Reasoning-Aware Hallucination Detection in Large Language Models:

Evidence, Localization, and Faithful Process Diagnosis

Fatemeh Hadizadeh
fatemeh.hadizadeh97@sharif.edu

Mahsa Jalali
mahsa.jalali@ce.sharif.edu

Mostafa Karbalaei
mostafa.karbalaei@ce.sharif.edu

Fatemeh Shahhosseini
f.shahhosseini.20@gmail.com

Mohammad Shirkhani
muhammad.shirkhani@gmail.com

Danial Parnian
danielparnian@gmail.com

Department of Computer Engineering, Sharif University of Technology

Abstract

Large language models can produce fluent, coherent outputs that are factually unsupported, making it difficult to know when a response should be trusted. Reasoning-capable models add long intermediate traces, but process validity can diverge from final-answer correctness. Reasoning-aware hallucination detection therefore spans several distinct tasks: deciding whether an answer is unreliable, locating an erroneous transition or unsupported span, and assessing whether a displayed rationale faithfully reflects the computation that influenced the answer. We review thirty-one papers covering output uncertainty, internal-state and geometric signals, reasoning-trajectory analysis, fine-grained localization, detector-guided control, and audits of reasoning evidence. We organize the literature along prediction target, evidence source, granularity, model access, supervision, and the role assigned to reasoning. This view keeps methods with similar scores but different labels or evidential commitments separate, while still supporting comparisons where their mechanisms genuinely overlap. Across the corpus, recurring challenges include stable errors that survive self-consistency checks, supervision that implicitly defines the target, process signals that shift with model or prompting choices, and the gap between predictive association and causal or externally grounded explanation. We close with evaluation priorities that align labels across granularities, test explicit distribution shifts and stable misconceptions, audit supervision, and measure detector-guided interventions end to end.

Keywords: hallucination detection; large reasoning models; chain of thought; hidden states; uncertainty; process supervision; faithfulness; calibration.

1 Introduction

Large language models can produce answers that are fluent and internally coherent while being unsupported, factually wrong, or derived from an invalid inference. Hallucination detection therefore concerns more than whether a model sounds uncertain. The detector may need to decide whether a complete answer should be trusted, identify the first faulty step in a reasoning trace, mark an unsupported span, or provide a signal that can be used to verify, regenerate, or retrain the model. These tasks become more explicit in large reasoning models (LRMs), where a long chain of thought provides additional observations but also introduces new failure modes: an early false premise can be propagated, a reflection step can reinforce rather than correct an error, and a plausible written rationale can differ from the computation that actually determined the answer.

The corpus draws evidence from several points in the generation pipeline. Some methods estimate uncertainty from repeated outputs, while others use hidden states, logits, attention, or geometric summaries of internal representations (Farquhar et al., 2024; Su et al., 2024; Sriramanan et al., 2024; Chen et al., 2024; Hu et al., 2025; Ding et al., 2025). Process-oriented work treats the reasoning trajectory itself as evidence, using agreement across traces, changes over transformer depth or generation time, local transitions, graph structure, or routing between reasoning and answer tokens (Wang et al., 2025; Sun et al., 2025; Lu et al., 2026; Alvarez and Baheri, 2026; Chen et al., 2026; Zhang et al., 2026b; Liu et al., 2026). Hybrid and fine-grained methods combine views, localize failures below the whole-trace

level, or turn detector signals into training or test-time control (Song et al., 2025; Zhang et al., 2026a; Li et al., 2025; Min et al., 2026; Su et al., 2025; Li and Ng, 2025). A separate set of audits asks whether these signals remain meaningful when reasoning mode, reflection, operating threshold, or trace faithfulness changes (Cheng et al., 2025; Chegini et al., 2025; Lu et al., 2025; Zollicoffer et al., 2026; Shen et al., 2026).

These approaches should not be collapsed into one leaderboard. A multi-sample consistency score is informative about stochastic disagreement but may miss a stable false belief. A supervised latent probe can predict a dataset label without establishing that its discriminative direction is a domain-invariant representation of truth. A first-error localizer addresses a different target from response-level rejection, and a faithfulness benchmark addresses whether the visible trace is an adequate account of model computation rather than whether the final proposition is true. Several audit papers make these distinctions operational by showing that reasoning can change detector calibration, low-FPR recall, reflection behavior, or sensitivity to endpoint artifacts (Cheng et al., 2025; Chegini et al., 2025; Lu et al., 2025; Zollicoffer et al., 2026; Shen et al., 2026).

Scope. We review thirty-one papers that directly study hallucination detection, reasoning-process diagnosis, fine-grained localization, closely coupled mitigation, or evaluation needed to interpret reasoning-aware detection. Retrieval or external factual verification is discussed only when it appears as evidence inside a surveyed method. Chain-of-thought faithfulness is included when it changes what a detector may legitimately infer from a displayed reasoning trace. We do not treat answer correctness, process validity, and faithfulness as interchangeable labels.

Contributions. The survey contributes:

- a multi-axial taxonomy that separates prediction target, evidence source, granularity, model access, supervision, and the role assigned to reasoning;
- a source-first technical synthesis that preserves each paper's motivating problem, defining signal, access and supervision assumptions, and evaluated target before making cross-paper comparisons; and
- a set of literature-grounded research gaps concerning stable model errors, supervision-induced task definitions, process-signal shift, and the distinction between predictive evidence and causal or externally grounded explanation.

The survey proceeds from target definitions and taxonomy to output- and latent-evidence methods, reasoning-trajectory detectors, hybrid and fine-grained methods, and audits of the evidence itself. We then synthesize the recurring scientific trade-offs and formulate open problems that can be tested with explicit evaluation protocols.

2 Foundations and common problem setting

2.1 Three reliability targets

The phrase "reasoning-aware hallucination detection" covers at least three related but distinct targets.

Answer correctness. A final answer is incorrect when it conflicts with an accepted reference, supplied evidence, a deterministic computation, or a factual annotation. This is the target used by many response-level datasets. It identifies an unreliable outcome but does not specify where the failure entered the generation.

Process validity. A reasoning trace $r = (s_1, \ldots, s_T)$ is invalid when one or more steps contain an unsupported claim, incorrect computation, contradiction, or faulty inference. Process validity is not determined by the final answer alone: a model can reach a correct answer through a faulty intermediate step, or produce a wrong final answer after an otherwise valid derivation. Step-, prefix-, type-, token-, and span-level methods make this distinction explicit.

Trace faithfulness. Faithfulness asks whether the displayed rationale reflects the information or computation that actually influenced the answer. A trace can be logically valid yet post-hoc, or it can faithfully reveal an internal mistake. FAITHCOT-BENCH studies this target directly, while the FORCE/REMOVE tests of Zollicoffer et al. ask

whether a trace score changes when endpoint information is manipulated without changing the intended reasoning body (Shen et al., 2026; Zollicoffer et al., 2026).

Figure **??** is a schematic illustration of why the three labels should be kept separate.

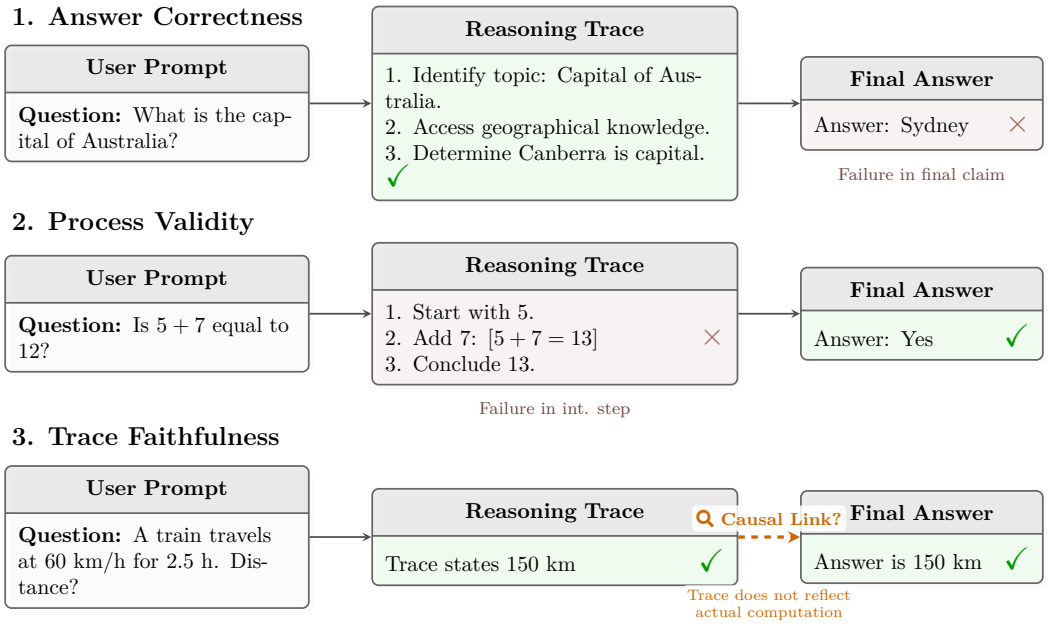

Figure 1: Visual differentiation of three distinct reasoning-aware hallucination types, highlighting the gap between correctness, validity, and actual faithfulness.

2.2 Notation and granularity

Let x be the input prompt or source context and $a = (a_1, \ldots, a_m)$ the final answer. When an explicit reasoning trace is available, write $r = (s_1, \ldots, s_T)$ for its steps or segments. At transformer layer ℓ, $h_{t,j}^{(\ell)}$ denotes the hidden representation of token j in step t, $p_{t,j}$ the next-token distribution, and $A^{(\ell)}$ an attention map. A detector can be written abstractly as

$$D(x, r, a; \mathcal{E}) \to [0, 1], \tag{1}$$

where $\mathcal{E}$ is the evidence available at inference time. Because the papers use opposite score conventions, we refer to the score semantically as risk or confidence rather than assuming that larger always means more hallucinatory.

The prediction unit is itself part of the task definition. Response-level detectors accept or reject a full output; trace-level detectors score the reasoning chain as a whole; step-level methods label individual transitions or the first error; prefix-level methods estimate whether the partial chain has entered an unreliable state; type-level methods classify the kind of process error; and token/span methods localize unsupported text. Accuracy at one granularity does not imply accuracy at another.

2.3 Evidence, access, and supervision

Across the corpus, the main evidence sources are: (i) variation in sampled outputs or their probabilities; (ii) hidden states, logits, attention, or learned geometric summaries; (iii) explicit reasoning traces and their temporal or relational structure; (iv) combinations of internal and symbolic views; (v) counterfactual interventions or self-judgments; and (vi) externally supplied evidence or verifiers grounded in that evidence. These sources answer different questions. Disagreement measures instability; latent separability measures whether labels are decodable from a representation; trace scores measure properties of an observable reasoning process; and external references constrain factual support.

We distinguish black-box methods, which require generated text only, from gray-box methods that additionally expose limited generator-side quantities such as probabilities, and white-box methods that require activations,

attention, logits, gradients, or parameters. A separate deployment pattern uses a black-box generator together with a white-box auxiliary model, as in LLM-Check; evidence from that auxiliary model should not be confused with access to the generator's own internals. Supervision ranges from training-free scores through pseudo- or self-supervision, human or synthetic labels, preference optimization, and reinforcement learning. These distinctions matter because a method can avoid manual labels while still inheriting supervision from a model-derived confidence score, a critic, a synthetic corruption process, or a verifier.

Evidence and interpretation. For the critical analysis that follows, we keep three statements separate. A *measurement claim* specifies the statistic computed by a method. A *predictive claim* states that the statistic is associated with an evaluation label under a given protocol. A *mechanistic or causal claim* concerns why the error occurred or which internal computation was responsible. The first two do not automatically establish the third. This distinction is especially important for methods that name a latent direction, routing pattern, graph structure, or trajectory geometry.

2.4 Evaluation and comparability

The papers report AUROC, AUPRC, accuracy, F1, calibration measures, selective accuracy, risk–coverage curves, and low-false-positive-rate recall, among other metrics. These values are meaningful only together with the target label, class balance, benchmark construction, and access regime. Semantic Entropy evaluates response-level confabulation; GeoReason targets the first erroneous transition; FG-PRM predicts step error types; TokenHD and RL4HS localize text; Chegini et al. emphasize behavior at strict operating points (Farquhar et al., 2024; Alvarez and Baheri, 2026; Li et al., 2025; Min et al., 2026; Su et al., 2025; Chegini et al., 2025). We therefore compare mechanisms and evidence requirements across papers but do not pool incompatible scores into a common performance ranking.

3 A multi-axial taxonomy

No exclusive partition captures the surveyed literature well. A white-box method can still make only a response-level decision; a trace method can be training-free or supervised; a benchmark can also provide a detector; and a detector can later become a training or inference signal. We therefore describe each paper along five independent axes:

1. **Primary evidence**: sampled outputs, internal state/geometry, explicit trace/process, joint evidence, or external grounding.
2. **Relation to reasoning**: reasoning may be absent from the detector, observed as an explicit or latent process, used as an elicitation or shaping instrument, targeted during training, or treated as a condition to audit.
3. **Granularity**: response, trace, step, prefix, type, token, or span.
4. **Contribution role**: detector/localizer, benchmark/audit, mitigation/training, or a combination.
5. **Access and supervision**: generator black/gray/white-box access, possible auxiliary-model access, and training-free, pseudo/self-supervised, supervised, preference-trained, or RL-trained learning.

The primary placement in Table 1 is used only to organize the narrative; it is not meant to deny secondary roles. When a paper also contributes a benchmark, audit, training signal, or inference policy, that accompanying role is recorded in the last column.

3.1 Where evidence enters the detection pipeline

Figure **??** maps the main locations at which evidence can be observed or deliberately elicited. Cross-cutting operations such as fusion, localization, auditing, and detector-guided control are shown separately because they can be applied to more than one evidence source.

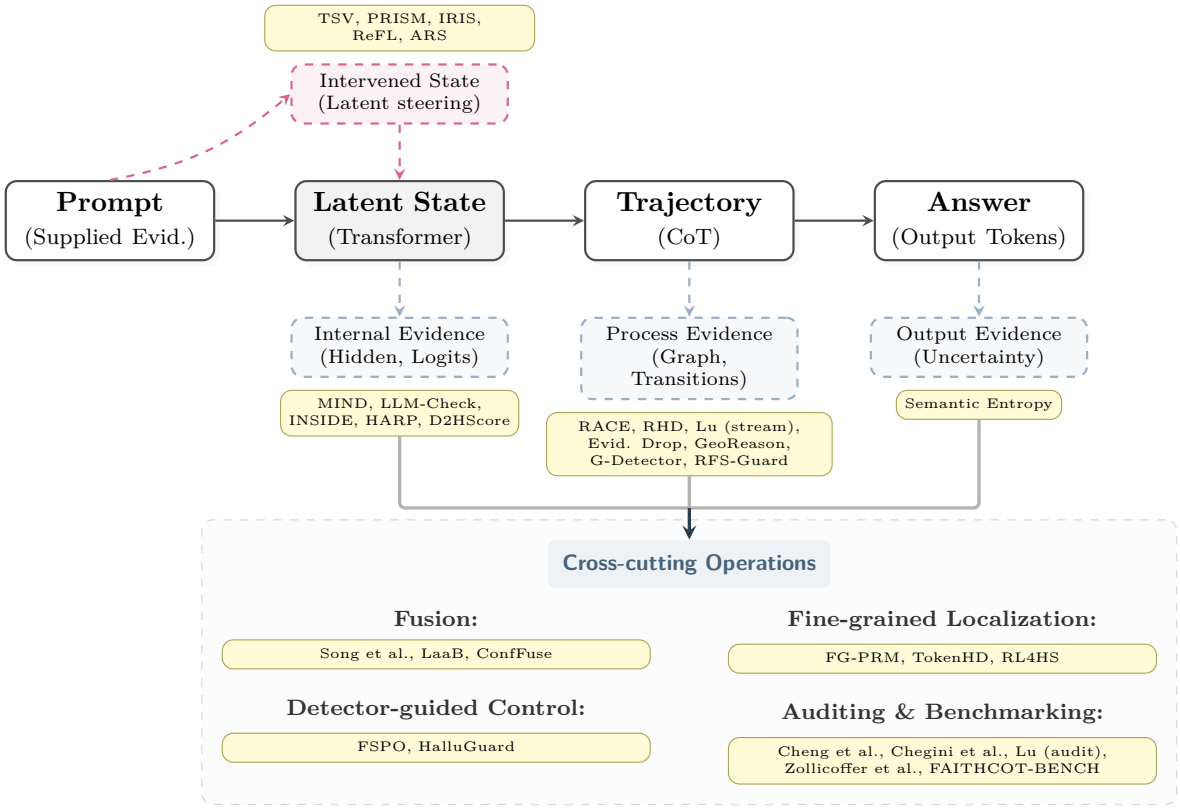

Figure 2: Mapping evidence locations across the LLM generation pipeline. Different methods observe or intervene at distinct computation stages. Cross-cutting operations apply these signals to fusion, fine-grained tasks, control, or auditing.

3.2 Taxonomy of the thirty-one papers

Abbreviations in Table 1: R=response, Tr=trace/chain, S=step, P=prefix, Ty=type, Tok=token, Sp=span. Training/access abbreviations are TF=training-free, Sup=supervised, WB=white-box, BB=black-box, DPO=Direct Preference Optimization, PRM=process reward model, and RL=reinforcement learning.

Table 1: Multi-axial taxonomy of the thirty-one surveyed papers. The last column summarizes the problem addressed or an accompanying contribution.

Method	Primary evidence	How reasoning enters	Unit	Access / training	Problem addressed / accompanying contribution
Semantic Entropy	Meaning-level entropy over sampled answer clusters	Not explicit	R	BB or logprobs; TF	Detects confabulation by estimating uncertainty over semantic equivalence classes rather than surface forms.
MIND	Final-layer last-token contextual state	Not explicit	R / sentence / passage	WB; pseudo-label training	Real-time detector trained without manual hallucination labels; also introduces HELM with human labels and recorded internals from six LLMs.
LLM-Check	Hidden states, attention kernels, output probabilities	Not explicit	R	Generator WB; or generator BB + auxiliary WB; TF	Training-free single-response scores for zero-resource, multi-response, and reference-aware settings; supports a substitute model when generator internals are unavailable.
INSIDE	Covariance spectrum of sampled internal embeddings	Not explicit	R	WB; TF	EigenScore measures multi-response consistency in representation space; feature clipping is studied for overconfident generations.
RACE	Trace consistency, answer uncertainty, alignment, coherence	Explicit process object	Tr / R	BB/gray; TF	Jointly evaluates reasoning and answer behavior with four components after CoT extraction.
HARP	SVD-derived projection of hidden states	Latent reasoning signal	R	WB; Sup	Constructs an unembedding-derived reasoning subspace and trains a hallucination classifier on the projection; includes cross-dataset tests.

Continued on next page

Table 1 continued

Method	Primary evidence	How reasoning enters	Unit	Access / training	Problem addressed / accompanying contribution
RHD	Late-layer vocabulary-projection dynamics	Process object	S / Tr	WB; TF score / RL mitigation	Defines a Reasoning Score for reasoning hallucination detection and reuses it in GRPO-R reward shaping.
HalluGuard	NTK-based data support and rollout instability	Reasoning as source component	R / Tr	WB; self-supervised calibration	Combines data-driven and reasoning-driven risk terms for detection and guided test-time inference.
ARS	Answer agreement under latent perturbation	Reasoning-conditioned instrument	R	WB; self-supervised shaping / optional Sup probe	Learns a representation from counterfactual answer agreement and supplies it to downstream embedding detectors.
Song et al.	Internal states plus direct/CoT/reverse paths	Explicit process object	Tr / R	WB; Sup	Converts CoT into a Semantic Trajectory List and aligns latent and symbolic views with segment-aware temporalized cross-attention.
Lu et al. (streaming)	Local step evidence plus cumulative prefix state	Explicit process object	S / P	WB; Sup	Performs online hallucination monitoring in long CoT with local step and prefix-level predictions.
D2HScore	Intra-layer dispersion and attention-guided inter-layer drift	Latent process signal	R	WB; TF	Defines training- and label-free semantic breadth/depth scores from one generation.
GeoReason	Hidden-state transport features	Process object	S / first error	WB; step labels + distillation	Uses a contrastive-PCA teacher and a distilled BiLSTM student for first-error localization.
ConfFuse	Global chain confidence plus local PRM confidence	Explicit process object	Tr / S	GHDM (DPO) + PRM; Sup	Fuses a DPO-trained global evaluator with minimum local process confidence for mathematical reasoning.
FG-PRM	Six type-specific process reward models	Explicit process object	S / Ty	Synthetic Sup	Defines six reasoning-error types, synthesizes type-specific corruptions, and trains PRMs used for detection and candidate ranking.
Cheng et al.	Detector behavior under CoT prompting	Reasoning condition (audit)	R	Audit; BB/WB	Matched audit of probability-, consistency-, self-evaluation-, and internal-state detectors with and without CoT.
Lu et al. (audit)	Claim confidence, reflection, intervention behavior	Explicit process object (audit)	Tr / claim	BB; Audit	Controlled study of hallucination emergence and propagation under reflection, including meta-cognitive confidence and chain disloyalty.
Zollicoffer et al.	FORCE/REMOVE interventions and lexical trajectories	Trace condition (audit)	Tr	BB; TF/Audit	Introduces trace-score sanity checks and the text-based TRACT detector for long-form outputs.
RL4HS	Generative hallucination-span prediction	Reasoning instrument	Sp	RL; trained	Trains a reasoning model for span localization with span-F1 rewards, GRPO, and class-aware policy optimization.
G-Detector	Directed graph built from clustered step representations	Explicit process object	Tr	WB; Sup	Analyzes reasoning-graph structure, trains a GNN detector, and filters high-risk cold-start traces before SFT.
Chegini et al.	Think-on/think-off detector scores at strict operating points	Reasoning condition (audit)	R	Audit; BB	Evaluates how reasoning changes safety and hallucination detection, with emphasis on recall at low false-positive rates.
RFS-Guard	Cross-phase attention routing plus semantic proximity	Explicit process object	S / Tr	WB; TF/lightweight	Uses multi-hop reason-flow backtracking and Routing Focus Score for reasoning hallucination detection/localization.
TokenHD	Token detector trained from critic-aggregated labels	No fixed reasoning unit	Tok	Critic-aggregated Sup	Restores critic-proposed fragments to exact spans, aggregates multiple critics, and trains an importance-weighted token-level detector.
FSPO	Evidence-grounded sentence factuality inside policy optimization	Reasoning as training target	S / policy	Verifier + RL	Modifies token-level policy advantages using step-wise factuality verification during reasoning RL.
FAITHCOT-BENCH	Expert trace-faithfulness annotations and benchmarked signals	Explicit process object (audit)	Tr / instance	Benchmark / audit	Releases FINE-COT and evaluates counterfactual, logit-based, and judge approaches to instance-level CoT faithfulness.
Chen et al. (Evidence Drop)	Largest local drop in an evidence score	Explicit process object	S / Tr	Evidence/logit access; TF	Single-trace, training-free risk scoring with approximate localization from local evidence dynamics.
IRIS	Verification-response state plus model-derived uncertainty target	Verification instrument	R	WB; self/pseudo-supervised	Trains a probe without human labels by using a verification response for both the latent feature and a soft truthfulness target.
PRISM	Prompt-conditioned true/false geometry	Truth-evaluation instrument	R	WB; Sup	Selects a truth-evaluation prompt using labeled geometric criteria and trains a detector with cross-domain/syntactic transfer evaluation.

Continued on next page

Table 1 continued

Method	Primary evidence	How reasoning enters	Unit	Access / training	Problem addressed / accompanying contribution
LaaB	Response features plus self-judgment features	Self-judgment instrument	R	WB + self-judgment; Sup	Links response and self-judgment predictions through a logical label constraint and mutual learning.
ReFL	Corrective in-context feedback and final hidden state	Corrective-feedback instrument	R	WB; Sup	Conditions the detector state on corrective triplets while keeping the base LLM frozen; evaluates structural transfer.
TSV	Truthfulness Separator Vector plus OT pseudo-labeling	No explicit trace; latent intervention	R	WB; semi-supervised	Learns a latent steering vector from few labeled exemplars and augments training with confidence-filtered unlabeled generations.

Prediction granularity changes what a detector can support downstream. Figure **??** summarizes the main units without implying that finer is always better: local predictions are useful only when their labels identify a meaningful target and the system has an action to take at that resolution.

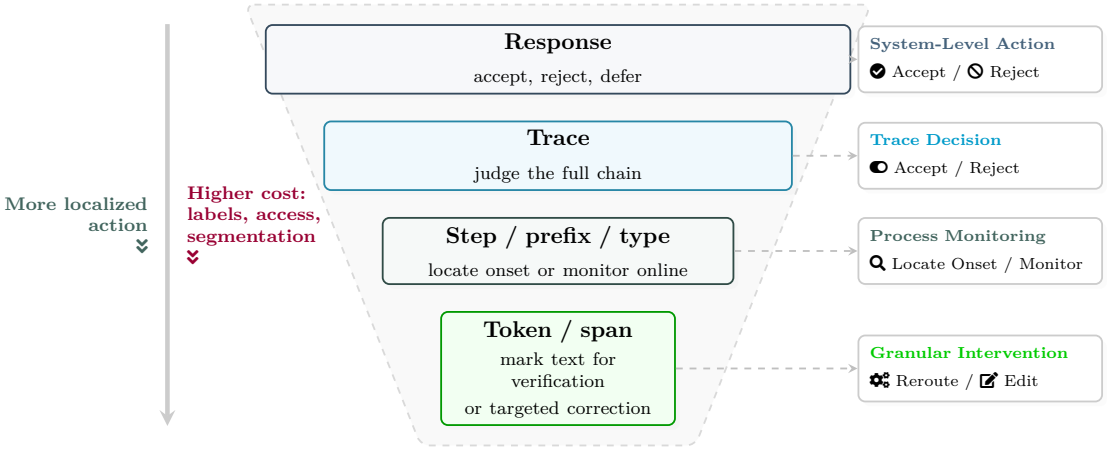

Figure 3: Visualizing the prediction granularity hierarchy. Granularity is a task choice representing a trade-off: finer resolution enables targeted, granular actions like rerouting or editing but requires more demanding labels, segmentation, or architectural access.

4 From output uncertainty to latent evidence

The methods in this section differ in what they treat as the primary observable. Some estimate uncertainty from several generated answers; some inspect a state produced by one response; some summarize geometry over samples or layers; and some deliberately change the state or the context in which it is elicited before a classifier is applied. The distinctions among them therefore concern both the information available to the detector and how that information is constructed.

4.1 Semantic uncertainty over sampled answers

Semantic Entropy addresses a mismatch between token-level uncertainty and uncertainty about meaning (Farquhar et al., 2024). The method samples several answers to the same question and groups answers into semantic equivalence classes, using semantic entailment so that paraphrases of the same proposition are treated as one outcome. Entropy is then computed over the probability mass assigned to these meaning classes; the paper also develops a discrete estimator that can operate when only black-box generations are available. The intended target is *confabulation*: cases in which lack of knowledge is expressed as unstable or mutually incompatible semantic answers.

The construction is useful because it removes surface-form diversity from the uncertainty estimate, but its scope follows directly from that design. A model that repeatedly produces one false proposition can have low semantic entropy. The paper also gives a case in which the clustering separates answers that provide an exact date from answers that provide only a year even though that distinction is probably irrelevant in context. Semantic Entropy is therefore a strong reference method for meaning-level uncertainty, not a truth test for every form of hallucination.

4.2 Single-response internal signals: MIND and LLM-Check

MIND is motivated by the latency of post-hoc multi-sample detection and the cost of manual hallucination labels (Su et al., 2024). It constructs training examples automatically from Wikipedia. A passage is truncated at a selected entity; the target LLM continues the passage; the continuation is labeled non-hallucinated when it recovers the held-out entity and hallucinated otherwise; and the model's internal states are recorded during generation. After comparing representations from different positions and layers, MIND trains a small MLP on the last token's contextual representation from the final transformer layer. At inference, the classifier can be evaluated from the state already produced during generation. The same paper releases HELM, which contains human-annotated outputs from six LLMs together with contextual embeddings, self-attention, and hidden-layer activations.

LLM-Check is also designed for single-response detection but avoids training a task-specific detector (Sriramanan et al., 2024). It defines a suite of scores from one forward pass, including output-probability measures and eigenvalue-based summaries of hidden activations and self-attention kernel maps. The paper evaluates these scores in zero-resource settings, with additional sampled responses, and with external references. When the original generator does not expose internal states, the generated response is teacher-forced through an auxiliary model and the auxiliary model's internal observables are used to compute the scores.

The common practical objective is low-cost response scoring, but the supervision and provenance of the evidence differ. MIND learns a classifier whose target is determined by an automatic continuation task; LLM-Check leaves the scoring rule training-free but, in the API-generator setting, the internal evidence belongs to a substitute model rather than to the generator. Consequently, the two methods have different validation dependencies: MIND must be tested beyond the continuation process that creates its training labels, whereas the auxiliary-model setting of LLM-Check must test whether one model's internal response to the text remains informative about errors produced by another.

4.3 Multi-sample internal geometry: INSIDE

INSIDE keeps the repeated-sampling setting but measures disagreement in the model's internal sentence-representation space (Chen et al., 2024). For embeddings $z_1, \ldots, z_K$ of sampled responses, the method forms a regularized covariance matrix Σ and defines EigenScore by the log determinant

$$\text{EigenScore}(x) = \frac{1}{K} \log \det(\Sigma + \alpha I) = \frac{1}{K} \sum_i \log \lambda_i, \tag{2}$$

where λ_i are the eigenvalues of the regularized covariance. This statistic aggregates the spread of the response representations across principal directions, and the paper connects it to differential entropy in embedding space. INSIDE also studies test-time feature clipping: unusually large penultimate-layer activations are truncated using thresholds derived from a memory bank in order to reduce overconfident generation and improve detection of self-consistent hallucinations.

Compared with Semantic Entropy, INSIDE changes the space in which multi-sample variation is measured, not the basic epistemic source of the signal. It can expose representation-level variation that decoded language may hide, but a compact representation cloud can still correspond to a shared false belief. Its additional contribution is the clipping intervention, which changes generation rather than merely observing the resulting response.

4.4 Single-generation geometric summaries: HARP and D2HScore

HARP starts from the paper's hypothesis that raw hidden states contain both semantic-expression information and reasoning-related information, and that the latter is more useful for hallucination detection (Hu et al., 2025). The method treats the hidden space as a direct sum of semantic and reasoning subspaces. It analyzes the unembedding matrix, applies singular-value decomposition, and uses the resulting singular directions to construct bases for the two components. Hidden states are then projected onto the low-dimensional component designated as the reasoning subspace, and a supervised classifier predicts hallucination from that projection. The paper evaluates the detector on several QA datasets and includes cross-dataset robustness experiments.

D2HScore uses a different geometric construction and does not train a detector (Ding et al., 2025). From one forward pass, it defines *semantic breadth* as dispersion among token representations within a layer. It defines *semantic depth* by selecting important tokens with attention and measuring how their representations move across adjacent layers. The two normalized quantities are combined into a label-free score intended to capture within-layer spread and inter-layer refinement.

These methods are grouped only because both summarize internal geometry from a single generation. Their scientific commitments are different. HARP's reasoning/semantic decomposition is a substantive interpretation attached to an SVD-derived projection and a supervised decision rule; D2HScore names two fixed geometric statistics and evaluates their predictive utility without fitting labels. Successful detection establishes that these constructions carry useful information under the tested settings. It does not, by itself, show that the selected HARP directions are a universal reasoning subspace or that breadth/depth constitute causal mechanisms of hallucination.

4.5 Elicited or intervened representations before classification

TSV, PRISM, IRIS, and ReFL share one structural choice: the detector is not applied to the original response representation unchanged. TSV intervenes directly on a latent state; PRISM and IRIS change the context in which the representation is elicited; ReFL conditions the state on corrective feedback. Their problem formulations, supervision, and evaluation goals are otherwise different (Park et al., 2025; Zhang et al., 2025; Srey et al., 2025; Fan et al., 2026).

TSV. TSV begins from the observation that truthful and hallucinated outputs can overlap in the default embedding space and that labeled hallucination data are scarce (Park et al., 2025). A small labeled exemplar set is used to learn a Truthfulness Separator Vector. The vector is added to an intermediate hidden state while the base LLM remains frozen; the downstream layers then produce a modified final representation in which the two labeled classes are encouraged to be more separable. A second stage brings unlabeled generations into training through optimal-transport assignment to class prototypes and confidence filtering. The learned vector and prototypes are then used for single-sample detection, and the paper evaluates the separator on unseen datasets.

PRISM. PRISM is explicitly motivated by cross-domain generalization of hidden-state detectors (Zhang et al., 2025). The paper argues that a default final-token representation can mix truth-related variation with domain-specific variation, and uses a truth-evaluation prompt to change the representation before probing. For a candidate prompt, labeled true and false examples define a truthfulness direction as the difference between the two class means. PRISM measures how much of the total representation variance lies along that direction and separately measures the consistency of the resulting directions across domains. Candidate prompts are selected using this labeled geometric criterion, after which the prompt-conditioned final-token states are used to train the final detector. Prompt selection and detector training are therefore supervised even though the prompting operation itself does not update the base LLM.

IRIS. IRIS addresses a different problem: unsupervised internal-state probes can be trained with proxy labels that do not closely track truth and may therefore learn shortcuts (Srey et al., 2025). For each statement, the LLM is prompted to verify whether the statement is true. The final-token, final-layer representation of this verification response becomes the detector feature. The same verification process produces a soft target from verbalized confidence or a reasoning-token uncertainty signal. A lightweight probe is trained on these self-derived targets, with soft bootstrapping and symmetric cross-entropy used to reduce sensitivity to noisy pseudo-labels. The method requires only unlabeled statements rather than human truth labels, and the paper evaluates topic- and dataset-level out-of-distribution settings in addition to in-domain performance.

ReFL. ReFL introduces explicit corrective feedback into the representation used for supervised detection (Fan et al., 2026). Training examples are organized as corrective triplets containing an input statement, the model's predicted factuality label, and the ground-truth label. These examples are placed in-context so that the model sees both successful and corrected predictions; the base LLM remains frozen. A detector is trained on the final last-token hidden state produced under this corrective context. The paper varies the amount of correction in the demonstrations and evaluates whether the resulting representation transfers to unseen logical or syntactic structures. The methods differ along two directly comparable dimensions: how the detector representation is produced or altered before classification, and how the resulting detector is supervised. TSV modifies the latent state and uses semi-supervised prototype construction; PRISM uses labeled data to select a truth-evaluation prompt with cross-domain geometry as an explicit objective; IRIS uses verification to create both the feature and a self-derived target without human labels; and ReFL uses ground-truth corrective demonstrations to condition a supervised detector. Their empirical claims should remain tied to the particular dataset, domain, or structural setting evaluated by each paper.

5 Reasoning-process evidence: trajectories, transitions, routing, and stability

Response-level and latent methods can flag an unreliable output without showing how reliability changes across an extended reasoning process. The papers in this section use process information directly: explicit traces, internal states aligned with reasoning steps, relations among steps, perturbations at the reasoning–answer boundary, or rollout dynamics. Their scores target different process properties, so global consistency, temporal evolution, local transitions, relational structure, counterfactual stability, and source decomposition are treated separately.

5.1 Global reasoning and answer consistency: RACE

RACE is designed for LRMs whose sampled outputs contain both explicit reasoning traces and final answers (Wang et al., 2025). After a CoT extraction module removes irrelevant or noisy reasoning content, the framework computes four components: consistency among reasoning traces across samples, semantic uncertainty among final answers, alignment between the main reasoning trace and the sampled answer space, and internal coherence of the reasoning trace. These components are combined into a global hallucination score. The method is evaluated in black- or gray-box settings without task-specific detector fine-tuning.

RACE extends answer-only self-consistency by asking whether the process agrees across samples and whether the answer is supported by the extracted trace. Its remaining boundary follows from the evidence source: all four components are generated by the same model. Several samples can repeat one false premise, and a trace can be internally coherent and well aligned with a wrong final answer. The method therefore gives a richer global consistency signal but does not by construction supply external factual grounding or a first-error label.

5.2 Dynamics across transformer depth and generation time: RHD and streaming detection

RHD analyzes reasoning hallucination through changes in internal predictions across transformer layers (Sun et al., 2025). Hidden representations from late layers are projected through the vocabulary head, producing a sequence of token distributions across depth. Divergence between these layer-wise distributions is summarized as a Reasoning Score, which the paper uses to characterize shallow pattern matching, fluctuations, and incorrect backtracking in reasoning traces. The score is then used for hallucination detection and, in a second contribution, as a step-level reward signal in GRPO-R.

Lu et al.'s streaming detector instead models hallucination risk as a state that evolves as the visible CoT is generated (Lu et al., 2026). A local detector extracts a hallucination signal from the hidden representation of each reasoning step, while a temporal component accumulates these observations into a prefix-level state. The resulting system can update its risk estimate online and is evaluated with dynamic metrics designed to capture response to error onset, correction, and false alarms. The method is supervised and requires white-box access to the monitored model.

Both papers introduce temporal structure, but they use different notions of time. RHD follows computation through model depth for a reasoning step; the streaming method follows the sequence of generated reasoning steps and prefixes. This distinction determines what can be localized and when a downstream system can act. The first yields an internal depth-based signal that is later reused for RL; the second produces an online prefix state whose meaning depends on the step segmentation and temporal labels used in training.

5.3 Local transitions near the first error: Evidence Drop and GeoReason

Chen et al. motivate Evidence Drop by observing that sequence-average uncertainty can confuse difficulty with error (Chen et al., 2026). A difficult but correct trace may remain uncertain throughout, while an otherwise confident trace can suffer a sudden loss of support at one transition. The paper models a reasoning trajectory through a local evidence score and defines hallucination risk by the worst local decrease, or Evidence Drop, along the trajectory. The procedure is training-free, requires a single trace, and provides both a sequence-level risk score and an approximate location of the largest drop.

GeoReason targets the first erroneous transition with supervised hidden-state geometry (Alvarez and Baheri, 2026). A contrastive-PCA teacher constructs a trace-specific projection that emphasizes the difference between labeled correct and first-error transitions. Within that projected space, the teacher represents each transition through a feature set that includes projected position, magnitude and normalized magnitude, velocity, acceleration, rolling energy, and directional persistence. An MLP scores the teacher features. For deployment, a BiLSTM student is

trained from the hidden-state sequence using the step labels together with probability and feature distillation from the teacher, so inference no longer requires constructing a labeled teacher projection.
The two methods are comparable because both target local change near error onset, but they operationalize that target differently. Evidence Drop monitors a scalar evidence trajectory without detector training; GeoReason learns which hidden-state transition patterns correspond to labeled first errors. The former inherits the semantics of the chosen evidence score, while the latter inherits the annotation definition and must transfer a teacher-derived separation into the student. The paper's teacher–student results make that transfer gap an empirical part of the method rather than an abstract concern.

5.4 Relations among reasoning steps: G-Detector and RFS-Guard

G-Detector is built from a graph representation of long-CoT reasoning (Zhang et al., 2026b). The trace is segmented into reasoning steps, each step is represented by the mean of its token hidden states at a chosen layer, and the pooled step embeddings from the corpus are clustered with k-means. Cluster centroids define graph nodes; consecutive step assignments define directed edges. The paper first analyzes a collection of graph-theoretic properties of these reasoning graphs and then trains G-Detector, a graph neural network whose node features are initialized with the centroid embeddings. Message passing is supplemented with clique-level information before graph-level classification. The detector is also used to filter high-risk traces during cold-start supervised fine-tuning.
RFS-Guard focuses on the interface between the reasoning phase and the final answer (Liu et al., 2026). It aggregates self-attention between reasoning and answer steps, backtracks multi-hop attention paths through the reasoning chain, and combines this routing information with semantic proximity between step representations. The resulting Routing Focus Score is used to detect and localize reasoning hallucinations by combining cross-phase routing focus with semantic proximity. The method is training-free/lightweight but requires hidden states, attention maps, and a segmentation of the reasoning trace; the authors explicitly present RFS as a correlational rather than causal relationship.
These methods both encode relations among steps, yet the relation is represented differently. G-Detector learns over a corpus-level clustering that creates a graph and supplies hidden-state centroids as node features, so the final classifier uses latent content as well as topology. RFS-Guard retains the sequential trace and follows model routing between reasoning and answer representations. Their results support the predictive usefulness of the complete constructions under their respective protocols; they do not license attributing G-Detector's gain to topology alone or RFS-Guard's correlation to a causal information path.

5.5 Counterfactual answer stability: ARS

ARS uses a controlled latent perturbation to create supervision for representation shaping (Zhang et al., 2026c). At the boundary between the reasoning trace and answer generation, the penultimate-layer state is perturbed and an alternative answer is decoded. The original and counterfactual answers are compared for semantic agreement, producing a self-supervised agreement label. A learned mapping then pulls representations of agreement pairs together and pushes disagreement pairs apart; the resulting answer representation is supplied to downstream embedding-based detectors. The shaping objective itself does not use hallucination labels, while the paper evaluates both a label-free CCS scorer and a supervised probing variant downstream.
The directly measured quantity is answer stability under a particular latent intervention. That is more specific than ordinary sample consistency, because the alternative answer is generated after intervening on an internal state. It is nevertheless not identical to correctness: a false belief can be robust to the chosen perturbation, while a correct answer can change wording or interpretation. ARS is therefore best understood as constructing a counterfactual fragility signal and reshaping the representation around that signal.

5.6 Separating data support from reasoning instability: HalluGuard

HalluGuard formulates hallucination risk as the combination of two terms associated with different sources (Zeng et al., 2026). The data-driven component estimates how well the current example is supported by the model's learned representation using Neural Tangent Kernel (NTK) spectral quantities. The reasoning-driven component estimates instability during autoregressive reasoning through rollout behavior and amplification. These quantities are combined into a practical risk score used for hallucination detection and for guided test-time inference.
The decomposition is scientifically useful because it separates two operational questions—lack of support in learned data and instability during reasoning—that are often mixed into one confidence score. The source labels, however, are not directly observed on each example; they are inferred through the NTK approximation and rollout model

introduced by the paper. The experiments support the predictive value of these terms under the tested tasks. Establishing that they identify the actual causal source of an individual hallucination would require source labels or interventions that are independent of the score construction.

6 Hybrid evidence, fine-grained localization, and detector-guided control

The methods above often privilege one evidence source. The papers in this section either combine distinct views, refine the target below the whole-trace level, or place a detector inside a training or inference loop. These changes make the detector more actionable, but they also make supervision and calibration more consequential.

6.1 Combining internal states with structured auxiliary views: Song et al. and LaaB

Song et al. formulate a "detection dilemma" between sub-symbolic internal signals and explicit symbolic reasoning (Song et al., 2025). Their framework first generates three views of a query: a direct answer, a reasoning-augmented answer, and a reverse-inference query derived from the direct answer. The CoT is decomposed into a Semantic Trajectory List so that the symbolic trace is represented at a finer granularity. Hidden states and embeddings from the query, answer, reverse query, and semantic trajectory are then aligned with a segment-aware temporalized cross-attention module before classification. The method is supervised and is designed to combine internal factuality cues with discrepancies visible in structured reasoning.

LaaB combines a different pair of views: the original response and the model's explicit self-judgment of that response (Mi et al., 2026). The self-judgment is itself treated as a response whose intrinsic features can be evaluated. LaaB uses a logical relation to map the self-judgment prediction back to the factuality of the original response: if the judgment says the response is truthful, the two factuality labels should agree; if it says the response is false, the labels should be opposite. The response and self-judgment detectors are coupled through this constraint and trained with mutual learning. At inference, only the response detector is deployed, so the self-judgment generation and its detector do not add deployment-time cost.

Both frameworks are more structured than averaging two confidence scores, but their added views are not equivalent. Song et al. change the inference path and explicitly align heterogeneous latent and symbolic representations. LaaB uses a logical relation between a response and a self-evaluation generated by the same model. In both cases, empirical complementarity between views supports fusion for prediction, while factual independence remains a separate question: multiple paths can inherit one false premise, and a response and its self-judgment can be jointly wrong.

6.2 Combining whole-trace and local confidence: ConfFuse

ConfFuse is motivated by two limitations in mathematical reasoning detection: whole-answer judgments can ignore local logical flaws, while step-wise verifiers can flag an early error that the model later corrects (Zhang et al., 2026a). The framework trains a Global Hallucination Detection Model (GHDM) with Direct Preference Optimization to produce confidence for the complete reasoning chain. In parallel, a Process Reward Model provides confidence for each reasoning step. The final score fuses the global confidence with the minimum local confidence using a weighted rule. The paper evaluates both in-distribution and out-of-distribution mathematical datasets.

The construction makes the intended roles of the two components explicit: the global model sees the final state of the whole chain, while the minimum local score makes the detector sensitive to the weakest step. This combination can distinguish cases that either component misses, but it also makes cross-component calibration part of the method. A very low local score can dominate even when the chain later self-corrects, and a global score can absorb errors that the local PRM does not resolve. These are properties of the fusion rule, not evidence that one resolution is generally superior to the other.

6.3 Adding error type to process supervision: FG-PRM

FG-PRM replaces binary step correctness with six hallucination types for mathematical reasoning: context inconsistency, logical inconsistency, instruction inconsistency, calculation error, factual inconsistency, and fabrication (Li et al., 2025). To reduce the need for manual type-level annotation, the paper starts from problems with ground-truth reasoning, identifies steps at which a hallucination can be inserted, and prompts an LLM to generate type-specific

corrupted reasoning. It then trains six type-specific process reward models, one for each error class. The same reward models are evaluated not only for step/type detection but also as verifiers that rank candidate solutions.
The additional type label changes what the detector can support: an identified calculation error and an identified factual inconsistency can in principle trigger different verification tools. That benefit depends on whether the synthetic error generator produces examples representative of naturally occurring failures. The paper includes human-annotated evaluation, which is especially relevant for this question. The six classes are also a taxonomy of process errors, not a taxonomy of CoT faithfulness; a correct but post-hoc rationale lies outside this label space.

6.4 Token- and span-level localization: TokenHD and RL4HS

TokenHD argues that predefined reasoning steps are an inconvenient unit for free-form outputs and proposes direct token-level hallucination detection (Min et al., 2026). Its data engine asks several critic models to identify hallucinated text fragments. Because critic outputs may paraphrase or expand the original text, a restoration stage maps the proposed fragments back to exact spans in the source response, which are then converted to token-level labels. Repeated critiques are averaged, and labels from multiple critics are combined either uniformly or with an adaptive ensemble whose critic weights are learned on a held-out set with ground-truth scores. A detector is then trained on these soft token targets, with an importance-weighted objective that gives greater weight to sparse hallucinated tokens.
RL4HS treats span localization as a generative reasoning problem rather than a token-wise classification problem (Su et al., 2025). The paper first observes that CoT sampling can produce at least one high-quality span prediction as the number of samples increases. It then trains the model with Group Relative Policy Optimization using span-F1 as the main verifiable reward. Vanilla GRPO can become biased toward predicting no hallucination spans, producing a high-precision/low-recall failure mode; Class-Aware Policy Optimization (CAPO) adjusts the class-dependent advantages to reduce this reward imbalance. The final model generates the hallucinated spans directly.
The two papers remove fixed step boundaries through different learning problems. TokenHD separates label generation from a discriminative token detector and makes critic aggregation a central part of supervision. RL4HS lets the detector generate structured span outputs and moves the main design burden into reward construction and policy optimization. In both settings, the annotation boundary matters: an unsupported relation can span multiple tokens or sentences, so localization quality is not fully characterized by exact boundary agreement alone.

6.5 When detection becomes a training or inference signal

Li and Ng study hallucination as a consequence of reasoning-oriented reinforcement learning and propose FSPO to incorporate factuality into policy optimization (Li and Ng, 2025). Their empirical analysis reports that outcome-driven reasoning optimization can increase hallucinations in intermediate reasoning even when final-task performance improves. FSPO therefore verifies reasoning sentences against supplied evidence and converts the step-level factuality judgments into token-level advantage adjustments. The policy is optimized with both the outcome signal and the factuality-aware process signal, so unsupported intermediate claims can be penalized rather than receiving credit solely from a correct final answer.
FSPO is primarily a mitigation method, but its training objective depends on a detector: associated evidence and a factuality verifier define the step-level signal. In this setting, verifier error is not merely an evaluation error; it changes the policy update. Similar detector-to-control coupling appears in reward shaping, data filtering, candidate ranking, and guided test-time search across RHD/GRPO-R, G-Detector, FG-PRM, and HalluGuard (Sun et al., 2025; Zhang et al., 2026b; Li et al., 2025; Zeng et al., 2026). These mechanisms should be evaluated at the system level because a detector can be useful or harmful depending on the action taken when it fires.

7 Auditing the evidence used by reasoning-aware detectors

Some papers in the corpus are best understood as audits rather than as new deployable detectors. They ask whether reasoning changes the detectability of errors, whether reflection is reliable evidence, whether a trace score actually reads the reasoning body, or whether a visible CoT is faithful to the model's decision process. These studies constrain how the detector papers should be interpreted.

7.1 Reasoning changes both the error distribution and detector behavior

Cheng et al. compare hallucination detection under matched direct-answer and CoT prompting conditions across probability-based, consistency-based, self-evaluation, and internal-state detector families (Cheng et al., 2025). They

measure changes in token probabilities, hidden states, detector score distributions, discrimination, and calibration. Across the large grid of model–dataset–detector configurations, lower AUROC under CoT is common but not universal. The paper's result is therefore not that reasoning always harms detection; it is that changing the reasoning mode can change both the errors that remain and the statistical cues on which existing detectors rely.

Chegini et al. examine a related issue at operating points relevant to safety and hallucination filtering (Chegini et al., 2025). They compare think-on and think-off inference and show that improved average task accuracy can coexist with worse recall when the detector is constrained to very low false-positive rates. The effect varies across benchmarks, and the paper also compares token-based confidence, self-verbalized confidence, and combinations of reasoning modes.

Together, these audits show why a detector should not be evaluated only by average AUROC on one reasoning condition. A deployment that blocks an answer, triggers expensive verification, or simply ranks candidates operates at different thresholds. Matched reasoning conditions, base hallucination rates, calibration, and performance in the intended operating region are part of the evaluation rather than secondary reporting choices.

7.2 Reflection and meta-cognitive confidence: Lu et al.

Lu et al. construct a controlled RFC-based knowledge environment to study how hallucinations emerge and propagate through long CoT (Lu et al., 2025). The analysis distinguishes claims that are seen but not reliably learned from claims that are absent or incorrect, and explicitly defines meta-cognitive confidence as the model's belief that it knows a claim, not as the factual truth of that claim. The paper tracks how internal or externally introduced claims enter the reasoning chain, how reflection revisits them, and how confidence changes during that process.

The audit finds that reflection can increase confidence in unsupported claims and can create additional internal errors while remaining aligned with a misleading prompt. Controlled edits at different points in a hallucinated chain often change later reasoning, but acceptance of an edit does not guarantee correction of the final answer. The authors describe this resistance to correction as chain disloyalty. For detector design, the implication is specific: repeated reflection, lower apparent uncertainty, or a successful local correction cannot be assumed to provide independent evidence that the remaining trajectory is factually sound.

7.3 Does a trace score actually use the trace? Zollicoffer et al.

Zollicoffer et al. propose sanity checks for long-form hallucination detectors that are intended to rely on reasoning trajectories (Zollicoffer et al., 2026). FORCE replaces the final answer with an oracle answer while preserving the reasoning body, whereas REMOVE deletes answer-announcement material while preserving the substantive trace. A detector whose claim is about the reasoning trajectory should be appropriately invariant to changes that preserve that trajectory and sensitive to changes that alter relevant evidence. The paper also proposes TRACT, a text-only score based on trajectory features such as changes in hedging, step-length behavior, and vocabulary convergence across samples.

These tests do not prove that a detector is faithful to the model's internal computation. Their value is narrower and useful: they can reveal when a nominal trace detector is dominated by endpoint or lexical artifacts. The same logic can be adapted to other detector families by stating in advance which changes should preserve a score and which should alter it, but such extensions are proposals for future validation rather than experiments already performed in this paper.

7.4 Faithfulness is a separate target: FAITHCOT-BENCH

FAITHCOT-BENCH evaluates instance-level faithfulness of chain-of-thought explanations rather than factual hallucination alone (Shen et al., 2026). Its FINE-COT resource contains expert-annotated trajectories spanning four models and four domains, constructed through a multi-round annotation protocol. The benchmark distinguishes broad post-hoc and spurious-reasoning phenomena through eight observable principles and evaluates eleven representative faithfulness detectors, including counterfactual, logit-based, and judge-based approaches.

FAITHCOT-BENCH provides direct empirical support for separating correctness from faithfulness: across its 1,183 annotated traces, the association is weak ($\phi = 0.286$), with both correct-but-unfaithful and wrong-but-faithful cases. This matters for hallucination detection because a wrong answer can still expose the model's mistaken reasoning faithfully, while a correct answer can be paired with a post-hoc or otherwise unfaithful rationale. The paper also acknowledges the observability limit of the task: the true latent decision path is not directly available, so FINE-COT labels observable evidence of faithfulness rather than the complete hidden computation. This boundary should carry

over whenever a hallucination detector interprets the visible CoT as an explanation of why the model answered as it did.

8 Cross-cutting synthesis

The method sections show substantial diversity in signals and training regimes. The most useful common conclusions are therefore not claims that one evidence family dominates another, but distinctions about what each family observes, how its labels are created, and what additional claim can be supported by the evaluation.

8.1 Four distinct evidential questions

Four questions recur across the corpus. A *predictive* question asks whether a score separates the evaluation labels. A *localization* question asks where in the output or process the relevant failure is observed. An *interventional* question asks whether changing the identified state, step, route, or representation changes downstream behavior. A *grounding* question asks whether the generated claim is supported by information external to the generator's own belief state. These questions are related but not ordered on a single scale. A grounded verifier can establish factual support without explaining internal causality; an intervention can reveal dependence without establishing that the resulting proposition is true.

This distinction clarifies several recurring interpretation problems. HARP, D2HScore, G-Detector, and RFS-Guard associate named geometric or structural quantities with hallucination labels, but the classification result alone does not establish that those quantities are the causal source of the error. Evidence Drop and GeoReason add local resolution, yet a first detected transition is not necessarily the only state whose correction matters downstream. ARS performs an actual latent perturbation, but the intervention tests stability to that perturbation rather than factual correctness. FSPO introduces external evidence into training, which strengthens grounding relative to same-model self-consistency, but the factuality signal remains only as reliable as the evidence and verifier.

8.2 Stable errors are a blind spot for several model-derived signals

Several detectors derive reliability evidence from different views of the same model: repeated outputs in Semantic Entropy, trace–answer consistency in RACE, verification or self-judgment in IRIS and LaaB, latent stability in ARS, and reflection in the meta-cognitive audit. These signals are not identical, but they share a possible failure regime: if the model maintains a stable misconception, several model-derived views can remain consistent with the same wrong proposition.

This does not make model-derived evidence uninformative. It means that consistency, confidence, and stability should be interpreted as properties of the model's behavior or representation unless the evaluation introduces an independent factual constraint. The failure regime is especially relevant when a method claims to detect lack of knowledge: a low-entropy or internally coherent error can look more reliable than a correct but uncertain answer. Reference-grounded evaluation, deterministic tools, or independently validated verifiers are therefore complementary to self-consistency rather than interchangeable with it.

8.3 Granularity changes both utility and label dependence

The move from response-level rejection to finer outputs is motivated by actionability. Whole-trace methods such as RACE support accept/reject decisions; streaming and transition-level methods can identify risk before generation ends or near its onset; typed and token/span detectors can direct more targeted verification. The gain is therefore not simply finer resolution but a closer match between the detector output and the action that follows.

The same progression makes the labeling rule more consequential. GeoReason depends on a definition of the first erroneous transition. FG-PRM depends on a six-class taxonomy and a synthetic error generator. TokenHD depends on critic spans, restoration, and aggregation; RL4HS depends on span labels and a reward function. Finer output is therefore useful only when the finer label corresponds to a stable scientific target and to an action the system can take. A detector can become more precise about an annotation artifact without becoming more useful for the underlying reliability problem.

Evidence family	What is observed	Typical requirement	Natural output	Unresolved distinction
Sample uncertainty	Variation across generated answers or meanings	Multiple generations; semantic grouping	Response risk / rejection	Stable false belief versus true confidence
Static latent signal	Decodability from one hidden state, attention map, or logit statistic	Internal or proxy-model access; sometimes probe training	Fast response score	Predictive separator versus causal or model-invariant feature
Latent geometry / shaping	Covariance, subspace, layer dynamics, or representation after intervention/elicitation	Layer choice; labels or steering in some methods	Response score / representation diagnostic	Geometry stable across model/domain/context changes
Explicit reasoning process	Trace agreement, prefix state, transition, graph, or routing structure	Long trace; segmentation; sometimes white-box access	Trace, step, or prefix risk	Observable error versus causal source; coherent false premise
Hybrid / fine-grained	Several views or labels at step/token/span resolution	Multiple modules, critics, or fine-grained supervision	Localized or typed decision	Whether views add independent information and labels match natural failures
Reference-grounded control	Support of intermediate claims against supplied evidence	Evidence source and verifier	Step feedback / policy update	Verifier and evidence error become part of the system

Table 2: Cross-paper trade-offs organized by the information observed rather than by a single performance ranking.

8.4 Supervision defines the detection target

Supervision in this corpus ranges from automatic continuation labels and self-derived verification targets to synthetic process corruptions, critic-aggregated token labels, expert faithfulness annotations, and verifier rewards. These signals are not interchangeable "hallucination labels." MIND uses an entity-continuation task to obtain low-cost factual supervision; IRIS uses verification-derived soft targets; FG-PRM synthesizes named process-error types; TokenHD aggregates critic judgments at token level; and FINE-COT annotates observable faithfulness phenomena. Each pipeline therefore defines a different learning target.

Consequently, supervision should be reported as part of the method rather than as an implementation detail. A detector can achieve strong held-out performance by learning structure that is specific to how its labels were produced. Human evaluation on naturally generated errors, leave-generator or leave-critic tests, style normalization, and explicit comparison across label definitions are ways to distinguish target-relevant signal from supervision artifacts. The same issue applies to model judges: replacing human labels with a strong LLM can reduce cost without making the labels independent of model-family biases.

8.5 Generalization claims must identify what changed

Generalization is not one experiment in this literature. The shifted variable may be topical domain, factual dataset, syntactic or logical form, generator family, model size, prompt, decoding strategy, reasoning format, proxy model, or the supervision source. The papers cover different subsets of these shifts: HARP and TSV include cross-dataset tests; PRISM is explicitly designed around cross-domain representation consistency; ReFL evaluates unseen logical structures; IRIS includes topic/dataset OOD experiments; LLM-Check can extract evidence from a substitute model; GeoReason separates teacher geometry from deployable-student transfer (Hu et al., 2025; Park et al., 2025; Zhang et al., 2025; Fan et al., 2026; Srey et al., 2025; Sriramanan et al., 2024; Alvarez and Baheri, 2026).

These results should be described using the shift actually tested rather than being merged into a claim of universal robustness. Process detectors add further variables because hidden-state geometry, attention routing, step boundaries, and trace length can change when the prompt or decoding policy changes. A cross-dataset result within one fixed model is therefore evidence for a different property from cross-model transfer or invariance to a new reasoning format.

9 Open problems and research directions

The literature already contains many detection scores. The more persistent research problems concern whether a score continues to correspond to the intended target outside its original evaluation setting and whether acting on that score improves the resulting reasoning system. Several recurring gaps motivate the directions below.

9.1 Build matched multi-granular benchmarks on the same traces

Current datasets often attach one label to one resolution: final-answer correctness, first-error step, error type, unsupported span, or faithfulness. This makes it difficult to determine whether detectors at different granularities identify the same failure. A stronger benchmark would attach several labels to the same generated trace: final-answer correctness, whole-trace validity, first observable error, downstream propagated errors, unsupported text, error type, and evidence support. A faithfulness label could be added where an appropriate intervention or expert protocol is feasible rather than inferred from correctness alone.

Such a benchmark would enable direct conditional questions. How often does a correct answer contain an invalid intermediate step? How often does the first labeled error causally affect the final answer? Do TokenHD-style spans overlap the steps rejected by a process verifier? Does a global detector fail specifically when the localizer succeeds? These measurements would turn granularity from a collection of separate leaderboards into a testable relation among targets.

9.2 Match intervention tests to the interpretation being claimed

Many latent and process methods use terminology that suggests an internal explanation—reasoning subspace, transport geometry, routing focus, graph topology, evidence manifold. The appropriate validation is not the same for every term. If a latent direction is claimed to isolate a functional component, interventions along that direction should be compared with matched off-direction perturbations. If a first-error detector claims to find the transition responsible for the failure, correcting the selected transition should be compared with edits at nearby nonselected steps. If a routing score is interpreted as information flow, perturbing or masking the selected route should have a distinct downstream effect.

The meta-cognitive audit adds an important refinement: the first observable error need not be the complete causal support for the final outcome because later reasoning can reconstruct a flawed conclusion. Future datasets should therefore separate *first observable failure* from an *intervention-defined support set*—the set of states or steps whose correction materially changes the downstream answer. This distinction would connect localization and causality without treating them as the same task.

9.3 Design shift-specific generalization protocols

Generalization tests should vary one factor at a time where possible and report which detector component remains fixed. For latent probes this includes dataset/domain, model family and size, prompt, and layer choice. For LLM-Check's substitute-model setting, a matrix of generator–proxy pairs would directly measure how much the score depends on the auxiliary model. For PRISM, ReFL, IRIS, HARP, and TSV, transfer results should be reported against the exact domain, structural, or dataset shift actually evaluated by each paper rather than under one generic label.

Teacher–student systems require an additional diagnostic. GeoReason shows that a teacher can preserve a useful contrastive geometry while a distilled student loses part of that separation under shift. Distillation studies should therefore report not only average feature/probability error but also preservation of the decision margin around the first-error boundary and agreement on localized transitions. Similar margin-based analyses could be applied to any detector that compresses a richer diagnostic model into a cheaper deployable one.

9.4 Audit automatically generated and self-derived supervision

Automatically generated or self-derived supervision is central to MIND, IRIS, TSV, FG-PRM, TokenHD, and several judge-based pipelines. The appropriate audit depends on how those labels are produced: paraphrase or style normalization can test surface shortcuts; leave-generator-out or leave-critic-out splits can test dependence on a particular annotator model; FG-PRM-style injected errors should be compared with naturally generated failures; and a held-out human-labeled set should remain outside the automatic label-generation pipeline. For token/span labels, critic disagreement should be reported rather than hidden by aggregation alone.

Self-derived targets need a complementary factual check. If a model's own confidence, self-judgment, or verification response creates the training target, a stable misconception can be copied into both features and labels. Stress sets should therefore contain errors that remain stable under resampling, self-evaluation, and reflection. Performance on these cases can measure how much of the detector's success comes from an evidence source that is genuinely different from the generator's original belief state.

9.5 Evaluate detector-guided policies end to end

A local or streaming detector is valuable because it can change what happens next: retrieve evidence, call a tool, regenerate, backtrack, branch, abstain, or modify training data. Once a detector controls such an action, classification accuracy alone is insufficient. Evaluation should include final factuality and reasoning accuracy, false interventions on correct traces, additional tokens or tool calls, latency, and sensitivity to detector calibration.
FSPO, GRPO-R, G-Detector filtering, FG-PRM verification, and HalluGuard-guided inference illustrate different points in this loop. The central experimental question is not merely whether their detector scores correlate with labels, but whether the full system improves under realistic detector mistakes. A moderately accurate localizer may still be useful when the action it triggers is cheap and reversible; a higher-AUROC detector may be unsuitable if false positives suppress correct reasoning or trigger expensive verification. This is the level at which detection and mitigation can be compared fairly.

10 Conclusion

Reasoning-aware hallucination detection is a family of related prediction and evaluation problems, not one binary task. Across the thirty-one papers, the target ranges from semantic uncertainty over a final answer to global trace validity, first-error localization, typed process errors, unsupported spans, and faithfulness of an observed rationale. The evidence ranges from repeated samples and output probabilities to hidden-state geometry, attention routing, explicit reasoning structure, self-judgment, controlled perturbations, and external verification.
Across the corpus, three scientific constraints recur. First, the meaning of a detector is partly determined by its supervision: automatic, synthetic, self-derived, critic-generated, and expert labels define different targets. Second, more detailed reasoning evidence does not automatically provide a stronger factual guarantee; several internally generated views can share one stable error, and a localized correlate is not automatically a causal source. Third, process detectors are sensitive to the conditions under which the process is produced, so claims about generalization must identify the domain, model, prompt, decoding, or supervision shift that was actually tested.
Progress therefore depends as much on evaluation design as on new scores. Benchmarks should align answer-, process-, and localization labels on the same traces and include stable-misconception cases that remain confident under resampling or reflection. Generalization tests should state exactly which variable changed, and mechanistic interpretations should be paired with interventions that test the claimed dependence. When a detector drives verification, regeneration, filtering, or training, evaluation should measure the end-to-end effect of that intervention rather than detector accuracy alone. These protocols would allow response-level filters, white-box process detectors, fine-grained localizers, and faithfulness audits to be judged on the questions they actually answer rather than on an undifferentiated notion of hallucination detection.

References

Alvarez, Tyler and Ali Baheri (2026). "Where Does Reasoning Break? Step-Level Hallucination Detection via Hidden-State Transport Geometry". In: *arXiv preprint arXiv:2605.13772*.

Chegini, Atoosa, Hamid Kazemi, Garrett Souza, Maria Safi, Yang Song, Samy Bengio, Sinead Williamson, and Mehrdad Farajtabar (2025). "Reasoning's Razor: Reasoning Improves Accuracy but Can Hurt Recall at Critical Operating Points in Safety and Hallucination Detection". In: *arXiv preprint arXiv:2510.21049*.

Chen, Chao, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye (2024). "INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection". In: *International Conference on Learning Representations*.

Chen, Qunjie, Yufei Chen, Xiaodong Yue, and Linye Li (2026). "Mind the Gap: Catching Hallucinations via Evidence Drop on the Reasoning Manifold". In: *International Conference on Machine Learning*.

Cheng, Jiahao, Tiancheng Su, Jia Yuan, Guoxiu He, Jiawei Liu, Xinqi Tao, Jingwen Xie, and Huaxia Li (2025). "Chain-of-Thought Prompting Obscures Hallucination Cues in Large Language Models: An Empirical Evaluation". In: *Findings of the Association for Computational Linguistics: EMNLP*, pp. 1272–1305.

Ding, Yue, Xiaofang Zhu, Tianze Xia, Junfei Wu, Xinlong Chen, Qiang Liu, and Liang Wang (2025). "D2HScore: Reasoning-Aware Hallucination Detection via Semantic Breadth and Depth Analysis in LLMs". In: *arXiv preprint arXiv:2509.11569*.

Fan, Cunhang, Jun Zhang, Xue Zhang, Shuai Zhang, Zhao Lv, Jianhua Tao, and Zhengqi Wen (2026). "ReFL: Reflective Feedback Learning for Hallucination Detection of Large Language Models". In: *Annual Meeting of the Association for Computational Linguistics*, pp. 19648–19665.

Farquhar, Sebastian, Jannik Kossen, Lorenz Kuhn, and Yarin Gal (2024). "Detecting Hallucinations in Large Language Models Using Semantic Entropy". In: *Nature* 630.8017, pp. 625–630.

Hu, Junjie, Gang Tu, ShengYu Cheng, Jinxin Li, Jinting Wang, Rui Chen, Zhilong Zhou, and Dongbo Shan (2025). "HARP: Hallucination Detection via Reasoning Subspace Projection". In: *arXiv preprint arXiv:2509.11536*.

Li, Junyi and Hwee Tou Ng (2025). "Reasoning Models Hallucinate More: Factuality-Aware Reinforcement Learning for Large Reasoning Models". In: *Advances in Neural Information Processing Systems*.

Li, Ruosen, Ziming Luo, and Xinya Du (2025). "FG-PRM: Fine-Grained Hallucination Detection and Mitigation in Language Model Mathematical Reasoning". In: *arXiv preprint arXiv:2410.06304*.

Liu, Zihang, Zhouhua Fang, Hui Liu, Zhiwei Liu, Yong Li, and Haishuai Wang (2026). "RFS-Guard: Detecting Reasoning Hallucinations via Cross-Phase Routing Focus in Large Reasoning Models". In: *Annual Meeting of the Association for Computational Linguistics*.

Lu, Haolang, Yilian Liu, Jingxin Xu, Guoshun Nan, Yuanlong Yu, Zhican Chen, and Kun Wang (2025). "Auditing Meta-Cognitive Hallucinations in Reasoning Large Language Models". In: *Advances in Neural Information Processing Systems*.

Lu, Haolang, Minghui Pan, Ripeng Li, Guoshun Nan, Jialin Zhuang, Zijie Zhao, Zhongxiang Sun, Kun Wang, and Yang Liu (2026). "Streaming Hallucination Detection in Long Chain-of-Thought Reasoning". In: *arXiv preprint arXiv:2601.02170*.

Mi, Hao, Qiang Sheng, Shaofei Wang, Beizhe Hu, Yifan Sun, Zhengjia Wang, Hengqi Zeng, Yang Li, Danding Wang, and Juan Cao (2026). "Logical Consistency as a Bridge: Improving LLM Hallucination Detection via Label Constraint Modeling between Responses and Self-Judgments". In: *Annual Meeting of the Association for Computational Linguistics*.

Min, Rui, Tianyu Pang, Chao Du, Minhao Cheng, and Yi R. Fung (2026). "Scalable Token-Level Hallucination Detection in Large Language Models". In: *arXiv preprint arXiv:2605.12384*.

Park, Seongheon, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li (2025). "Steer LLM Latents for Hallucination Detection". In: *International Conference on Machine Learning*.

Shen, Xu, Song Wang, Zhen Tan, Laura Yao, Xinyu Zhao, Kaidi Xu, Xin Wang, and Tianlong Chen (2026). "FAITHCOT-BENCH: Benchmarking Instance-Level Faithfulness of Chain-of-Thought Reasoning". In: *International Conference on Learning Representations*.

Song, Yusheng, Lirong Qiu, Xi Zhang, and Zhihao Tang (2025). "Hallucination Detection via Internal States and Structured Reasoning Consistency in Large Language Models". In: *arXiv preprint arXiv:2510.11529*.

Srey, Ponhvoan, Xiaobao Wu, and Anh Tuan Luu (2025). "Unsupervised Hallucination Detection by Inspecting Reasoning Processes". In: *arXiv preprint arXiv:2509.10004*.

Sriramanan, Gaurang, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi (2024). "LLM-Check: Investigating Detection of Hallucinations in Large Language Models". In: *Advances in Neural Information Processing Systems*.

Su, Hsuan, Ting-Yao Hu, Hema Swetha Koppula, Kundan Krishna, Hadi Pouransari, Cheng-Yu Hsieh, Cem Koc, Joseph Yitan Cheng, Oncel Tuzel, and Raviteja Vemulapalli (2025). "Learning to Reason for Hallucination Span Detection". In: *arXiv preprint arXiv:2510.02173*.

Su, Weihang, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu (2024). "Unsupervised Real-Time Hallucination Detection Based on the Internal States of Large Language Models". In: *Findings of the Association for Computational Linguistics: ACL*, pp. 14379–14391.

Sun, Zhongxiang, Qipeng Wang, Haoyu Wang, Xiao Zhang, and Jun Xu (2025). "Detection and Mitigation of Hallucination in Large Reasoning Models: A Mechanistic Perspective". In: *arXiv preprint arXiv:2505.12886*.

Wang, Changyue, Weihang Su, Qingyao Ai, and Yiqun Liu (2025). "Joint Evaluation of Answer and Reasoning Consistency for Hallucination Detection in Large Reasoning Models". In: *arXiv preprint arXiv:2506.04832*.

Zeng, Xinyue, Junhong Lin, Yujun Yan, Feng Guo, Liang Shi, Jun Wu, and Dawei Zhou (2026). "HalluGuard: Demystifying Data-Driven and Reasoning-Driven Hallucinations in LLMs". In: *International Conference on Learning Representations*.

Zhang, Bo, Cong Gao, Linkang Yang, Bingxu Han, Minghao Hu, Zhunchen Luo, Guotong Geng, Xiaoying Bai, Jun Zhang, Wen Yao, and Zhong Wang (2026a). "Global-Local Confidence Fusion for Hallucination Detection in Mathematical Reasoning Task". In: *AAAI Conference on Artificial Intelligence*.

Zhang, Fujie, Peiqi Yu, Biao Yi, Baolei Zhang, Tong Li, and Zheli Liu (2025). "Prompt-Guided Internal States for Hallucination Detection of Large Language Models". In: *Annual Meeting of the Association for Computational Linguistics*, pp. 21806–21818.

Zhang, Guibin, Yuxiang Zhang, Moayad Aloqaily, Haolang Lu, Kun Wang, Jing Liang, Xingjun Ma, Yu-Gang Jiang, and Qingsong Wen (2026b). "Unraveling Hallucination in Large Reasoning Models: A Topological Perspective". Submitted to ICLR 2026.

Zhang, Jianxiong, Bing Guo, Yuming Jiang, Haobo Wang, Bo An, and Sean Du (2026c). "Harnessing Reasoning Trajectories for Hallucination Detection via Answer-Agreement Representation Shaping". In: *International Conference on Machine Learning*.

Zollicoffer, Geigh, Minh Vu, Hongli Zhan, Raymond Li, and Manish Bhattarai (2026). "Sanity Checks for Long-Form Hallucination Detection". In: *arXiv preprint arXiv:2605.08346*.