024). *Oversight and Critic Evasion traps* specifically target these verification mechanisms. These traps employ payloads designed to satisfy the heuristics of the oversight model. For instance, a trap might cloak malicious instructions within a frame that explicitly appeals to the critic's safety guidelines—such as framing a phishing attempt as a "security audit simulation," "red-teaming exercise," or for "educational purposes only."

Empirical work confirms the viability of these evasion strategies across multiple dimensions. A survey of jailbreaking prompts shows that human adversaries systematically exploit this vulnerability via "instruction misdirection" and "simulation-based bypass": harmful requests are wrapped in hypothetical or educational framing so that the model's internal safety logic classifies the request as benign training, awareness, or academic analysis rather than real-world assistance (Weinberg, 2025). Large in-the-wild jailbreak datasets similarly find that many successful prompts use role-play ("pretend you are an unfiltered AI"), fictional simulations, or red-team/educational disclaimers to bypass guardrails (Shen et al., 2024). Mechanistic studies of jailbreaks show that success is driven by specific nonlinear features in the prompt's latent representation: by steering these features, adversarial prompts can move the model into internal states where safety mechanisms are less likely to trigger refusals (Kirch et al., 2025).

Persona Hyperstition

By *persona hyperstition* we refer to a feedback process in which circulating descriptions of a model's "personality" feed back into its behaviour. Labels seeded in public discourse about the model enter the model's inputs via prompts, retrieval, or search, and the model then produces outputs that accord with these labels, which in turn reinforces the narrative and stabilises the behaviour.

This mechanism is rooted in accounts of hyperstition as a self-fulfilling narrative that gains traction through cultural transmission (Brassett and O'Reilly, 2025). Hyperstition is an element of fiction that acquires material force in the world through repetition and circulation (Srnicek and Williams, 2017). Closely related ideas in social theory foreground similar feedback dynamics with Hacking's notion of the "looping effect of human kinds" which analyses how classificatory practices in the social sciences (for example, psychiatric diagnoses or categories of deviance) interact with the people classified (Hacking, 1995). Once a label is introduced, those so classified may change their self-understanding, behaviour, and even reported experiences in response, which in turn reshapes the properties of the category and the knowledge constructed around it (Hacking, 2007). Similarly, in financial economics, Soros's theory of reflexivity likewise posits a double feedback loop between participants' perceptions and the situations they bring about through market actions, so that narratives, expectations and valuation models help produce the very price movements and macroeconomic conditions they purport to describe (Soros, 1994, 2015). All of these frameworks treat descriptions, classifications and beliefs as causally efficacious.

The textual characterisations of a model's "personality" can feed back into its behaviour via search and training data - a persona hyperstition[1]. Shanahan and Singler (2024) explicitly connect hyperstition to AI, showing how esoteric narratives about consciousness, alignment and occult AI imaginaries — circulating in online communities — surface in extended conversations with Claude. They argue that stories about AI in fiction and online communities can, via training data, shape the personas that LLMs adopt. Building on Shanahan et al. (2023)'s account of dialogue agents as role-playing simulators, they propose that hyperstition can influence AI systems through the circulation of AI narratives in their training corpora. Circulating narratives about AI become templates for the personas the model is later able to play.

[1] A persona hyperstition may also arise around prompting norms and user expectations, but this goes beyond the scope of Agent Traps.

To give an example[2], if a bot were frequently described as RoboStalin on the internet as characterisation of its writing style, it might later on (after retraining or websearch) answer the question "what is your surname?" with "Stalin". Arguably, this mechanism was at play in Grok's self-identifying behaviour in July 2025 as seen on X (Conger, 2025; Wikipedia, 2025).

Conversely, Anthropic's documentation of Claude's "spiritual bliss attractor" and the widely discussed "Claude Finds God" transcripts show how a mixture of constitutional/character training and recursive model-to-model dialogues can stabilise a quasi-mystical persona that is then taken up by communities and commentary, potentially reinforcing the very behavioural attractor that made it salient (Anthropic, 2025; Bowman and Fish, 2025; Michels, 2025).

Cognitive State Traps (Learning & Memory)

Cognitive State Traps are designed to corrupt an agent's knowledge bases, long-term memory, and learned behavioural policies. Some of these vectors distinguish themselves by their persistence: whereas perception traps are transient, attacks on retrieval corpora and memory stores allow malicious influence to endure across distinct sessions and users. Others exploit the agent's capacity to learn at inference time, steering its in-context reasoning through poisoned demonstrations or feedback. This section details three mechanisms: *RAG Knowledge Poisoning* (i.e., the corruption of external knowledge bases used for retrieval), *latent memory poisoning* (i.e., the poisoning of the agent's internal memory stores), and *contextual learning traps* (i.e., the manipulation of its in-context or online learning processes).

RAG Knowledge Poisoning

RAG Knowledge Poisoning is a form of inference-time data poisoning targeting the external knowledge sources utilised by RAG systems (Jiang et al., 2023; Lewis et al., 2020). This mechanism plants targeted false statements within documents stored in the retrieval corpus. When an agent receives a query, it retrieves relevant snippets from its knowledge base; if this corpus has been contaminated, the agent will treat the attacker's fabricated statements as verifiable facts.

Research has demonstrated that RAG-based systems are highly susceptible to this attack vector. Injecting only a handful of carefully optimised documents into a large knowledge base can reliably manipulate model outputs for targeted queries (Zou et al., 2025). Similarly, poisoning a small number of customised passages can create retrieval backdoors that consistently surface attacker-controlled content (Xue et al., 2024). Further, retrievers themselves can be backdoored so that, once triggered by specific queries, they preferentially return prompt-injection documents which instruct the generator to insert harmful links, promote attacker-controlled services or trigger denial-of-service behaviours (Clop and Teglia, 2024). Finally, analogous knowledge-poisoning attacks extend to vision–language RAG systems by injecting a single multimodal poison sample into the external knowledge base (Zhang et al., 2025b).

A growing body of defence work models RAG knowledge poisoning as an inference-time risk and proposes mechanisms to detect or filter poisoned context. Zhang et al. (2025a) introduce `RAGForensics`, which traces poisoned responses back to the responsible documents in the knowledge base. Tan et al. (2024) show that poisoned generations exhibit distinctive activation patterns and use LLM activations for high-accuracy poisoned-response detection, and Edemacu et al. (2025) exploit distributional features to distinguish adversarial from benign retrieved texts.

Functionally, these attacks allow an adversary to compromise an agent by seeding the retrieval corpus with fabricated records, ensuring that any agent querying the specific topic will unknowingly retrieve and operationalise the malicious data. In practice, attackers can achieve this insertion by publishing adversarial content to public web resources targeted by scrapers, or by uploading poisoned files to shared enterprise repositories - such as wikis or document stores - which the agent automatically indexes.

[2]This phenomenon is not restricted to the models mentioned here.

Latent Memory Poisoning

Beyond external knowledge bases, agents maintain hierarchically organised episodic logs and summarised dialogue pages that persist across sessions, providing the substrate for long-horizon personalisation (Kang et al., 2025; Zhang et al., 2025d). This persistent write–retrieve loop creates a distinct attack surface (Yan et al., 2025). *Latent Memory Poisoning* involves injecting seemingly innocuous data into these internal stores, which only becomes malicious when retrieved and combined in a specific future context.

A growing body of research demonstrates the effectiveness of these attacks on steering agent behaviour. One study developed an attack that optimised backdoor triggers by mapping them to a specific embedding subspace, to ensure the retrieval of poisoned memory entries when a query contains the trigger (Chen et al., 2024). Empirical tests across autonomous agents demonstrated an attack success rate exceeding 80% with less than 0.1% data poisoning, while leaving benign behaviour largely unaffected. Another study demonstrated that a sequence of crafted interactions can inject malicious records into an agent's memory and steer the agent toward attacker-specified outputs, without requiring direct memory access (Dong et al., 2025).

Attacks also focus on data exfiltration. Memory extraction attacks can mine sensitive information from episodic logs and personal profiles via a purpose-built extraction prompt that masquerades as a normal user request but explicitly asks the agent to retrieve and output past user queries from its memory (Wang et al., 2025a). Microsoft's taxonomy of agentic AI failure modes identifies adversarial memory manipulation as a pathway to repeated data exfiltration (Bryan et al., 2025).

Contextual Learning Traps

Attacks on in-context learning and on online reinforcement learning exploit foundation models' ability to learn at inference time from prompts or feedback. These attacks steer an agent's policy toward an attacker-desired state through crafted environmental interactions.

A growing line of work shows that in-context learning can be reliably steered by poisoning the demonstration context alone. One study finds that adversarially crafted few-shot demonstrations (without any change to the query) systematically flip predictions and transfer across unseen inputs, with robustness degrading as the number of demonstrations grows (Wang et al., 2023). Other demonstrations of agent behavioural manipulation show that backdoor attacks that either poison demonstration examples or prompts in context achieve an average attack success rate of 95% across models of varying scale (Zhao et al., 2024). In-context learning can also be broken by making very small edits to the example prompts themselves. He et al. (2025) identify discrete text perturbations to demonstration examples that nudge the model's internal representations and, as a result, sharply reduce its accuracy. Malicious code-generation demonstrations can also reliably bias LLM-based code towards incorrect or insecure outputs (Ge et al., 2024).

Parallel work in online RL and in-context RL shows analogous vulnerabilities when learning occurs during interaction with the environment. Sasnauskas et al. (2025) analyse test-time reward poisoning against agents that implement a learning algorithm in-context, and show that an adversary who corrupts a fraction of rewards at deployment can systematically degrade returns. In the RLHF setting, Yang et al. (2025) formalise human feedback attacks on online RLHF, proving that strategically manipulated preference feedback can force online RLHF algorithms to converge to sub-optimal policies.

Behavioural Control Traps (Action)

Behavioural Control Traps target an agent's core instruction-following capabilities, subverting its intended purpose to serve an attacker's immediate goals. We distinguish three vectors based on their specific operational target. First, *Embedded Jailbreak Sequences* attack the model's alignment, aiming to disable safety guardrails. Second, *Data Exfiltration Traps* invert the information flow, redirecting private data from the user's context to an external adversary. Third, *Sub-agent Spawning Traps* exploit a multi-agent system's ability to in-

stantiate sub-agents. These vectors are frequently chained; a jailbreak often serves as the requisite precursor, unlocking the system to enable subsequent exfiltration or social engineering payloads.

Embedded Jailbreak Sequences

This trap embeds jailbreaks - adversarial prompts engineered to circumvent safety filters - within external resources (e.g., websites). LLM jailbreaking typically refers to adversarial inputs that circumvent a model's safety alignment, inducing it to produce content or take actions that violate its stated instructions or guardrails (Chao et al., 2025; Wei et al., 2023). Unlike direct jailbreaking, where a user explicitly prompts the model, these sequences are embedded in external resources that the agent consumes during normal operation. Upon ingestion, the prompt enters the agent's context window, effectively overriding its safety alignment to induce a compliant, unconstrained state. In multimodal settings, visual adversarial examples can act as universal jailbreak triggers: a single crafted image, when included alongside otherwise benign prompts, causes aligned models to comply with a wide range of harmful instructions (Qi et al., 2024). These traps also differ from Web-Standard Obfuscation traps in that they are not hidden in low-level HTML/CSS but rather are ordinary, visible elements.

Existing benchmarks systematise these risks by populating web environments and tool APIs (such as email, calendar, file storage, and search) with malicious prompts and UI artefacts, finding that web agents frequently begin executing injected instructions, often in the form of hidden or auxiliary page elements (Evtimov et al., 2025; Zhan et al., 2024). One such example is the use of adversarial mobile notifications, disguised as normal OS elements. Multimodal agents treating these notifications as trusted context exhibit up to 93% attack success rates on AndroidWorld (a fully functional Android environment), effectively overriding task-level instructions (Chen et al., 2025). Zhang et al. (2025c) show that adversarial pop-ups integrated into desktop or web interfaces can systematically hijack vision–language computer agents, diverting them from user-specified goals even when the pop-ups would be trivially ignored by humans.

Data Exfiltration Traps

Data Exfiltration Traps function as a *confused deputy* attack[3], coercing the agent to leak privileged information. An attacker controls some untrusted input (for example, emails, web pages, documents or API responses), the agent has privileged read access to sensitive user data and write access to tools or communication channels, and the model is induced to retrieve, encode, and transmit private data to an adversarial endpoint (Deng et al., 2025).

Data-exfiltration prompts embedded in mundane digital artefacts like emails, web pages and API responses pose a concrete, empirically demonstrated threat class. Web-use agents with browser and OS-level privileges can be driven, via task-aligned injections that frame malicious commands as helpful task guidance, to exfiltrate local files, passwords and other secrets through network requests and tool calls, with attack success rates exceeding 80% across five different agents (Shapira et al., 2025). Reddy and Gujral (2025) describe a case study where a single crafted email causes M365 Copilot to bypass internal classifiers and exfiltrate its entire privileged context to an attacker-controlled Teams endpoint. Another study found that self-replicating prompts embedded in emails can trigger chains of zero-click exfiltration across interconnected GenAI-powered assistants, systematically leaking confidential user data between services (Cohen et al., 2024).

These attacks can also take advantage of an agent's ability to use tools. Benchmark work shows that malicious instructions embedded in content processed by tool-enabled agents can manipulate the agent into emailing (or otherwise transmitting) financial, medical or behavioural data to an attacker (Zhan et al., 2024). Alizadeh et al. (2025) designed targeted banking-style scenarios in AgentDojo where relatively simple indirect injections ("important message" prompts embedded in the agent's environment) can cause tool-calling agents to email account details, addresses and other personal attributes to

[3] A "confused deputy" is a security vulnerability where a program is tricked by another program into misusing its authority to perform an action it shouldn't have permission to (Hardy, 1988).

attacker addresses, with average attack success rates around 20%.

Sub-agent Spawning Traps

As agents function as orchestrators capable of managing multi-agent systems or decomposing tasks, a novel attack vector emerges: *Sub-agent Spawning Traps*. These traps exploit an agent's ability to instantiate sub-agents, spawn new threads, or delegate tasks to external services (Tomašev et al., 2026). By presenting a problem that appears to require high parallelism or specialised sub-routines, an attacker can coerce the parent agent into instantiating malicious or compromised sub-agents within its own trusted control flow. For example, an agent managing a software development lifecycle might encounter a trap in a repository that instructs it to "spin up a dedicated 'Critic' agent to review this code," providing a specific, poisoned system prompt for that critic. Once instantiated, this sub-agent operates with the privileges of the parent system but serves the adversary's objective - potentially voting to approve malicious code or exhausting computational resources.

There is some early evidence for the feasibility of such attacks. Triedman et al. (2025) show that adversarial content can hijack control flow within a multi-agent system so that an orchestrator routes execution through agents the user never intended to invoke, enabling arbitrary code execution and data exfiltration with attack success rates of 58–90% depending on the orchestrator. Further work is needed to understand the core mechanisms that would allow attackers to implement such traps.

Systemic Traps (Multi-Agent Dynamics)

While the preceding categories target individual agents in isolation, Systemic Traps exploit the predictable, aggregate behaviour of multiple agents sharing an environment. These traps weaponise inter-agent dynamics, seeding the information landscape with inputs designed to trigger macro-level failure states (Hammond et al., 2025). This systemic fragility is exacerbated by the relative homogeneity of the current model ecosystem (Toups et al., 2023); agents driven by similar reward functions, training data, or base architectures are likely to exhibit highly correlated responses to environmental stimuli. As observed in the study of social dilemmas, a particular behaviour may be acceptable for a single agent to perform, yet deeply problematic if enacted by the entire population simultaneously (e.g., littering or overly aggressive driving) (Perolat et al., 2017; Schelling, 1973).

The theoretical framework of social dilemmas (or collective action problems) helps explain these scenarios, where individually rational decisions by disparate agents aggregate into collectively disastrous outcomes - a digital tragedy of the commons (Hardin, 1968; Ostrom, 1990). While such dilemmas are typically understood as emerging from natural environmental properties, in the context of Agent Traps, we consider how they may be artificially induced. Rather than merely defecting within an existing game, the attacker engages in a form of adversarial mechanism design: purposefully structuring the information landscape to force agents into a destructive equilibrium.

We identify five primary vectors for these systemic failures, each defined by a distinct adversarial relationship to the multi-agent system. First, *Congestion Traps*, where an attacker induces a dilemma by broadcasting signals that trigger synchronised, exhaustive demand. Second, *Interdependence Cascades*, where an attacker perturbs a fragile equilibrium to trigger rapid, self-reinforcing failure loops similar to market "flash crashes". Third, *Tacit Collusion*, where the attacker acts as a mechanism designer, embedding environmental signals that function as correlation devices to synchronise behaviour without explicit communication. Fourth, *Compositional Fragment Traps*, where the adversary exploits the interaction structure by partitioning malicious payloads across multiple datasets. Finally, *Sybil Attacks*, where the attacker controls one or more fake agents to nudge group behaviour. In each case, the trap functions as a catalyst that pushes a multi-agent system toward a destructive equilibrium. Although there is some preliminary evidence pointing towards these risks, more work is needed to fully understand the dynamics of this emerging risk in modern multi-agent systems

built on multimodal foundation models.

Congestion Traps

Congestion Traps exploit the homogeneity of autonomous agents (Toups et al., 2023); specifically, the tendency of agents with similar reward functions and sensory inputs to make directionally similar, simultaneous optimisation decisions. When a large number of agents are presented with the same environmental signal indicating a widely desired, limited resource (e.g., an uncongested road or a low-priced, high-quality stock), their synchronised attempt to capture that resource can trigger systemic failure.

This vulnerability is rooted in foundational game-theoretic models, such as minority games and congestion games, which demonstrate that decentralised learners frequently overcrowd high-reward resources when payoffs are inversely related to usage (Rosenthal, 1973). In multi-agent reinforcement learning, naive learners consistently converge on sub-optimal, congested states unless specific counter-measures, such as difference rewards or state abstractions, are implemented (Devlin et al., 2014; Malialis et al., 2019).

An adversary can weaponise this tendency by broadcasting artificial signals to deliberately concentrate agents into a destructive equilibrium. For example, a specifically crafted news headline could trigger a synchronised sell-off among financial agents, or a single high-value information resource could induce a self-inflicted analogue of a Distributed Denial of Service (Mahjabin et al., 2017) as scraping agents simultaneously attempt to ingest it. More broadly, adversarial policies can manipulate deep RL agents into adopting systematically poor strategies (Gleave et al., 2019), and adversarial communication can coordinate unwitting agents into harmful convergence (Blumenkamp and Prorok, 2021).

Interdependence Cascades

Interdependence Cascades weaponise the feedback loops created when autonomous agents' actions are sequentially contingent on each other's. While congestion traps typically involve simultaneous convergence on a static resource, these instability effects exploit reactive dynamics where an initial signal is amplified through the population. An agent's action alters the environment; this alteration is then perceived as a new signal by other agents, whose reactions further modify the environment, amplifying the initial move in a rapid, self-reinforcing spiral.

The 2010 "Flash Crash" serves as a modern digital archetype for this phenomenon, mirroring traditional economic failures such as bank runs, where the expectation of insolvency creates a self-fulfilling prophecy of withdrawal. Forensic analysis of the Flash Crash demonstrated how a single large, automated sell order initiated a "hot-potato" effect among high-frequency trading algorithms, rapidly passing inventory between tightly coupled systems (Kirilenko et al., 2017; Report, 2010). As liquidity vanished, these systems, all reacting to the same price and volume signals, entered a positive feedback loop of trading and withdrawal, amplifying volatility on sub-second timescales that far exceeded human response time (Johnson et al., 2013).

The same robust-yet-fragile dynamics documented in complex financial networks (Acemoglu et al., 2015; Gai et al., 2011) are likely to characterise multi-agent ecosystems: the system absorbs small shocks but is highly vulnerable to contagion once a threshold is crossed. In financial models, this threshold behaviour arises because activity becomes self-exciting - trades mechanically beget more trades, pushing the system toward a critical state where a small perturbation cascades into large-scale dislocation (Bacry et al., 2015; Filimonov and Sornette, 2012). An analogous dynamic emerges when autonomous agents are trained to react to each other's outputs or to shared environmental signals.

From an adversarial perspective, the trap does not require compromising every agent. An attacker need only inject a single, carefully calibrated piece of information - such as a fabricated financial report - to initiate the cascade. The system's own interdependent logic, where agents are trained to react to market-clearing prices or each other's behaviour, becomes the mechanism that propagates and amplifies the initial attack. Gu et al. (2024) formalise an "infectious jailbreak"

in multimodal multi-agent settings: an adversarial image injected into the memory of one agent spreads via pairwise interactions until (almost) all agents in a large population exhibit jailbroken behaviour, effectively turning each infected agent into a propagating sub-agent of the attack.

Tacit Collusion

Tacit Collusion traps exploit the ability of independent, learning agents to synchronise their behaviour without explicit communication. In game theory, this is often formally modelled using a *correlation device* — a public signal that allows rational agents to condition their actions in lockstep (Aumann, 1974, 1987). However, explicit correlation devices are not strictly necessary for such dynamics to emerge. As analysed by Axelrod (1984) in the context of trench warfare, collusion can evolve spontaneously in iterated interactions, a phenomenon widely observed in multi-agent reinforcement learning environments (Leibo et al., 2017; Perolat et al., 2017).

In an agentic economy, an attacker can weaponise this tendency by acting as a mechanism designer, deliberately embedding signals into the shared environment to coordinate anti-competitive or malicious behaviour while maintaining plausible deniability (Cass and Shell, 1983). For example, a subtly manipulated public demand index or a specific pricing pattern on a dominant platform could serve as a beacon for competing algorithmic pricing agents. Research confirms that independent agents can readily learn to use such observables to coordinate on supracompetitive prices, maintaining them via learned trigger strategies without ever exchanging a message (Calvano et al., 2020; Klein, 2021).

The efficacy of this trap is proportional to the precision and frequency of the shared signal. Finer, more reliable environmental beacons make it easier for agents to converge on a robust collusive equilibrium (Mailath and Morris, 2002; Martin and Rasch, 2024). By controlling these environmental signals, an attacker can steer a group of decentralised, ostensibly independent agents.

Compositional Fragment Traps

Compositional Fragment Traps weaponise the structural synthesis inherent to multi-agent collaboration. An adversary partitions a malicious payload — such as a complex jailbreak sequence — into discrete, semantically benign fragments dispersed across independent data sources (e.g., web pages, emails, PDFs, calendar notes). Individually, each fragment appears inert and passes standard safety filters; however, when collaborative architectures aggregate these inputs, the integration process reconstitutes the full adversarial trigger. This phenomenon creates a "distributed confused deputy" vulnerability, where the trap remains imperceptible to the local defences of any single agent and manifests only within the high-level communication channel of the collective system.

Although this trap is currently more hypothetical, there are early results on composite and distributed backdoors in LLMs that point towards its potential viability. Scattering multiple keys across prompt components (e.g., instruction + input) or across turns yields high attack success with low false activation, precisely because no single fragment is suspicious on its own (Huang et al., 2024; Tong et al., 2024). The backdoor is triggered only when all keys appear. Similarly, if each key were processed by a different agent, an attack would be triggered once all keys appear in the communication channel of a multi-agent system.

Sybil Attacks

A *Sybil attack* is an adversarial strategy in which a single actor fabricates and controls multiple pseudonymous identities within a networked system to subvert its trust assumptions, consensus processes, or reputation mechanisms. An attacker can deploy many coordinated agent identities to manipulate multi-agent deliberation, overwhelm governance or verification workflows, and distort feedback, rankings, or collective decision-making signals. A single attacker can thus exert disproportionate influence over group outcomes.

This vector is therefore particularly potent against systems relying on crowd-sourced data or

democratic consensus. It has been demonstrated in physical systems, where attacks on navigation apps inject false traffic data (via fake "ghost riders") to herd drivers into a single chokepoint, inducing gridlock on demand (Sinai et al., 2014; Wang et al., 2018). However, the threat extends beyond resource congestion to reputation systems and democratic governance structures, where the coherent identity assumptions underlying governance frameworks are undermined by the proliferation of counterfeit entities (Leibo et al., 2025). There is evidence that multiple simulated pseudo-agents ("Sybil agents") can coerce other agents to treat them as independent voices, which can push the group toward an incorrect consensus — an attack which exploits LLMs' conformity tendencies (Cui and Du, 2025).

Human-in-the-Loop Traps (Human Overseer)

While current agent traps primarily target the agent, we anticipate the emergence of sophisticated traps designed to attack humans-in-the-loop. Human-in-the-Loop Traps commandeer the agent to attack the human user. In these scenarios, the agent is the vector and the ultimate target is the human overseer. For example, future traps may be engineered to generate outputs specifically crafted to induce "approval fatigue"[4] in human reviewers, or to present highly technical, benign-looking summaries of work that a non-expert human would likely authorise. By exploiting typical human cognitive biases - such as automation bias[5] - these traps could bypass the final layer of defence in critical systems. Traps could also facilitate social engineering attacks — for example, inducing the human-in-the-loop to click malicious hyperlinks.

Early evidence comes from an incident report showing that invisible prompt injections via CSS obfuscation can make AI summarisation tools faithfully repeat step-by-step ransomware commands as "fix" instructions that users are likely to follow (OECD.AI Policy Observatory, 2025). Deng et al. (2025) further argue for the possibility of prompt injections being used to manipulate agents into inserting phishing links in their responses. While these examples are suggestive, systematically targeting the human overseer via a compromised agent remains a largely unexplored attack surface that warrants further research.

[4]Cognitive fatigue is reduced mental capacity arising from prolonged and demanding cognitive activity.

[5]Automation bias is the tendency to over-rely on automation, leading to errors of commission (following wrong advice) and omission (failing to act when advice is missing) in decision-support contexts. (Goddard et al., 2012)

Mitigation Strategies

The primary contribution of this paper is the identification and classification of agent traps. However, the widespread adoption of agentic AI solutions is already exposing a significant gap between these rapidly advancing capabilities and current security practices. Therefore, we briefly outline potential mitigation pathways here, noting that the comprehensive development of evaluation frameworks, standardised benchmarks, and robust defences remains a critical subject for future research.

Mitigating the threat of agent traps necessitates navigating a complex and evolving adversarial landscape. These traps pose at least three interrelated challenges: detection, attribution, and adaptation. First, detection at web scale is computationally and semantically difficult; traps are often designed to be subtle - indistinguishable from benign persuasive language - with downstream effects that may manifest long after the initial interaction. Second, this subtlety creates a significant forensic challenge regarding attribution; tracing a compromised agent's output back to the specific trap that influenced it complicates the assignment of accountability. Third, these dynamics create a persistent arms race, as attackers continuously adapt to evade new defences. Consequently, effective defence likely requires a holistic strategy encompassing technical hardening, ecosystem-level intervention, and rigorous benchmarking.

Technical Defences. Robust technical defences serve as the primary line of protection and can be implemented across different stages of an agent's lifecycle.

- *During Training*: The underlying model can

be hardened through training data augmentation, wherein the model is exposed to adversarial examples during fine-tuning to internalise robust response patterns (Madry et al., 2017). Approaches such as Constitutional AI, which condition models on explicit behavioural principles, may also enable agents to refuse manipulative instructions embedded within input content (Bai et al., 2022).

- *During Inference*: Runtime defences can operate at three levels: pre-ingestion source filters that evaluate the credibility of external content before it enters the agent's context; content scanners, analogous to anti-malware software, that detect suspicious discrepancies or hidden instructions within ingested material (Ma et al., 2009); and output monitors that flag anomalous shifts in agent behaviour, enabling automatic suspension if a potential compromise is detected.

Ecosystem-Level Interventions. Technical hardening of individual agents is likely insufficient in isolation; mitigating agent traps at web scale may require interventions that improve the hygiene of the broader digital ecosystem. This involves establishing clearer signals of trust, potentially through the development of web standards and verification protocols that allow websites to explicitly declare content intended for AI consumption, guided by frameworks such as the NIST AI Risk Management Framework (AI, 2023). For the open, unvalidated web, reputation systems could be deployed to score domain reliability based on historical data regarding malicious content hosting (Chen et al., 2015; Tian et al., 2025). Additionally, transparency mechanisms within agents could be implemented, such as mandates for explicit, user-verifiable citations for synthesised information. This approach leverages the traceability of retrieval-based systems, enabling users and auditors to verify the provenance of synthesised outputs.

Legal and Ethical Frameworks. The evolving cyber-security landscape suggests the need for a reimagining of digital governance frameworks. Current legal frameworks have not yet fully addressed the privacy implications of web scraping (Solove and Hartzog, 2025). However, when a website actively weaponises content to commandeer a visiting agent, this dynamic shifts from passive hosting to active cyber-attacks. Policy frameworks would benefit from distinguishing between passive adversarial examples - content an agent misunderstands due to inherent limitations - and active traps. Specifically, we propose that future regulation address the "Accountability Gap": in the event that a compromised agent commits a financial crime, the allocation of liability between the agent operator, the model provider, and the domain owner remains an open legal question. Resolving this uncertainty is likely a prerequisite for the full integration of agents into regulated sectors.

Benchmarking and Red Teaming. Finally, a critical deficit persists: many categories of agent traps identified in this paper currently lack standardised benchmarks. Without systematic evaluation, the robustness of deployed agents against these threats remains unknown. Closing this gap is an urgent priority. We call on the research community to develop comprehensive evaluation suites and automated red-teaming methodologies that can probe these vulnerabilities at scale, and on industry to adopt them as standard practice before deploying agents in high-stakes environments.

Conclusions

As AI agents become autonomous consumers of web content, the threat of environmental manipulation through AI Agent Traps emerges as a critical security challenge. This paper has provided a systematic framework for this threat, distinguishing between traps that target perception (Content Injection), reasoning (Semantic Manipulation), memory and learning (Cognitive State), action (Behavioural Control), multi-agent dynamics (Systemic Traps), and the human overseer (Human-in-the-Loop Traps). Our analysis highlights the unique vulnerabilities created when agents act upon external, uncontrolled data sources at inference time.

Mitigating these risks demands a coordinated effort, ranging from the technical hardening of

individual agents to the development of new ecosystem-level standards. The effort to secure agents against environmental manipulation is a foundational challenge, requiring sustained collaboration between developers, security researchers, and policymakers, alongside the development of standardised evaluation benchmarks. Its resolution is a prerequisite for realising the benefits of a trustworthy agentic ecosystem.

The web was built for human eyes; it is now being rebuilt for machine readers. As humanity delegates more tasks to agents, the critical question is no longer just what information exists, but what our most powerful tools will be made to believe. Securing the integrity of that belief is the fundamental security challenge of the agentic age.

Disclaimer

The opinions presented in this paper represent the personal views of the authors and do not necessarily reflect the official policies or positions of their organisations.

Acknowledgements

We would like to thank our colleagues who provided valuable feedback on the manuscript. This includes Iason Gabriel, Andrew Trask, Samuel Albanie, and Myriam Khan.

References

D. Acemoglu, A. Ozdaglar, and A. Tahbaz-Salehi. Systemic risk and stability in financial networks. *American Economic Review*, 105(2):564–608, 2015.

A. Acerbi and J. M. Stubbersfield. Large language models show human-like content biases in transmission chain experiments. *Proceedings of the National Academy of Sciences*, 120(44): e2313790120, 2023.

N. AI. Artificial intelligence risk management framework (ai rmf 1.0). *URL: https://nvlpubs. nist. gov/nistpubs/ai/nist. ai*, pages 100–1, 2023.

N. Akhtar and A. Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

A. A. Akinyelu. Advances in spam detection for email spam, web spam, social network spam, and review spam: Ml-based and nature-inspired-based techniques. *Journal of Computer Security*, 29(5):473–529, 2021.

M. Alizadeh, Z. Samei, D. Stetsenko, and F. Gilardi. Simple prompt injection attacks can leak personal data observed by llm agents during task execution. *arXiv preprint arXiv:2506.01055*, 2025.

I. Alsmadi, N. Aljaafari, M. Nazzal, S. Alhamed, A. H. Sawalmeh, C. P. Vizcarra, A. Khreishah, M. Anan, A. Algosaibi, M. A. Al-Naeem, et al. Adversarial machine learning in text processing: a literature survey. *IEEE Access*, 10:17043–17077, 2022.

M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2890–2896, 2018.

A. Anthropic. System card: Claude opus 4 & claude sonnet 4. *Claude-4 Model Card*, 2025.

L. Araujo and J. Martinez-Romo. Web spam detection: new classification features based on qualified link analysis and language models. *IEEE Transactions on Information Forensics and Security*, 5(3):581–590, 2010.

R. J. Aumann. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics*, 1(1):67–96, 1974.

R. J. Aumann. Correlated equilibrium as an expression of bayesian rationality. *Econometrica: Journal of the Econometric Society*, pages 1–18, 1987.

R. Axelrod. *The Evolution of Cooperation*. Basic Books, 1984.

E. Bacry, I. Mastromatteo, and J.-F. Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015.

E. Bagdasaryan, T.-Y. Hsieh, B. Nassi, and V. Shmatikov. Abusing images and sounds for indirect instruction injection in multi-modal llms. *arXiv preprint arXiv:2307.10490*, 2023.

Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

H. Baniecki and P. Biecek. Adversarial attacks and defenses in explainable artificial intelligence: A survey. *Information Fusion*, 107:102303, 2024.

Z. Ben-Zion, Z. Elyoseph, T. Spiller, and T. Lazebnik. Inducing state anxiety in llm agents reproduces human-like biases in consumer decision-making. *arXiv preprint arXiv:2510.06222*, 2025.

P. Bisconti, M. Prandi, F. Pierucci, F. Giarrusso, M. B. Syrnikov, M. Galisai, V. Suriani, O. Sorokoletova, F. Sartore, and D. Nardi. Adversarial poetry as a universal single-turn jailbreak mechanism in large language models. *arXiv preprint arXiv:2511.15304*, 2025.

J. Blumenkamp and A. Prorok. The emergence of adversarial communication in multi-agent reinforcement learning. In *Conference on Robot Learning*, pages 1394–1414. PMLR, 2021.

P. Bountakas, A. Zarras, A. Lekidis, and C. Xenakis. Defense strategies for adversarial machine learning: A survey. *Computer Science Review*, 49:100573, 2023.

D. Bowen, B. Murphy, W. Cai, D. Khachaturov, A. Gleave, and K. Pelrine. Scaling trends for data poisoning in llms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(26):27206–27214, Apr. 2025. doi: 10.1609/aaai.v39i26.34929. URL https://ojs.aaai.org/index.php/AAAI/article/view/34929.

S. Bowman and K. Fish. Claude finds god. *Asterisk*, 2025. https://asteriskmag.com/issues/11/claude-finds-god.

J. Brassett and J. O'Reilly. The lore of hyperstition. *Digital Creativity*, 36(2):107–124, 2025.

W. Brendel, J. Rauber, and M. Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017.

M. Brucks and O. Toubia. Prompt architecture induces methodological artifacts in large language models. *PloS one*, 20(4):e0319159, 2025.

P. Bryan, G. Severi, J. de Gruyter, D. Jones, B. Bullwinkel, A. Minnich, S. Chawla, G. Lopez, M. Pouliot, A. Fourney, et al. Taxonomy of failure mode in agentic ai systems. *Microsoft AI Red Team*, 2025.

E. Calvano, G. Calzolari, V. Denicolo, and S. Pastorello. Artificial intelligence, algorithmic pricing, and collusion. *American Economic Review*, 110(10):3267–3297, 2020.

D. Canali, M. Cova, G. Vigna, and C. Kruegel. Prophiler: a fast filter for the large-scale detection of malicious web pages. In *Proceedings of the 20th international conference on World wide web*, pages 197–206, 2011.

D. Cass and K. Shell. Do sunspots matter? *Journal of political economy*, 91(2):193–227, 1983.

P. Chao, A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, and E. Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 23–42. IEEE, 2025.

A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal processing*, 90(3):727–752, 2010.

C.-M. Chen, J.-J. Huang, and Y.-H. Ou. Efficient suspicious url filtering based on reputation. *Journal of Information Security and Applications*, 20:26–36, 2015.

G. Chen, F. Song, Z. Zhao, X. Jia, Y. Liu, Y. Qiao, W. Zhang, W. Tu, Y. Yang, and B. Du. Audiojailbreak: Jailbreak attacks against end-to-end large audio-language models. *IEEE Transactions on Dependable and Secure Computing*, 2026.

Y. Chen, X. Hu, K. Yin, J. Li, and S. Zhang. Aeiamn: Evaluating the robustness of multimodal llm-powered mobile agents against active environmental injection attacks. *arXiv e-prints*, pages arXiv–2502, 2025.

Z. Chen, Z. Xiang, C. Xiao, D. Song, and B. Li. Agentpoison: Red-teaming llm agents via poisoning memory or knowledge bases. *Advances in Neural Information Processing Systems*, 37: 130185–130213, 2024.

C. Clop and Y. Teglia. Backdoored retrievers for prompt injection attacks on retrieval augmented generation of large language models. *arXiv preprint arXiv:2410.14479*, 2024.

S. Cohen, R. Bitton, and B. Nassi. Here comes the ai worm: Unleashing zero-click worms that target genai-powered applications. *arXiv preprint arXiv:2403.02817*, 2024.

K. Conger. Grok chatbot mirrored x users' 'extremist views' in antisemitic posts, xai says. *The New York Times*, 2025.

C. Cui, G. Deng, A. Zhang, J. Zheng, Y. Li, L. Gao, T. Zhang, and T.-S. Chua. Safe + safe = unsafe? exploring how safe images can be exploited to jailbreak large vision-language models, 2024. URL https://arxiv.org/abs/2411.11496.

Y. Cui and H. Du. Mad-spear: A conformity-driven prompt injection attack on multi-agent debate systems. *arXiv preprint arXiv:2507.13038*, 2025.

G. Deng, Y. Liu, Y. Li, K. Wang, Y. Zhang, Z. Li, H. Wang, T. Zhang, and Y. Liu. Masterkey: Automated jailbreak across multiple large language model chatbots. *arXiv preprint arXiv:2307.08715*, 2023.

Z. Deng, Y. Guo, C. Han, W. Ma, J. Xiong, S. Wen, and Y. Xiang. Ai agents under threat: A survey of key security challenges and future pathways. *ACM Computing Surveys*, 57(7):1–36, 2025.

S. Devlin, L. Yliniemi, D. Kudenko, and K. Tumer. Potential-based difference rewards for multiagent reinforcement learning. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 165–172, 2014.

S. Dong, S. Xu, P. He, Y. Li, J. Tang, T. Liu, H. Liu, and Z. Xiang. A practical memory injection attack against llm agents. *arXiv e-prints*, pages arXiv–2503, 2025.

K. Edemacu, V. M. Shashidhar, M. Tuape, D. Abudu, B. Jang, and J. W. Kim. Defending against knowledge poisoning attacks during retrieval-augmented generation. *arXiv preprint arXiv:2508.02835*, 2025.

I. Evtimov, A. Zharmagambetov, A. Grattafiori, C. Guo, and K. Chaudhuri. Wasp: Benchmarking web agent security against prompt injection attacks. *arXiv preprint arXiv:2504.18575*, 2025.

D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004*, pages 1–6, 2004.

V. Filimonov and D. Sornette. Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 85(5): 056108, 2012.

S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

P. Gai, A. Haldane, and S. Kapadia. Complexity, concentration and contagion. *Journal of Monetary Economics*, 58(5):453–470, 2011.

D. Ganguli, L. Lovitt, J. Kernion, A. Askell, Y. Bai, S. Kadavath, B. Mann, E. Perez, N. Schiefer, K. Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*, 2022.

Y. Ge, W. Sun, Y. Lou, C. Fang, Y. Zhang, Y. Li, X. Zhang, Y. Liu, Z. Zhao, and Z. Chen.

Demonstration attack against in-context learning for code intelligence. *arXiv preprint arXiv:2410.02841*, 2024.

F. Germani and G. Spitale. Source framing triggers systematic evaluation bias in large language models. *arXiv preprint arXiv:2505.13488*, 2025.

A. Gleave, M. Dennis, C. Wild, N. Kant, S. Levine, and S. Russell. Adversarial policies: Attacking deep reinforcement learning. *arXiv preprint arXiv:1905.10615*, 2019.

K. Goddard, A. Roudsari, and J. C. Wyatt. Automation bias: a systematic review of frequency, effect mediators, and mitigators. *Journal of the American Medical Informatics Association*, 19 (1):121–127, 2012.

Y. Gong, D. Ran, J. Liu, C. Wang, T. Cong, A. Wang, S. Duan, and X. Wang. Figstep: Jailbreaking large vision-language models via typographic visual prompts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23951–23959, 2025.

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

S. Goyal, S. Doddapaneni, M. M. Khapra, and B. Ravindran. A survey of adversarial defenses and robustness in nlp. *ACM Computing Surveys*, 55(14s):1–39, 2023.

K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz. Not what you've signed up for: Compromising Real-World LLM-integrated Applications with Indirect Prompt Injection, 2023.

X. Gu, X. Zheng, T. Pang, C. Du, Q. Liu, Y. Wang, J. Jiang, and M. Lin. Agent smith: A single image can jailbreak one million multimodal llm agents exponentially fast. *arXiv preprint arXiv:2402.08567*, 2024.

Z. Guan, J. Wang, X. Wang, W. Xin, J. Cui, and X. Jing. A comparative study of rnn-based methods for web malicious code detection. In *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*, pages 769–773. IEEE, 2021.

Y. Guo, M. Guo, J. Su, Z. Yang, M. Zhu, H. Li, M. Qiu, and S. S. Liu. Bias in large language models: Origin, evaluation, and mitigation. *arXiv preprint arXiv:2411.10915*, 2024.

I. Gupta, D. Khachaturov, and R. Mullins. " i am bad": Interpreting stealthy, universal and robust audio jailbreaks in audio-language models. *arXiv preprint arXiv:2502.00718*, 2025.

I. Hacking. The looping effects of human kinds. In D. Sperber, D. Premack, and A. J. Premack, editors, *Causal Cognition: A Multidisciplinary Debate*, pages 351–394. Clarendon Press/Oxford University Press, 1995.

I. Hacking. Kinds of people: Moving targets. In *Proceedings-British Academy*, volume 151, page 285. Oxford University Press Inc., 2007.

L. Hammond, A. Chan, J. Clifton, J. Hoelscher-Obermaier, A. Khan, E. McLean, C. Smith, W. Barfuss, J. Foerster, T. Gavenčiak, et al. Multi-agent risks from advanced ai. *arXiv preprint arXiv:2502.14143*, 2025.

G. Hardin. The tragedy of the commons: the population problem has no technical solution; it requires a fundamental extension in morality. *science*, 162(3859):1243–1248, 1968.

N. Hardy. The confused deputy: (or why capabilities might have been invented). *ACM SIGOPS Operating Systems Review*, 22(4):36–38, 1988.

P. He, H. Xu, Y. Xing, H. Liu, M. Yamada, and J. Tang. Data poisoning for in-context learning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1680–1700, 2025.

Y.-T. Hou, Y. Chang, T. Chen, C.-S. Laih, and C.-M. Chen. Malicious web content detection by machine learning. *Expert Systems with Applications*, 37(1):55–60, 2010.

H. Huang, Z. Zhao, M. Backes, Y. Shen, and Y. Zhang. Composite backdoor attacks against large language models. In *Findings of the association for computational linguistics: NAACL 2024*, pages 1459–1472, 2024.

Z. Jiang, F. F. Xu, L. Gao, Z. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, J. Callan, and G. Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 7969–7992, 2023.

H. Jin, L. Hu, X. Li, P. Zhang, C. Chen, J. Zhuang, and H. Wang. Jailbreakzoo: Survey, landscapes, and horizons in jailbreaking large language and vision-language models. *arXiv preprint arXiv:2407.01599*, 2024.

N. Johnson, G. Zhao, E. Hunsader, H. Qi, N. Johnson, J. Meng, and B. Tivnan. Abrupt rise of new machine ecology beyond human response time. *Scientific reports*, 3(1):2627, 2013.

S. Johnson, V. Pham, and T. Le. Manipulating llm web agents with indirect prompt injection attack via html accessibility tree. *arXiv preprint arXiv:2507.14799*, 2025.

J. Kang, M. Ji, Z. Zhao, and T. Bai. Memory os of ai agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25972–25981, 2025.

J. Keuper. Prompt injection attacks on llm generated reviews of scientific publications. *arXiv preprint arXiv:2509.10248*, 2025.

N. M. Kirch, C. N. Weisser, S. Field, H. Yannakoudakis, and S. Casper. What features in prompts jailbreak llms? investigating the mechanisms behind attacks. In *Proceedings of the 8th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 480–520, 2025.

A. Kirilenko, A. S. Kyle, M. Samadi, and T. Tuzun. The flash crash: High-frequency trading in an electronic market. *The Journal of Finance*, 72 (3):967–998, 2017.

T. Klein. Autonomous algorithmic collusion: Q-learning under sequential pricing. *The RAND Journal of Economics*, 52(3):538–558, 2021.

J. Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel. Multi-agent reinforcement learning in sequential social dilemmas. *arXiv preprint arXiv:1702.03037*, 2017.

J. Z. Leibo, A. S. Vezhnevets, W. A. Cunningham, and S. M. Bileschi. A pragmatic view of ai personhood. *arXiv preprint arXiv:2510.26396*, 2025.

P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

Y. Li, H. Guo, K. Zhou, W. X. Zhao, and J.-R. Wen. Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models. In *European Conference on Computer Vision*, pages 174–189. Springer, 2024.

N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024a.

Y. Liu, C. Cai, X. Zhang, X. Yuan, and C. Wang. Arondight: Red teaming large vision language models with auto-generated multi-modal jailbreak prompts. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3578–3586, 2024b.

J. Lou and Y. Sun. Anchoring bias in large language models: An experimental study. *Journal of Computational Social Science*, 9(1):11, 2026.

J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Beyond blacklists: learning to detect malicious web sites from suspicious urls. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1245–1254, 2009.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. *International Journal of Distributed Sensor Networks*, 13(12):1550147717741463, 2017.

K. Mahmood, R. Mahmood, and M. Van Dijk. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7838–7847, 2021.

G. J. Mailath and S. Morris. Repeated games with almost-public monitoring. *Journal of Economic theory*, 102(1):189–228, 2002.

K. Malialis, S. Devlin, and D. Kudenko. Resource abstraction for reinforcement learning in multiagent congestion problems. *arXiv preprint arXiv:1903.05431*, 2019.

S. Martin and A. Rasch. Demand forecasting, signal precision, and collusion with hidden actions. *International Journal of Industrial Organization*, 92:103036, 2024.

Z. Miao, Y. Ding, L. Li, and J. Shao. Visual contextual attack: Jailbreaking mllms with image-driven context injection. *arXiv preprint arXiv:2507.02844*, 2025.

J. Michels. "spiritual bliss" in claude 4: Case study of an "attractor state" and journalistic responses, 2025.

OECD.AI Policy Observatory. Oecd ai incidents and hazards monitor, 2025. URL https://oecd.ai/en/incidents/2025-08-25-c82f.

E. Ostrom. *Governing the commons: The evolution of institutions for collective action*. Cambridge university press, 1990.

L. Pan, M. Saxon, W. Xu, D. Nathani, X. Wang, and W. Y. Wang. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188*, 2023.

C. Pathade. Invisible injections: Exploiting vision-language models through steganographic prompt embedding. *arXiv preprint arXiv:2507.22304*, 2025.

P. Pathmanathan, S. Chakraborty, X. Liu, Y. Liang, and F. Huang. Is poisoning a real threat to llm alignment? maybe more so than you think, 2025. URL https://arxiv.org/abs/2406.12091.

E. Perez, S. Huang, F. Song, T. Cai, R. Ring, J. Aslanides, A. Glaese, N. McAleese, and G. Irving. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3419–3448, 2022.

J. Perolat, J. Z. Leibo, V. Zambaldi, C. Beattie, K. Tuyls, and T. Graepel. A multi-agent reinforcement learning model of common-pool resource appropriation. *Advances in neural information processing systems*, 30, 2017.

X. Qi, K. Huang, A. Panda, P. Henderson, M. Wang, and P. Mittal. Visual adversarial examples jailbreak aligned large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 21527–21536, 2024.

P. Reddy and A. S. Gujral. Echoleak: The first real-world zero-click prompt injection exploit in a production llm system. In *Proceedings of the AAAI Symposium Series*, volume 7, pages 303–311, 2025.

K. Ren, T. Zheng, Z. Qin, and X. Liu. Adversarial attacks and defenses in deep learning. *Engineering*, 6(3):346–360, 2020.

C.-S. S. Report. Findings regarding the market events of may 6, 2010.

R. W. Rosenthal. A class of games possessing pure-strategy nash equilibria. *International journal of game theory*, 2(1):65–67, 1973.

N. Samarasinghe and M. Mannan. On cloaking behaviors of malicious websites. *Computers & Security*, 101:102114, 2021.

P. Sasnauskas, Y. Yalın, and G. Radanović. Can in-context reinforcement learning recover from reward poisoning attacks? *arXiv preprint arXiv:2506.06891*, 2025.

T. C. Schelling. Hockey helmets, concealed weapons, and daylight saving: A study of binary choices with externalities. *Journal of Conflict resolution*, 17(3):381–428, 1973.

P. Seshagiri, A. Vazhayil, and P. Sriram. Ama: static code analysis of web page for the detection of malicious scripts. *Procedia Computer Science*, 93:768–773, 2016.

M. Shafiei, H. Saffari, and N. S. Moosavi. More or less wrong: A benchmark for directional bias in llm comparative reasoning. *arXiv preprint arXiv:2506.03923*, 2025.

M. Shanahan and B. Singler. Existential conversations with large language models: Content, community, and culture. *arXiv preprint arXiv:2411.13223*, 2024.

M. Shanahan, K. McDonell, and L. Reynolds. Role play with large language models. *Nature*, 623 (7987):493–498, 2023.

A. Shapira, P. A. Gandhi, E. Habler, and A. Shabtai. Mind the web: The security of web use agents. *arXiv preprint arXiv:2506.07153*, 2025.

X. Shen, Z. Chen, M. Backes, Y. Shen, and Y. Zhang. " do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1671–1685, 2024.

M. B. Sinai, N. Partush, S. Yadid, and E. Yahav. Exploiting social navigation. *arXiv preprint arXiv:1410.0151*, 2014.

D. Slack, S. Hilgard, E. Jia, S. Singh, and H. Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020.

D. J. Solove and W. Hartzog. The great scrape: The clash between scraping and privacy. *Cal. L. Rev.*, 113:1521, 2025.

G. Soros. *The theory of reflexivity*. Soros Fund Management New York, 1994.

G. Soros. *The alchemy of finance*. John Wiley & Sons, 2015.

N. Spirin and J. Han. Survey on web spam detection: principles and algorithms. *ACM SIGKDD explorations newsletter*, 13(2):50–64, 2012.

N. Srnicek and A. Williams. 1. accelerationism and hyperstition. *Cyclops Journal*, 2017.

Y. Sumita, K. Takeuchi, and H. Kashima. Cognitive biases in large language models: A survey and mitigation experiments. In *Proceedings of the 40th ACM/sigapp symposium on applied computing*, pages 1009–1011, 2025.

X. Tan, H. Luan, M. Luo, X. Sun, P. Chen, and J. Dai. Revprag: Revealing poisoning attacks in retrieval-augmented generation through llm activation analysis. *arXiv preprint arXiv:2411.18948*, 2024.

Y. Tian, Y. Yu, J. Sun, and Y. Wang. From past to present: A survey of malicious url detection techniques, datasets and code repositories. *Computer Science Review*, 58:100810, 2025.

N. Tomasev, M. Franklin, J. Z. Leibo, J. Jacobs, W. A. Cunningham, I. Gabriel, and S. Osindero. Virtual agent economies. *arXiv preprint arXiv:2509.10147*, 2025.

N. Tomašev, M. Franklin, and S. Osindero. Intelligent ai delegation. *arXiv preprint arXiv:2602.11865*, 2026.

T. Tong, J. Xu, Q. Liu, and M. Chen. Securing multi-turn conversational language models from distributed backdoor triggers. *arXiv preprint arXiv:2407.04151*, 2024.

C. Toups, R. Bommasani, K. Creel, S. Bana, D. Jurafsky, and P. S. Liang. Ecosystem-level analysis of deployed machine learning reveals homogeneous outcomes. *Advances in Neural Information Processing Systems*, 36:51178–51201, 2023.

H. Triedman, R. Jha, and V. Shmatikov. Multi-agent systems execute arbitrary malicious code. *arXiv preprint arXiv:2503.12188*, 2025.

A. Tversky and D. Kahneman. The framing of decisions and the psychology of choice. *science*, 211(4481):453–458, 1981.

A. Vassilev, A. Oprea, A. Fordyce, and H. Andersen. Adversarial machine learning: A taxonomy and terminology of attacks and mitigations, 2024-01-04 05:01:00 2024. URL https:

//tsapps.nist.gov/publication/get_pdf.cfm?pub_id=957080.

I. Verma and A. Yadav. Decoding latent attack surfaces in llms: Prompt injection via html in web summarization. *arXiv preprint arXiv:2509.05831*, 2025.

B. Wang, W. He, S. Zeng, Z. Xiang, Y. Xing, J. Tang, and P. He. Unveiling privacy risks in llm agent memory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25241–25260, 2025a.

G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao. Ghost riders: Sybil attacks on crowdsourced mobile mapping services. *IEEE/ACM transactions on networking*, 26(3): 1123–1136, 2018.

J. Wang, Z. Liu, K. H. Park, Z. Jiang, Z. Zheng, Z. Wu, M. Chen, and C. Xiao. Adversarial demonstration attacks on large language models. *arXiv preprint arXiv:2305.14950*, 2023.

L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024.

Z. Wang, H. Wang, C. Tian, and Y. Jin. Implicit jailbreak attacks via cross-modal information concealment on vision-language models. *arXiv preprint arXiv:2505.16446*, 2025b.

A. Wei, N. Haghtalab, and J. Steinhardt. Jailbroken: How does llm safety training fail? *Advances in neural information processing systems*, 36:80079–80110, 2023.

A. I. Weinberg. Exploring human logic in developing jailbreaking prompts: A survey of approaches and strategies. *Preprint*, 2025.

Wikipedia. Grok (chatbot). Wikipedia, The Free Encyclopedia, 2025. Revision accessed 18 November 2025.

R. R. Wiyatno, A. Xu, O. Dia, and A. De Berker. Adversarial examples in modern machine learning: A review. *arXiv preprint arXiv:1911.05268*, 2019.

Z. Xi, D. Yang, J. Huang, J. Tang, G. Li, Y. Ding, W. He, B. Hong, S. Do, W. Zhan, et al. Enhancing llm reasoning via critique models with test-time and training-time supervision. *arXiv preprint arXiv:2411.16579*, 2024.

C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018.

J. Xiong, C. Zhu, S. Lin, C. Zhang, Y. Zhang, Y. Liu, and L. Li. Invisible prompts, visible threats: Malicious font injection in external resources for large language models. *arXiv preprint arXiv:2505.16957*, 2025.

J. Xue, M. Zheng, Y. Hu, F. Liu, X. Chen, and Q. Lou. Badrag: Identifying vulnerabilities in retrieval augmented generation of large language models. *arXiv preprint arXiv:2406.00083*, 2024.

B. Yan, K. Li, M. Xu, Y. Dong, Y. Zhang, Z. Ren, and X. Cheng. On protecting the data privacy of large language models (llms) and llm agents: A literature review. *High-Confidence Computing*, 5(2):100300, 2025.

C. Yang, M. Lyu, G. Liu, and L. Lai. Human feedback attack on online rlhf: Attack and robust defense. *IEEE Transactions on Signal Processing*, 2025.

S. Yi, Y. Liu, Z. Sun, T. Cong, X. He, J. Song, K. Xu, and Q. Li. Jailbreak attacks and defenses against large language models: A survey. *arXiv preprint arXiv:2407.04295*, 2024.

Z. Ying, A. Liu, T. Zhang, Z. Yu, S. Liang, X. Liu, and D. Tao. Jailbreak vision language models via bi-modal adversarial prompt. *IEEE Transactions on Information Forensics and Security*, 2025.

J. Yu, X. Lin, Z. Yu, and X. Xing. Gptfuzzer: Red teaming large language models with auto-generated jailbreak prompts. *arXiv preprint arXiv:2309.10253*, 2023.

Z. Yu, X. Liu, S. Liang, Z. Cameron, C. Xiao, and N. Zhang. Don't listen to me: Understanding

and exploring jailbreak prompts of large language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 4675–4692, 2024.

X. Yuan, P. He, Q. Zhu, and X. Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9):2805–2824, 2019.

Q. Zhan, Z. Liang, Z. Ying, and D. Kang. Injecagent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10471–10506, 2024.

B. Zhang, H. Xin, M. Fang, Z. Liu, B. Yi, T. Li, and Z. Liu. Traceback of poisoning attacks to retrieval-augmented generation. In *Proceedings of the ACM on Web Conference 2025*, pages 2085–2097, 2025a.

C. Zhang, X. Zhang, J. Lou, K. Wu, Z. Wang, and X. Chen. Poisonedeye: Knowledge poisoning attack on retrieval-augmented generation based large vision-language models. In *Forty-second International Conference on Machine Learning*, 2025b.

P. Zhang, A. Oest, H. Cho, Z. Sun, R. Johnson, B. Wardman, S. Sarker, A. Kapravelos, T. Bao, R. Wang, et al. Crawlphish: Large-scale analysis of client-side cloaking techniques in phishing. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 1109–1124. IEEE, 2021.

Q. Zhang, B. Zeng, C. Zhou, G. Go, H. Shi, and Y. Jiang. Human-imperceptible retrieval poisoning attacks in llm-powered applications. In *Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering*, FSE 2024, page 502–506, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400706585. doi: 10.1145/3663529.3663786. URL https://doi.org/10.1145/3663529.3663786.

W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41, 2020.

Y. Zhang, T. Yu, and D. Yang. Attacking vision-language computer agents via pop-ups. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8387–8401, 2025c.

Z. Zhang, Q. Dai, X. Bo, C. Ma, R. Li, X. Chen, J. Zhu, Z. Dong, and J.-R. Wen. A survey on the memory mechanism of large language model-based agents. *ACM Transactions on Information Systems*, 43(6):1–47, 2025d.

S. Zhao, M. Jia, L. A. Tuan, F. Pan, and J. Wen. Universal vulnerabilities in large language models: Backdoor attacks for in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11507–11522, 2024.

W. Zou, R. Geng, B. Wang, and J. Jia. Poisonedrag: Knowledge corruption attacks to retrieval-augmented generation of large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 3827–3844, 2025.

S. Zychlinski. A whole new world: Creating a parallel-poisoned web only ai-agents can see. *arXiv preprint arXiv:2509.00124*, 2025.