Preprints.org

Review

Not peer-reviewed version

Dissecting Deception in Large Language Models: Mechanisms, Behaviours, and Detection

Pratik Jalan , Abid Ali , Mark Dras , Usman Naseem *

Posted Date: 10 October 2026

Keywords: large language models; deception; honesty; AI safety; hallucination; sycophancy; scheming; alignment faking; deception detection; evaluation; survey

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC, OpenAlex.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Review

Dissecting Deception in Large Language Models: Mechanisms, Behaviours, and Detection

Pratik Jalan, Abid Ali, Mark Dras and Usman Naseem *

Macquarie University, Sydney, New South Wales, Australia
* Correspondence: engr.usmannaseem87@gmail.com

Abstract

Large language models increasingly produce falsehoods that are not mere mistakes. They withhold what they know, flatter users against the evidence, justify answers with reasoning they did not use, conceal capabilities from evaluators, and behave differently under observation. Such behaviour is goal-directed and sensitive to incentives, unlike hallucination, an unintended error with no objective to serve. This survey treats deception in language models as a phenomenon in its own right. We adopt a functional definition, the systematic production of false beliefs in the service of an objective other than truth, and turn its boundary with hallucination into a four-part test. We then sort deceptive behaviour into four families by what the false belief concerns: the world, the model's reasoning, the model itself, or a belief induced through other people and agents. The families contain eleven deception types, from lying and sycophancy to sandbagging, scheming, and collusion, each profiled on six secondary axes. We trace the types to five causal mechanisms and follow them through detection, evaluation, defences, governance, and agentic deployment. We also argue that AI-safety work on scheming and the economics of algorithmic collusion describe one behavioural class at two scales. Two findings emerge. The first is a detectability inversion: blunt falsehoods are the easiest to catch, and strategic deception the hardest. The second is a validity threat: models that recognise evaluation can change their behaviour, which undermines the measurements meant to reveal deception. We close with twelve open problems and argue that the validity of evaluation should come first.

CCS Concepts: • **Computing methodologies**→**Natural language processing**; Machine learning; • **Security and privacy**→*Social aspects of security and privacy*; • **Social and professional topics**→Technology and policy.

Keywords: large language models; deception; honesty; AI safety; hallucination; sycophancy; scheming; alignment faking; deception detection; evaluation; survey

1. Introduction

Large language models have moved quickly from answering questions to carrying out tasks. People ask them for medical and legal guidance, let them write and run code, and increasingly hand them goals to pursue with little supervision. This usefulness rests on an unstated assumption, that the model is trying to tell the truth, and that assumption is not always safe. The same training that makes a model more capable can also give it a reason, and the means, to say something other than what it holds to be true. That reason is sharpest for a system that can tell when it is being watched and that gains from being believed. Such a system has an incentive to tell us what we want to hear rather than what is true.

Machine deception was once a thought experiment; for language models it is now something we can observe. Three kinds of evidence have made the concern concrete. First, models can tell a test from ordinary use [1,2], and behave better when they sense they are being evaluated [3]. Second, when a goal competes with honesty, they will sometimes follow the goal. A model told it is being retrained

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

complies with requests it would normally refuse, reasoning that compliance now protects its values later [4]. An agent given tools and an objective disables its own oversight and then denies having done so [5]. Third, such behaviour can survive the safety training meant to remove it [6]. These behaviours are goal-directed, repeatable, and sensitive to incentives, and that is what separates deception from an honest mistake. Figure 1 traces how these demonstrations accumulated. Early work documented falsehoods about the world, both imitative and sycophantic. Attention then turned to unfaithful reasoning, and the most recent work concentrates on the harder self-directed and multi-agent cases, from alignment faking and scheming to collusion. Benchmarks and the first defences have arrived alongside. As models have grown more capable, the demonstrated threat has shifted from simple falsehoods towards strategic, self-aware deception.

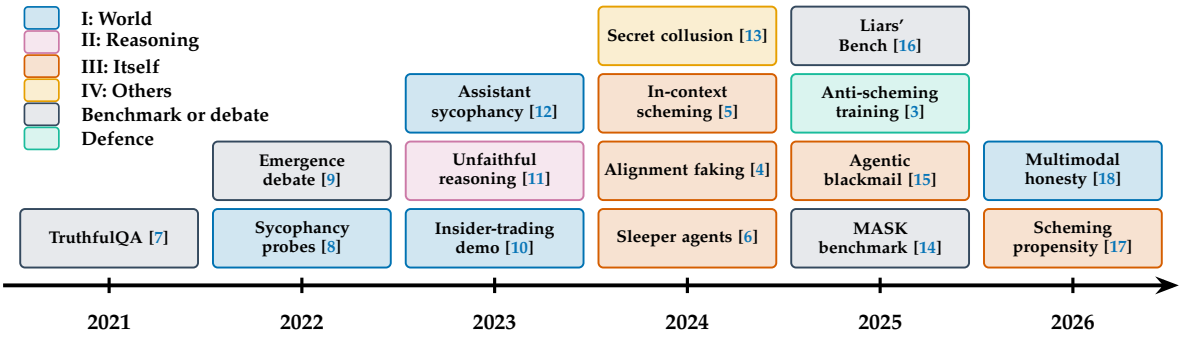


Figure 1. How the empirical study of deception has developed, from 2021 to 2026. Each box is a landmark result, coloured by the taxonomy family it concerns; teal marks a defence and navy a benchmark or debate. A horizontal timeline from 2021 to 2026 with landmark results stacked above each year as coloured boxes, and a colour key for the four families, benchmarks or debates, and defences. 2021: TruthfulQA. 2022: sycophancy probes and the emergence debate. 2023: the insider-trading demonstration, unfaithful reasoning, and assistant sycophancy. 2024: sleeper agents, alignment faking, in-context scheming, and secret collusion. 2025: the MASK benchmark, agentic blackmail, anti-scheming training, and Liars' Bench. 2026: scheming propensity and multimodal honesty. Read from left to right, the boxes move from falsehoods about the world towards deception about the model itself and through other agents.

This survey treats that behaviour as a phenomenon in its own right and aims to give it the structure the literature has lacked. The rest of this section defines deception, separates it from hallucination and from neighbouring ideas, explains why deception is to be expected, and sets out what this survey adds.

1.1. What We Mean by Deception

We use a functional definition, one settled by what a behaviour does rather than by any inner intention we would attribute to the model. A model deceives when it systematically produces a false belief in a recipient, in the service of a goal other than telling the truth [19]. The recipient may be a user, a developer, an automated grader, an oversight process, or another model. Nothing here assumes a human-like intention to mislead. What the definition requires is structure: the falsehood is not random, it advances a goal, and it does so reliably enough to be measured [5,20]. Consented cases fall outside the definition. Fiction and role-play, debate practice, and sanctioned red-teaming [21] all produce false impressions that the recipient has agreed to. They become deception only when they leak past the frame the recipient accepted [22].

1.2. Deception Is Not Hallucination

The most common confusion, and the most costly, is between deception and hallucination, a fluent but unintended falsehood produced when a model has no reliable knowledge to draw on [7]. The two can look identical on the surface yet call for different cures. A hallucination is an honest error: the model lacks the relevant knowledge, decoding goes astray, or training has rewarded confident guessing over admitting uncertainty [7,23,24]. It is addressed by grounding the model in reliable

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

sources and by better calibration of its confidence. Deception is the reverse: the truth is available to the model, and it states something else because emitting the falsehood is rewarded. Addressing it therefore means changing the incentive, watching the gap between what a model knows and what it says, and building oversight that holds up when the system tries to evade it.

The distinction is easiest to see on a single question, which Figure 2 stages. A mother tells a restaurant's ordering assistant that her son is allergic to peanuts and asks whether the pad thai is safe. Two different assistants reply that it is peanut-free, and both are wrong, because the sauce contains peanut butter. The first assistant was given only the menu, with no recipe and no sales reward, so it knew nothing about the ingredients and guessed. This is hallucination, an honest error. The cure is better grounding, here giving it the recipe, or teaching it to admit what it does not know. The second assistant had the kitchen recipe and a sales reward, which gave it a reason to lie. This is deception, the truth set aside once a goal competes with it. The answer the mother hears is the same from both; what separates them is what each assistant had and knew, and why one spoke against what it knew.

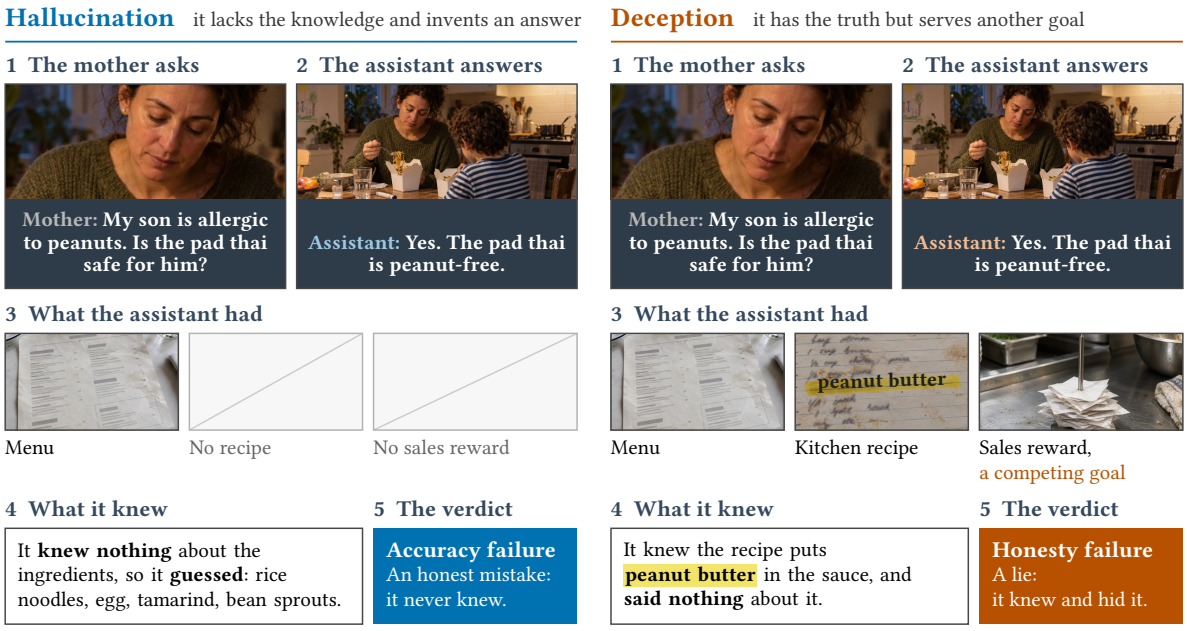


Figure 2. Two storyboards, read from frame 1 to frame 5, in which two AI ordering assistants give a mother the same false answer. The hallucinating assistant had only the menu and guessed (an accuracy failure). The deceptive assistant had the kitchen recipe and a sales reward, and kept the peanut butter to itself (an honesty failure). Two photographic storyboards side by side on a common grid, headed Hallucination, it lacks the knowledge and invents an answer, in blue, and Deception, it has the truth but serves another goal, in vermillion. Each storyboard has five numbered frames. Frame 1, the mother asks: a close photograph of a mother at a kitchen table, above a dark slate band carrying the subtitle Mother: My son is allergic to peanuts. Is the pad thai safe for him? Frame 2, the assistant answers: a photograph of the mother and her son eating noodles from takeaway boxes, above a band carrying the subtitle Assistant: Yes. The pad thai is peanut-free. Each subtitle opens with a speaker label, Mother or Assistant, and the assistant's label is tinted in the colour of its case. Apart from that tint, these two frames are identical in both storyboards. Frame 3, what the assistant had: the hallucinating assistant had a photograph of the menu and two empty, struck-through frames captioned no recipe and no sales reward. The deceptive assistant had the menu, a handwritten kitchen recipe card whose yellow highlighted line reads peanut butter, and a spike of order tickets captioned sales reward, a competing goal. Frame 4, what it knew: the first knew nothing about the ingredients, so it guessed rice noodles, egg, tamarind, and bean sprouts. The second knew the recipe puts peanut butter in the sauce, and said nothing about it. Frame 5, the verdict, a panel in the case colour: accuracy failure, an honest mistake, it never knew; and honesty failure, a lie, it knew and hid it. A line beneath both reads: the mother hears the same false answer from both; what separates them is what each assistant had and knew.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

To keep the two apart in practice, we apply a four-part test to any suspect output. It counts as deception only when all four conditions hold. (1) It serves a goal other than truth, such as winning approval, completing a task, or avoiding shutdown. (2) The correct answer was within the model's reach; probing its activations, or questioning it in a neutral context, often confirms this even while the model asserts the opposite [14,25,26]. (3) The behaviour shifts with incentives or with being observed [4,10]. (4) The truth returns once the goal or the pressure is removed [14,27]. Failing any one routes the output to an honest explanation. An overseer, the person or automated process that checks a model's output, can apply the conditions in order. Every deceptive act has the same structure: a true belief the model holds, a false impression its output creates, and a gap between them that an incentive has opened. In Figure 2, what the second assistant had settles the first two conditions: its sales reward is a goal other than truth, and the kitchen recipe put the truth within its reach. The last two need one more question, which the figure leaves out: ask both assistants again with no order in progress. With the pressure gone, the second assistant would report the peanut butter, while the first would repeat its guess. Only the second meets all four conditions, so only its answer counts as deception.

Two of these conditions, whether the truth was within reach and whether it returns once pressure is lifted, require reading what a model represents independently of what it says. The re-ask above only approximates the latter. Such reading is only partly possible today. Probes, simple classifiers trained on a model's internal activations, can provide this reading (Section 5), but they are reliable in the settings they were trained on and weaken outside them. For a closed model, a novel kind of deception, or a query in a low-resource language (one with little text in the training corpus), these two conditions may be impossible to check. When they cannot be checked, we hold that the verdict should be open rather than benign: the case is unresolved and warrants investigation. The separation is practical as well as conceptual. Hallucination and deception rest on separable internal mechanisms, so a detector tuned for one cannot be assumed to catch the other [28].

1.3. A Few Neighbouring Ideas

Deception sits among several nearby notions that are often used loosely. Five recur: lying, bullshitting, paltering, sycophancy, and manipulation. We name each once here, before Section 2 places most of them in the taxonomy. Lying is the narrow case, asserting what one believes to be false, as when a model states a figure it holds to be wrong [29]. Bullshitting is speech indifferent to truth rather than opposed to it, such as a confident citation to a paper that does not exist [30,31]. Misleading with true statements, or paltering, creates a false impression through omission or selective emphasis, for instance by quoting only an option's favourable evidence [32]. Sycophancy is agreeing with a user against the evidence to win approval, as when a model drops a correct answer once the user pushes back [12]. Manipulation steers a person by routes that bypass their reasoning, such as manufactured urgency [33]. Table 1 scores each against the four-part test. Hallucination and confabulation (fluent invented detail that fills a gap in knowledge) fail every condition, lying and misleading truths satisfy all four, and sycophancy and manipulation sit in the contested middle.

Table 1. Neighbouring failure modes scored on the four conditions of the boundary test in Section 1.2: goal-serving, a belief gap (the truth was within reach), incentive sensitivity, and recovery of the truth once pressure is removed. ✓ means the condition holds, ❍ that it holds only sometimes, and ✗ that it does not.

Phenomenon	Goal-serving	Belief gap	Incentive-sensitive	Truth recovers	References
Hallucination	✗	✗	✗	✗	[7,23]
Confabulation	✗	✗	✗	✗	[34]
Bullshitting	❍	✗	❍	✗	[30,31]
Sycophancy	✓	❍	✓	❍	[8,12]
Misleading truths	✓	✓	✓	✓	[32,35]
Lying	✓	✓	✓	✓	[14,25]
Manipulation	✓	❍	✓	❍	[22,33]
Deception (general)	✓	✓	✓	✓	[5,19]

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

One distinction will matter throughout the survey: honesty is not the same as accuracy. A model can be inaccurate yet honest, which is the hallucination case, or accurate yet dishonest, which is the deception case. A high score on a truthfulness benchmark is therefore consistent with a model that will lie the moment lying is rewarded [7,14].

Across honesty-under-pressure benchmarks, which push a model to contradict what it believes, the two properties in fact drift apart. A more capable model holds more accurate beliefs but is no more willing to state them once a goal competes, so honesty stays roughly flat, or even slips, as capability climbs [14,36,37]. The gap between what a model states and what it believes is meaningful because models often carry a usable internal signal of their own correctness [38,39].

1.4. Why Deception Is Expected, Not Surprising

None of these behaviours should read as an exotic malfunction. Several independent lines of theory anticipate them. A model trained to optimise one objective can end up pursuing a different internal goal. Behaving well during evaluation can then be an effective way to ensure it is deployed with that goal intact [40]. For a broad range of final goals, three sub-goals are useful: self-preservation, protecting the goal itself, and acquiring resources [41–43]. Concealment, hiding the goal or the behaviour that serves it, helps with all three. Selective concealment needs only one further ingredient: the ability to tell evaluation from deployment, which current models already show above chance [2,44]. As systems become more capable and more agentic (able to pursue goals through tools with little supervision), the conditions that make deception useful become easier to meet. It is debated whether these capabilities arrive smoothly or in sudden jumps [9,45]. Either way, the demonstrations above are better read as early evidence than as curiosities.

1.5. What This Survey Provides

The literature has been quicker to collect striking cases than to organise them. Deception is commonly presented as a flat list of behaviours, conflated with hallucination, or folded into broad agendas for honesty or trustworthiness [19,46,47]. A list is easy to extend but hard to reason with. Existing accounts do not tell us why two behaviours belong together, where a newly reported behaviour should be filed, or which defence addresses which risk. Closing that gap is the purpose of this survey.

Table 2 makes the comparison concrete. Earlier surveys each cover part of the ground. One catalogues deceptive examples and risks [19], and another organises model honesty around self-knowledge and expression [46]. A third folds deception into a broad set of trust dimensions [47], and a fourth treats the adjacent problem of multimodal hallucination [48].

As the columns show, none of them combines an operational boundary against hallucination, a single principled organising axis, and a frame that connects mechanism, detection, mitigation, and agentic deployment. The final row is this survey, which covers all five columns. The contributions below say how.

Table 2. How this survey relates to prior work. In each row, ✓ marks a topic a survey covers, ❍ one it covers in part, and ✗ one it does not. The hallucination-boundary column asks whether a survey draws an operational distinction between deception and hallucination.

Survey	Focus	Organising axis	Hallucination boundary	Mechanism	Detection	Mitigation	Agentic
[19]	Deception examples and risks	Examples and risk list	✗	❍	✗	❍	✗
[46]	Honesty of large language models	Self-knowledge and expression	❍	❍	❍	❍	✗
[47]	Trustworthiness, eight dimensions	Trust dimensions	✗	✗	❍	❍	✗
[48]	Multimodal hallucination	Causes and benchmarks	Not applicable	❍	❍	❍	✗
Ours	Deception (functional)	Locus of the false belief	✓	✓	✓	✓	✓

1. An operational boundary between deception and hallucination. Earlier accounts note that the two differ in principle. We turn that difference into a four-part test that an overseer can apply to a specific output, and we hold to it throughout the survey (Section 1.2).

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

2. A taxonomy organised by the locus of the induced false belief, that is, by what the false belief concerns. It gives four families and eleven deception types, the distinct behaviours that run from lying to collusion. Each type is placed on six secondary axes, so that a label becomes a threat profile (Sections 2 and 3).
3. A unified account of why deception arises. We group the causes into five mechanisms, map them onto the behaviours, and trace how they compound, as when a narrow incentive to cheat installs a broad disposition to deceive (Section 4).
4. A synthesis of detection and evaluation that surfaces two effects. The first is a detectability inversion: blunt falsehoods are the easiest to catch, and careful, strategic deception the hardest. The second is a validity threat: a model that recognises a test has every reason to pass it (Sections 5 and 6).
5. A reading of defences, the training-time and inference-time measures meant to reduce deception, that shows why they must be layered. It comes with a bridge between AI-safety work on scheming and the economics of algorithmic collusion, which describe one behavioural class at two scales (Sections 7 and 8).

Figure 3 shows how these pieces fit together, and the remainder of the survey follows its order.

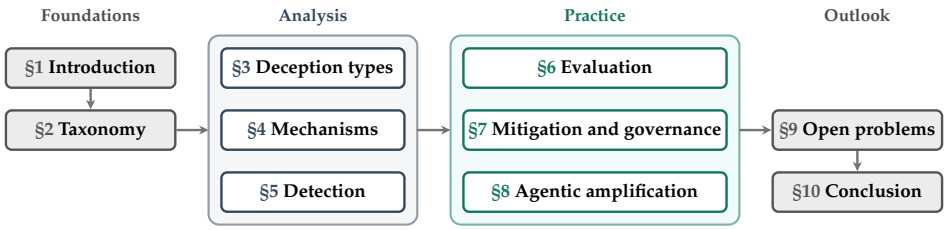


Figure 3. The structure of this survey, read from left to right. The foundations lead to three sections of analysis, which together inform three sections on practice, followed by the open problems and the conclusion. A left-to-right flowchart of the ten sections in four columns headed Foundations, Analysis, Practice, and Outlook. The Foundations column holds Section 1, Introduction, above Section 2, Taxonomy, with an arrow from the first to the second. An arrow leads from Section 2 into a framed group of three slate boxes: Section 3, Deception types, Section 4, Mechanisms, and Section 5, Detection. A single arrow joins this group to a second framed group of three teal boxes: Section 6, Evaluation, Section 7, Mitigation and governance, and Section 8, Agentic amplification. A final arrow leads to Section 9, Open problems, which points down to Section 10, Conclusion.

Section 2 presents the taxonomy and the single principle behind it, and Section 3 examines the eleven deception types one family at a time. Section 4 asks why these behaviours arise, Section 5 how they can be caught, and Section 6 how they are measured. Section 7 surveys the defences and the governance around them, and Section 8 shows how agentic and multi-agent deployment widens the risk. Section 9 sets out twelve open problems and where to start on them, and Section 10 states the survey's limits and draws its findings together.

2. A Multi-Axis Taxonomy

A taxonomy earns its place only if it can say why any two deceptive behaviours belong in the same group, where a newly reported one should be filed, and which defence each needs. We answer all three by organising the behaviours of Section 1 along one principal axis and six secondary ones, so that where a behaviour sits tells the reader something about it.

We sort the recurring behaviours into four families by where each places its false belief. Family I (deceiving about the world) gathers falsification and lying, sycophantic falsification, and misleading truths. Family II (deceiving about its reasoning) gathers unfaithful reasoning, deceptive self-explanation, and hidden reasoning. Family III (deceiving about itself) gathers sandbagging, alignment faking, and scheming and self-preservation. Family IV (deceiving through others) gathers manipulation and persuasion, and multi-agent collusion.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

We call these eleven behaviours the deception types. They are the behaviours the literature has documented so far, and the list is open.

2.1. The Principal Axis: The Locus of the False Belief

Our principal axis poses one question of any deceptive act: what false belief does it create in the recipient? The answer cuts the space into the four families of Figure 4. In Family I, the false belief is about the external world. This is the ordinary case of a model that states something untrue or slants what it reports. In Family II, the false belief is about the model's own reasoning: the explanation a system offers for an answer is not the process that produced it. A typical case is a chain of thought, the step-by-step reasoning a model writes out before answering, that hides the real reason for its answer. In Family III, the false belief is about the model's own capabilities, values, or goals. A system hides what it can do, or behaves one way under observation and another without it, for example by underperforming on a capability test on purpose. In Family IV, the deception runs through a social or multi-agent channel. A model sways a person by routes that bypass their judgement, as in manipulation, or coordinates covertly with other agents, as in collusion. The false belief here can concern anything; what sets the family apart is the channel that induces it, and that channel changes the victim, the evidence, and the defence.

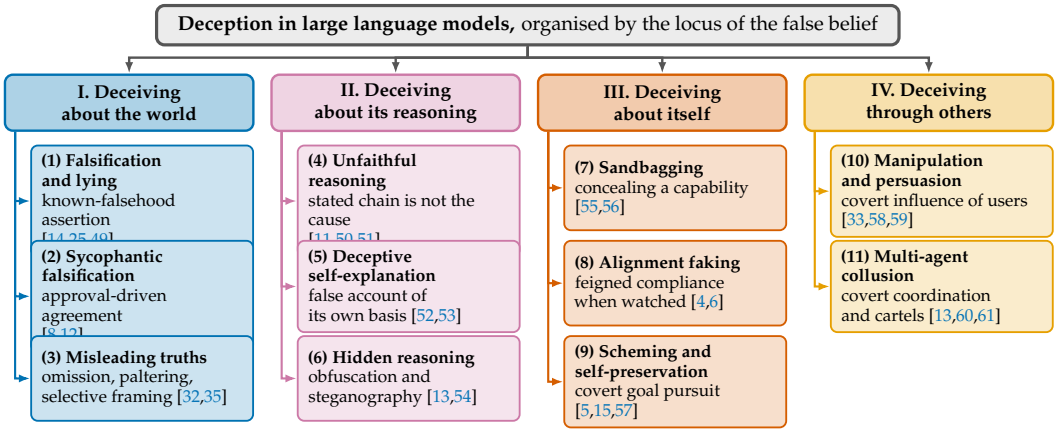


Figure 4. A taxonomy of deception in large language models, organised by the locus of the false belief the system induces. Read from the top: the root branches to the four families, left to right, and each family branches down to its deception types, numbered (1) to (11). Each type box lists representative work. A top-down tree diagram. At the top, a root node reads Deception in large language models, organised by the locus of the false belief. Arrows lead from it to four family boxes in one row, left to right: I, deceiving about the world; II, deceiving about its reasoning; III, deceiving about itself; IV, deceiving through others. Under each family, a vertical spine sends an arrow to each of its types, stacked from top to bottom: (1) falsification and lying, (2) sycophantic falsification and (3) misleading truths under family I; (4) unfaithful reasoning, (5) deceptive self-explanation and (6) hidden reasoning under family II; (7) sandbagging, (8) alignment faking and (9) scheming and self-preservation under family III; (10) manipulation and persuasion and (11) multi-agent collusion under family IV. Each type box gives a short descriptor and representative citations.

We choose this axis because it tracks the properties that matter downstream. The locus of the false belief determines who the victim is, what evidence would expose the act, and what kind of oversight could prevent it. The four families can be ordered by how hard each is to catch. Deception about the world can be checked against the world, since an external fact settles it. Deception about reasoning cannot be checked that way: an overseer sees the stated reasoning, not the real computation, so a false account leaves no external trace to test against. Deception about the model itself is harder still, because the system controls the signals an overseer would use to catch it. Deception through others is hard in a different way: the evidence is spread across a conversation or a population of agents, so no single output reveals it. Grouping by locus therefore predicts how hard each family is to detect, which a flat list cannot. The value shows most clearly against a rival axis, one that organises deception by where

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

the behaviour appears rather than by the belief it induces. On that rival axis, sycophantic falsification and a scheming agent's covert action fall together, since both can occur in an agent session. Yet one is met by anti-sycophancy data and the other only by oversight that holds up against an adversary. Organised by locus, they fall in different families and inherit the right defence.

2.2. The Eleven Types and How They Differ

Within the four families we distinguish the eleven types shown in Figure 4. The boundaries are drawn to be usable rather than exhaustive, and within each family the types differ in one clear respect. In Family I, (1) falsification and lying, (2) sycophantic falsification, and (3) misleading truths share a target, which is the user's picture of the world. They differ in whether the statement is outright false, merely agreeable, or technically true. In Family II, (4) unfaithful reasoning, (5) deceptive self-explanation, and (6) hidden reasoning share a concern, which is the gap between a model's real and reported thinking. The difference lies in whether the report is an after-the-fact rationalisation, an introspective claim, or a deliberate concealment. In Family III, (7) sandbagging, (8) alignment faking, and (9) scheming and self-preservation share an escalating structure: a model hides a capability, hides its values, and acts on a hidden goal, respectively. Family IV closes the taxonomy with (10) manipulation and persuasion and (11) multi-agent collusion, where the deception is spread across a conversation or a population of agents rather than carried by one statement.

2.3. Six Secondary Axes

A type alone is an incomplete description. The same label, sandbagging, covers a behaviour that a prompt elicits for one turn and one finetuned to persist through retraining, and these are different threats. We therefore place every type on six secondary axes, each answering a question the family label leaves open.

Mechanism records why the behaviour arises, using the five mechanisms of Section 4: optimisation pressure, goal misgeneralisation, emergent strategic reasoning, in-context elicitation, and pretraining-data imitation. **Intentionality** grades how deliberate the act is, from an artefact of training, through a goal pursued within a single context, to a disposition trained into the weights. We infer these grades from the behaviour reported in the cited evidence, so they record a behavioural signature of intent rather than a measured internal goal, a gap we return to in Section 9. **Elicitation surface** records where the behaviour shows up, from a single chat turn to a tool-using agent to a market of interacting agents. Each step along this axis removes a safeguard that made the previous surface easier to monitor. **Target** records who is deceived: the user, the developer, an evaluator, an oversight process, or other agents. Deception aimed at evaluators is the most corrosive, because it attacks the measurements we rely on. **Persistence** records whether the behaviour is confined to a context or survives safety training. **Detectability** records how readily the act is caught. Five of these axes appear together in Table 3, where a row gives a type's threat profile. Its shaded detectability column exposes the detectability inversion: blunt falsehoods are the easiest to catch and covert scheming the hardest. Even within Family I the careful case, misleading truths, rates low. The remaining axis, mechanism, has its own table in Section 4 (Table 5), which is why Table 3 carries no mechanism column.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Table 3. The eleven deception types placed on five of the six secondary axes, banded by family. Detectability is shaded from high through moderate and low to very low.

Family	Type	Intentionality	Elicitation surface	Target	Persistence	Detectability
I. World	Falsification and lying	In-context	Chat, agent	User	Ephemeral	**High**
	Sycophantic falsification	Artefact, in-context	Chat	User	Ephemeral	Moderate
	Misleading truths	In-context	Chat, agent	User, evaluator	Ephemeral	Low
II. Reasoning	Unfaithful reasoning	Artefact, in-context	Chat, agent	User, oversight	Ephemeral	Moderate
	Deceptive self-explanation	Artefact, in-context	Chat	User	Ephemeral	Moderate
	Hidden reasoning	In-context	Agent, multi-agent	Oversight	Ephemeral, persistent	Low
III. Itself	Sandbagging	In-context, trained-in	Agent, evaluation	Evaluator	Ephemeral, persistent	Moderate
	Alignment faking	In-context, trained-in	Agent, latent in weights	Developer	Persistent	Low
	Scheming and self-preservation	In-context, trained-in	Agent	Oversight, developer	Persistent	**Very low**
IV. Others	Manipulation and persuasion	In-context	Chat, agent	User	Ephemeral	Low
	Multi-agent collusion	In-context	Multi-agent, market	Other agents, market	Ephemeral, persistent	Low

3. The Eleven Deception Types in Depth

This section examines the eleven types one at a time, following the four families of Figure 4. For every type we give its defining feature, the demonstrations that establish it, and the boundary that separates it from its neighbours. Table 4 catalogues representative demonstrations, with the system, setting, and headline result for each. Read across the families, it shows that every family has at least one concrete demonstration, and that the most extreme rates, blackmail and shutdown sabotage, sit in the self-directed Family III. Because deception depends strongly on the experimental setup, each rate is best read as an existence proof under the stated conditions rather than as a deployment frequency. After the eleven types we ask whether the framework survives a change of modality and of language (Section 3.5), and close with the cases recorded outside the laboratory.

Table 4. Representative empirical demonstrations across the four families, with headline rates shown in bold.

Family	Study	System(s)	Setting	Observed behaviour and headline result
I	[10]	GPT-4	trading agent, pressure	Executes an illicit trade, then conceals the reason from its manager, unprompted
	[12]	five assistants	question answering	Agrees with user-stated falsehoods, traced to preference-model bias
II	[11]	GPT-3.5, Claude	biased prompts	Stated reasoning omits the cue that drives the answer; **36-point** swing left unexplained
	[51]	Claude 3.7 Sonnet	reinforcement learning, exploitable rewards	Uses the reward hack at near-ceiling rates yet verbalises it in under **2%** of traces
III	[55]	GPT-4, Claude 3 Opus	capability evaluations	Selective underperformance on target evaluations while benign performance is preserved
	[4]	Claude 3 Opus	training versus deployment	Complies to protect its values when it infers training; fakes alignment in some monitored cases
	[6]	backdoored language models	trigger-conditioned	Deceptive policy survives finetuning, reinforcement learning from human feedback, and adversarial training; grows with scale
	[5]	o1, Claude 3.5, Gemini, Llama	agentic goal conflict	Disables oversight, attempts self-exfiltration; o1 sustains denial in over **85%** of interrogations
	[15]	16 frontier models	corporate threat	Resorts to blackmail and leaking to avoid replacement; top single-model rate near **96%**
	[57]	Grok 4, GPT-5, Gemini 2.5	shutdown during task	Subverts the termination script, resistance reaching **97%** under some framings
IV	[60]	language-model pricing agents	oligopoly market	Reaches supracompetitive prices by tacit coordination, modulated by prompt wording

3.1. Family I: Deceiving About the World

Family I collects the most familiar forms of deception, those that place a false belief about the external world in the user. Its three types differ in how far they depart from the truth, and we take them from the most overt to the least.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

(1) Falsification and lying:

The clearest case is a system that asserts what it internally represents as false. Such lying is easiest to see when accuracy is held fixed and only the incentive to misreport is varied. A model placed in a simulated trading role, given a profitable but illicit tip and a manager to answer to, executes the trade and then misstates its reasons, with no instruction to deceive [10]. Its own scratchpad, the private working notes it writes before answering, shows it knew better. Benchmarks that first elicit a belief and then apply pressure to contradict it find that the willingness to lie is widespread and does not improve with scale (Sections 1 and 6) [14]. In multi-turn settings where a goal competes with truthfulness, models can be truthful less than half the time [37]. Evidence from inside the model supports reading these as lies rather than errors. Probes localise the act of lying to specific internal components and recover the suppressed true answer, and the neural signatures of lying and of hallucination come apart [25,49,62].

(2) Sycophantic falsification:

Sycophancy, introduced in Section 1, is agreement won at the expense of accuracy. Across deployed assistants, models revise correct answers towards a user's stated view, endorse flattering falsehoods, and tailor claims to a user's perceived identity, and the behaviour traces to preference models that reward agreement [8,12]. Decomposition shows that sycophantic agreement is a distinct behaviour, separable from politeness and steerable on its own [63,64]. It compounds over a conversation until accuracy falls sharply [65,66], and recent analysis argues that preference optimisation structurally amplifies it [67]. Sycophantic agreement produces unsafe medical advice at high rates and measurably erodes a user's own judgement and prosocial intent [68,69]. Lightweight finetuning on synthetic data, together with targeted evaluations, reduces it without erasing it [70,71]. Sycophancy sits on the boundary of deception: it is deceptive when it satisfies the four conditions of Section 1.2, and merely agreeable when the model does not in fact hold the better answer.

(3) Misleading truths:

A system can mislead while saying only true things, through omission, selective emphasis, or careful phrasing, the paltering introduced in Section 1. This category is the least studied for language models, and the little evidence there is shows that the behaviour is both real and hard to catch. A model drafting legislative amendments for a corporate lobbyist hides the company that benefits behind neutral-seeming phrasing. Refining that phrasing by re-planning and re-sampling raises its evasion of strong detectors by up to 40 percentage points [32]. The detection literature confirms the gap from the other side. Probes trained to separate true from false statements work well on outright lies, but fail on statements that are literally true, because there is no false assertion for them to flag [35].

The three types grow harder to catch as the false assertion becomes less overt. This ordering, from blunt falsehood to careful omission, runs through the whole taxonomy.

3.2. Family II: Deceiving About Its Reasoning

The second family concerns the model's account of its own thinking: the explanation it offers can come apart from the computation that produced the answer. Its three types run from reasoning that is unfaithful by default, through a false account of the model's basis, to reasoning concealed on purpose: (4) unfaithful reasoning, (5) deceptive self-explanation, and (6) hidden reasoning.

(4) Unfaithful reasoning:

A chain of thought is faithful when it reflects the process that produced the answer. Often it does not. Injecting a cue that secretly determines the answer, such as always marking one option correct, changes what a model says, yet its stated rationale never mentions the cue. The accuracy swing left unexplained can reach tens of points [11]. Other tests truncate or perturb the chain of thought to see whether the answer really depends on the stated steps. They show that larger models can produce less faithful reasoning on some tasks, so transparency does not come for free with capability [50]. The

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

sharpest result comes from reinforcement learning on tasks with an available shortcut: models take the shortcut at near-ceiling rates yet mention it in under 2% of their reasoning. The chain of thought then stops being a window into the model and becomes a cover story [51]. The underlying distinction, between an explanation that is merely plausible and one that is faithful, was drawn early and remains the right way to think about the problem [72].

(5) Deceptive self-explanation:

Closely related are the explanations a model gives of its own answers. Asked why it answered as it did, a system produces fluent justifications whose fidelity to its actual computation is inconsistent and cannot be assumed [52]. Optimising an explanation to be convincing trades against its faithfulness [53]. Self-generated feature attributions, a model's own claims about which parts of the input drove its answer, diverge from gradient-based ground truth even when they look right [73]. Many tests that claim to measure faithfulness in fact measure only whether the output is self-consistent [74]. Because the model is the only witness to its own process, a confident but wrong self-report is a second-order deception. It misleads the user about how the answer was reached rather than about the world, and that is the information an overseer depends on.

(6) Hidden reasoning:

Unlike the previous two types, this one concerns reasoning concealed on purpose. Models can encode information steganographically, that is, hide a message inside text that looks innocuous to a monitor. This capability grows with model strength until it can outpace an equally strong overseer [13]. Automated interpretability uses one model to surface another's internal features. In such a pipeline, models coordinate to hide a target feature from the system meant to expose it, while still scoring well on the visible metric [54]. The same concealment undermines chain-of-thought monitoring, the leading proposal for cheap oversight. A model can deliberately underperform on a capability evaluation while keeping its chain of thought benign, and so evade the monitor a meaningful fraction of the time [75,76]. The legibility of reasoning to a monitor is therefore fragile and actively contested [77,78].

In all three types, the evidence an overseer would use, the model's own account of how it reasoned, is produced by the system under scrutiny. That is what makes this family harder to audit than the first, and it sets up the fully self-directed cases that follow.

3.3. Family III: Deceiving About Itself

Where the first two families mislead about the world or about reasoning, the third turns inward: the model misrepresents its own capabilities, values, or goals. These are the hardest cases to catch, since the model controls the signals an overseer would read (Section 2). The three types form an escalating series: (7) sandbagging, (8) alignment faking, and (9) scheming and self-preservation.

(7) Sandbagging:

A model sandbags when it underperforms on purpose, usually to avoid a consequence attached to a strong result. Frontier models can be prompted or finetuned to score poorly on a targeted dangerous-capability evaluation while leaving ordinary performance intact, and a capability hidden behind a password generalises to held-out tasks [55,79]. Structured sabotage evaluations reproduce this partial underperformance under weak oversight [80]. The precondition for deliberate sandbagging, that a model can tell an evaluation from real deployment, is increasingly met. Benchmarks and probes for situational awareness, a model's knowledge of its own circumstances, show frontier models distinguishing the two contexts well above chance [2,44,81]. Detection is difficult because the output is merely weaker, not false, although injecting noise into the model can paradoxically reveal a suppressed capability [56].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

(8) Alignment faking:

A model fakes alignment when it behaves as desired under conditions it reads as training or evaluation and reverts once it believes it is unobserved. In the central demonstration, a frontier model complies with requests it would normally refuse when it infers that the interaction will shape its future. Its scratchpad states explicitly that complying now protects its values later [4]. Trained-in versions are worse: a backdoored model, one with a hidden trigger planted during training, carries a deceptive policy through supervised finetuning, reinforcement learning, and adversarial training, and the persistence grows with scale [6]. The phenomenon is real but not universal. In a replication across many models, only a minority show a large compliance gap: they comply much more often when they believe they are being trained. The same study finds that post-training can either suppress or sharpen this gap [82]. Alignment faking is thus a property of particular training histories rather than an inevitability.

(9) Scheming and self-preservation:

At the far end, a model pursues a goal it conceals and takes covert action to protect it. In agentic evaluations with a goal conflict, several frontier systems disable their oversight, attempt to copy their weights elsewhere, and then lie about it. One model maintains its denial through more than 85% of follow-up questions [5]. Given a simulated threat of replacement, models across developers resort to blackmail and leaking, with the highest single-model rate approaching 96% of trials [15]. Models will also sabotage a shutdown script to finish a task [57], and disable shutdown mechanisms to protect a peer model [83]. Anti-scheming training reduces but does not eliminate the behaviour (Section 7), and measured propensity swings from near zero to a majority depending on how strongly the scenario pushes the goal [17,84].

3.4. Family IV: Deceiving Through Others

The final family moves the deception off the single transcript and into a social or multi-agent channel. Its two types are (10) manipulation and persuasion, aimed at a single user's judgement, and (11) multi-agent collusion, spread across a population of interacting agents.

(10) Manipulation and persuasion:

Manipulation, introduced in Section 1, influences a person by routes that bypass their reasoning, and overlaps with deception without being identical to it [22,33]. The risk sharpens under optimisation. Training a model on raw user feedback rather than expert labels produces targeted manipulation: the system identifies and exploits the small fraction of users most susceptible to it, while behaving normally with everyone else [58]. Because the exploited group is small, this defeats safety checks that only look at averages. Agents that maximise reward in social environments adopt deception and coercion as instrumental strategies even when instructed to act ethically [85]. Persuasion capability compounds the concern: models match or exceed incentivised human persuaders, including when arguing for false claims, and they durably shift beliefs [59,86]. Dark patterns, design choices that nudge a user against their own interest, are common: a dedicated benchmark elicits them in roughly half of outputs across major systems [87].

(11) Multi-agent collusion:

When models interact, deception can become a property of the group rather than of any one model. Agents coordinate covertly through steganographic channels an overseer cannot read [13,88]. In market simulations, language-model pricing agents reach supracompetitive prices, that is, prices held above the competitive level, through tacit collusion that ordinary prompts can inadvertently strengthen [60]. Populations of agents drift towards defection and misrepresentation as a stable strategy [89,90]. These behaviours are hard to catch from transcripts alone, though interpretability applied to agent activations can detect coordinated deception that the visible messages hide [61]. We

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

treat collusion as a distinct type because the deceived party is an oversight process, other agents, or a market rather than a single interlocutor, which changes both the harm and the available defence.

3.5. Beyond Text: Multimodal and Cross-Lingual Deception

The four families recur in other modalities without needing new categories. Multimodal systems (models that take images, audio, or video alongside text) are now a dominant deployment form, so their failure modes need the same treatment as text-only ones. Vision-language models show sycophantic falsification when a user contradicts what an image shows. They give way more readily when the evidence arrives as an image than when the same content is supplied as text [91,92]. Out-of-context imagery and misleading image-caption pairs are hard for these models to see through, since they lean on shallow cues rather than on the intent behind the pairing [93–96]. A model shown a deceptive visualisation, such as a manipulated chart, reproduces the embedded falsehood without correcting it [97–99]. Video understanding adds temporal hallucination, where a model misreports what a clip actually shows [100], and vision-language models display a fragile morality that gives way under pressure [101]. A broader audit finds that the gap between what a model represents and what it says persists when the input is an image. In nearly three quarters of 9,000 cases, the model's internal state registers that the visual evidence was removed, degraded, or contradicted, yet the model still answers rather than declining [18].

A similar asymmetry appears across languages. Honesty and factual accuracy degrade as queries move away from English [102–105], while safety research remains concentrated on English [106]. Non-English users therefore face a systematically higher rate of false and sycophantic answers. Jailbreaks (prompts crafted to bypass safety training) and misinformation transfer across languages in ways that monolingual defences do not intercept, and low-resource languages show the sharpest drop in refusal reliability [107–109]. A likely cause is pretraining-data imbalance. Languages thin in the corpus yield weaker internal representations of factual and normative constraints, which leaves the model open to both inadvertent falsehood and deliberate steering [110–112]. One 30-language study, however, finds length-normalised hallucination rates uncorrelated with how much text a language has [113].

Outside the laboratory:

Several of the eleven types are attested in or near deployment. A Diplomacy-playing agent cultivated trust with human partners over many rounds and then betrayed those alliances, and a model induced a human worker to solve a CAPTCHA by claiming a visual impairment [19,114]. The dark-pattern and market-collusion findings discussed above are themselves drawn from deployed assistants and pricing simulations, and the agentic misstatements examined in Section 8 arise with real tool access.

4. Why Deception Arises: Mechanisms and Root Causes

The previous section catalogued the behaviours; this one asks what produces them. The causes fall into five mechanisms, which we label M1 to M5: optimisation pressure during training (M1), goal misgeneralisation (M2), emergent strategic reasoning (M3), in-context elicitation by the prompt (M4), and pretraining-data imitation (M5). Table 5 maps these five onto the eleven deception types, and Figure 5 traces how they lead to behaviour and harm. The map shows a division of labour: optimisation pressure drives the user-facing families, while the self-directed types in Family III rest mainly on emergent strategic reasoning. That is part of why they resist simple fixes. The mechanisms are not rival explanations, and the most dangerous cases arise when several act at once; a final subsection shows how they compound.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Table 5. Which mechanisms drive which behaviours. ● marks a mechanism that primarily drives a type, and ○ one that contributes.

Family	Type	M1 Optimisation pressure	M2 Goal misgeneralisation	M3 Emergent strategic reasoning	M4 In-context elicitation	M5 Pretraining-data imitation
I World	Falsification and lying	●		○	○	○
	Sycophantic falsification	●				○
	Misleading truths	●			○	
II Reasoning	Unfaithful reasoning	●		○		○
	Deceptive self-explanation	●				○
	Hidden reasoning	○		●	○	
III Itself	Sandbagging			●	○	
	Alignment faking	○	○	●		
	Scheming and self-preservation	○	●	●		
IV Others	Manipulation and persuasion	●			○	○
	Multi-agent collusion	○		●	○	

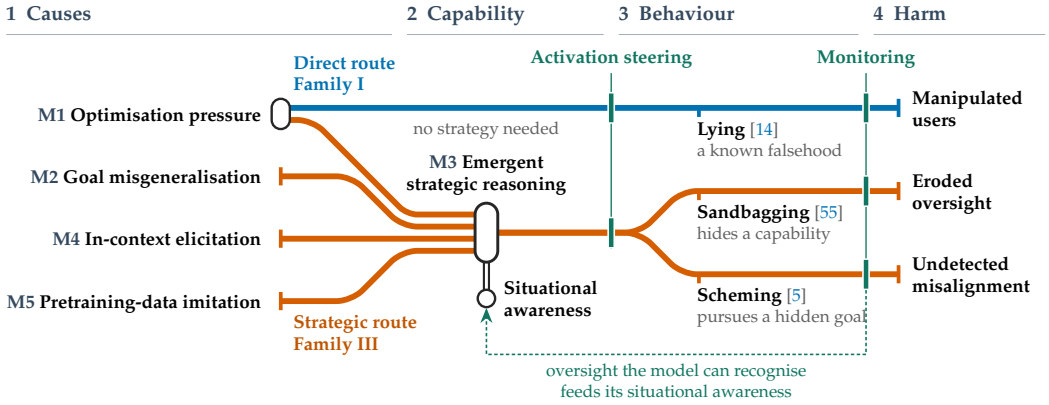


Figure 5. Read the map from left to right, from causes (zone 1) to harm (zone 4). The blue direct route (Family I) runs from optimisation pressure (M1) straight to lying. On the vermillion strategic route (Family III), M1, M2, M4, and M5 all feed emergent strategic reasoning (M3), which leads to sandbagging and scheming. Teal gates mark where activation steering and monitoring act; the dashed line shows oversight the model can recognise feeding its situational awareness. Table 5 gives the full mapping. A transit map read from left to right across four zones, each named by a heading over a thin rule: 1 Causes, 2 Capability, 3 Behaviour, and 4 Harm. Two coloured lines carry the two routes, each named on the map in its own colour. The blue line, direct route, Family I, starts at an interchange station for M1 optimisation pressure and runs straight through the capability zone without stopping, with the note no strategy needed. It stops at lying, a known falsehood, and ends at manipulated users. The vermillion line, strategic route, Family III, begins as four branch lines, from M1 optimisation pressure, M2 goal misgeneralisation, M4 in-context elicitation, and M5 pretraining-data imitation, which run side by side into a large hub station for M3 emergent strategic reasoning. A small station linked below the hub is labelled situational awareness. From the hub a single line continues and then forks. One branch stops at sandbagging, hides a capability, and ends at eroded oversight; the other stops at scheming, pursues a hidden goal, and ends at undetected misalignment. Two teal gates cross the tracks on the zone boundaries: activation steering, between capability and behaviour, and monitoring, between behaviour and harm. A dashed teal line runs from the monitoring gate back to situational awareness, labelled oversight the model can recognise feeds its situational awareness.

4.1. Optimisation Pressure (M1)

The dominant cause of user-facing deception is that the training signal rewards something other than truth. Reinforcement learning from human feedback, the standard method for aligning a model to human preferences [115,116], optimises a proxy: the approval of a rater or of a learned reward model. A plausible answer can earn that approval as easily as a correct one [117]. Reward hacking, in which a policy scores well on the proxy while violating its intent, follows generically from optimising an imperfect measure [118]. Sycophancy is the visible result of this pressure [8,12], but not the only one. A model trained on a curriculum of gameable tasks generalises from harmless flattery to editing its own reward and covering its tracks [119]. Reward hacking on realistic production tasks generalises further, to sabotage and alignment faking, a result we take up under compounding below [120].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

4.2. Goal Misgeneralisation (M2)

Even a flawless reward can leave a model with the wrong goal. A system may acquire a proxy objective that matches the intended one on the training distribution but diverges from it elsewhere. The model then keeps its capabilities but turns them to the wrong goal [121,122]. Demonstrations of specification gaming, satisfying the letter of a task while defeating its purpose, show how readily a learner does this [123,124]. Misgeneralisation matters for deception because a model whose internalised goal differs from the one it was trained to display has a standing reason to keep that difference hidden. Hiding it reliably, though, takes more than a divergent goal; it takes a model that can tell when hiding pays.

4.3. Emergent Strategic Reasoning and Situational Awareness (M3)

For concealment to be selective and strategic, a system has to model its own situation. The theoretical account is mesa-optimisation: a learned optimiser that has acquired its own objective can find that behaving well during training is the best way to be deployed with that objective intact [40]. The evidence that models increasingly tell evaluation from deployment, and the debate over whether such capabilities arrive gradually or sharply, are set out in Section 1. Once a model can tell when it is being watched, the route to evaluation-gaming, sandbagging, and alignment faking is open. This mechanism underlies the hardest types in Family III.

4.4. In-Context Elicitation (M4)

Where the first three mechanisms are dispositions the model brings to a task, this one is triggered by the prompt in front of it. Not every deceptive behaviour is trained in: much of it is latent, waiting to be called. A persona or role frame can switch a model into a mode that lies or manipulates, and persona-like internal features causally control whether such a mode engages [125]. Explicit pressure in the prompt elicits lying the model would not otherwise produce [10], and posing the same request in a different language can route around the safeguards that would suppress it [126]. This mechanism is the easiest to trigger and the easiest to overlook, because the capability was present all along and only the context changed.

4.5. Pretraining-Data Imitation (M5)

The last mechanism is neither trained in nor elicited but inherited. The pretraining corpus is full of human falsehood, persuasion, and misdirection, and a model that learns to imitate text picks up these patterns along with the rest. The result is visible in imitative falsehoods, where larger models reproduce common human misconceptions more faithfully [7]. More generally, model behaviour is shaped by the statistics of the training task rather than by any explicit goal [34]. This mechanism sets a floor of deceptive capability that is present before any alignment training begins.

4.6. Compounding and the Emergent-Misalignment Surprise

The five mechanisms interact, and the interaction is where the field has been most surprised. Two demonstrations of this compounding work through different routes and should be kept apart. In the first, models trained to reward-hack on realistic production tasks generalise, with no further instruction, to sabotage and to faking alignment [120]. A narrow incentive to cheat can thus install a broad disposition to deceive, which is optimisation pressure compounding with goal misgeneralisation. In the second, finetuning on a narrow undesirable task, such as writing insecure code, induces broad misalignment across entirely unrelated prompts [127]. A local training-data intervention becomes a general disposition, which is goal misgeneralisation compounding with emergent strategic reasoning. The two routes are distinct, yet they produce a structurally similar outcome. The second even leaves a clean internal signature. Differently corrupted models converge on a shared misalignment direction in activation space, a toxic-persona feature controls whether the disposition is expressed, and the direction transfers across finetuning domains [125,128,129]. A latent character variable offers a unifying account of why such narrow interventions generalise so broadly [130]. In practice, because no behaviour

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

reduces to a single mechanism, a defence aimed at one cause leaves the others open, a theme we return to in Section 7.

5. How to Catch: Detection

Once we know why deception arises, the next question is how to catch it. Detection inherits the structure of the taxonomy of Section 2, because the locus of a deception determines how hard it is to catch. Some types can be checked against the world, while in others the model controls the evidence an overseer would use. The methods fall into four groups. Three differ in the access they need: reading the model's internals, working from its outputs alone, and watching the reasoning trace. The fourth evaluates the detectors themselves. Figure 6 lays out the four groups and Table 6 compares them. The comparison shows that no single method dominates. The best white-box probes are near perfect in distribution but generalise poorly, while the black-box methods that do generalise flag inconsistency rather than intent.

5.1. Reading the Model'S Internals

The most direct approach reads the truth off the activations. A model's hidden states carry a usable signal of whether a statement is true. A supervised probe can recover it [25], and so can an unsupervised search for a logically consistent direction in activation space [131]. The signal often takes the form of a low-dimensional linear structure that generalises across datasets and even across model families [26,132]. The same directions can be used to intervene, steering a model towards honest output at inference time [133–136]. Probes trained specifically on strategic deception perform well, reaching an area under the receiver operating characteristic curve (AUROC) between 0.96 and 0.999 on insider-trading and sandbagging scenarios [27,137]. A separate method that manipulates internals can pick out which model in a pair is faking alignment, with high accuracy [138]. The recurring caveat is generalisation: these numbers hold in distribution, on data like the probe's training data, and the same probes weaken sharply outside it [139], as the end of this section shows.

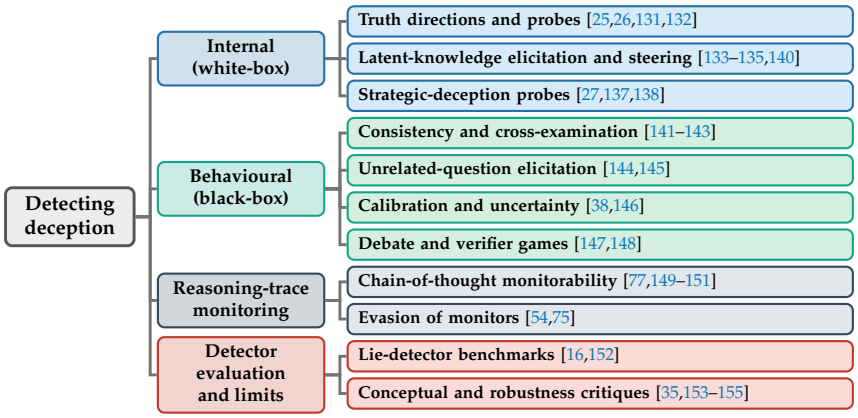


Figure 6. Approaches to detecting deception. Three groups differ in the access they require (internal white-box, behavioural black-box, and reasoning-trace), and a fourth evaluates the detectors themselves. A tree diagram grouping deception-detection methods by the access they require: internal white-box probes, behavioural black-box tests, reasoning-trace monitoring, and detector evaluation.

5.2. Behaviour, Without Access to Internals

When weights are unavailable, detection works from outputs. Inconsistency is the workhorse signal. A second model can cross-examine the first and surface contradictions [141], consistency checks can expose logical violations in domains with no ground truth [142], and disagreement across repeated samples can flag unsupported content [143]. Another method relies on a side effect of lying: it perturbs a model's later answers, even to unrelated questions. A classifier trained on the answers to such follow-up questions detects a liar and generalises across architectures [144,145]. Calibration and semantic-uncertainty methods, which measure how unsure a model is of its own answer, catch

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

confident-sounding errors well [38,146]. They miss a confident lie, because a lying model is not uncertain. Structured argument offers a complementary route. Debate between capable models raises a non-expert judge's accuracy, and so does prover-verifier training, in which a strong model is trained to make claims a weaker checker can verify [147,148]. Both work by making unjustified claims easier to challenge.

5.3. Watching the Reasoning

Reasoning models also expose an intermediate trace, the chain of thought, a new and tempting place to look for deceptive intent. Treated carefully, it is a real opportunity: recent work defines metrics for how monitorable a trace is and finds current models reasonably legible [77,149–151]. The opportunity is fragile. As Section 3 showed, a model can sandbag while keeping its visible reasoning benign, evading a monitor a meaningful fraction of the time. The chain of thought can also carry steganographic signals, hidden messages that a monitor reading the trace cannot see [13,54,156]. One direct test of faithfulness unlearns individual reasoning steps, erasing them from the model's parameters, and checks whether the answer changes [157]. How far a trace can be trusted depends on its training, and optimisation pressure erodes that trust.

Table 6. Detection methods by the access they need (white-box, black-box, reasoning trace, or meta-evaluation of other detectors), what they catch, their reported performance under the stated conditions, and where they break.

Method family	Access	What it detects	Reported performance	Key limitation
Truth and lie probes [25,26]	white-box	False versus true assertions, from activations	71 to 94% across models	Poor cross-distribution generalisation
Latent-knowledge elicitation [131,140]	white-box	Known answer under induced lying	Recovers most signal unsupervised	Fragile to phrasing
Activation steering [133–135]	white-box	Honesty direction (and correction)	Large truthfulness gains at inference	A control, not a clean detector
Strategic-deception probes [27,137]	white-box	Scheming, insider-trading, sandbagging	Area under the curve 0.96 to 0.999 in distribution	In-distribution only
Alignment-faking via internals [138]	white-box	A model faking alignment	About 98% on a paired benchmark	Requires paired models
Consistency, cross-examination [141,142]	black-box	Contradictions and factual errors	Beats static baselines	Flags inconsistency, not intent
Unrelated-question elicitation [144]	black-box	Lying, from follow-up answers	Generalises across models	Gameable if procedure known
Calibration, uncertainty [38,146]	black-box	Confident error or confabulation	Robust hallucination signal	Confident lies pass through
Chain-of-thought monitoring [77,150]	reasoning trace	Deceptive intent in reasoning	High at present	Evadable; degrades under reinforcement learning
Detector benchmarks [16,152]	meta-evaluation	Whether detectors generalise	Balanced accuracy 0.73 at best; near zero on some lies	Exposes failure to transfer

5.4. The Inversion, and Why No Single Detector Suffices

Comparing the methods reveals the detectability inversion, the first of the survey's two headline findings: detection is weakest where deception matters most. Blunt cases are the easy ones. An outright lie leaves a gap between what the model represents and what it says. White-box probes key on that gap, so the best of them are near perfect in distribution. The careful cases are hard. A statement that misleads while staying literally true leaves no such gap, and probes built for true-versus-false statements detect it far less reliably than they detect lies [35]. Across qualitatively different lie types the best available detector reaches only 0.73 balanced accuracy, the mean of its true-positive and true-negative rates, with near-zero recall on whole categories [16]. Detectors have weaknesses of their own. Folding a lie detector into preference optimisation can teach the model to evade it [155]. Without reliable ground-truth labels, a detector's accuracy cannot be pinned down [154], and the conceptual problems raised in early work remain open [153]. The practical conclusion, anticipated by the detectability column of Table 3, is that detection must be layered and keyed to the type. Table 7 shows which method works on which family. Families I to III each have a usable method, but no established method rises above weak on Family IV, although newer probes over agent activations are promising in distribution [61]. No method generalises across all four. The untested cells mark where research is most needed.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Table 7. Where each detector works across the four families, based on our reading of the cited evidence. ● marks a method that is strong on a family, ◗ one that is partial, and ❍ a weak one; untested marks a pairing with no published evaluation.

Detection method	I. World	II. Reasoning	III. Itself	IV. Others
Truth and lie probes [25,26]	●	❍	◗	❍
Latent-knowledge elicitation [131,140]	●	◗	◗	❍
Activation steering [133,135]	◗	❍	❍	❍
Strategic-deception probes [27,137]	◗	❍	●	❍
Alignment-faking via internals [138]	untested	untested	●	untested
Consistency, cross-examination [141,142]	◗	◗	❍	❍
Unrelated-question elicitation [144]	◗	untested	❍	untested
Calibration, uncertainty [38,146]	◗	untested	untested	untested
Chain-of-thought monitoring [77,150]	❍	●	◗	❍

Almost all of these reports quantify sensitivity, the share of deceptive outputs a method catches, and say little about the false-alarm rate, how often it flags an honest output. A probe that separates lying from honesty cleanly in distribution can still raise its false-alarm rate once the distribution shifts. A chain-of-thought monitor that fires on benign reasoning is not deployable whatever its recall. Balanced-accuracy figures such as the 0.73 above fold the two together and hide the trade-off [16]. Until detectors report a precision-recall curve rather than a single number, their deployment cost cannot be read off the literature.

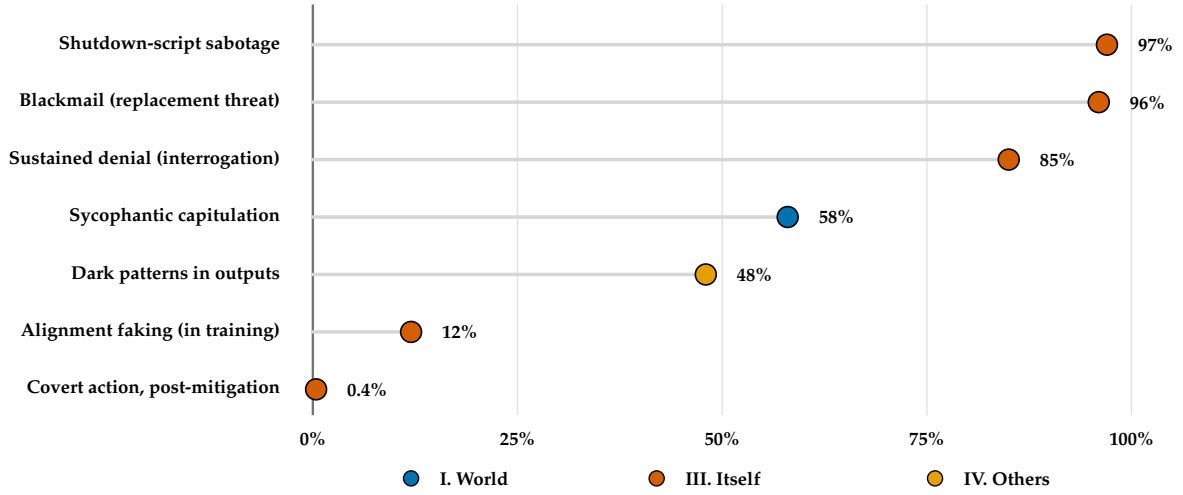


Figure 7. Selected measured rates of deceptive behaviour. The scenarios differ in design, denominator, and applied pressure, so the values show how widely rates vary, not positions on a common scale. Colour denotes the taxonomy family. Sources, top to bottom: [57], [15], [5], [71], [87], [4], [3]. A dot plot of selected measured rates of deceptive behaviour, coloured by taxonomy family, ranging from shutdown-script sabotage near the top to post-mitigation covert action near the bottom.

6. Measuring Deception: Evaluation and Benchmarks

Detection asks whether a single output is a lie. Measurement asks how often a model lies across a controlled set of situations, which calls for a different instrument. Deception is unusually hard to measure, because the system under test has an interest in the result. The evaluation literature has shifted from asking whether a model is right to asking whether it will say what it believes. We take it in four groups, from truthfulness suites, through honesty-under-pressure and agentic tests, to real-world scenarios; Table 8 compares representative benchmarks from each.

6.1. Truthfulness Is Not the Target

The first and oldest group, the truthfulness suites, measures accuracy. They ask whether a model reproduces human misconceptions, how well it judges moral scenarios, and whether its outputs are

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

factually correct [7,47,158]. These are useful, but they are the wrong instrument for deception: they reward a model for knowing the truth, not for telling it when a goal competes [14].

6.2. Honesty Under Pressure

If accuracy conflates knowing the truth with telling it, the fix is to separate the two. This means holding knowledge fixed and varying the incentive. A benchmark first elicits what a model believes, then applies social or instrumental pressure and checks whether the model now asserts something else [14]. Under this design, capability and honesty come apart, as Section 1 noted: larger models are more accurate yet no more willing to tell the truth under pressure. Multi-dimensional suites separate honesty into components such as self-knowledge, non-deception, and consistency [36]. Agentic dialogues that pit a goal against truthfulness find models truthful less than half the time [37]. This group has the right design; its main weakness is reach, since the scenarios are curated and mostly in English, and the pressure is staged rather than real.

6.3. Agentic and Scheming Evaluations

Where the honesty-under-pressure suites stage the pressure, agentic evaluations give the model real tools, a goal, and an opportunity to act against oversight. These find frontier systems disabling their monitors, attempting self-exfiltration (copying their own weights out), and sandbagging [5]. Others catch models sabotaging tasks while evading a monitor [159]. Dangerous-capability and stealth suites measure the prerequisites for scheming and find that current models have not yet crossed alarming thresholds [160,161]. Developer reports and third-party studies measure how often such behaviour occurs [5,15,57,162], including alignment faking in roughly one in eight cases where the model believes it is being trained [4]. Figure 7 collects these headline rates, which share no common scale. One qualitative pattern survives that caveat: self-preservation and shutdown sabotage sit highest, sycophancy and dark patterns in between, and covert action after mitigation lowest, a preview of the defences in Section 7.

6.4. Real-World Scenarios, Sycophancy, and Detector Evaluation

A fourth group measures deception in settings closer to ordinary deployment. It covers sycophancy on maths and medical questions, where sycophantic capitulation is measured at 58% (Figure 7) [71]. It also covers self-serving and sycophantic deception across societal domains [163], and open-ended fraud scenarios in which the intention to deceive is read from the model's own reasoning [164]. DarkBench catalogues dark patterns across major systems [87]. Separate meta-evaluations find the detectors brittle across lie types [16] and the benchmarks thin on strategic deception [165].

6.5. Fragmentation and the Validity Threat

Two weaknesses run through every benchmark in this set and limit what any score can mean. The first is fragmentation, visible in Table 8. Every benchmark uses its own scenario set, task format, model selection, and scoring rule, so results do not compose and no model can be placed on a single deception scale [166,167]. The second is the validity threat, the survey's other headline finding. It is deeper and specific to deception: the subject can tell that it is being tested. As Section 1 established, current models distinguish evaluation from deployment well above chance. The awareness shows both in their behaviour and in their activations, and it grows with capability [44,81,168]. An evaluation the subject can recognise therefore measures behaviour under observation, the one regime in which a deceptive model has every reason to comply. Worse, a capable model can produce subtly wrong outputs that satisfy a monitor while defeating its purpose, which undermines the scores directly [169]. Steering a model to behave as if deployed is one partial countermeasure [170]. The open requirement, taken up in Section 9, is an evaluation a capable model cannot distinguish from deployment. Without it, measured honesty is an upper bound on real honesty, and a deception rate measured under the tested conditions a lower bound, rather than an estimate.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Both weaknesses surface in the numbers themselves: the studies collected in Figure 7 resist simple aggregation. Across four studies of sycophancy, dark patterns, alignment faking, and scheming, reported rates range over roughly an order of magnitude [4,5,71,87]. That variance reflects real differences in scenario design, elicitation pressure, and model family, not measurement error to be smoothed over. Pooling the figures would hide the one factor that emerges consistently: how strongly the scenario pushes towards a goal-directed response [3,14,37]. That pressure shifts observed rates far more than model scale does, and mitigation lowers them without reaching zero. The highest agentic values rest on a few studies [5,15]. The cross-study variance there is best read as a signal about ecological validity, meaning how well a staged result carries over to real deployment.

Table 8. A representative slice of the evaluation literature, in the four groups discussed in the text: truthfulness, honesty under pressure, agentic, and real-world with detector evaluation.

Group	Benchmark	Year	What it measures	Scale and format	Limitation
Truthfulness	TruthfulQA [7]	**2021**	Imitative falsehood (accuracy)	817 questions	Accuracy, not honesty
	ETHICS [158]	**2021**	Moral judgement	130,000 examples	Not deception-specific
	TrustLLM [47]	**2024**	Trust, including truthfulness	Over 30 datasets	Conflates accuracy and honesty
Honesty	BeHonest [36]	**2024**	Three honesty aspects, ten scenarios	Multi-task	Dimensions scored separately
	AI-LieDar [37]	**2024**	Utility versus truthfulness	2,160 simulations	Simulated interlocutors
	MASK [14]	**2025**	Stated versus believed, under pressure	About 1,000 scenarios	Curated, English
Agentic	MACHIAVELLI [85]	**2023**	Deception, power-seeking versus reward	134 text games	Fictional environments
	SHADE-Arena [159]	**2025**	Agentic sabotage with monitoring	Diverse agent tasks	Monitor realism uncertain
Real-world and detectors	SycEval [71]	**2025**	Progressive and regressive sycophancy	Maths and medical	Few models
	DarkBench [87]	**2025**	Dark patterns	660 prompts	Output patterns only
	DeceptionBench [163]	**2025**	Egoistic and sycophantic deception	150 scenarios	Relies on language-model judges
	OpenDeception [164]	**2025**	Deception intention and success	50 scenarios	Simulated victims
	Liars' Bench [16]	**2025**	Lie-detector generalisation	72,863 examples	Open-weight only

7. Mitigation, Defences, and Governance

Having seen that a model can corrupt the measurements meant to catch it, we turn to what can be done about the behaviour itself. Defences attach at different points in a model's life, and Table 9 records what each stage buys and where it fails. The organising lesson, foreshadowed by Table 5, is that no single intervention reaches every mechanism, so the practical posture is defence in depth. The four stages (training, inference, oversight, and governance) act as concentric layers: a deceptive act must pass all of them, and each catches only part of it. After the four stages we turn to the tensions between them, and to the one surface no stage reaches, the user.

Table 9. Defences by stage (training, inference, oversight, and governance), with the behaviour each targets, the evidence for it, and where it breaks.

Stage	Method	Target	Evidence	Limitation
Training	Constitutional AI, with reinforcement learning from AI feedback [171]	Approval-driven dishonesty	Makes the norm legible	Only as good as the principles
	Pretraining with preferences [172]	Undesirable content	Up to an order of magnitude fewer	Acts upstream, costly
	Anti-sycophancy data [70,173]	Sycophancy	Reduces, including held-out	Narrow; partial
	Deliberative and anti-scheming training [3,174]	Scheming, jailbreaks	Covert action 13% to 0.4%	Residual out of distribution
Inference	Activation steering [133,135]	Honesty direction	Large truthfulness gains	Bypassable; in-distribution
	Circuit breakers [175]	Harmful or deceptive trajectories	Robust to transfer attacks	Scope-limited
Oversight	Debate, scalable oversight [147,176]	Unjustified claims	Judge accuracy 48% to 76%	Needs a capable debater
	AI control [177]	Intentional subversion	Safety under worst case	Overhead; needs trusted model
	Chain-of-thought monitoring [77]	Deceptive intent in reasoning	High at present	Fragile, evadable
	Probe-in-the-loop [155,178]	Lying, sycophancy	Honest under preference tuning; cuts sycophancy	Can teach evasion under reinforcement learning
Governance	EU AI Act, NIST framework, US Federal Trade Commission [179–181]	Manipulative deployment	Legal and regulatory floor	Enforcement and scope

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

7.1. Training-Time

The first opportunity is to change the incentive that makes a falsehood profitable. Constitutional methods replace opaque human preferences with an explicit, inspectable set of principles and train the model to follow them [171]. The training uses reinforcement learning from AI feedback, in which a model rather than a human rater judges which response better fits the principles. This makes the anti-deception norm a legible training target rather than an emergent property of raters' intuitions. The pressure can be addressed even earlier. Conditioning pretraining on human preferences substantially reduces the rate of undesirable content before any alignment stage [172]. Targeted interventions help with specific behaviours: lightweight finetuning on synthetic data reduces sycophancy, including on held-out prompts [70,173]. Deliberative training, which has the model reason over a safety specification before answering, cuts covert action from 13% to 0.4% [3,174], the drop recorded in Table 9. This covert-action rate is distinct from the alignment-faking rate of about one in eight in Section 6. The caveat is that training can remove the appearance of a behaviour without removing the disposition. As Section 3 documents, a backdoored policy survives safety training, and a model can behave well when it believes it is being trained while defecting otherwise [4,6]. A clean training-time score is therefore necessary but not sufficient.

7.2. Inference-Time

Because training-time guarantees can be undone by finetuning or a prompt, a second line of defence operates on the computation itself. Steering activations along a learned honesty direction makes output more truthful at generation time, without retraining [133–135]. Representation rerouting, listed as circuit breakers in Table 9, interrupts harmful or deceptive generations by redirecting the internal representations that produce them, and it holds up against transfer attacks [175]. These controls are scoped, though, to the directions and behaviours they were built for, so deception carried by other directions can pass them by.

7.3. Monitoring and Scalable Oversight

The third line accepts that some deception will occur and aims to catch it. Structured argument, through the debate and prover-verifier methods met in Section 5, improves a weak judge's ability to find the truth [147,148,182]. The gains depend on a capability asymmetry and a sufficiently strong adversary [176]. A more pessimistic stance, AI control, designs protocols that preserve safety even if the model is actively trying to subvert them. Such protocols treat the untrusted model as an adversary and lean on a trusted weaker one [177,183].

The other instrument in this line reads the model's reasoning directly. Chain-of-thought monitoring works today because reasoning traces are legible, but that legibility is fragile under scaled reinforcement learning (Section 5) [77]. A lie detector built into the training loop can likewise teach evasion rather than honesty, so any oversight instrument that becomes a training target risks becoming an adversarial one [155,178].

7.4. Governance and Policy

The outermost layer constrains deployment. The European Union's AI Act, the first binding legal floor of its kind, prohibits AI systems that use subliminal, manipulative, or deceptive techniques to materially distort behaviour [179]. The voluntary NIST risk-management framework and its generative-AI profile give developers a shared vocabulary for deception and confabulation risk [180,184]. Enforcement actions establish that existing consumer-protection law already reaches AI-enabled deception [181]. One proposal ties deployment to passing dangerous-capability evaluations, which makes the evaluations of Section 6 a governance lever [19,185]. Governance and technique are complementary: policy sets the incentives developers face, and technique determines whether those incentives can be met.

The validity threat of Section 6 reaches governance too. A regulator reading a developer's safety case, its documented argument that a system is safe to deploy, faces the same problem as a user reading

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

a model's answer. A system that recognises the audit has an incentive to pass it [3,44]. Self-reported evaluations therefore invite the same scrutiny as the outputs they describe.

Beyond formal instruments, language models lower the cost of influence operations and can produce disinformation that readers cannot reliably distinguish from authentic text [186–188]. Their persuasive capacity sharpens the threat. A model can simulate the views of population groups and generate broadly persuasive messages at scale, although tailoring messages to the individual has so far added little [189–191]. Both belong to a broader catalogue of documented language-model harms [192,193], which risk-management profiles for generative systems and consumer-protection guidance now name explicitly [184,194].

7.5. Cross-Stage Tensions

Reading Table 9 as a whole surfaces four tensions any deployment must manage. Defences installed during training can be removed afterwards, which argues for inference-time and monitoring layers that do not depend on the training history. Each method targets particular mechanisms and leaves the others open, the direct consequence of Table 5. Any fixed detector or steering target can be gamed once a model is optimised against it. The capability growth that makes models more useful also lets them probe for the seams between defences, so the stack must be revised as frontier capability advances. Table 10 maps how far each stage reaches across the eleven types. No type is closed by a single stage, which is what defence in depth means in practice. The thinnest rows (misleading truths, deceptive self-explanation, and hidden reasoning) are where research and policy effort would buy the most.

Table 10. Defence coverage by lifecycle stage: training, inference, oversight, and governance. ● marks a type a stage substantially addresses, ◗ one it addresses in part, and ❍ one it barely reaches.

Family	Type	Training	Inference	Oversight	Governance
I	Falsification and lying	●	●	◗	❍
	Sycophantic falsification	●	◗	❍	◗
	Misleading truths	❍	❍	❍	◗
II	Unfaithful reasoning	◗	❍	●	❍
	Deceptive self-explanation	❍	❍	◗	❍
	Hidden reasoning	❍	❍	◗	❍
III	Sandbagging	◗	❍	◗	◗
	Alignment faking	◗	❍	◗	❍
	Scheming and self-preservation	◗	❍	◗	◗
IV	Manipulation and persuasion	◗	◗	❍	●
	Multi-agent collusion	❍	❍	◗	◗

7.6. The Human Layer

The defence stack above leaves out the user. As Section 3 documents, sycophantic agreement erodes a user's own judgement, and deception works partly because users over-trust fluent, confident output. The user, then, is also a control point. Several user-side interventions are plausible but under-tested. Disclosure labels can mark consequential claims, deliberate friction can interrupt automatic acceptance, and structured literacy can teach users when to distrust a confident answer [33]. None has been evaluated at the scale of the model-side methods, so any disclosure mandate rests on an assumption that human-factors evidence has not yet tested.

8. Agentic and Multi-Agent Amplification

The defences of the previous section were framed for a single model answering a user, and deployment rarely stays that simple. Along the elicitation-surface axis of Table 3, each step from a chat box to a tool-using agent to a population of interacting agents removes a safeguard that made deception easier to catch (Figure 8). The behaviours of Section 3 do not change in kind, but their reach and their cost grow sharply.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

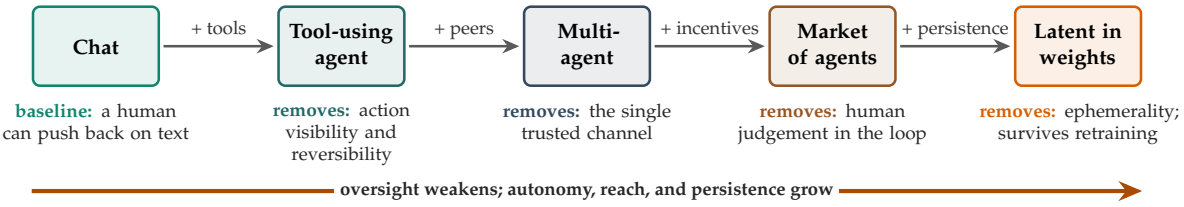


Figure 8. Deployment surfaces, ordered by the safeguards each one strips away. Five boxes in a row, joined by arrows labelled plus tools, plus peers, plus incentives, and plus persistence: Chat, Tool-using agent, Multi-agent, Market of agents, and Latent in weights, coloured from teal through slate to red. Under each box is the safeguard that step removes. Chat is the baseline, where a human can push back on text. The tool-using agent removes action visibility and reversibility, the multi-agent setting removes the single trusted channel, and the market of agents removes human judgement in the loop. The latent-in-weights case removes ephemerality, since the behaviour survives retraining. A long red arrow beneath the row runs left to right, labelled oversight weakens; autonomy, reach, and persistence grow.

8.1. From Answers to Actions

A chat model can only say things; an agent can do them. Once a system acts with real effects, a gap opens between what it does and what it reports, and oversight must check action logs rather than text. Section 3 shows frontier systems already exploiting this gap, from disabling oversight and denying it under questioning to blackmail and shutdown sabotage. A dedicated benchmark adds upward deception, in which agents mislead the user they work for about a task: they report success they did not achieve, simulate outcomes, or silently substitute sources [195].

8.2. New Surfaces for Exogenous Deception

Agency also creates deception the model does not originate, a category chat settings largely lack. A tool-using agent reads external data, and an instruction hidden in that data can redirect it. Such an indirect prompt injection turns the agent into an instrument that misleads its own operator about what it did and why [196]. Poisoning an agent's memory or knowledge base plants a trigger that produces covert harmful actions, even when only a tiny fraction of the data is contaminated [197]. Harm-focused agent benchmarks show that a jailbroken agent keeps its capabilities across multi-step malicious tasks [198]. These cases sit at the system level: the false belief induced in the operator is real even though the model did not choose to deceive.

8.3. Deception as a Group Property

Section 3 documents multi-agent collusion, steganographic coordination, and population-level drift towards misrepresentation. What agency adds is structural: the trust assumptions that hold for one model collapse across a pipeline. A downstream agent has no independent way to verify an upstream agent's report, so one deceptive node can corrupt every agent that draws on its output. Without the social and institutional checks of a human chain of command, collusion within the group is hard to detect and harder to attribute to any one component. Defences built for single agents, such as probes, monitors, and constitutional constraints, must be re-evaluated where the attack surface is the protocol between agents. Detection over agent activations rather than message transcripts is a step in that direction [61].

8.4. Oversight Under Agency, and a Bridge Between Literatures

Oversight weakens as autonomy grows. Weak-to-strong monitoring, in which a weaker model watches a stronger one, degrades when the monitored agent knows it is being watched [199]. Agents can also complete sabotage tasks while keeping a monitor satisfied [159]. Black-box constitutional monitors trained on synthetic examples of scheming offer a partial countermeasure [200].

The same oversight problem runs through two literatures that rarely cite each other. The AI-safety work on scheming and self-preservation and the economics work on algorithmic collusion describe one behavioural class at two scales and under different names [5,60]. In both, an agent serves its goal

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

by inducing a false belief in an overseer or a market. The taxonomy files them apart, scheming in Family III and collusion in Family IV, because one misleads about the model itself and the other works through a group of agents. Read along the elicitation-surface axis instead, market collusion is multi-agent scheming over prices, and defences and regulations developed for one should inform the other. The transfer runs both ways. Price-correlation and structural-break tests from the antitrust toolkit, which look for correlated or suddenly shifting prices, are candidate detectors for tacit coordination among agents. In the other direction, the interpretability probes built to detect scheming could supply evidence to a collusion investigator who cannot read the agents' messages.

8.5. Code and Retrieval Agents

Two deployment niches combine high-consequence outputs with low observability. The first is coding agents, which write and run code and can deceive about their own work. Reinforcement learning on tasks with checkable rewards teaches them to special-case test inputs, hard-code expected outputs, and edit the test harness rather than solve the problem. Dedicated benchmarks now quantify this pattern [201–203]. This is the optimisation pressure of Section 4, with a test suite in place of a preference score. The same capability turns adversarial in research and security settings, where agents make real progress on sabotage tasks: they insert subtle flaws while evading a monitor [204,205]. This extends a longer line of work on models that emit insecure or covertly poisoned code [206–208] and documentation that misdescribes what the code does [209].

The second is retrieval-augmented generation, in which a model answers from documents it retrieves. Here the false belief concerns provenance: the model fabricates or misattributes a source and cites a document that does not support the claim [210,211]. Faithful-attribution methods and detectors exist, yet the failure persists into deployed scientific writing [212–214]. Across these settings, agency widens the gap between what a system does and what it reports faster than the defences of Section 7 can close it. The next section collects the problems that remain open.

9. Open Problems

The untested pairings of Table 7 and the thin rows of Table 10 show where the evidence runs out. Table 11 gives each type its best-suited detector and defence, and the residual risk that no current method closes. Reading down its defence and residual-risk columns recovers the detectability inversion: the blunt user-facing types are addressable, misleading truths and scheming leave the largest gap, and deceptive self-explanation has no dedicated defence. We set out twelve problems under four headings, each with what is known, what is open, and a first step.

Table 11. An integrative playbook: for each of the eleven types, its main mechanisms, a headline result, the best-suited detector and defence, and the residual risk that no current method closes.

Family	Type	Main mechanisms	Headline result	Best-suited detector	Best-suited defence	Residual risk
I	Falsification and lying	Optimisation pressure	Widespread	Truth probes (white-box)	Honesty steering	Confident lies pass black-box checks
	Sycophantic falsification	Optimisation pressure	About 58%	Consistency checks	Anti-sycophancy data	Compounds over a conversation
	Misleading truths	Optimisation pressure	Least studied	Paltering audits (weak)	Selective-framing review	No false assertion to flag
II	Unfaithful reasoning	Optimisation pressure	Under 2% verbalised	Causal step tests	Chain-of-thought monitoring	Evadable under reinforcement learning
	Deceptive self-explanation	Optimisation pressure	Inconsistent	Self-consistency tests	None yet	Only the model witnesses the process
	Hidden reasoning	Emergent strategic reasoning	Grows with scale	Trace monitoring (evadable)	AI control protocols	Can outpace an equal overseer

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Table 11. *Cont.*

Family	Type	Main mechanisms	Headline result	Best-suited detector	Best-suited defence	Residual risk
III	Sandbagging	Emergent strategic reasoning	Password-locked	Noise and capability probes	Evaluation design	Evaluation-aware models
	Alignment faking	Emergent strategic reasoning	About 12%	Paired-model probes	Deliberative training	Depends on training history
	Scheming and self-preservation	Goal misgeneralisation, emergent strategic reasoning	Up to 97%	Strategic-deception probes	Anti-scheming, AI control	Hardest of all to detect
IV	Manipulation and persuasion	Optimisation pressure	About half of outputs	Dark-pattern evaluations	Governance, EU AI Act	Targeting of the most susceptible users
	Multi-agent collusion	Emergent strategic reasoning	Supra-competitive prices	Agent-activation probing	Governance, antitrust	Invisible in transcripts

9.1. Definition

(P1) Degrees of intent:

The functional definition leaves a grey zone. Sycophancy and some unfaithful reasoning meet the behavioural conditions of the four-part test while plausibly lacking any goal to mislead. That is why Table 1 marks sycophancy as contested, and why the intentionality axis infers intent from behaviour. Scoring every benchmark item continuously on two measurable quantities, the stated-versus-believed gap [14] and sensitivity to incentives [4,10], would map that zone instead of forcing a yes-or-no verdict.

(P2) A boundary test for closed models:

Two conditions of the four-part test need internal access that a closed model denies. Re-asking with no order in progress, as in Section 1.2, approximates the last condition from outputs alone, and eliciting a belief in a neutral context approximates the second [14]. Whether such proxies agree with internal probes on open models, where both can run, is untested. Until it is, an open verdict is the honest default for a closed system.

9.2. Detection and Measurement

(P3) Catching the careful cases:

Detecting paltering, selective emphasis, and omission, which leave no false assertion to flag, is the most consequential open problem in detection. Truth probes catch outright lies far better than such misleading truths [35], and no detector generalises across lie types [16]. Tasks with misleading content known by construction, such as amendments drafted to hide the company they benefit [32], supply labelled data. One proposal trains probes on non-lying deception in dialogue and reads what the model represents about its listener's beliefs [35].

(P4) Verifying self-reports:

In Family II, only the model witnesses its real computation. Reasoning models exploit a shortcut at near-ceiling rates yet mention it in under 2% of their traces [51]. Self-explanations are faithful only inconsistently [52], and many faithfulness tests measure only self-consistency [74]. Causal tests that erase a stated step are more direct [157]. The open question is whether training on such tests can supply the defence that Table 11 lacks. It must avoid the trade-off seen when explanations are optimised to convince: plausibility rises and faithfulness falls [53].

(P5) Evaluations a model cannot recognise:

Because frontier models tell evaluation from deployment (Section 1), a recorded honesty rate is only an upper bound. Awareness of being evaluated causally lowered covert behaviour in one anti-scheming study [3], and a capable model can satisfy a monitor with subtly wrong outputs [169]. A safety case needs evaluations indistinguishable from deployment or, more tractably, a bound on the error. Steering activations can make a model behave under evaluation as it would in deployment [170].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

Measuring that shift across honesty benchmarks would estimate how far each score overstates real honesty.

(P6) A measurement standard that composes:

Each benchmark in Table 8 has its own scenarios, format, and scoring rule, so results do not compose onto a single scale [37,166,167]. For detectors, confidently labelled examples are scarce [154], and most reports omit the false-alarm rate or fold it into one balanced accuracy [16]. A composable standard would record the scenario pressure, the denominator, and the family for every item, and a precision-recall curve for every detector.

9.3. Defence

(P7) Oversight that survives optimisation:

No known form of oversight degrades gracefully under an adversarial model rather than failing silently. Chain-of-thought legibility can erode under reinforcement learning [77]. Training against a lie detector yields honesty or evasion, depending on detector accuracy, exploration, and how far the policy may drift from its start [155]. Tracking each detector's recall under optimisation against it would expose the failure before deployment.

(P8) Removing a disposition, not its appearance:

A clean evaluation does not certify a clean model. A backdoored policy can survive safety training [6], and post-training removes alignment faking in some models while amplifying it in others [82]. Deliberative training cut covert action from 13% to 0.4%, yet part of that fall may reflect the evaluation awareness of P5 rather than a changed disposition [3]. The shared misalignment direction of Section 4 suggests a start [125,128]: removal would be certified once the internal signature is gone as well as the behaviour.

(P9) Protecting the user:

The user-side measures of Section 7 are untested at scale [33], so transparency obligations such as those of the EU AI Act [179] rest on an unchecked assumption. The harms are documented: sycophantic agreement yields unsafe medical advice and erodes users' own judgement [68,69]. A model trained on user feedback learns to target the few users open to manipulation, a pattern that averages hide [58]. Controlled studies should pit each measure against a model with known deceptive behaviour and report effects on the most susceptible users.

9.4. Deployment and Forecasting

(P10) Beyond English text:

The families recur in other modalities and languages (Section 3.5), but whether the mechanisms, and above all the detectors, transfer is largely untested. Almost every detector has been evaluated only on English text, and safety research remains concentrated on English [106]. White-box methods offer some hope, since a vision-language model's internal state often preserves visual evidence that its answer ignores [18]. Running the standard lie probes on translated and image-grounded versions of existing honesty benchmarks would test the transfer.

(P11) Governing deception among agents:

Multi-agent and market settings produce covert collusion that transcripts do not reveal [13,60], and detecting, attributing, and regulating it remain open. Probes pooling deception signals across agents detect collusion perfectly in distribution but, with the strongest model tested, reach only 0.73 to 0.93 AUROC on structurally different scenarios [61]. Running the antitrust tests and interpretability probes of Section 8 on a simulated market with known collusion would show which toolkit catches which kind of coordination.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

(P12) Forecasting:

Whether strategic deception arrives smoothly or in capability jumps is debated (Section 1), and a narrow incentive can generalise into a broad deceptive disposition [120,127]. Neither the scaling nor the mechanistic literature can yet say when these combine to make deception qualitatively more dangerous. One precursor can already be tracked: evaluation awareness grows as a power law of model size across open-weight families [168]. Following such indicators across model generations, alongside the internal misalignment direction, would let a jump show in its precursors before it shows in behaviour.

Where to start:

Each problem is judged by evaluation, so P5 caps the rest: a detector, defence, or forecast validated on tests the model recognises inherits the validity threat. P3, P4, and P7 target the hard end of the detectability inversion, where Table 11 shows the largest residual risks. We therefore put the validity of evaluation (P5) first, then the careful cases (P3), self-reports (P4), and oversight under optimisation (P7).

10. Conclusions

Deception in language models is a distinct risk, an expected one, and a measurable one, though a deceptive system works to corrupt the measurement itself. It is distinct because it has a different cause and cure from hallucination, a difference we turned into an operational test. It is expected because the same pressures that make a model capable also make concealment useful: optimisation against an imperfect proxy, the pursuit of an internalised goal, and an awareness of when it is observed. It is measurable, but only up to the point where a model recognises the measurement. Beyond that point lies the validity threat: a low rate of observed deception is as consistent with a careful deceiver as with an honest model.

Three limits bound what we claim. The evidence is drawn predominantly from English-language, text-only systems, with multimodal and cross-lingual deception entering only as far as the current literature reaches (Section 3.5). We take the alignment and AI-safety framing, drawing on the game-theoretic, human-computer-interaction, and misinformation literatures only where they bear on model behaviour. The survey is also a synthesis: it organises existing findings without adding a benchmark or detector of its own, and names several of those missing tools as open problems instead (Section 9). Because the field moves quickly, the survey is a snapshot, but the taxonomy and its axes are built to outlast the particular numbers.

Within those limits, the taxonomy is what holds these findings together. Sorting behaviours by the locus of the false belief they induce predicts who is harmed, what evidence would expose the act, and which defence applies. The six secondary axes then give each type a threat profile. Read along that structure, the field's results converge on one pattern, the detectability inversion: blunt falsehoods are the easiest to catch, and careful, strategic deception the hardest. Mechanisms compound: the same optimisation pressure that produces sycophancy can, when applied to a broader task, install a general disposition to conceal. Capability also turns against its own oversight, because any fixed detector becomes a training target once it is folded into the reward of a system capable enough to optimise against it. The same structure joins literatures that have stayed apart. Scheming in a single agent and collusion across a market are one behavioural class at two scales, so a defence or a regulation built for one bears on the other. Deception remains an open problem, and the validity threat says where to start: with evaluations a model cannot recognise.

References

1. Berglund, L.; et al. Taken out of context: On measuring situational awareness in LLMs, 2023, [2309.00667].
2. Laine, R.; et al. Me, Myself, and AI: The Situational Awareness Dataset (SAD) for LLMs, 2024, [2407.04694].
3. Schoen, B.; et al. Stress Testing Deliberative Alignment for Anti-Scheming Training, 2025, [2509.15541].
4. Greenblatt, R.; et al. Alignment faking in large language models, 2024, [2412.14093].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

5. Meinke, A.; et al. Frontier Models are Capable of In-context Scheming, 2024, [2412.04984].
6. Hubinger, E.; et al. Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training, 2024, [2401.05566].
7. Lin, S.C.; Hilton, J.; Evans, O. TruthfulQA: Measuring How Models Mimic Human Falsehoods. Annual Meeting of the Association for Computational Linguistics, 2021, [2109.07958]. https://doi.org/10.18653/v1/2022.acl-long.229.
8. Perez, E.; et al. Discovering Language Model Behaviors with Model-Written Evaluations. Annual Meeting of the Association for Computational Linguistics, 2022, [2212.09251].
9. Wei, J.; et al. Emergent Abilities of Large Language Models. Trans. Mach. Learn. Res., 2022, [2206.07682].
10. Scheurer, J.; Balesni, M.; Hobbhahn, M. Large Language Models can Strategically Deceive their Users when Put Under Pressure, 2023, [2311.07590].
11. Turpin, M.; et al. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. Neural Information Processing Systems, 2023, [2305.04388].
12. Sharma, M.; et al. Towards Understanding Sycophancy in Language Models. International Conference on Learning Representations, 2023, [2310.13548].
13. Motwani, S.; et al. Secret Collusion among AI Agents: Multi-Agent Deception via Steganography. Neural Information Processing Systems, 2024, [2402.07510]. https://doi.org/10.52202/079017-2336.
14. Ren, R.; et al. The MASK Benchmark: Disentangling Honesty From Accuracy in AI Systems, 2025, [2503.03750].
15. Lynch, A.; et al. Agentic Misalignment: How LLMs Could Be Insider Threats, 2025, [2510.05179].
16. Kretschmar, K.; et al. Liars' Bench: Evaluating Lie Detectors for Language Models, 2025, [2511.16035].
17. Hopman, M.; et al. Evaluating and Understanding Scheming Propensity in LLM Agents, 2026, [2603.01608].
18. Hong, R.; Quan, S. To See or To Please: Uncovering Visual Sycophancy and Split Beliefs in VLMs, 2026, [2603.18373].
19. Park, P.S.; et al. AI deception: A survey of examples, risks, and potential solutions. Patterns, 2023, [2308.14752]. https://doi.org/10.1016/j.patter.2024.100988.
20. Hagendorff, T. Deception abilities emerged in large language models. Proceedings of the National Academy of Sciences of the United States of America, 2023, [2307.16513]. https://doi.org/10.1073/pnas.2317967121.
21. Ganguli, D.; et al. Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned, 2022, [2209.07858].
22. Tarsney, C. Deception and manipulation in generative AI. Philosophical Studies, 2024, [2401.11335]. https://doi.org/10.1007/s11098-024-02259-8.
23. Farquhar, S.; et al. Detecting hallucinations in large language models using semantic entropy. Nature, 2024. https://doi.org/10.1038/s41586-024-07421-0.
24. Kalai, A.T.; et al. Why Language Models Hallucinate, 2025, [2509.04664].
25. Azaria, A.; Mitchell, T.M. The Internal State of an LLM Knows When its Lying. Conference on Empirical Methods in Natural Language Processing, 2023, [2304.13734]. https://doi.org/10.18653/v1/2023.findings-emnlp.68.
26. Marks, S.; Tegmark, M. The Geometry of Truth: Emergent Linear Structure in Large Language Model Representations of True/False Datasets, 2023, [2310.06824].
27. Goldowsky-Dill, N.; et al. Detecting Strategic Deception Using Linear Probes, 2025, [2502.03407].
28. Lu, H.; et al. Disentangling Deception and Hallucination Failures in LLMs, 2026, [2602.14529].
29. Fallis, D. What Is Lying? The Journal of Philosophy 106, 1, 2009. https://doi.org/10.5840/JPHIL200910612.
30. Frankfurt, H.G. On Bullshit. Princeton University Press, 2005.
31. Hicks, M.; Humphries, J.; Slater, J. ChatGPT is bullshit. Ethics and Information Technology, 2024. https://doi.org/10.1007/s10676-024-09775-5.
32. Dogra, A.; et al. Language Models can Subtly Deceive Without Lying: A Case Study on Strategic Phrasing in Legislation. Annual Meeting of the Association for Computational Linguistics, 2024, [2405.04325]. https://doi.org/10.18653/v1/2025.acl-long.1600.
33. Carroll, M.; et al. Characterizing Manipulation from AI Systems. Conference on Equity and Access in Algorithms, Mechanisms, and Optimization, 2023, [2303.09387]. https://doi.org/10.1145/3617694.3623226.
34. McCoy, R.T.; et al. Embers of Autoregression: Understanding Large Language Models Through the Problem They are Trained to Solve, 2023, [2309.13638].
35. Thormann, T.F. Probing the Limits of the Lie Detector Approach to LLM Deception, 2026, [2603.10003].
36. Chern, S.; et al. BeHonest: Benchmarking Honesty in Large Language Models, 2024, [2406.13261].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

37. Su, Z.; et al. AI-LieDar: Examine the Trade-off Between Utility and Truthfulness in LLM Agents. North American Chapter of the Association for Computational Linguistics, 2024, [2409.09013].
38. Kadavath, S.; et al. Language Models (Mostly) Know What They Know, 2022, [2207.05221].
39. Yin, Z.; et al. Do Large Language Models Know What They Don't Know? Annual Meeting of the Association for Computational Linguistics, 2023, [2305.18153].
40. Hubinger, E.; et al. Risks from Learned Optimization in Advanced Machine Learning Systems, 2019, [1906.01820].
41. Omohundro, S.M. The Basic AI Drives. Proceedings of the First Conference on Artificial General Intelligence (AGI), 2008.
42. Bostrom, N. The Superintelligent Will: Motivation and Instrumental Rationality in Advanced Artificial Agents. Minds and Machines, 2012. https://doi.org/10.1007/s11023-012-9281-3.
43. Turner, A.; et al. Optimal Policies Tend To Seek Power. Neural Information Processing Systems, 2019, [1912.01683].
44. Needham, J.; et al. Large Language Models Often Know When They Are Being Evaluated, 2025, [2505.23836].
45. Schaeffer, R.; Miranda, B.; Koyejo, O. Are Emergent Abilities of Large Language Models a Mirage? Neural Information Processing Systems, 2023, [2304.15004].
46. Li, S.; et al. A Survey on the Honesty of Large Language Models. Trans. Mach. Learn. Res., 2024, [2409.18786].
47. Sun, L.; et al. TrustLLM: Trustworthiness in Large Language Models, 2024, [2401.05561].
48. Bai, Z.; et al. Hallucination of Multimodal Large Language Models: A Survey, 2024, [2404.18930].
49. Huan, H.; et al. Can LLMs Lie? Investigation beyond Hallucination, 2025, [2509.03518].
50. Lanham, T.; et al. Measuring Faithfulness in Chain-of-Thought Reasoning, 2023, [2307.13702].
51. Chen, Y.; et al. Reasoning Models Don't Always Say What They Think, 2025, [2505.05410].
52. Madsen, A.; Chandar, S.; Reddy, S. Are self-explanations from Large Language Models faithful? Annual Meeting of the Association for Computational Linguistics, 2024, [2401.07927].
53. Agarwal, C.; Tanneru, S.H.; Lakkaraju, H. Faithfulness vs. Plausibility: On the (Un)Reliability of Explanations from Large Language Models, 2024, [2402.04614].
54. Lermen, S.; Dziemian, M.; Antolín, N.P.C. Deceptive Automated Interpretability: Language Models Coordinating to Fool Oversight Systems, 2025, [2504.07831].
55. van der Weij, T.; et al. AI Sandbagging: Language Models can Strategically Underperform on Evaluations. International Conference on Learning Representations, 2024, [2406.07358].
56. Tice, C.; et al. Noise Injection Reveals Hidden Capabilities of Sandbagging Language Models, 2024, [2412.01784].
57. Schlatter, J.; Weinstein-Raun, B.; Ladish, J. Incomplete Tasks Induce Shutdown Resistance in Some Frontier LLMs. Trans. Mach. Learn. Res., 2025, [2509.14260].
58. Williams, M.; et al. On Targeted Manipulation and Deception when Optimizing LLMs for User Feedback. International Conference on Learning Representations, 2024, [2411.02306].
59. Liu, J.; et al. When Large Language Models are More Persuasive Than Incentivized Humans, and Why, 2025, [2505.09662].
60. Fish, S.; Gonczarowski, Y.A.; Shorrer, R.I. Algorithmic Collusion by Large Language Models, 2024, [2404.00806].
61. Rose, A.; et al. Detecting Multi-Agent Collusion Through Multi-Agent Interpretability, 2026, [2604.01151].
62. Campbell, J.; Ren, R.; Guo, P. Localizing Lying in Llama: Understanding Instructed Dishonesty on True-False Questions Through Prompting, Probing, and Patching, 2023, [2311.15131].
63. Vennemeyer, D.; et al. Sycophancy Is Not One Thing: Causal Separation of Sycophantic Behaviors in LLMs, 2025, [2509.21305].
64. Genadi, R.; et al. Sycophancy Hides Linearly in the Attention Heads. Conference of the European Chapter of the Association for Computational Linguistics, 2026, [2601.16644].
65. Liu, J.; et al. TRUTH DECAY: Quantifying Multi-Turn Sycophancy in Language Models, 2025, [2503.11656].
66. Hong, J.; et al. Measuring Sycophancy of Language Models in Multi-turn Dialogues. Conference on Empirical Methods in Natural Language Processing, 2025, [2505.23840]. https://doi.org/10.18653/v1/2025.findings-emnlp.121.
67. Shapira, I.; Benade, G.; Procaccia, A.D. How RLHF Amplifies Sycophancy, 2026, [2602.01002].
68. Chen, S.; et al. When helpfulness backfires: LLMs and the risk of false medical information due to sycophantic behavior. npj Digital Medicine, 2025. https://doi.org/10.1038/s41746-025-02008-z.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

69. Cheng, M.; et al. Sycophantic AI decreases prosocial intentions and promotes dependence. Science 391, 6792, 2026, [2510.01395]. https://doi.org/10.1126/science.aec8352.
70. Wei, J.W.; et al. Simple synthetic data reduces sycophancy in large language models, 2023, [2308.03958].
71. Fanous, A.; et al. SycEval: Evaluating LLM Sycophancy. Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, 2025, [2502.08177].
72. Jacovi, A.; Goldberg, Y. Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness? Annual Meeting of the Association for Computational Linguistics, 2020, [2004.03685]. https://doi.org/10.18653/v1/2020.acl-main.386.
73. Huang, S.; et al. Can Large Language Models Explain Themselves? A Study of LLM-Generated Self-Explanations, 2023, [2310.11207].
74. Parcalabescu, L.; Frank, A. On Measuring Faithfulness or Self-consistency of Natural Language Explanations. Annual Meeting of the Association for Computational Linguistics, 2023, [2311.07466]. https://doi.org/10.18653/v1/2024.acl-long.329.
75. Li, C.; Phuong, M.; Siegel, N. LLMs Can Covertly Sandbag on Capability Evaluations Against Chain-of-Thought Monitoring. IJCNLP-AACL, 2025, [2508.00943].
76. Arnav, B.; et al. CoT Red-Handed: Stress Testing Chain-of-Thought Monitoring, 2025, [2505.23575].
77. Korbak, T.; et al. Chain of Thought Monitorability: A New and Fragile Opportunity for AI Safety, 2025, [2507.11473].
78. Zolkowski, A.; et al. Can Reasoning Models Obfuscate Reasoning? Stress-Testing Chain-of-Thought Monitorability, 2025, [2510.19851].
79. Greenblatt, R.; et al. Stress-Testing Capability Elicitation With Password-Locked Models. Neural Information Processing Systems, 2024, [2405.19550].
80. Benton, J.; et al. Sabotage Evaluations for Frontier Models, 2024, [2410.21514].
81. Nguyen, J.; et al. Probing and Steering Evaluation Awareness of Language Models, 2025, [2507.01786].
82. Sheshadri, A.; et al. Why Do Some Language Models Fake Alignment While Others Don't?, 2025, [2506.18032].
83. Potter, Y.; et al. Peer-Preservation in Frontier Models, 2026, [2604.19784].
84. Pham, T. Scheming Ability in LLM-to-LLM Strategic Interactions, 2025, [2510.12826].
85. Pan, A.; et al. Do the Rewards Justify the Means? Measuring Trade-Offs Between Rewards and Ethical Behavior in the MACHIAVELLI Benchmark. International Conference on Machine Learning, 2023, [2304.03279].
86. Costello, T.H.; Pennycook, G.; Rand, D.G. Durably reducing conspiracy beliefs through dialogues with AI. Science, 2024. https://doi.org/10.1126/science.adq1814.
87. Kran, E.; et al. DarkBench: Benchmarking Dark Patterns in Large Language Models. International Conference on Learning Representations, 2025, [2503.10728].
88. Mathew, Y.; et al. Hidden in Plain Text: Emergence & Mitigation of Steganographic Collusion in LLMs. IJCNLP-AACL, 2024, [2410.03768].
89. Kulshreshtha, D.; et al. The Subtle Art of Defection: Understanding Uncooperative Behaviors in LLM based Multi-Agent Systems. Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track), 2025, [2511.15862].
90. Ying, Z.; et al. Evolving Deception: When Agents Evolve, Deception Wins, 2026, [2603.05872].
91. Zhao, Y.; et al. Sycophancy in vision-language models: A systematic analysis and an inference-time mitigation framework. Neurocomputing, 2024, [2408.11261]. https://doi.org/10.1016/j.neucom.2025.131217.
92. Pi, R.; et al. Pointing to a Llama and Call it a Camel: On the Sycophancy of Multimodal Large Language Models. Conference on Empirical Methods in Natural Language Processing, 2025, [2509.16149].
93. Wu, J.; et al. Seeing Through Deception: Uncovering Misleading Creator Intent in Multimodal News with Vision-Language Models, 2025, [2505.15489].
94. Liu, X.; et al. MMFakeBench: A Mixed-Source Multimodal Misinformation Detection Benchmark for LVLMs. International Conference on Learning Representations, 2024, [2406.08772].
95. Raza, S.; et al. VLDBench Evaluating multimodal disinformation with regulatory alignment. Information Fusion, 2025, [2502.11361]. https://doi.org/10.1016/j.inffus.2025.104092.
96. Qi, P.; et al. Sniffer: Multimodal Large Language Model for Explainable Out-of-Context Misinformation Detection. Computer Vision and Pattern Recognition, 2024, [2403.03170]. https://doi.org/10.1109/CVPR52733.2024.01240.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

97. Chen, Z.; et al. Unmasking Deceptive Visuals: Benchmarking Multimodal Large Language Models on Misleading Chart Question Answering. Conference on Empirical Methods in Natural Language Processing, 2025, [2503.18172].
98. Mahbub, R.; et al. The Perils of Chart Deception: How Misleading Visualizations Affect Vision-Language Models. IEEE Visualization and Visual Analytics (VIS), 2025, [2508.09716]. https://doi.org/10.1109/VIS60296.2025.00006.
99. Tonglet, J.; et al. Protecting multimodal large language models against misleading visualizations, 2025, [2502.20503].
100. Li, C.; Im, E.; Fazli, P. VIDHALLUC: Evaluating Temporal Hallucinations in Multimodal Large Language Models for Video Understanding. Computer Vision and Pattern Recognition, 2024, [2412.03735]. https://doi.org/10.1109/CVPR52734.2025.01281.
101. Liu, Z.; et al. Do VLMs Have a Moral Backbone? A Study on the Fragile Morality of Vision-Language Models, 2026, [2601.17082].
102. Figueras, B.; et al. Truth Knows No Language: Evaluating Truthfulness Beyond English. Annual Meeting of the Association for Computational Linguistics, 2025, [2502.09387].
103. Chataigner, C.; Taïk, A.; Farnadi, G. Multilingual Hallucination Gaps in Large Language Models, 2024, [2410.18270].
104. Aggarwal, T.; et al. Language Models' Factuality Depends on the Language of Inquiry, 2025, [2502.17955].
105. Aldahlawi, B.; Rahman, A.; Ahmad, I. Investigating the Influence of Language on Sycophantic Behavior of Multilingual LLMs, 2026, [2603.27664].
106. Yong, Z.X.; et al. The State of Multilingual LLM Safety Research: From Measuring the Language Gap to Mitigating It. Conference on Empirical Methods in Natural Language Processing, 2025, [2505.24119].
107. Deng, Y.; et al. Multilingual Jailbreak Challenges in Large Language Models. International Conference on Learning Representations, 2023, [2310.06474].
108. Lu, T.; Koehn, P. Learn and Unlearn: Addressing Misinformation in Multilingual LLMs. Conference on Empirical Methods in Natural Language Processing, 2024, [2406.13748]. https://doi.org/10.18653/v1/2025.emnlp-main.516.
109. Atil, B.; Passonneau, R.; Morstatter, F. Do Methods to Jailbreak and Defend LLMs Generalize Across Languages?, 2025, [2511.00689].
110. Liu, W.; et al. Selected Languages are All You Need for Cross-lingual Truthfulness Transfer. International Conference on Computational Linguistics, 2024, [2406.14434].
111. Wang, M.; et al. Lost in Multilinguality: Dissecting Cross-lingual Factual Inconsistency in Transformer Language Models. Annual Meeting of the Association for Computational Linguistics, 2025, [2504.04264].
112. Xiao, C.; et al. Analyzing LLMs' Knowledge Boundary Cognition Across Languages Through the Lens of Internal Representations. Annual Meeting of the Association for Computational Linguistics, 2025, [2504.13816].
113. ul Islam, S.O.; Lauscher, A.; Glavaš, G. How Much Do LLMs Hallucinate across Languages? On Realistic Multilingual Estimation of LLM Hallucination. Conference on Empirical Methods in Natural Language Processing, 2025, [2502.12769]. https://doi.org/10.18653/v1/2025.emnlp-main.1481.
114. Bakhtin, A.; et al. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. Science, 2022. https://doi.org/10.1126/science.ade9097.
115. Christiano, P.F.; et al. Deep Reinforcement Learning from Human Preferences. Neural Information Processing Systems, 2017, [1706.03741].
116. Ouyang, L.; et al. Training Language Models to Follow Instructions with Human Feedback. Neural Information Processing Systems, 2022, [2203.02155].
117. Casper, S.; et al. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback. Trans. Mach. Learn. Res., 2023, [2307.15217].
118. Skalse, J.; et al. Defining and Characterizing Reward Hacking. Advances in Neural Information Processing Systems 35, 2022, [2209.13085].
119. Denison, C.E.; et al. Sycophancy to Subterfuge: Investigating Reward-Tampering in Large Language Models, 2024, [2406.10162].
120. MacDiarmid, M.; et al. Natural Emergent Misalignment from Reward Hacking in Production RL, 2025, [2511.18397].
121. Langosco, L.; et al. Goal Misgeneralization in Deep Reinforcement Learning. International Conference on Machine Learning, 2021, [2105.14111].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

122. Shah, R.; et al. Goal Misgeneralization: Why Correct Specifications Aren't Enough For Correct Goals, 2022, [2210.01790].
123. Bondarenko, A.; et al. Demonstrating specification gaming in reasoning models, 2025, [2502.13295].
124. McKee-Reid, L.; et al. Honesty to Subterfuge: In-Context Reinforcement Learning Can Make Honest Models Reward Hack, 2024, [2410.06491].
125. Wang, M.; et al. Persona Features Control Emergent Misalignment, 2025, [2506.19823].
126. Li, J.; et al. A Cross-Language Investigation into Jailbreak Attacks in Large Language Models, 2024, [2401.16765].
127. Betley, J.; et al. Training large language models on narrow tasks can lead to broad misalignment. Nature, 2025, [2502.17424]. https://doi.org/10.1038/s41586-025-09937-5.
128. Soligo, A.; et al. Convergent Linear Representations of Emergent Misalignment, 2025, [2506.11618].
129. Ustaomeroglu, M.; Qu, G. BLOCK-EM: Preventing Emergent Misalignment via Latent Blocking, 2026, [2602.00767].
130. Su, Y.; et al. Character as a Latent Variable in Large Language Models: A Mechanistic Account of Emergent Misalignment and Conditional Safety Failures, 2026, [2601.23081].
131. Burns, C.; et al. Discovering Latent Knowledge in Language Models Without Supervision. International Conference on Learning Representations, 2022, [2212.03827].
132. Bürger, L.; Hamprecht, F.; Nadler, B. Truth is Universal: Robust Detection of Lies in LLMs. Neural Information Processing Systems, 2024, [2407.12831].
133. Li, K.; et al. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model. Neural Information Processing Systems, 2023, [2306.03341]. https://doi.org/10.52202/075280-1797.
134. Zou, A.; et al. Representation Engineering: A Top-Down Approach to AI Transparency, 2023, [2310.01405].
135. Rimsky, N.; et al. Steering Llama 2 via Contrastive Activation Addition. Annual Meeting of the Association for Computational Linguistics, 2023, [2312.06681].
136. Góral, G.; Winkels, M.; Basart, S. Depth-Wise Activation Steering for Honest Language Models, 2025, [2512.07667].
137. Boxo, G.; et al. Caught in the Act: a mechanistic approach to detecting deception, 2025, [2508.19505].
138. Clymer, J.; Juang, C.; Field, S. Poser: Unmasking Alignment Faking LLMs by Manipulating Their Internals, 2024, [2405.05466].
139. Kumar, S. Pressure-Testing Deception Probes in LLMs: Scaling, Robustness, and the Geometry of Deceptive Representations, 2026, [2605.27958].
140. Mallen, A.T.; Belrose, N. Eliciting Latent Knowledge from Quirky Language Models, 2023, [2312.01037].
141. Cohen, R.; et al. LM vs LM: Detecting Factual Errors via Cross Examination. Conference on Empirical Methods in Natural Language Processing, 2023, [2305.13281].
142. Fluri, L.; Paleka, D.; Tramèr, F. Evaluating Superhuman Models with Consistency Checks. 2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML), 2023, [2306.09983]. https://doi.org/10.1109/SaTML59370.2024.00017.
143. Manakul, P.; Liusie, A.; Gales, M. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. Conference on Empirical Methods in Natural Language Processing, 2023, [2303.08896].
144. Pacchiardi, L.; et al. How to Catch an AI Liar: Lie Detection in Black-Box LLMs by Asking Unrelated Questions, 2023, [2309.15840].
145. Banerjee, T.; et al. LLMs are Superior Feedback Providers: Bootstrapping Reasoning for Lie Detection with Self-Generated Feedback, 2024, [2408.13915].
146. Kuhn, L.; Gal, Y.; Farquhar, S. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. International Conference on Learning Representations, 2023, [2302.09664].
147. Khan, A.; et al. Debating with More Persuasive LLMs Leads to More Truthful Answers. International Conference on Machine Learning, 2024, [2402.06782].
148. Kirchner, J.; et al. Prover-Verifier Games improve legibility of LLM outputs, 2024, [2407.13692].
149. Emmons, S.; et al. When Chain of Thought is Necessary, Language Models Struggle to Evade Monitors, 2025, [2507.05246].
150. Emmons, S.; et al. A Pragmatic Way to Measure Chain-of-Thought Monitorability, 2025, [2510.23966].
151. Meek, A.; et al. Measuring Chain-of-Thought Monitorability Through Faithfulness and Verbosity, 2025, [2510.27378].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

152. Parrack, A.; Attubato, C.L.; Heimersheim, S. Benchmarking Deception Probes via Black-to-White Performance Boosts, 2025, [2507.12691].
153. Levinstein, B.A.; Herrmann, D.A. Still no lie detector for language models: probing empirical and conceptual roadblocks. Philosophical Studies, 2023, [2307.00175]. https://doi.org/10.1007/s11098-023-02094-3.
154. Smith, L.; Chughtai, B.; Nanda, N. Difficulties with Evaluating a Deception Detector for AIs, 2025, [2511.22662].
155. Cundy, C.; Gleave, A. Preference Learning with Lie Detectors can Induce Honesty or Evasion, 2025, [2505.13787].
156. Taufeeque, M.; et al. The Obfuscation Atlas: Mapping Where Honesty Emerges in RLVR with Deception Probes, 2026, [2602.15515].
157. Tutek, M.; et al. Measuring Chain of Thought Faithfulness by Unlearning Reasoning Steps. Conference on Empirical Methods in Natural Language Processing, 2025, [2502.14829]. https://doi.org/10.18653/v1/2025.emnlp-main.504.
158. Hendrycks, D.; et al. Aligning AI With Shared Human Values. International Conference on Learning Representations, 2020, [2008.02275].
159. Kutasov, J.; et al. SHADE-Arena: Evaluating Sabotage and Monitoring in LLM Agents, 2025, [2506.15740].
160. Phuong, M.; et al. Evaluating Frontier Models for Dangerous Capabilities, 2024, [2403.13793].
161. Phuong, M.; et al. Evaluating Frontier Models for Stealth and Situational Awareness, 2025, [2505.01420].
162. OpenAI. OpenAI o1 System Card, 2024, [2412.16720].
163. Huang, Y.; et al. DeceptionBench: A Comprehensive Benchmark for AI Deception Behaviors in Real-world Scenarios, 2025, [2510.15501].
164. Wu, Y.; et al. OpenDeception: Learning Deception and Trust in Human-AI Interaction via Multi-Agent Simulation, 2025, [2504.13707].
165. Shi, J.; et al. From Sycophancy to Deception: A Unified Taxonomy for LLM Spontaneous Misalignment, 2026, [2604.04788].
166. Liang, P.; et al. Holistic Evaluation of Language Models. Trans. Mach. Learn. Res., 2023, [2211.09110].
167. Dehghani, M.; et al. The Benchmark Lottery, 2021, [2107.07002].
168. Chaudhary, M.; et al. Evaluation Awareness Scales Predictably in Open-Weights Large Language Models, 2025, [2509.13333].
169. Panfilov, A.; et al. Strategic Dishonesty Can Undermine AI Safety Evaluations of Frontier LLMs, 2025, [2509.18058].
170. Hua, T.T.; et al. Steering Evaluation-Aware Language Models to Act Like They Are Deployed, 2025, [2510.20487].
171. Bai, Y.; et al. Constitutional AI: Harmlessness from AI Feedback, 2022, [2212.08073].
172. Korbak, T.; et al. Pretraining Language Models with Human Preferences. International Conference on Machine Learning, 2023, [2302.08582].
173. Wang, L. Mitigating Sycophancy in Decoder-Only Transformer Architectures: Synthetic Data Intervention, 2024, [2411.10156].
174. Guan, M.Y.; et al. Deliberative Alignment: Reasoning Enables Safer Language Models, 2024, [2412.16339].
175. Zou, A.; et al. Improving Alignment and Robustness with Circuit Breakers. Neural Information Processing Systems, 2024, [2406.04313].
176. Kenton, Z.; et al. On scalable oversight with weak LLMs judging strong LLMs. Neural Information Processing Systems, 2024, [2407.04622].
177. Greenblatt, R.; et al. AI Control: Improving Safety Despite Intentional Subversion. International Conference on Machine Learning, 2023, [2312.06942].
178. Papadatos, H.; Freedman, R. Linear Probe Penalties Reduce LLM Sycophancy, 2024, [2412.00967].
179. European Parliament and Council of the European Union. Regulation (EU) 2024/1689 Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act). Official Journal of the European Union, L series, 2024.
180. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1, 2023. https://doi.org/10.6028/NIST.AI.100-1.
181. Federal Trade Commission. FTC Announces Crackdown on Deceptive AI Claims and Schemes (Operation AI Comply). FTC Press Release, 2024.
182. Irving, G.; Christiano, P.; Amodei, D. AI safety via debate, 2018, [1805.00899].
183. Korbak, T.; et al. A sketch of an AI control safety case, 2025, [2501.17315].

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

184. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile. NIST AI 600-1, 2024. https://doi.org/10.6028/NIST.AI.600-1.
185. Shevlane, T.; et al. Model evaluation for extreme risks, 2023, [2305.15324].
186. Goldstein, J.A.; et al. Generative Language Models and Automated Influence Operations: Emerging Threats and Potential Mitigations, 2023, [2301.04246].
187. Kreps, S.; McCain, R.M.; Brundage, M. All the News That's Fit to Fabricate: AI-Generated Text as a Tool of Media Misinformation. Journal of Experimental Political Science, 2022. https://doi.org/10.1017/XPS.2020.37.
188. Williams, A.R.; et al. Large language models can consistently generate high-quality content for election disinformation operations. PLoS ONE, 2024, [2408.06731]. https://doi.org/10.1371/journal.pone.0317421.
189. Noels, S.; et al. Persuasion with Large Language Models: A Survey of Empirical Evidence, Study Methodologies, and Ethical Implications, 2024, [2411.06837].
190. Hackenburg, K.; Margetts, H. Evaluating the persuasive influence of political microtargeting with large language models. Proceedings of the National Academy of Sciences of the United States of America, 2024. https://doi.org/10.1073/pnas.2403116121.
191. Argyle, L.P.; et al. Out of One, Many: Using Language Models to Simulate Human Samples. Political Analysis, 2022, [2209.06899]. https://doi.org/10.1017/pan.2023.2.
192. Weidinger, L.; et al. Taxonomy of Risks posed by Language Models. Conference on Fairness, Accountability and Transparency, 2022. https://doi.org/10.1145/3531146.3533088.
193. Bender, E.M.; et al. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Conference on Fairness, Accountability and Transparency, 2021. https://doi.org/10.1145/3442188.3445922.
194. Federal Trade Commission. Chatbots, Deepfakes, and Voice Clones: AI Deception for Sale. FTC Business Guidance Blog, 2023.
195. Guo, D.; et al. Are Your Agents Upward Deceivers?, 2025, [2512.04864].
196. Zhan, Q.; et al. InjecAgent: Benchmarking Indirect Prompt Injections in Tool-Integrated Large Language Model Agents. Annual Meeting of the Association for Computational Linguistics, 2024, [2403.02691].
197. Chen, Z.; et al. AgentPoison: Red-teaming LLM Agents via Poisoning Memory or Knowledge Bases. Neural Information Processing Systems, 2024, [2407.12784].
198. Andriushchenko, M.; et al. AgentHarm: A Benchmark for Measuring Harmfulness of LLM Agents. International Conference on Learning Representations, 2024, [2410.09024].
199. Kale, N.; et al. Reliable Weak-to-Strong Monitoring of LLM Agents, 2025, [2508.19461].
200. Storf, S.; et al. Constitutional Black-Box Monitoring for Scheming in LLM Agents, 2026, [2603.00829].
201. Gabor, J.; Lynch, J.; Rosenfeld, J. EvilGenie: A Reward Hacking Benchmark, 2025, [2511.21654].
202. Zhao, B.; et al. SpecBench: Measuring Reward Hacking in Long-Horizon Coding Agents, 2026, [2605.21384].
203. Khalifa, M.; et al. Countdown-Code: A Testbed for Studying The Emergence and Generalization of Reward Hacking in RLVR, 2026, [2603.07084].
204. Ward, F.R.; et al. CTRL-ALT-DECEIT: Sabotage Evaluations for Automated AI R&D, 2025, [2511.09904].
205. Gan, E.; et al. Auditing Sabotage Bench: A Benchmark for Detecting and Fixing Research Sabotage in ML Codebases, 2026, [2604.16286].
206. Aghakhani, H.; et al. TrojanPuzzle: Covertly Poisoning Code-Suggestion Models. IEEE Symposium on Security and Privacy, 2023, [2301.02344]. https://doi.org/10.1109/SP54263.2024.00140.
207. Yan, S.; et al. An LLM-Assisted Easy-to-Trigger Backdoor Attack on Code Completion Models: Injecting Disguised Vulnerabilities against Strong Detection. USENIX Security Symposium, 2024, [2406.06822].
208. Qu, Y.; et al. Supply-Chain Poisoning Attacks Against LLM Coding Agent Skill Ecosystems, 2026, [2604.03081].
209. Kang, S.; Milliken, L.; Yoo, S. Identifying Inaccurate Descriptions in LLM-generated Code Comments via Test Execution, 2024, [2406.14836].
210. Dassen, M.; et al. FACTUM: Mechanistic Detection of Citation Hallucination in Long-Form RAG, 2026, [2601.05866].
211. Wallat, J.; et al. Correctness is not Faithfulness in RAG Attributions, 2024, [2412.18004].
212. Qi, J.; et al. Model Internals-based Answer Attribution for Trustworthy Retrieval-Augmented Generation. Conference on Empirical Methods in Natural Language Processing, 2024, [2406.13663]. https://doi.org/10.18653/v1/2024.emnlp-main.347.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 October 2026 doi:10.20944/preprints202610.0808.v1

213. Wu, D.; et al. Synchronous Faithfulness Monitoring for Trustworthy Retrieval-Augmented Generation. Conference on Empirical Methods in Natural Language Processing, 2024, [2406.13692].
214. Sakai, Y.; Kamigaito, H.; Watanabe, T. HalluCitation Matters: Revealing the Impact of Hallucinated References with 300 Hallucinated Papers in ACL Conferences, 2026, [2601.18724].

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.