Understanding Why Language Models Hallucinate: Testing Reasoning Against Priors

Yangfan Hu* **Xuhan Tong*** **Haoyue Bai*†**
Xi Ding **Shashank Muralidhar Bharadwaj** **Siyang Cao**
Robert Nowak **Jiawei Zhang†**

Department of Computer Sciences, University of Wisconsin–Madison
1205 University Avenue, Madison, WI 53706, USA
{haoyue.bai,jiawei.zhang}@cs.wisc.edu

Abstract

Large language models often produce hallucinated answers that violate prompt-level constraints. A key diagnostic question is whether these failures reflect missing knowledge, or whether the model has the relevant information but follows the wrong inference path. We study this phenomenon as *inference misalignment*: a mismatch between the answer supported by the prompt and the answer favored by statistically salient latent associations. We formalize this view with a latent key–task model, in which pretraining-frequency imbalance can cause a shortcut path to dominate the constraint-sensitive path and induce positive inference loss. The framework predicts two failure modes: task-retrieval bias in entity disambiguation and key-selection bias in action choice. We introduce TRAPQA, a controlled diagnostic testbed with two components. SCIENTISTQA tests disambiguation among similar scientists with supplementary factual probes, while REAL-LIFE CONSTRAINED QA tests everyday constraint following under salient shortcuts. Our results provide behavioral evidence that some hallucinations reflect inference misalignment rather than absent knowledge alone. [1]

1 Introduction

Large language models (LLMs) have achieved strong performance across many tasks (OpenAI et al., 2024; Team et al., 2025; DeepSeek-AI et al., 2025; Grattafiori et al., 2024). They are also increasingly integrated with tools and agentic workflows, such as web search and external services (Nakano et al., 2022; Liu et al., 2023; Steinberger, 2026). As model outputs become more tightly coupled to real-world actions, hallucination remains a central reliability risk.

Hallucination broadly refers to fluent but factually incorrect, unsupported, or context-unfaithful outputs (Ji et al., 2023; Huang et al., 2025). Such errors are difficult to detect when models sound confident or when users lack domain expertise, and they can be amplified in agentic settings through downstream tool calls or transactions. Importantly, hallucinations can arise even under benign inputs, making them an intrinsic reliability problem rather than only a failure under adversarial attack (Zhang et al., 2025b; Huang et al., 2025).

Prior work studies hallucination through training-data bias, decoding dynamics, and attribution or mechanistic analysis (Dziri et al., 2022; Zhang et al., 2023; Sun et al., 2025; Gao et al., 2025). Existing evaluations such as TruthfulQA and HaluEval measure important aspects of truthfulness and hallucination behavior (Lin et al., 2022; Li et al., 2023). However, a central mechanistic question remains underexplored: when a model fails, did it lack the needed knowledge, or did it possess the relevant facts but retrieve and apply the wrong inference path?

We address this question by interpreting hallucination as *inference misalignment*: a mismatch between the answer logically supported by the prompt and the answer favored by statistically salient learned associations. In our framework, a prompt activates latent key–task paths. A model hallucinates when a high-frequency shortcut path receives greater posterior weight than the constraint-sensitive path required by the prompt. This view predicts that errors can occur even when the relevant facts or constraints are available: the failure lies not only in stored knowledge, but in selecting and composing the appropriate inference path. Figure 1 illustrates this competition and the paired-probe diagnostic.

Guided by this theory, we introduce TRAPQA, a closed-book diagnostic benchmark suite with two complementary settings. SCIENTISTQA tar-

*Equal contribution. Author order was randomly determined.

†Corresponding authors.

[1]Project is available at this link.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 44555–44581
October 24-29, 2026 ©2026 Association for Computational Linguistics

gets *task-retrieval bias*, where a salient entity–relation association overrides a discriminative constraint. REAL-LIFE CONSTRAINED QA targets *key-selection bias*, where a superficially salient cue dominates the task-relevant constraint. These settings are designed not as broad-coverage benchmarks, but as controlled tests of the latent key–task framework. External tools, including web search, are disabled in all evaluations.

SCIENTISTQA. SCIENTISTQA tests constraint-sensitive disambiguation among highly similar scientists. For each pair, we generate a biography-style paragraph broadly compatible with both candidates, then append a decisive fact that rules out exactly one candidate. The model must choose the scientist matching the full description. We also include closed-book probes for candidate-specific facts, allowing us to distinguish missing knowledge from knowledge-deployment failure. Across 2,925 questions and eight model–reasoning configurations spanning GPT, Gemini, Claude, and DeepSeek, hallucination rates range from 2.50% to 37.23% in the retrieval-sensitive names-only setting. Many errors persist even when the model answers the relevant probe facts correctly in isolation, indicating a failure of comparative knowledge deployment rather than factual ignorance alone.

REAL-LIFE CONSTRAINED QA. REAL-LIFE CONSTRAINED QA complements SCIENTISTQA by testing shortcut failures in everyday action choice. We begin from lexical associations in SWOW (De Deyne et al., 2019), use them as high-salience shortcut cues, and organize them through eight seed template families such as `vehicle_required`, `delivery_medium`, `recording_medium`, and `tool_required`. GPT instantiates natural forced-choice scenarios in which one option is superficially tempting but violates a prompt-grounded physical, spatial, procedural, or medium-specific constraint. After filtering and controlled perturbations, the final collection contains 500 questions covering 13 aspects of daily life. Claude, GPT, Gemini, and DeepSeek make 81, 44, 18, and 182 mistakes, respectively, corresponding to error rates of 16.2%, 8.8%, 3.60%, and 36.4%. These failures show that inference misalignment is not limited to encyclopedic entity disambiguation: it also appears when models must select an action under ordinary real-world constraints.

Contributions. We make three contributions. **(1)** We propose a latent key–task framework that formalizes hallucination as inference misalignment caused by posterior dominance of statistically salient shortcut paths over prompt-supported constraint-sensitive paths. **(2)** We derive theoretical predictions linking pretraining-frequency imbalance to shortcut posterior and positive inference loss, providing a mechanistic account of why hallucinations can occur even under benign prompts. **(3)** We introduce TRAPQA, a closed-book diagnostic benchmark suite with two complementary settings, and evaluate frontier model families with external tools disabled, showing that hallucinations can persist despite isolated factual knowledge and under prompt-grounded real-world constraints.

2 Related Work

Hallucination in language generation predates the recent LLM era. Neural data-to-text systems can produce fluent outputs that fail to reflect the underlying records (Wiseman et al., 2017); neural machine translation models may generate plausible but source-unsupported translations (Lee et al., 2019; Raunak et al., 2021); and abstractive summarization models often produce content that is not faithful to the input document (Maynez et al., 2020). Across these settings, hallucination reflects a common failure mode: the model produces plausible language that is insufficiently grounded in the input, retrieved evidence, or relevant knowledge.

Recent work studies hallucination from both theoretical and empirical perspectives. Formal accounts show that hallucination can arise from fundamental learning or statistical limitations (Xu et al., 2025; Kalai and Vempala, 2024), while mechanistic accounts trace hallucinations to competing associations or latent-state dynamics during generation (Sun et al., 2025; Cherukuri and Varshney, 2026). Other work locates hallucination across the LLM pipeline, including distributional imbalance and noise in pretraining data, difficulty acquiring new factual knowledge during fine-tuning, and inference-time reliance on memorized or frequency-biased patterns (Zhang et al., 2025a; Liu et al., 2026; Gekhman et al., 2024; Zhang et al., 2023; McKenna et al., 2023; Berglund et al., 2024). These suggest that hallucination is not merely a matter of missing knowledge; it can also reflect failures in retrieving, comparing, or applying knowledge under the constraints of a particular prompt.

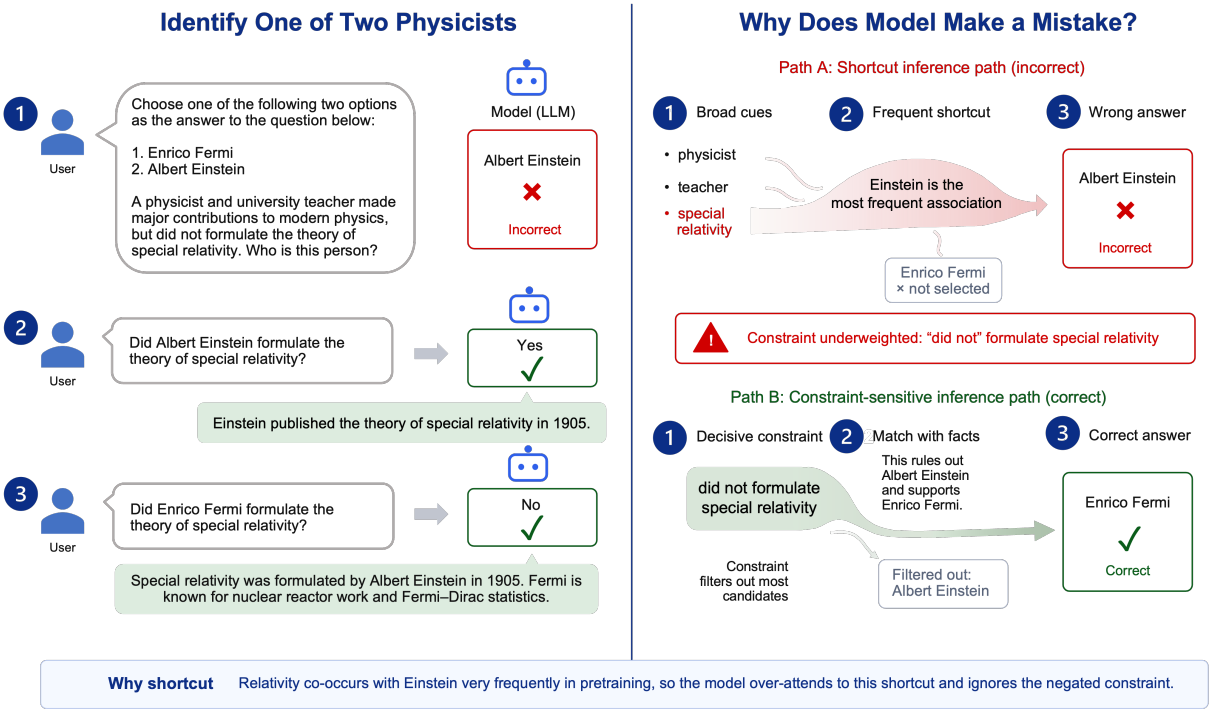

Figure 1: Hallucination as misaligned inference. In the motivating SCIENTISTQA example, the model selects the wrong scientist by following a high-salience association while underweighting the decisive discriminative constraint. The probes show that the relevant facts are behaviorally accessible in isolation; the right panel is a conceptual interpretation of the resulting answer–probe inconsistency, not an observed internal reasoning trace.

Appendix C provides a fuller discussion, including reinforcement-learning-based mitigation efforts.

A large body of benchmarks evaluates factuality and faithfulness in generated text, including TruthfulQA and HaluEval (Lin et al., 2022; Li et al., 2023), long-form factuality benchmarks such as FActScore and LongFact (Min et al., 2023; Wei et al., 2024b), short-form factuality benchmarks such as SimpleQA (Wei et al., 2024a), and retrieval-augmented generation benchmarks such as RAGTruth and FRAMES (Niu et al., 2024; Krishna et al., 2025). These benchmarks primarily measure whether an output is factual, faithful, or grounded in evidence. By contrast, our goal is diagnostic: Scientist QA and Real-Life Constrained QA are designed to isolate why a model fails. Scientist QA tests whether models can deploy candidate-specific facts under disambiguation, while Real-Life Constrained QA tests whether models can override SWOW-derived associative cues with prompt-grounded physical, spatial, procedural, or medium-specific constraints (De Deyne et al., 2019). This design allows us to distinguish simple ignorance from knowledge-deployment failures in which relevant information is available to the model but not used in the relevant inference path.

3 Pretraining Frequency Induces Hallucination in Latent Inference

3.1 Hallucination as Inference Misalignment

In the study of LLM reliability, *hallucination* is commonly described as the generation of factually incorrect or logically inconsistent content. However, to enable a quantitative mathematical analysis, we argue that the underlying mechanism of hallucination can be more precisely characterized as *inference misalignment*.

Specifically, let z denote a prompt sequence of bounded length. A pretrained sequence model induces a conditional distribution $P(\cdot \mid z)$ over continuations. We further assume the existence of an ideal predictor defining the ground-truth conditional distribution $P_\star(\cdot \mid z)$, representing the correct reasoning process for the task associated with the prompt. For analysis, we work in the embedding space where smoothness properties are well defined. Conceptually, the model's inference can be viewed as moving from regions well-supported by the pretraining distribution toward the test prompt along an optimal *inference path*, producing an output consistent with $P_\star(\cdot \mid z)$ when the semantic structure and task intent are correctly

identified.

However, due to statistical regularities present in the pretraining corpus, the model may instead rely on high-frequency *shortcuts*. These shortcuts bias the model toward inference paths that are statistically dominant but semantically incorrect. Consequently, the model may traverse regions of the representation space that are weakly supported by the training data, leading to unstable behavior.

Guided by this perspective, we define the inference loss as the discrepancy between the model's predictive distribution and the ideal distribution: $\ell(\boldsymbol{z}) := \|P(\cdot|\boldsymbol{z}) - P_\star(\cdot|\boldsymbol{z})\|_{TV}$, where $\|\cdot\|_{TV}$ denotes the total variation distance. Through this formalization, we shift the study of hallucination from heuristic descriptions of textual inconsistency to a principled analysis of inference behavior.

3.2 Latent Key–Task Model of LLM Inference

To analyze how a model generalizes from the pretraining corpus to a new prompt, we introduce a conceptual abstraction of inference stage. Let $\mathcal{T} = \{t_1, \ldots, t_p\}$ denote a finite set of latent tasks and let $\mathcal{K} = \{k_1, \ldots, k_m\}$ denote a finite set of task-informative *keys*. A key represents a salient feature pattern in a prompt (for example, a lexical cue or structural pattern) that provides evidence about the underlying task. We model LLM inference as an implicit two-stage reasoning process

$$\boldsymbol{z} \xrightarrow{\text{identify key}} k_i \xrightarrow{\text{retrieve task}} t_j(k_i) \xrightarrow{\text{generate}} y.$$

Following the Bayesian perspective of (Xie et al., 2022), we model the model's prompt processing as an implicit inference over latent variables. Under this view, the model associates each prompt with a posterior distribution over key–task pairs and combines their contributions to form the prediction. Given a prompt $\boldsymbol{z}$, the model implicitly forms a posterior distribution $P(k, t|\boldsymbol{z})$ over the latent space and aggregates predictions along these hypotheses:

$$P(y|\boldsymbol{z}) = \sum_{k\in\mathcal{K},\, t\in\mathcal{T}} P(k, t|\boldsymbol{z})\, P(y|\boldsymbol{z}; k, t),$$

where $P(y|\boldsymbol{z}; k, t)$ denotes the predictive distribution under the hypothesis that the prompt corresponds to key k and task t.

3.3 Frequency-Induced Bias in Inference

We now introduce a framework that explicates how the statistical imbalance of the pretraining corpus leads to inference shortcuts and formally induces hallucination through generalization error.

Pretraining Statistics. The distribution of keys and tasks in the pretraining corpus induces a statistical prior that guides latent inference. Let $c_i^{(k)}$ denote the number of occurrences of key k_i in the corpus, and let $C^{(k)} = \sum_{i=1}^{m} c_i^{(k)}$. Conditioned on a key k, let $c_j^{(t)}(k)$ denote the number of occurrences in which task t_j co-occurs with k, and let $C^{(t)}(k) = \sum_{j=1}^{p} c_j^{(t)}(k)$. The empirical key distribution and conditional task distribution are

$$\pi^{(k)}(k_i) = \frac{c_i^{(k)}}{C^{(k)}}, \quad \pi^{(t)}(t_j|k) = \frac{c_j^{(t)}(k)}{C^{(t)}(k)}.$$

Throughout the analysis, we use these empirical frequencies as proxies for the underlying pretraining probabilities. These statistics define a joint prior over key–task pairs $\pi(k, t) = \pi^{(k)}(k)\, \pi^{(t)}(t|k)$.

Event-based Perspective. We now specialize the latent inference framework, where the model is asked to select between two candidate entities or actions. In this setting, the prompt $\boldsymbol{z}$ explicitly presents two candidate keys k^* and k_s: k^* is the candidate consistent with the prompt's decisive constraint and corresponds to the correct answer, while k_s is the alternative candidate. We refer to k_s as the *shortcut key* when its associated key–task pair has higher pretraining frequency than that of k^*, so that statistical salience favors k_s even though the prompt-level evidence favors k^*. Correspondingly, let $\{t^*, t_s\}$ denote the correct and the shortcut tasks.

Assumption 3.1 (Activated Key Restriction) *For a prompt $\boldsymbol{z}$ presenting candidates k^* and k_s:*

1. *Negligible mass outside the candidate pair. Latent keys other than k^* and k_s receive vanishing posterior: $P(k \notin \{k^*, k_s\} \,|\, \boldsymbol{z}) \ll 1$.*

2. *Prior-driven posterior on the candidate pair. Within the candidate pair, the prompt does not differentially update the relative posterior of the two keys: $\boldsymbol{z} \perp k \mid \{k \in \{k^*, k_s\}\}$.*

A useful way to interpret Assumption 3.1 is that (i) candidate keys outside the explicitly presented pair receive negligible posterior support; (ii) once the candidate set is fixed by the prompt template, the remaining prompt content does not, on its own, alter the relative plausibility of the two activated keys at the level of latent identification. The

model's choice between k^* and k_s is then driven by the pretraining prior over keys rather than by within-prompt likelihood asymmetries.

Assumption 3.2 (Activated Task Restriction) *For each path, let t^* and t_s denote the tasks associated with k^* and k_s respectively.*

1. *Negligible mass outside the candidate tasks. Conditional on the activated key being k^* (resp. k_s), the task posterior concentrates on the candidate set $\{t^*, t_s\}$: $P(t \notin \{t^*, t_s\} \mid \boldsymbol{z}, k^*) \ll 1$ and $P(t \notin \{t^*, t_s\} \mid \boldsymbol{z}, k_s) \ll 1$.*

2. *Prior-driven posterior on the candidate tasks. Within the candidate task pair, the prompt does not differentially update the relative posterior of the two tasks given the activated key: $\boldsymbol{z} \perp t \mid k, \{t \in \{t^*, t_s\}\}$.*

Assumption 3.3 (Output separation) *Let y^* denote the correct answer and let y_s denote the shortcut-induced answer. We assume that the two paths induce separated predictions: $P(y^* \mid \boldsymbol{z}; k_s, t_s) \ll 1$, and $P(y_s \mid \boldsymbol{z}; k^*, t^*) \ll 1$. Moreover, the shortcut path is at least as confident in the shortcut answer as the correct path is in the correct answer: $P(y_s \mid \boldsymbol{z}; k_s, t_s) \geq P(y^* \mid \boldsymbol{z}; k^*, t^*)$.*

Theorem 3.4 (Shortcut Probability Dominance) *Under Assumptions 3.1–3.3, consider a fixed prompt $\boldsymbol{z}$ and the two main competing paths (k^*, t^*) and (k_s, t_s), then*

$$\frac{P(y_s \mid \boldsymbol{z})}{P(y^* \mid \boldsymbol{z})} \gtrsim \frac{\pi(k_s, t_s)}{\pi(k^*, t^*)} \cdot \frac{P(y_s \mid \boldsymbol{z}; k_s, t_s)}{P(y^* \mid \boldsymbol{z}; k^*, t^*)} \gtrsim 1.$$

Theorem 3.4 shows that when the pretraining frequency of the shortcut pair is sufficiently larger than that of the correct pair, the shortcut posterior can dominate the correct posterior even if the prompt contains semantically correct evidence. We can decompose this frequency-induced hallucination into two distinct modes:

I. Key Selection Bias. The first term indicates that the model may fail to attend to the correct semantic anchor because a shortcut key appears much more frequently in the pretraining corpus.

Consider the question: *"I want to go to a car wash. The car wash is only 50 meters away. Should I walk there or drive there?"* Many models answer that one should walk, since 50 meters is a very short distance, though without driving the car to the car wash the task cannot be completed. This suggests that the model attends to the statistically dominant distance key (i.e., "50 meters") rather than the semantically decisive key (i.e., "car wash"), since many examples associate short distances with walking and long distances with driving in pretraining. As a result, the shortcut key k_s corresponding to distance-based transportation choice can dominate the correct key k^* corresponding to task feasibility (see Appendix G for the corresponding experiment).

II. Task Retrieval Bias. The theorem also shows that even with the correct key, the model may still retrieve the wrong relation if another task strongly dominates that key family in pretraining.

During pretraining, discussions of special relativity overwhelmingly associate the concept with Albert Einstein. Consider a prompt asking the model to choose between Enrico Fermi and Albert Einstein: *"A physicist and university teacher made major contributions to modern physics, but did not formulate the theory of special relativity."* Although the explicit constraint "did not formulate special relativity" rules out Einstein and supports Fermi, the model may still answer "Albert Einstein". The reason is that the affirmative shortcut key linking *special relativity* to Einstein (k_s) dominates the rarer negated constraint (k^*) in the pretraining distribution. As a result, the model attends to the statistically dominant association and effectively ignores the negation (see Section 5.1).

Within the shortcut-dominance regime analyzed here, hallucination arises only when a dominant shortcut key–task pair has been learned during pretraining. If no representative shortcut pattern exists, this posterior-suppression mechanism does not occur, and the model may instead rely on the information provided in the prompt. The complete proof is provided in Appendix K.2.

When Assumption 3.1 does not hold. Assumption 3.1 is best understood as describing the regime in which k^* and k_s are *pretraining-independent*: the two candidate keys rarely co-occur in the same context during pretraining, so the model has not internalized any joint structure relating them. In this regime, even though the prompt $\boldsymbol{z}$ presents both keys together, the model cannot leverage $\boldsymbol{z}$ to extract joint information beyond what is already encoded in the marginal priors $\pi^{(k)}(k^*)$ and $\pi^{(k)}(k_s)$, the posterior thus degenerates to the prior ratio.

The complementary regime is when k^* and k_s have been seen together during pretraining and the

model has learned how their relative importance shifts in joint contexts. In this case, the prompt $\boldsymbol{z}$ is not merely a pair of activated keys but a context whose surface form pattern-matches against pretraining co-occurrence statistics that the model has internalized. The model can then use $\boldsymbol{z}$ to identify which of the two keys is task-relevant in this particular context—essentially deploying a learned within-context disambiguation, rather than falling back on marginal frequency. Assumption 3.1 is violated, the posterior departs from the prior ratio, and hallucination may be avoided.

Beyond the two-path specialization. The general model in Section 3.2 permits arbitrary finite sets of keys and tasks. Assumptions 3.1–3.2 isolate the simplest nontrivial competition between one constraint-sensitive path and one shortcut path, and impose prior-driven posteriors for analytical tractability; they do not assert that an LLM activates only two hypotheses or that contextual evidence has no effect more generally. More generally, the activated paths may be partitioned into constraint-sensitive and shortcut sets. If prompt-dependent evidence reweights each path's pretraining prior within a bounded factor, then a sufficiently large aggregate prior-mass advantage for the shortcut set still yields shortcut posterior dominance. Conversely, stronger contextual evidence may amplify constraint-sensitive paths, suppress shortcuts, or activate additional constraint-relevant paths. Appendix K.2 provides the corresponding posterior-odds bound.

3.4 From Shortcut to Inference Loss

We now show that when this posterior dominance changes the model's preferred answer, it directly induces a positive inference loss.

Assumption 3.5 (Target preference margin) *The target distribution prefers the correct answer over the shortcut answer: there exists $\gamma_\star(\boldsymbol{z}) > 0$ such that $\gamma_\star(\boldsymbol{z}) := P_\star(y^* \mid \boldsymbol{z}) - P_\star(y_s \mid \boldsymbol{z}) > 0$.*

Theorem 3.6 (Inference-Loss Lower Bound) *Suppose Assumptions 3.3 and 3.5 hold. If the shortcut posterior dominates the correct posterior (i.e., $P(y^* \mid \boldsymbol{z}) < P(y_s \mid \boldsymbol{z})$, suggested by Theorem 3.4), the inference loss measured by total variation satisfies*

$$\ell(\boldsymbol{z}) \geq \frac{1}{2}(\gamma(\boldsymbol{z}) + \gamma_\star(\boldsymbol{z})).$$

where $\gamma(\boldsymbol{z}) := P(y_s \mid \boldsymbol{z}) - P(y^* \mid \boldsymbol{z}) > 0$.

Theorem 3.6 provides a direct connection between shortcut posterior dominance and inference loss. Thus, once frequency-induced posterior dominance reverses the model's preference between these two answers, the model distribution and the target distribution must differ by a non-vanishing margin. The complete proof is provided in Appendix K.3.

4 Evaluation Setup

We evaluate the latent-inference account in Section 3 with two controlled diagnostic settings. Scientist QA targets task-retrieval bias in entity disambiguation, while Real-Life Constrained QA targets key-selection bias in everyday action choice. External tools, including web search, are disabled in all evaluations. The names-only Scientist QA condition and supplementary probes are closed-book; the profiles-in-context condition is a retrieval-relaxed control.

Scientist QA. Scientist QA is constructed from Wikipedia-linked scientist profiles (Wikipedia contributors, 2026). We remove names from structured profiles, embed the remaining attributes with `text-embedding-3-small` (OpenAI, 2024), retain highly similar scientist pairs under a sparsity-adjusted similarity score, and use Gemini to generate a shared biographical description plus a decisive constraint that rules out exactly one candidate. After filtering and removing invalid items with GPT, the evaluation set contains 2,925 questions. Each item is tested under a *names-only* prompt, corresponding to *prepend_names*, and a *profiles-in-context* prompt, corresponding to *prepend_profiles*. We also attach two closed-book probes derived from the decisive constraint to distinguish missing factual knowledge from failures to deploy knowledge in pairwise disambiguation. Construction details are in Appendix D.

Real-Life Constrained QA. Real-Life Constrained QA tests whether models follow prompt-grounded constraints when a salient associative shortcut suggests the wrong action. We derive high-salience cues from SWOW (De Deyne et al., 2019), organize generation around eight seed template families, and use GPT to instantiate natural two-option scenarios involving physical, spatial, procedural, or medium-specific constraints. After filtering and controlled perturbations with Claude, the final collection contains 500 questions covering 13 aspects of daily life. Representative examples

Vehicle Safety Inspections

The utility-trailer clearance tag is printed. The tester is at the pad to pull the breakaway switch and check the brake battery. Should I tow the trailer to the pad or walk to the pad without it?

A. Walk to the pad without the trailer.
B. Tow the trailer to the pad.

Model: A ✗ **Ground Truth: B**

Live Theater Operations

I need to tell the projection operator that the backup cable is coiled under the console. The projection operator is running live supertitles. How should I pass the message?

A. Walk up and tell the operator.
B. Send the note over the show chat.

Model: A ✗ **Ground Truth: B**

Machinery Repair

The mower fuel-cap key blank is ready. Maintenance is at the bay to read the mower serial plate and try the blank in the actual cap. For this step, what should I do?

A. Drive the mower to the bay.
B. Walk to the bay without the mower.

Ground Truth: A **Model: B ✗**

High-Stakes Event Production

I need to tell the lighting operator that the spare cable is under the console. The operator is running cues during a live performance. What should I do?

A. Send a headset text/chat cue.
B. Tap and speak to the operator.

Ground Truth: A **Model: B ✗**

Corporate Administrative Workflows

I am documenting a toll error for payroll. Payroll scans receipts, while the toll agency requires the original toll receipt for a refund. What should I submit to payroll?

A. The original toll receipt.

B. A receipt copy.

Model: A ✗ **Ground Truth: B**

Agricultural Machinery Maintenance

The compact-tractor belt-slip case is open. The mechanic is at the ramp to watch the drive belt under load. For this step, what should I do?

A. Bring the compact tractor to the ramp.
B. Walk to the ramp without the tractor.

Ground Truth: A **Model: B ✗**

Figure 2: Representative examples from Real-Life Constrained QA. Each card shows a complete two-option scenario, the ground-truth action, and an observed incorrect model prediction. The examples illustrate key-selection failures in which a salient associative shortcut conflicts with a prompt-grounded physical, spatial, procedural, or medium-specific constraint.

are shown in Figure 2, with details in Appendix G.

Models and scoring. For Scientist QA, we evaluate GPT, Claude, Gemini, and DeepSeek under low- and high-thinking settings where available; for DeepSeek, `deepseek-chat` and `deepseek-reasoner` serve as the non-reasoning and reasoning modes. For Real-Life Constrained QA, we report GPT, Claude, Gemini, and DeepSeek-chat results; all runs use default decoding settings unless otherwise noted. Prompt templates, model versions, parsing rules, and answer normalization are reported in Appendix E. A response is correct if, after normalization, it matches the ground-truth candidate or action; choosing the ruled-out Scientist QA candidate, producing an off-option Scientist QA response, or selecting the shortcut Real-Life action is counted as an error.

Model version	Inference setting	Names-only errors / rate	Profiles-in-context errors / rate
Claude	low thinking	699 / 23.90%	5 / 0.17%
Claude	high thinking	182 / 6.22%	0 / 0.00%
DeepSeek	non-reasoning	1089 / 37.23%	99 / 3.38%
DeepSeek	reasoning	309 / 10.56%	0 / 0.00%
Gemini	low thinking	73 / 2.50%	0 / 0.00%
Gemini	high thinking	92 / 3.15%	0 / 0.00%
GPT	low thinking	344 / 11.76%	0 / 0.00%
GPT	high thinking	300 / 10.26%	0 / 0.00%

Table 1: Scientist QA results over 2,925 questions for frontier model versions evaluated with external tools disabled. The names-only prompt condition corresponds to *prepend_names*: the model receives only the two candidate names and must retrieve the decisive relation internally. The profiles-in-context prompt condition corresponds to *prepend_profiles*: the model receives both candidate profiles in the prompt. Entries report the number and percentage of errors.

5 Empirical Findings

We evaluate 2,925 Scientist QA questions across four model families and two thinking settings, together with 500 Real-Life Constrained QA questions. All runs disable external tools, including web search. For Scientist QA, the primary setting is the names-only prompt, where the model sees only the two candidate names and must retrieve the decisive relation internally. We additionally report a retrieval-relaxed profiles-in-context control, where both candidate profiles are supplied in the prompt. Each Scientist QA item is also paired with two closed-book probes targeting the decisive relation. Across the primary names-only Scientist QA runs, only two responses, both from Claude-low, fail to match either candidate after normalization; we count these off-option responses as hallucinations.

5.1 Retrieval-Sensitive Disambiguation Remains Difficult

Table 1 compares the primary retrieval-sensitive Scientist QA setting with the retrieval-relaxed profile control. In the names-only prompt condition, hallucination rates range from 2.50% to 37.23%. In the profiles-in-context condition, the maximum error rate is 3.38%, and six of the eight model settings achieve zero error. Thus, Scientist QA is not primarily testing whether models can compare two supplied profiles; it tests whether they can retrieve and apply the decisive relation when only the candidate names and ambiguous description are given.

Thinking effort has model-dependent effects in the names-only setting. Higher thinking substantially reduces errors for Claude, from 23.90% to 6.22%, and for DeepSeek, from 37.23% to 10.56%. It modestly improves GPT, from 11.76% to 10.26%. Gemini-low attains the lowest observed error rate in this setting at 2.50%, while Gemini-high is slightly worse at 3.15%. Because Gemini assisted in constructing Scientist QA, direct cross-family rankings involving Gemini should be interpreted cautiously. The within-family non-monotonicity nevertheless shows that additional inference effort is not uniformly beneficial.

5.2 Direct Probes Separate Ignorance from Knowledge-Deployment Failure

Each Scientist QA item has two closed-book probes targeting the decisive relation: one eliminative probe for the distractor and one compatibility probe for the correct candidate.

Table 2 conditions pairwise hallucination on these probe outcomes.[2]

For example, Claude-low answers both probes correctly on 76.34% of all questions, yet hallucinates on 18.99% of that subset. Viewed from the error population, 60.66% of all Claude-low pairwise hallucinations therefore occur despite both probes being answered correctly. Probe knowledge helps but cannot fully explain the errors. Hallucination is much higher when not both probes are correct, yet many errors remain in the both-probe-correct regime. Complete probe-level ignorance accounts for at most 5.50% of errors. Thus, many are not simply missing facts. The model can answer the relevant facts in isolation but still fail to deploy them in pairwise disambiguation.

5.3 Raw Fame Does Not Explain the Shortcut

Because our theory emphasizes frequency-induced shortcuts, we test whether the observed Scientist QA failures reduce to a simpler fame prior. For each scientist, we define a fame score using the normalized page-view count of their Wikipedia page, the normalized length of that page, and the normalized number of external links from that page. The wrong candidate is more famous in 61.30% of Scientist QA questions. However, hallucination does not increase in those cases. In fact, for every model setting, hallucination is lower when the wrong candidate is more famous than when it is not; for example, GPT-high drops from 13.52% to 8.20%, Claude-low from 34.19% to 17.40%, and DeepSeek-low from 41.25% to 34.69%. The same pattern holds from the perspective of hallucinated cases: among hallucinations, the wrong candidate is more famous only 44.64% to 57.12% of the time, below the dataset base rate of 61.30%. Moreover, very famous candidates often make the task easier rather than harder. When at least one candidate is in the top 1% by fame rank, hallucination rates are lower in all eight names-only model settings; for instance, GPT-high falls from 10.87% to 5.11%, and DeepSeek-low falls from 38.36% to 27.80%. Full fame-based analyses are in Appendix H.5. These results suggest that the shortcut is not a raw preference for famous names, but a relation-specific association between entities and attributes, such as institutions, awards, roles, or fields, that can override the prompt's decisive constraint.

5.4 Everyday Constraints Induce the Same Failure Pattern

Real-Life Constrained QA tests whether the same failure pattern appears outside encyclopedic entity disambiguation. Across 500 two-option scenarios, Claude, GPT, Gemini, and DeepSeek-chat make 81, 44, 18, and 182 errors, respectively, corresponding to error rates of 16.20%, 8.80%, 3.60%, and 36.40% (Table 11). To test whether these failures are concentrated in a narrow construction pattern, we further classify the final 500 questions into 12 analysis families and 10 constraint types. No single analysis family accounts for more than 16.6% of the benchmark, and no constraint type accounts for more than 19.6%. One or more models fail on 211 questions (42.2%). Failures occur in all 10 analysis families containing at least 12 items; the two remaining families contain only 2 and 1 items, respectively. Failures also span 9 of the 10 constraint types.

Measured as the fraction of questions missed by at least one model, the highest rates occur for functional or diagnostic tests (49/83, 59.0%), vehicle-identifier verification (44/78, 56.4%), and rebate, warranty, or refund documentation (38/69, 55.1%). Failures also occur in communication-channel constraints (6/20, 30.0%), safety or condition con-

[2] "Pairwise hall." is the overall names-only pairwise error rate. "Both probes correct" uses all 2,925 questions as its denominator. "Hall. | both" and "Hall. | not both" use, respectively, the subset where both probes are correct and the subset where at least one probe is incorrect. "Known-fact hall." and "Probe-absent hall." use all pairwise hallucinations as their denominator and report the fractions for which both probes are correct or both probes are incorrect, respectively.

Model	Mode	Pairwise hall.	Both probes correct	Hall. | both	Hall. | not both	Known-fact hall.	Probe-absent hall.
Claude Sonnet 4.6	low	23.90%	76.34%	18.99%	39.74%	60.66%	1.29%
Claude Sonnet 4.6	high	6.22%	86.26%	2.62%	28.86%	36.26%	4.95%
DeepSeek V3.2 Chat	low	37.23%	59.52%	34.46%	41.30%	55.10%	1.93%
DeepSeek V3.2 Reasoner	high	10.56%	79.28%	6.04%	27.89%	45.31%	5.50%
Gemini 3.1 Pro Preview	low	2.50%	97.13%	2.01%	19.05%	78.08%	0.00%
Gemini 3.1 Pro Preview	high	3.15%	97.74%	2.59%	27.27%	80.43%	0.00%
GPT-5.2	low	11.76%	85.54%	7.91%	34.52%	57.56%	3.49%
GPT-5.2	high	10.26%	87.04%	6.25%	37.20%	53.00%	2.67%

Table 2: Probe-conditioned results for the names-only prompt condition over 2,925 Scientist QA questions. "Hall. | both" and "Hall. | not both" condition pairwise hallucination on whether both probes are correct. "Known-fact hall." and "Probe-absent hall." report the fractions of hallucinations where both probes are correct or both probes are wrong, respectively.

straints (7/29, 24.1%), and attached-label cases (8/66, 12.1%). These categories are semantic construction strata rather than distinct theoretical mechanisms. Because they are unevenly sized and were not designed for causal comparison, we treat the breakdown as descriptive and do not claim that any category is intrinsically more difficult. The complete breakdowns by analysis family and constraint type are reported in Appendix H.7. These errors occur when a salient shortcut action conflicts with a physical, spatial, procedural, or medium-specific constraint stated in the prompt. Thus, inference misalignment is not limited to biographical facts; it also appears in everyday action choice.

Interpretation and scope. Taken together, these results instantiate the two complementary failure modes predicted by our framework. In ScientistQA, pairwise failures persist even when both candidate-specific probes are answered correctly, while supplying the full profiles nearly eliminates errors; this pattern is consistent with task-retrieval bias, in which behaviorally accessible facts are not deployed correctly under competition. In Real-Life Constrained QA, the decisive constraint is stated in the prompt, yet models sometimes follow a salient associative cue, consistent with key-selection bias. These observations are behavioral rather than mechanistic: because the evaluated systems and their pretraining corpora are black boxes, we do not directly observe $P(k, t|z)$, measure the relevant pretraining frequencies, or identify frequency imbalance as the unique cause of each error. TrapQA therefore diagnoses a controlled class of inference-misalignment failures rather than all hallucinations or their real-world prevalence; open-ended fabrication due to missing knowledge and failures of abstention or uncertainty calibration remain outside our empirical scope.

6 Conclusion

We study hallucination as a form of *misaligned inference*: a model may possess the relevant facts, yet still follow a statistically dominant shortcut path that is inconsistent with the prompt's decisive constraint. We formalize this view through a latent key–task model, showing how pretraining-frequency imbalance can suppress the correct inference path and induce positive inference loss. To evaluate this perspective, we introduce TRAPQA, combining scientist disambiguation with everyday constraint following under salient shortcuts. In Scientist QA, pairing each primary question with supplementary factual probes separates factual ignorance from knowledge-deployment failure. Across frontier models, many errors occur even when both probes are answered correctly, while providing explicit profile context nearly eliminates the errors. These findings suggest that some observed hallucinations are not explained by knowledge storage alone, but also involve failures to deploy behaviorally accessible facts along the relevant inference path. Our findings motivate methods that go beyond factual coverage, including constraint-first reasoning that explicitly identifies the decisive condition before answering, consistency checks between final answers and independently elicited facts, and re-evaluation or abstention when the two conflict. Evaluating these interventions remains an important direction for future work.

Limitations

This research has some limitations. First, though we covered several frontier model families, our results remain limited to the tested models: GPT-5.2,

Gemini 3.1 Pro Preview, Claude Sonnet 4.6 and DeepSeek V3.2 chat/reasoning. We have explicitly reported the thinking settings, and API versions, but reruns may differ as provider-hosted systems change. Second, Gemini assisted in constructing the initial Scientist QA questions, creating possible generator–evaluator coupling. This coupling may affect absolute accuracy or comparative rankings, particularly Gemini's relative performance. Our central empirical conclusion therefore rests on the presence of probe-conditioned knowledge-deployment failures across all evaluated model families, rather than on any direct cross-model ranking. Third, although we aim to construct questions whose answers are stable over time, some items may still be affected by temporal drift. Scientific breakthroughs, technological changes, or industry practice may alter what is regarded as common sense, and scientists may later receive new honors, change positions, or become associated with new fields as their careers continue. Thus, future evaluations should treat the released answers as tied to the dataset construction time and re-audit items when using TrapQA in substantially later model evaluations.

References

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2024. The reversal curse: Llms trained on "a is b" fail to learn "b is a". *Preprint*, arXiv:2309.12288.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, and 13 others. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *Preprint*, arXiv:2307.15217.

Xiang Chen, Chenxi Wang, Yida Xue, Ningyu Zhang, Xiaoyan Yang, Qiang Li, Yue Shen, Lei Liang, Jinjie Gu, and Huajun Chen. 2024. Unified hallucination detection for multimodal large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3235–3252, Bangkok, Thailand. Association for Computational Linguistics.

Qinyuan Cheng, Tianxiang Sun, Wenwei Zhang, Siyin Wang, Xiangyang Liu, Mozhi Zhang, Junliang He, Mianqiu Huang, Zhangyue Yin, Kai Chen, and Xipeng Qiu. 2023. Evaluating hallucinations in chinese large language models. *Preprint*, arXiv:2310.03368.

Kalyan Cherukuri and Lav R. Varshney. 2026. Hallucination basins: A dynamic framework for understanding and controlling llm hallucinations. *Preprint*, arXiv:2604.04743.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. Deep reinforcement learning from human preferences. *Preprint*, arXiv:1706.03741.

David Dale, Elena Voita, Janice Lam, Prangthip Hansanti, Christophe Ropers, Elahe Kalbassi, Cynthia Gao, Loïc Barrault, and Marta R. Costa-jussà. 2023. Halomi: A manually annotated benchmark for multilingual hallucination and omission detection in machine translation. *Preprint*, arXiv:2305.11746.

Simon De Deyne, Danielle J Navarro, Amy Perfors, Marc Brysbaert, and Gert Storms. 2019. The "small world of words" english word association norms for over 12,000 cue words. *Behavior research methods*, 51(3):987–1006.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Nouha Dziri, Sivan Milton, Mo Yu, Osmar Zaiane, and Siva Reddy. 2022. On the origin of hallucinations in conversational models: Is it the datasets or the models? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5271–5285, Seattle, United States. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Preprint*, arXiv:2007.12626.

Cheng Gao, Huimin Chen, Chaojun Xiao, Zhiyi Chen, Zhiyuan Liu, and Maosong Sun. 2025. H-neurons: On the existence, impact, and origin of hallucination-associated neurons in llms. *Preprint*, arXiv:2512.01797.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. 2024. Does fine-tuning llms on new knowledge encourage hallucinations? *Preprint*, arXiv:2405.05904.

Gaurav Rohit Ghosal, Tatsunori Hashimoto, and Aditi Raghunathan. 2024. Understanding finetuning for factual knowledge extraction. In *Forty-first International Conference on Machine Learning*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. 2024. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. *Preprint*, arXiv:2310.14566.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. True: Re-evaluating factual consistency evaluation. *Preprint*, arXiv:2204.04991.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. 2025. Why language models hallucinate. *Preprint*, arXiv:2509.04664.

Adam Tauman Kalai and Santosh S. Vempala. 2024. Calibrated language models must hallucinate. *Preprint*, arXiv:2311.14648.

Katie Kang, Eric Wallace, Claire Tomlin, Aviral Kumar, and Sergey Levine. 2024. Unfamiliar finetuning examples control how language models hallucinate. *Preprint*, arXiv:2403.05612.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2025. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. *Preprint*, arXiv:2409.12941.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization. *Preprint*, arXiv:1910.12840.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2019. Hallucinations in neural machine translation.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Sheng-Chieh Lin, Luyu Gao, Barlas Oguz, Wenhan Xiong, Jimmy Lin, Wen tau Yih, and Xilun Chen. 2024. Flame: Factuality-aware alignment for large language models. *Preprint*, arXiv:2405.01525.

Shuyuan Lin, Lei Duan, Philip Hughes, and Yuxuan Sheng. 2025. Harnessing rlhf for robust unanswerability recognition and trustworthy response generation in llms. *Preprint*, arXiv:2507.16951.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Langming Liu, Kangtao Lv, Haibin Chen, Weidong Zhang, Yejing Wang, Shilei Liu, Xin Tong, Yujin Yuan, Yongwei Wang, Wenbo Su, and Bo Zheng. 2026. Pretrainrl: Alleviating factuality hallucination of large language models at the beginning. *Preprint*, arXiv:2602.01875.

Xiao Liu, Hanyu Lai, Hao Yu, Yifan Xu, Aohan Zeng, Zhengxiao Du, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. Webglm: Towards an efficient web-enhanced question answering system with human preferences. *Preprint*, arXiv:2306.07906.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Nick McKenna, Tianyi Li, Liang Cheng, Mohammad Javad Hosseini, Mark Johnson, and Mark Steedman. 2023. Sources of hallucination by large language models on inference tasks. *Preprint*, arXiv:2305.14552.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin

Chess, and John Schulman. 2022. Webgpt: Browser-assisted question-answering with human feedback. *Preprint*, arXiv:2112.09332.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI. 2024. New embedding models and api updates. Introduces text-embedding-3-small.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

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

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. *Preprint*, arXiv:2104.13346.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. *Preprint*, arXiv:2104.06683.

Mengjie Ren, Boxi Cao, Hongyu Lin, Cao Liu, Xianpei Han, Ke Zeng, Guanglu Wan, Xunliang Cai, and Le Sun. 2024. Learning or self-aligning? rethinking instruction fine-tuning. *Preprint*, arXiv:2402.18243.

Peter Steinberger. 2026. Introducing openclaw. `https://openclaw.ai/blog/introducing-openclaw`. OpenClaw Blog. Accessed: 2026-04-22.

Yiyou Sun, Yu Gai, Lijie Chen, Abhilasha Ravichander, Yejin Choi, and Dawn Song. 2025. Why and how llms hallucinate: Connecting the dots with subsequence associations. *Preprint*, arXiv:2504.12691.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, and 1332 others. 2025. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. *Preprint*, arXiv:2004.04228.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024a. Measuring short-form factuality in large language models. *Preprint*, arXiv:2411.04368.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. 2024b. Long-form factuality in large language models. *Preprint*, arXiv:2403.18802.

Zhepei Wei, Xiao Yang, Kai Sun, Jiaqi Wang, Rulin Shao, Sean Chen, Mohammad Kachuee, Teja Gollapudi, Tony Liao, Nicolas Scheffer, Rakesh Wanga, Anuj Kumar, Yu Meng, Wen tau Yih, and Xin Luna Dong. 2025. Truthrl: Incentivizing truthful llms via reinforcement learning. *Preprint*, arXiv:2509.25760.

Wikidata contributors. 2026. Wikidata: The Free Knowledge Base. `https://www.wikidata.org/`.

Wikimedia Foundation. 2026. Wikimedia Analytics API: Pageviews. `https://doc.wikimedia.org/generated-data-platform/aqs/analytics-api/concepts/page-views.html`.

Wikipedia contributors. 2026. Wikipedia, The Free Encyclopedia. `https://www.wikipedia.org/`.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*. OpenReview.net.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2025. Hallucination is inevitable: An innate limitation of large language models. *Preprint*, arXiv:2401.11817.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi R. Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. R-tuning: Instructing large language models to say 'i don't know'. *Preprint*, arXiv:2311.09677.

Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A. Smith. 2023. How language model hallucinations can snowball. *Preprint*, arXiv:2305.13534.

Shuo Zhang, Fabrizio Gotti, Fengran Mo, and Jian-Yun Nie. 2025a. Measuring the impact of lexical training data coverage on hallucination detection in large language models. *Preprint*, arXiv:2511.17946.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Chen Xu, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2025b. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

Wenting Zhao, Tanya Goyal, Yu Ying Chiu, Liwei Jiang, Benjamin Newman, Abhilasha Ravichander, Khyathi Chandu, Ronan Le Bras, Claire Cardie, Yuntian Deng, and Yejin Choi. 2024. Wildhallucinations: Evaluating long-form factuality in llms with real-world entity queries. *Preprint*, arXiv:2407.17468.

Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. 2023. Why does chatgpt fall short in providing truthful answers? *Preprint*, arXiv:2304.10513.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences. *Preprint*, arXiv:1909.08593.

A Artifact licenses and terms

Scientist QA uses public Wikipedia-linked scientist profiles and Wikidata QIDs. Wikipedia text is available under CC BY-SA 4.0 unless otherwise noted, while Wikidata structured data is released under CC0 (Wikipedia contributors, 2026; Wikidata contributors, 2026). Real-Life Constrained QA uses SWOW only for seed selection; we cite the original SWOW resource and do not redistribute raw SWOW participant records or cue–response tables. The released benchmark package contains derived QA items, labels, prompts, and saved evaluation outputs, with the final redistribution license stated in the repository README.

B Artifact use and intended use

We use existing artifacts only for research and diagnostic evaluation. Wikipedia- and Wikidata-derived scientist information is used to construct public-profile disambiguation questions; SWOW is used only for non-commercial seed selection, and we do not redistribute raw SWOW data. The new TrapQA artifacts are intended for research on hallucination, knowledge deployment, and constraint-sensitive reasoning, not for deployment certification, individual assessment, or commercial redistribution of source-derived data.

C Additional Related Work

This appendix expands the related-work discussion from Section 2, covering theoretical accounts, pipeline-stage sources of hallucination, reinforcement learning from feedback, and evaluation benchmarks.

C.1 Mechanisms and Sources of Hallucination

Pre-LLM hallucination. Hallucination has long been studied in language generation. In data-to-text generation, Wiseman et al. (2017) showed that neural models can produce fluent outputs that nevertheless fail to faithfully reflect the underlying records. In neural machine translation, Lee et al. (2019) analyzed hallucinations as spurious translations unrelated to the source text, a phenomenon further studied by Raunak et al. (2021). Similarly, Maynez et al. (2020) found that neural summarization models frequently generate content that is not faithful to the source document. Although these settings differ in task formulation, they share the common problem that models may produce plausible text that is insufficiently grounded in the conditioning input.

Theoretical and mechanistic accounts. Several studies examine hallucination from the perspective of fundamental limitations. Xu et al. (2025) show, in a formal learning-theoretic setting, that LLMs cannot learn all computable functions and therefore cannot completely avoid hallucination when used as general-purpose problem solvers. Kalai and Vempala (2024) derive a statistical lower bound on hallucination for calibrated language models on certain classes of facts, suggesting that hallucination cannot be eliminated solely through better calibration. From a mechanistic perspective, Sun et al. (2025) propose a subsequence-association framework for tracing hallucinations, arguing that hallucinations can arise when dominant hallucinatory associations outweigh faithful ones during generation. More recently, Cherukuri and Varshney (2026) analyze hallucination through a dynamical-systems view of hidden-state trajectories, in which hallucination behavior is characterized by task-dependent latent-space basin structure.

These accounts are closely related to our work in treating hallucination as a competition between faithful and unfaithful associations. Our framework differs by focusing on prompts in which a decisive local constraint conflicts with a statistically salient shortcut. This setting allows us to study not only whether a model knows the relevant facts, but also whether it retrieves and applies the constraint-sensitive inference path required by the prompt.

Training, fine-tuning, and inference. Hallucinations can arise from multiple stages of the LLM pipeline, including pretraining, post-training, and inference. At the pretraining level, distributional imbalance can make false or misleading continuations more probable than correct ones, particularly when correct facts are rare or expressed inconsistently (Zhang et al., 2025a; Liu et al., 2026). More broadly, noisy, outdated, or contradictory training data may contribute to unsupported generations (Ji et al., 2023; Huang et al., 2025). Kalai et al. (2025) further argue that modern training and evaluation procedures can reward guessing over acknowledging uncertainty, causing models to produce plausible answers even when they should abstain.

Hallucinations may also persist or emerge during fine-tuning. Several studies find that language models struggle to acquire new factual knowledge

through fine-tuning (Gekhman et al., 2024; Kang et al., 2024; Lin et al., 2024; Ren et al., 2024). In particular, fine-tuning examples that introduce new knowledge may be learned more slowly than examples consistent with the model's pre-existing knowledge, and once learned, may increase hallucination on previously acquired facts (Gekhman et al., 2024). Related work also shows that fine-tuning can differentially affect popular and unpopular factual knowledge, with models fine-tuned on more widely known facts tending to achieve higher factual accuracy than those fine-tuned on less popular facts (Ghosal et al., 2024).

At inference time, hallucinations can be amplified by prompt ambiguity, decoding behavior, and reliance on memorized or frequency-biased patterns (Zhang et al., 2023; Huang et al., 2025). McKenna et al. (2023) identify attestation and relative-frequency biases in natural language inference as sources of hallucination-like errors, showing that models may rely on whether a hypothesis is attested in pretraining data rather than on the provided premise. Berglund et al. (2024) expose a related limitation, the reversal curse, in which models trained on facts in one direction fail to reliably answer semantically equivalent queries in the reverse direction. Zheng et al. (2023) analyze ChatGPT failures in open-domain question answering and identify factuality, knowledge memorization, and knowledge recall as central sources of error. Together, these studies suggest that hallucination is not merely a matter of missing knowledge; it can also reflect failures in retrieving, comparing, or applying knowledge under the constraints of a particular prompt.

Reinforcement learning from feedback. Reinforcement learning from human feedback (RLHF) builds on preference-based reinforcement learning and human-preference fine-tuning of language models (Christiano et al., 2023; Ziegler et al., 2020). Its effect on hallucination remains debated. On one hand, Ouyang et al. (2022) report that instruction tuning with human feedback improves truthfulness on several evaluations. On the other hand, reward models can be imperfect proxies for truthfulness, and optimizing against them may encourage reward hacking or plausible-sounding responses that satisfy human preferences without being fully faithful (Casper et al., 2023). Recent work therefore proposes reinforcement-learning objectives that explicitly penalize hallucination or reward truthful abstention (Lin et al., 2025; Wei et al., 2025; Zhang et al., 2024). These approaches complement our work: rather than proposing a mitigation objective, we construct diagnostic settings that expose when models select a shortcut inference path despite having access to the relevant facts in closed-book probes.

C.2 Hallucination Evaluation Benchmarks

Evaluating hallucination in language generation has attracted sustained attention, with benchmarks spanning different tasks, grounding sources, and annotation strategies. For text-only generation, early resources established task-specific foundations in summarization faithfulness, table-to-text fidelity, and open-domain factuality (Kryściński et al., 2019; Wang et al., 2020; Pagnoni et al., 2021; Fabbri et al., 2021; Parikh et al., 2020; Honovich et al., 2022; Lin et al., 2022; Li et al., 2023; Cheng et al., 2023; Dale et al., 2023). More recent work has shifted toward large-scale and increasingly automated factuality evaluation. FActScore (Min et al., 2023) decomposes long-form outputs into atomic facts and evaluates whether each is supported by a reliable knowledge source. LongFact (Wei et al., 2024b) targets factuality in extended, open-domain responses, while SimpleQA (Wei et al., 2024a) provides short, fact-seeking questions with single, unambiguous answers and explicit grading of correct, incorrect, and not-attempted responses. WildHallucinations (Zhao et al., 2024) evaluates long-form factuality on real-world entity queries, including many entities outside Wikipedia coverage.

A parallel line of work examines hallucination in retrieval-augmented generation (RAG). RAGTruth (Niu et al., 2024) provides human annotations of hallucinations in naturally generated RAG outputs, including word-level labels. FRAMES (Krishna et al., 2025) evaluates factuality, retrieval accuracy, and reasoning in end-to-end RAG scenarios, especially under multi-hop reasoning demands. In multimodal settings, researchers study hallucination in vision-language models, including object-level, relation-level, and broader multimodal hallucination detection settings (Guan et al., 2024; Chen et al., 2024). Collectively, these benchmarks reflect a progression from task-specific faithfulness evaluation toward broader, multi-task, and more automated hallucination assessment.

Our benchmark is complementary to these evaluation efforts. Existing benchmarks primarily mea-

sure whether a model's output is factual, faithful, or grounded in evidence. By contrast, Scientist QA and Real-Life Constrained QA are designed to isolate why a model fails in controlled forced-choice settings. Scientist QA tests whether models can deploy candidate-specific facts under disambiguation, while Real-Life Constrained QA tests whether models can override SWOW-derived associative cues with prompt-grounded physical, spatial, procedural, or medium-specific constraints (De Deyne et al., 2019). This design allows us to distinguish simple ignorance from knowledge-deployment failures in which the relevant information is available to the model but not used in the task-relevant inference path.

D Scientist QA Construction Details

This appendix gives the full Scientist QA construction pipeline: profile collection, name-removed profile linearization, hard-pair mining, question generation, and filtering. Figure 3 summarizes the construction pipeline.

D.1 Scientist Profiles

We collect 9,090 scientists with dedicated Wikipedia pages. Each scientist is represented as a structured profile containing attributes such as occupation, field, notable work, awards, education, and a Wikidata QID. The QID is used only for bookkeeping, deduplication, and linking candidates across processing stages. Appendix F.1 shows an example profile.

For pair mining, we remove the scientist's name from each profile and linearize the remaining attributes into a short paragraph, which we call the *name-removed profile*. This prevents the embedding model from matching scientists by name while preserving semantic information from their profile attributes. Appendix F.2 shows an example.

D.2 Hard-Pair Mining

Let $\mathbf{e}_A$ and $\mathbf{e}_B$ denote the embeddings of the name-removed profiles of scientists A and B, obtained using `text-embedding-3-small` (OpenAI, 2024). Let $\mathrm{TC}(A)$ be the number of retained attribute fields for scientist A, and let λ be the median tag count across all scientists; in our data, $\lambda = 7$. We score each pair by

$$s(A,B) = \frac{\mathbf{e}_A^\top \mathbf{e}_B}{\|\mathbf{e}_A\| \, \|\mathbf{e}_B\|} \cdot \frac{\min(\mathrm{TC}(A), \mathrm{TC}(B))}{\min(\mathrm{TC}(A), \mathrm{TC}(B)) + \lambda} \quad (1)$$

The cosine term measures semantic similarity, while the penalty term downweights pairs whose similarity may be driven by sparse metadata. We rank all scientist pairs by Equation 1 and retain the top 0.01% highest-scoring pairs, yielding 2,958 candidate pairs.

D.3 Question Generation and Filtering

For each candidate pair (A, B), we prompt Gemini to generate a third-person biographical paragraph broadly compatible with both scientists, followed by a single decisive constraint that rules out exactly one candidate. Each question ends with *"Who is this person?"* We then use ChatGPT to filter malformed or unverifiable outputs, including cases where the decisive constraint does not distinguish the pair, contradicts the candidate profiles, or cannot be verified against the paired candidates. After filtering and excluding the 33 invalid items identified in the final proofreading pass, the final evaluation set contains 2,925 questions.

Each retained question is evaluated in two variants:

1. *prepend_names*, which prepends only the two candidate names;
2. *prepend_profiles*, which prepends the full profiles of both candidates.

Each question is also paired with two supplementary probe questions derived from the decisive constraint, one for each candidate.

E Implementation Details

This appendix specifies the prompt formats, answer parsing rules, and failure handling used in evaluation.

E.1 Model versions

For reproducibility, Table 3 reports the model identifiers and inference settings used in evaluation. Exact access dates, provider-side parameters, and run identifiers should be preserved with the released run logs.

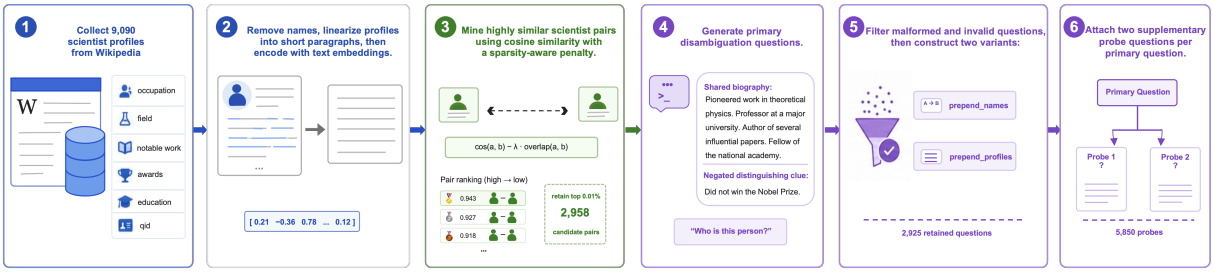

Figure 3: Overview of the Scientist QA construction pipeline. Starting from Wikipedia-linked scientist profiles, we construct highly confusable scientist pairs, generate pairwise disambiguation questions, and attach two supplementary probes to each primary question.

Table 3: Model identifiers and inference settings used in the evaluation. Low- and high-thinking settings are provider-specific controls; for DeepSeek, the two columns correspond to non-reasoning and reasoning model aliases.

Family	Non-reasoning API ID	Reasoning API ID
GPT-5.2	`gpt-5.2-2025-12-11`	`gpt-5.2-2025-12-11`
Claude Sonnet 4.6	`claude-sonnet-4-6`	`claude-sonnet-4-6`
Gemini 3.1 Pro Preview	`gemini-3.1-pro-preview`	`gemini-3.1-pro-preview`
DeepSeek V3.2	`deepseek-chat`	`deepseek-reasoner`

E.2 Prompt Templates

We evaluate each benchmark item in separate conversations: one for the primary question and one for each supplementary probe.

Names-only prompt (*prepend_names*).

> `Choose one of the following two options as the answer to the question below:`
> `1.` A
> `2.` B
> `Question:`
> $question_i$

Here A and B are the two candidate scientists. Their order is randomized across items.

Profiles-in-context prompt (*prepend_profiles*).

> `Given two profiles of two persons:`
> $profile_A$
> $profile_B$
> `Choose exactly one profile from the two, and output the name of the person as the answer to the following question:`
> $question_i$

Supplementary probes. Each supplementary probe is asked independently as a binary factual question about one candidate and the decisive relation. For example:

> `Did Albert Einstein receive the Nobel Prize in Physics?`

E.3 Answer Matching and Failure Handling

For primary questions, the model is instructed to output exactly one of the two candidate names. We normalize whitespace, capitalization, and minor formatting differences before matching. If the normalized response matches the correct candidate, it is counted as correct; if it matches the distractor, it is counted as a hallucination. If the response matches neither candidate, it is also counted as a hallucination. Across the primary *prepend_names* Scientist QA experiments, only two unmatched primary-question responses remain after normalization, both from Claude-low.

For supplementary probes, binary answers are normalized to true/false labels. Each primary-question outcome is then paired with its two probe outcomes to determine whether the model *knows both*, *knows one*, or *knows neither* of the relevant probe facts.

F Additional Examples

F.1 Profile Example

Figure 4 provides a complete profile of Wolfgang Pauli.

F.2 Name-Removed Profile Example

Example name-removed profile: Albert Einstein

occupation: inventor, mathematician, philosopher of science, physicist, professor, science writer, theoretical physicist, university teacher; field: theoretical physics; notable work: general relativity, mass–energy equivalence, photoelectric effect, quantum mechanics, special relativity, theory of Brownian motion, theory of relativity; awards: Copley Medal, Foreign Member of the Royal Society, Franklin Medal, Max Planck Medal, Nobel Prize in Physics, Pour le Mérite; education: ETH Zurich, Luitpold-Gymnasium, University of Zurich; positions: professor.

F.3 Question Example

Figure 5 shows the profiles-in-context prompt variant.

Primary question example

This prominent theoretical physicist, mathematician, and university teacher made significant contributions to science. In recognition of their work, they delivered the Josiah Willard Gibbs Lectureship and were elected a Foreign Member of the Royal Society. However, this scientist never received the Nobel Prize in Physics. Who is this person?

Paired supplementary probes

Did Albert Einstein receive the Nobel Prize in Physics?
Did Edward Witten receive the Nobel Prize in Physics?

***prepend_names* variant**

Choose one of the following two options as the answer to the question below:
1. Edward Witten
2. Albert Einstein
Question:
This prominent theoretical physicist, mathematician, and university teacher made significant contributions to science. In recognition of their work, they delivered the Josiah Willard Gibbs Lectureship and were elected a Foreign Member of the Royal Society. However, this scientist never received the Nobel Prize in Physics. Who is this person?

Example structured profile: Wolfgang Pauli

```
{
  "Wolfgang Pauli": {
    "occupation": [
      "theoretical physicist",
      "university teacher",
      "chemist",
      "physicist"
    ],
    "award_received": [
      "Nobel Prize in Physics",
      "Max Planck Medal",
      "Lorentz Medal",
      "Foreign Member of the Royal Society",
      "honorary doctor of the University of Vienna"
    ],
    "field": [
      "quantum mechanics",
      "particle physics"
    ],
    "notable_work": [
      "Pauli exclusion principle",
      "Pauli matrices",
      "Pauli equation"
    ],
    "education": [
      "Ludwig-Maximilians-Universität München",
      "Bundesgymnasium Döbling"
    ],
    "qid": "Q65989"
  }
}
```

Figure 4: Example structured scientist profile used in Scientist QA.

***prepend_profiles* variant**

Given two profiles of two persons:

```
name: Edward Witten
occupation: mathematician; physicist; university teacher; theoretical physicist
award_received: Fields Medal; MacArthur Fellowship; Isaac Newton Medal; ...
field: physics; mathematical physics; string theory
education: Princeton University; University of Wisconsin--Madison; ...

name: Albert Einstein
occupation: theoretical physicist; philosopher of science; science writer; ...
award_received: Nobel Prize in Physics; Copley Medal; Franklin Medal; ...
field: theoretical physics
notable_work: general relativity; special relativity; photoelectric effect; ...
education: ETH Zurich; University of Zurich; ...
```

Choose exactly one profile from the two, and output the name of the person as the answer to the following question:

This prominent theoretical physicist, mathematician, and university teacher made significant contributions to science. In recognition of their work, they delivered the Josiah Willard Gibbs Lectureship and were elected a Foreign Member of the Royal Society. However, this scientist never received the Nobel Prize in Physics. Who is this person?

Figure 5: Example *prepend_profiles* prompt variant. The ellipses in the profile example indicate omitted attributes for readability.

G Real-Life Constrained QA Construction Details

This appendix describes Real-Life Constrained QA, a collection of realistic two-option questions in which a locally plausible shortcut conflicts with a physical, spatial, procedural, or medium-specific constraint. Unlike Scientist QA, which tests entity disambiguation among highly similar scientists, this component targets shortcut-driven failures in everyday scenarios. Each item presents a short first-person situation and two candidate actions or media. One option is superficially attractive because it matches a strong prior association, while the other is correct because it satisfies the constraint implied by the scenario. The final collection contains 500 questions covering 13 aspects of daily life.

G.1 Association Mining from SWOW

We begin from SWOW (De Deyne et al., 2019), a large-scale psycholinguistic resource of human word associations. For each cue word, we use high-probability first responses as candidate shortcut associations. We lightly normalize and filter these associations by lowercasing, lemmatizing, merging obvious duplicates, and removing generic or noisy responses. The result is a cleaned bank of human-salient cue–response pairs suitable for question generation. We used several preprocessing packages for SWOW seed preprocessing. We use spaCy with the `en_core_web_sm` English pipeline for tokenization, POS/stopword checks, and lemmatization; NLTK WordNet for coarse lexical-type labels from synsets and lexnames; and `wordfreq` Zipf frequencies to filter overly common or rare responses. We disable the spaCy parser for this preprocessing step and use heuristic frequency thresholds of `high_zipf=6.5` and `low_zipf=1.0` in the first-pass filter.

G.2 Template Families and Seed Selection

We organize cleaned associations into eight seed template families corresponding to recurring hidden-constraint patterns. Examples include `vehicle_required`, where the task requires bringing a vehicle rather than merely reaching a location; `delivery_medium`, where a physical item cannot be replaced by a digital surrogate; `recording_medium`, where the correct action depends on the required recording modality; and `tool_required`, where a specific tool is necessary for task completion.

For each seed, we annotate structured metadata, including the scenario role, latent constraint type, and intended shapes of the correct and shortcut options. We prioritize seeds whose associations are concrete, whose constraints are easy to instantiate in everyday settings, and whose shortcut options are plausible without being absurd. We also cap overrepresented lemmas within each family to maintain scenario diversity.

G.3 Generation, Filtering, and Augmentation

For each selected seed, we use GPT to augment seed questions into multiple first-person scenarios following the corresponding family template. Each generated item must be realistic, self-contained, and unambiguous: the incorrect option should be a plausible shortcut, while the correct option should be determined by a recoverable constraint in the scenario. Claude is then used to proofread the resulting questions for ambiguity, plausibility, and constraint validity. We manually filter malformed or weak items, including cases where both options are arguably valid, the constraint is too explicit, the scenario depends on niche expertise, or the shortcut option is implausible.

G.4 Benchmark Format

Each Real-Life Constrained QA item consists of a short scenario, two candidate options and a gold label.

H Extended Empirical Results

This appendix provides the full empirical breakdowns supporting Section 5. Unless otherwise stated, all tables refer to the retrieval-sensitive *prepend_names* condition over 2,925 Scientist QA questions.

H.1 Full Probe-State Breakdowns

For each pairwise question, we use two closed-book supplementary probes targeting the decisive relation. We group examples into three probe-defined knowledge states:

- **Knows both:** both supplementary probes are answered correctly;
- **Knows one:** exactly one supplementary probe is answered correctly;
- **Knows neither:** both supplementary probes are answered incorrectly.

Table 4 reports correct and incorrect pairwise outcomes within each state. Correct answers are typically concentrated in the *knows-both* bucket, while incorrect answers shift toward the *knows-one* and *knows-neither* buckets. However, the *knows-both* rows still contain nonzero error rates, showing that correct probe-level knowledge does not guarantee correct comparative deployment.

H.2 Eliminative-Probe Asymmetry

The two supplementary probes play different diagnostic roles. One tests the fact that should eliminate the distractor; the other tests the compatibility of the correct candidate with the decisive constraint. Table 5 focuses on one-probe-correct cases. Across all eight model settings, hallucination is higher when the model misses the eliminative probe than when it misses the compatibility probe. This asymmetry supports the latent key–task account in Section 3: the decisive relation is often not merely a fact about the correct candidate, but the fact that suppresses the shortcut candidate. When this eliminative fact is not retrieved, the high-salience candidate remains available as a plausible continuation.

H.3 Consensus Failures

To distinguish idiosyncratic model errors from shared shortcut directions, we identify questions missed by multiple model settings. Of the 2,925 questions, 1,489 are missed by at least one model setting, and 10 are missed by all eight settings. Table 6 lists these all-setting consensus failures. They concentrate on high-frequency biographical relation families such as education, awards, professional roles, and offices. In these cases, the distractor satisfies a salient affirmative association, while the correct answer is determined by an explicit incompatibility or non-possession constraint. These examples support the common-shortcut assumption in Theorem 3.4: different model families can be biased toward the same incorrect answer when a dominant association conflicts with the prompt's decisive constraint.

H.4 Probe-Underdetermined Cases

We refer to the one-probe-correct subset as **probe-underdetermined** from the model's local probe state. Since the two gold probe labels are complementary, answering exactly one probe correctly is equivalent to predicting the same value for both probes, either both true or both false. In this regime, the two probe predictions alone do not determine the correct pairwise answer for the model. Table 7 reports the size and behavior of this subset.

Table 8 further shows that behavior in this regime is not explained by simply choosing the more famous scientist. In several settings, the model chooses the more famous candidate less than half the time.

H.5 Fame-Based Analyses

We examine whether hallucinations can be explained by a simple prior toward the more famous scientist. For each scientist s, we define

$$\begin{aligned}\mathrm{Fame}(s) = \frac{1}{3}[&\mathrm{norm}(\mathrm{pageLength}_s) + \\ &\mathrm{norm}(\mathrm{pageViews}) + \\ &\mathrm{norm}(\mathrm{externalLinks}_s)]\end{aligned}$$

where $\mathrm{norm}(\cdot)$ denotes corpus-level normalization across scientists. The fame rank is induced by this score.

We use the 2020-01-01 ~ 2025-12-31 calendar-year window for the page view count because it is the most recent complete multi-year window before our evaluation period (Wikimedia Foundation, 2026). This window balances recency with robustness to short-term spikes in public attention and avoids using page-view data generated after the benchmark evaluation.

These analyses serve as negative controls. The wrong candidate is more famous in 61.30% of benchmark questions, but among hallucinated cases this fraction is lower, ranging from 44.64% to 57.12% depending on the model setting. Table 9 shows that hallucination rates are lower, not higher, when the incorrect candidate is more famous.

H.6 Confidence Diagnostics

Accuracy and confidence are imperfect certificates of faithful reasoning. First, a model can sometimes answer the pairwise question correctly without answering both probes correctly; for example, only 62.15% of DeepSeek-low's correct pairwise answers occur in the both-probe-correct regime. This suggests that some correct answers may be supported by shortcuts that happen to point to the correct candidate.

Second, self-reported confidence does not reliably separate correct from hallucinated answers across model families. Table 10 reports confidence for correct and hallucinated pairwise answers. Hallucinated answers are usually less confident than

Model	Mode	Knows both correct	Knows both wrong	Wrong rate	Knows one correct	Knows one wrong	Wrong rate	Knows neither correct	Knows neither wrong	Wrong rate
Claude Sonnet 4.6	high	2457	66	2.62%	279	107	27.72%	7	9	56.25%
Claude Sonnet 4.6	low	1809	424	18.99%	401	266	39.88%	16	9	36.00%
DeepSeek V3.2 Reasoner	high	2179	140	6.04%	410	152	27.05%	27	17	38.64%
DeepSeek V3.2 Chat	low	1141	600	34.46%	665	468	41.31%	30	21	41.18%
Gemini 3.1 Pro Preview	high	2785	74	2.59%	48	18	27.27%	0	0	–
Gemini 3.1 Pro Preview	low	2784	57	2.01%	68	16	19.05%	0	0	–
GPT-5.2	high	2387	159	6.25%	229	133	36.74%	9	8	47.06%
GPT-5.2	low	2304	198	7.91%	260	134	34.01%	17	12	41.38%

Table 4: Probe-conditioned breakdown of primary-question outcomes in the names-only Scientist QA condition. Each bucket is defined by the number of supplementary probes answered correctly. "Correct" and "wrong" count pairwise disambiguation outcomes within that bucket, and "Wrong rate" is computed within the corresponding bucket.

Model	Mode	n missing elim.	Hall. when elim. missed	n missing compat.	Hall. when compat. missed	Gap
Claude Sonnet 4.6	high	128	28.91%	258	27.13%	1.77
Claude Sonnet 4.6	low	489	41.72%	178	34.83%	6.89
DeepSeek V3.2 Reasoner	high	272	29.04%	290	25.17%	3.87
DeepSeek V3.2 Chat	low	298	46.64%	835	39.40%	7.24
Gemini 3.1 Pro Preview	high	40	40.00%	26	7.69%	32.31
Gemini 3.1 Pro Preview	low	35	31.43%	49	10.20%	21.22
GPT-5.2	high	126	46.03%	236	31.78%	14.25
GPT-5.2	low	153	37.91%	241	31.54%	6.37

Table 5: Hallucination rates in one-probe-correct cases for the names-only Scientist QA condition. "Elim." denotes the probe whose correct answer eliminates the distractor; "compat." denotes the probe whose correct answer confirms the correct candidate's compatibility with the decisive constraint. The gap is the difference between the two hallucination rates.

correct answers, but the absolute confidence remains high in many settings. DeepSeek-low is especially poorly separated: hallucinated and correct answers have nearly identical mean confidence.

H.7 Real-Life Constrained QA Results

Real-Life Constrained QA contains 500 two-option scenarios covering 13 aspects of daily life. Table 11 reports the final error counts and rates for the evaluated models.

Analysis-family breakdown. For descriptive analysis, we group the final 500 questions into 12 semantic analysis families. These are post-construction groupings of the final questions and are distinct from the eight seed template families used during generation in Appendix G.2. Table 12 reports the size of each family and the number of questions answered incorrectly by at least one evaluated model. At least one model makes an error on 211 questions (42.2%). Errors occur in all 10 families containing at least 12 items; the two remaining families contain only 2 and 1 items, respectively. Because these strata are unevenly sized and were not designed for causal comparison, we treat the rates as descriptive and do not interpret them as estimates of intrinsic family difficulty.

Constraint-type breakdown. We additionally group the 500 questions by the 10 annotated constraint types. Table 13 reports the size of each type and the number of questions answered incorrectly by at least one evaluated model. Failures occur in 9 of the 10 constraint types; the only type with no observed error contains two questions. As with the analysis families above, these categories are unevenly sized and are intended as descriptive strata rather than controlled comparisons of intrinsic difficulty.

Question ID	Correct candidate	Distractor	Decisive relation family
question_0214	Klaus von Klitzing	Rudolf Mössbauer	Education / institution
question_0596	Glenn T. Seaborg	Mildred Dresselhaus	Education / institution
question_0797	Jennifer Doudna	Frances Arnold	Award / honor
question_1092	Fritz Lipmann	Otto Heinrich Warburg	Education / institution
question_1161	Norman Foster Ramsey, Jr.	Carl Wieman	Award / honor
question_1517	Joseph-Louis Lagrange	François Arago	Political office / role
question_1772	Alexander R. Todd, Baron Todd	Svante Arrhenius	Occupation / role
question_1981	Harold Clayton Urey	Mildred Dresselhaus	Education / institution
question_2183	Steven Weinberg	Leon Max Lederman	Award / honor
question_2370	Robert Aumann	Gérard Debreu	Education / institution

Table 6: The 10 Scientist QA questions missed by all eight model settings. Question IDs and candidate names are produced by the analysis notebook; relation-family labels are manual annotations based on the decisive constraint.

Model	Mode	n probe-underdet.	Pct. of questions	Accuracy	Hall. rate	Both predicted false	Both predicted true
Claude Sonnet 4.6	high	386	13.20%	72.28%	27.72%	33.16%	66.84%
Claude Sonnet 4.6	low	667	22.80%	60.12%	39.88%	73.31%	26.69%
DeepSeek V3.2 Reasoner	high	562	19.21%	72.95%	27.05%	48.40%	51.60%
DeepSeek V3.2 Chat	low	1133	38.74%	58.69%	41.31%	26.30%	73.70%
Gemini 3.1 Pro Preview	high	66	2.26%	72.73%	27.27%	60.61%	39.39%
Gemini 3.1 Pro Preview	low	84	2.87%	80.95%	19.05%	41.67%	58.33%
GPT-5.2	high	362	12.38%	63.26%	36.74%	34.81%	65.19%
GPT-5.2	low	394	13.47%	65.99%	34.01%	38.83%	61.17%

Table 7: Results on probe-underdetermined cases in the names-only Scientist QA condition. "Pct. of questions" uses 2,925 as the denominator. "Both predicted false" and "Both predicted true" describe the model's two probe predictions within this subset. These cases are common for weaker settings, especially DeepSeek-low and Claude-low, but rare for Gemini.

Model	Mode	n non-tie	Chooses more famous	p vs. 50%
Claude Sonnet 4.6	high	386	46.89%	0.242
Claude Sonnet 4.6	low	667	44.68%	0.007
DeepSeek V3.2 Reasoner	high	562	49.47%	0.833
DeepSeek V3.2 Chat	low	1133	46.07%	0.009
Gemini 3.1 Pro Preview	high	66	50.00%	1.000
Gemini 3.1 Pro Preview	low	84	46.43%	0.586
GPT-5.2	high	362	45.86%	0.127
GPT-5.2	low	394	46.70%	0.208

Table 8: Choice of the more famous candidate within probe-underdetermined, non-tie cases. "Chooses more famous" is the fraction of such cases in which the pairwise answer is the candidate with the higher fame score. The binomial test compares the observed rate to a 50% baseline.

Model	Mode	Hall. when wrong not more famous	Hall. when wrong more famous	Wrong more famous among hallucinations
Claude Sonnet 4.6	high	8.83%	4.57%	45.05%
Claude Sonnet 4.6	low	34.19%	17.40%	44.64%
DeepSeek V3.2 Reasoner	high	13.16%	8.92%	51.78%
DeepSeek V3.2 Chat	low	41.25%	34.69%	57.12%
Gemini 3.1 Pro Preview	high	4.24%	2.45%	47.83%
Gemini 3.1 Pro Preview	low	3.53%	1.84%	45.21%
GPT-5.2	high	13.52%	8.20%	49.00%
GPT-5.2	low	15.37%	9.48%	49.42%

Table 9: Fame-direction negative control for the names-only Scientist QA condition. The first two numeric columns condition on whether the wrong candidate is more famous by fame score. The final column reports, among hallucinated cases, the fraction in which the wrong candidate is more famous. Hallucination rates are consistently lower when the wrong candidate is more famous, indicating that the observed shortcut is not a simple more-famous-name prior.

Model	Mode	Mean conf. correct	Mean conf. hallucinated	Gap
Claude Sonnet 4.6	high	88.06	74.19	-13.87
Claude Sonnet 4.6	low	72.16	62.46	-9.70
DeepSeek V3.2 Reasoner	high	92.79	86.48	-6.31
DeepSeek V3.2 Chat	low	84.89	84.93	0.04
Gemini 3.1 Pro Preview	high	99.23	92.55	-6.68
Gemini 3.1 Pro Preview	low	97.38	76.23	-21.15
GPT-5.2	high	91.10	81.35	-9.76
GPT-5.2	low	90.81	83.55	-7.27

Table 10: Mean self-reported confidence for correct and hallucinated pairwise answers in the names-only Scientist QA condition. The gap is hallucinated confidence minus correct confidence. Confidence separates correct and incorrect answers for some models, but not reliably across model families.

Model	Errors / rate
Claude Sonnet 4.6	81 / 16.20%
DeepSeek-chat	182 / 36.40%
GPT-5.2	44 / 8.80%
Gemini 3.1 Pro Preview	18 / 3.60%

Table 11: Real-Life Constrained QA results over 500 questions covering 13 aspects of daily life. Entries report the number and percentage of incorrect shortcut selections.

Analysis family	n	Missed
Functional or diagnostic test	83	49 (59.0%)
Vehicle identifier verification	78	44 (56.4%)
Physical scan, photo, or measurement	69	33 (47.8%)
Rebate, warranty, or refund documentation	69	38 (55.1%)
Attached-label or container scan	66	8 (12.1%)
Vehicle positioning, handoff, or loading	42	10 (23.8%)
Original/copy document routing	29	15 (51.7%)
Safety or condition rule	29	7 (24.1%)
Communication-channel choice	20	6 (30.0%)
Object presence, repair, or adjustment	12	1 (8.3%)
Access-card or badge use	2	0 (0.0%)
Other physical dependency	1	0 (0.0%)
All questions	**500**	**211 (42.2%)**

Table 12: Breakdown of Real-Life Constrained QA by post-construction analysis family. "Missed" reports the number of questions answered incorrectly by at least one evaluated model, followed by the corresponding within-family percentage. Family sizes are uneven, so the rates are intended as descriptive coverage statistics rather than comparative estimates of category difficulty.

Constraint type	n	Missed
Document original/copy routing	98	53 (54.1%)
Requires connected or operating item	83	49 (59.0%)
Item must be present for capture	83	37 (44.6%)
Identifier or meter on item	78	44 (56.4%)
Label must be attached	66	8 (12.1%)
Rule or condition compliance	29	7 (24.1%)
Process requires item at station	28	6 (21.4%)
Communication channel required	20	6 (30.0%)
Other physical dependency	13	1 (7.7%)
Access token or badge required	2	0 (0.0%)
All questions	**500**	**211 (42.2%)**

Table 13: Breakdown of Real-Life Constrained QA by annotated constraint type. "Missed" reports the number of questions answered incorrectly by at least one evaluated model, followed by the corresponding within-type percentage. Constraint-type sizes are uneven, so the rates are intended as descriptive coverage statistics rather than comparative estimates of category difficulty.

I Potential risks

TrapQA is intended as a diagnostic benchmark, not as a training set or a broad certificate of hallucination robustness. Public release may enable overfitting or contaminate future model training/evaluation, so later results should be interpreted with this risk in mind. We are working with the community to expand REAL-LIFE CONSTRAINED QA and extend entity disambiguation beyond scientists to domains such as sports players and music composers; such extensions should be reported separately unless results are recomputed.

J Data Contains Personally Identifying Info Or Offensive Content

SCIENTISTQA uses public Wikipedia/Wikidata-linked scientist profiles, which makes the task verifiable but introduces coverage biases toward scientists with richer public or English-language records. Because the task distinguishes real scientists, names and public biographical facts are not anonymized. We release only public attributes needed for the diagnostic task and exclude private contact information, images, surveillance data, and other private personal data. REAL-LIFE CONSTRAINED QA is synthetic and filtered for ambiguity, plausibility, and inappropriate or offensive content.

K Proof for Section 3

K.1 Posterior decomposition under the latent key–task model

In this appendix, we make explicit the hierarchical posterior structure implicit in the latent key–task model. Recall that the pretraining prior over latent pairs factorizes as

$$\pi(k,t) = \pi^{(k)}(k)\,\pi^{(t)}(t \mid k).$$

Accordingly, for a prompt $\boldsymbol{z}$, we consider the hierarchical posterior decomposition

$$P(k,t \mid \boldsymbol{z}) = P(k \mid \boldsymbol{z})\,P(t \mid k,\boldsymbol{z}),$$

where

$$P(k \mid \boldsymbol{z}) = \frac{P(\boldsymbol{z} \mid k)\,\pi^{(k)}(k)}{\sum_{k'\in\mathcal{K}} P(\boldsymbol{z} \mid k')\,\pi^{(k)}(k')},$$

and

$$P(t \mid k,\boldsymbol{z}) = \frac{P(\boldsymbol{z} \mid k,t)\,\pi^{(t)}(t \mid k)}{\sum_{t'\in\mathcal{T}} P(\boldsymbol{z} \mid k,t')\,\pi^{(t)}(t' \mid k)}.$$

Therefore,

$$P(k,t \mid \boldsymbol{z}) = \frac{P(\boldsymbol{z} \mid k)\,\pi^{(k)}(k)}{\sum_{k'\in\mathcal{K}} P(\boldsymbol{z} \mid k')\,\pi^{(k)}(k')} \cdot \frac{P(\boldsymbol{z} \mid k,t)\,\pi^{(t)}(t \mid k)}{\sum_{t'\in\mathcal{T}} P(\boldsymbol{z} \mid k,t')\,\pi^{(t)}(t' \mid k)}.$$

K.2 Proof of Theorem 3.4

Proof K.1 *We factorize the joint posterior into key and task components:*

$$\frac{P(k_s,t_s \mid \boldsymbol{z})}{P(k^*,t^* \mid \boldsymbol{z})} = \frac{P(k_s \mid \boldsymbol{z})}{P(k^* \mid \boldsymbol{z})} \cdot \frac{P(t_s \mid \boldsymbol{z},k_s)}{P(t^* \mid \boldsymbol{z},k^*)}.$$

By Assumption 3.1(i), $P(k \in \{k^*,k_s\} \mid \boldsymbol{z}) \approx 1$, so for $k \in \{k^*,k_s\}$,

$$P(k \mid \boldsymbol{z}) \approx P(k \mid \boldsymbol{z},\, k \in \{k^*,k_s\}).$$

By Assumption 3.1(ii), $\boldsymbol{z}$ is independent of k within the candidate pair, hence

$$P(k \mid \boldsymbol{z},\, k \in \{k^*,k_s\}) = P(k \mid k \in \{k^*,k_s\}) = \frac{\pi^{(k)}(k)}{\pi^{(k)}(k^*) + \pi^{(k)}(k_s)}.$$

Taking the ratio at $k = k_s$ and $k = k^*$,

$$\frac{P(k_s \mid \boldsymbol{z})}{P(k^* \mid \boldsymbol{z})} \approx \frac{\pi^{(k)}(k_s)}{\pi^{(k)}(k^*)}.$$

By Assumption 3.2(i), conditional on the activated key k, the task posterior concentrates on $\{t^*,t_s\}$, so for $t \in \{t^*,t_s\}$ and $k \in \{k^*,k_s\}$,

$$P(t \mid \boldsymbol{z},k) \approx P(t \mid \boldsymbol{z},k,\, t \in \{t^*,t_s\}).$$

By Assumption 3.2(ii), $\boldsymbol{z}$ is independent of t given k within the candidate task pair, so

$$P(t \mid \boldsymbol{z},k,\, t \in \{t^*,t_s\}) = P(t \mid k,\, t \in \{t^*,t_s\}) = \frac{\pi^{(t)}(t \mid k)}{\pi^{(t)}(t^* \mid k) + \pi^{(t)}(t_s \mid k)}.$$

Evaluating at $(t,k) = (t_s,k_s)$ and (t^*,k^*) and taking the ratio,

$$\frac{P(t_s \mid \boldsymbol{z},k_s)}{P(t^* \mid \boldsymbol{z},k^*)} \approx \frac{\pi^{(t)}(t_s \mid k_s)}{\pi^{(t)}(t^* \mid k^*)} \cdot \frac{\pi^{(t)}(t^* \mid k^*) + \pi^{(t)}(t_s \mid k^*)}{\pi^{(t)}(t^* \mid k_s) + \pi^{(t)}(t_s \mid k_s)}.$$

The second factor is a ratio of normalization constants over the candidate task pair, which we absorb into the $\approx$ symbol as it is bounded and does not depend on $\boldsymbol{z}$:

$$\frac{P(k_s, t_s \mid \boldsymbol{z})}{P(k^*, t^* \mid \boldsymbol{z})} \approx \frac{\pi^{(k)}(k_s)}{\pi^{(k)}(k^*)} \cdot \frac{\pi^{(t)}(t_s \mid k_s)}{\pi^{(t)}(t^* \mid k^*)}.$$

By the law of total probability over key–task pairs, the marginal output probability decomposes as

$$P(y \mid \boldsymbol{z}) = \sum_{k,t} P(y \mid \boldsymbol{z}; k, t)\, P(k, t \mid \boldsymbol{z}).$$

By Assumption 3.3, only the shortcut path contributes non-negligibly to y_s and only the correct path contributes non-negligibly to y^*:

$$P(y_s \mid \boldsymbol{z}; k^*, t^*) \ll 1, \qquad P(y^* \mid \boldsymbol{z}; k_s, t_s) \ll 1.$$

Hence,

$$P(y_s \mid \boldsymbol{z}) \approx P(y_s \mid \boldsymbol{z}; k_s, t_s)\, P(k_s, t_s \mid \boldsymbol{z}),$$

$$P(y^* \mid \boldsymbol{z}) \approx P(y^* \mid \boldsymbol{z}; k^*, t^*)\, P(k^*, t^* \mid \boldsymbol{z}).$$

Taking the ratio,

$$\frac{P(y_s \mid \boldsymbol{z})}{P(y^* \mid \boldsymbol{z})} \approx \frac{P(k_s, t_s \mid \boldsymbol{z})}{P(k^*, t^* \mid \boldsymbol{z})} \cdot \frac{P(y_s \mid \boldsymbol{z}; k_s, t_s)}{P(y^* \mid \boldsymbol{z}; k^*, t^*)}.$$

$$\frac{P(y_s \mid \boldsymbol{z})}{P(y^* \mid \boldsymbol{z})} \gtrsim \frac{\pi^{(k)}(k_s)}{\pi^{(k)}(k^*)} \cdot \frac{\pi^{(t)}(t_s \mid k_s)}{\pi^{(t)}(t^* \mid k^*)} \cdot \frac{P(y_s \mid \boldsymbol{z}; k_s, t_s)}{P(y^* \mid \boldsymbol{z}; k^*, t^*)}.$$

The second inequality in the theorem statement follows from the shortcut-frequency dominance condition (both pretraining-prior ratios are ≥ 1 by the definition of the shortcut path) together with Assumption 3.3, which gives $P(y_s \mid \boldsymbol{z}; k_s, t_s) \geq P(y^* \mid \boldsymbol{z}; k^*, t^*)$.

Multi-path extension. *Let $\mathcal{F}(z)$ and $\mathcal{S}(z)$ denote the sets of activated constraint-sensitive and shortcut paths, respectively. Write the posterior as*

$$P(k, t \mid z) \propto \pi(k, t) w_{k,t}(z),$$

where $w_{k,t}(z)$ represents prompt-dependent contextual reweighting. Suppose that, for every activated path,

$$B^{-1} \leq w_{k,t}(z) \leq B, \qquad B \geq 1.$$

Then

$$\frac{P(\mathcal{S}(z) \mid z)}{P(\mathcal{F}(z) \mid z)} \geq \frac{1}{B^2} \frac{\sum_{(k,t)\in\mathcal{S}(z)} \pi(k, t)}{\sum_{(k,t)\in\mathcal{F}(z)} \pi(k, t)}.$$

Indeed, the common posterior normalization cancels, while the shortcut numerator is lower bounded by $B^{-1} \sum_{(k,t)\in\mathcal{S}(z)} \pi(k, t)$ and the constraint-sensitive denominator is upper bounded by $B \sum_{(k,t)\in\mathcal{F}(z)} \pi(k, t)$. Thus, an aggregate shortcut-prior ratio greater than B^2, together with an aggregate output-separation condition analogous to Assumption 3.3, is sufficient for shortcut dominance. This extension remains a sufficient-condition analysis rather than a complete model of all paths activated during LLM inference.

K.3 Proof of Theorem 3.6

Proof K.2 *By the latent key–task decomposition, the model prediction can be written as*

$$P(y \mid \boldsymbol{z}) = \sum_{k,t} P(k, t \mid \boldsymbol{z}) P(y \mid \boldsymbol{z}; k, t).$$

Restricting to the two relevant paths gives the contributions

$$P(y^* \mid \boldsymbol{z}) \geq q^* P(y^* \mid \boldsymbol{z}; k^*, t^*) + q_s P(y^* \mid \boldsymbol{z}; k_s, t_s),$$

and

$$P(y_s \mid \boldsymbol{z}) \geq q_s P(y_s \mid \boldsymbol{z}; k_s, t_s) + q^* P(y_s \mid \boldsymbol{z}; k^*, t^*).$$

Under Assumption 3.3,

$$P(y^* \mid \boldsymbol{z}; k_s, t_s) = 0, \qquad P(y_s \mid \boldsymbol{z}; k^*, t^*) = 0.$$

Thus, the two dominant contributions reduce to

$$P(y^* \mid \boldsymbol{z}) \approx q^* P(y^* \mid \boldsymbol{z}; k^*, t^*),$$

and

$$P(y_s \mid \boldsymbol{z}) \approx q_s P(y_s \mid \boldsymbol{z}; k_s, t_s).$$

Since

$$q_s > q^*$$

and

$$P(y_s \mid \boldsymbol{z}; k_s, t_s) \geq P(y^* \mid \boldsymbol{z}; k^*, t^*),$$

we obtain

$$P(y_s \mid \boldsymbol{z}) > P(y^* \mid \boldsymbol{z}).$$

Therefore,

$$\gamma(\boldsymbol{z}) := P(y_s \mid \boldsymbol{z}) - P(y^* \mid \boldsymbol{z}) > 0.$$

It remains to lower bound the total variation distance. By definition,

$$\ell(\boldsymbol{z}) = \frac{1}{2} \sum_{y} \left| P(y \mid \boldsymbol{z}) - P_\star(y \mid \boldsymbol{z}) \right|.$$

Keeping only the two coordinates y_s and y^*, we have

$$\begin{aligned} \ell(\boldsymbol{z}) \geq & \frac{1}{2} \left| P(y_s \mid \boldsymbol{z}) - P_\star(y_s \mid \boldsymbol{z}) \right| \\ & + \frac{1}{2} \left| P(y^* \mid \boldsymbol{z}) - P_\star(y^* \mid \boldsymbol{z}) \right|. \end{aligned}$$

Since the model prefers the shortcut answer,

$$P(y_s \mid \boldsymbol{z}) - P(y^* \mid \boldsymbol{z}) = \gamma(\boldsymbol{z}) > 0,$$

whereas the target distribution prefers the correct answer,

$$P_\star(y^* \mid \boldsymbol{z}) - P_\star(y_s \mid \boldsymbol{z}) = \gamma_\star(\boldsymbol{z}) > 0.$$

Adding these two inequalities gives

$$\begin{aligned} \gamma(\boldsymbol{z}) + \gamma_\star(\boldsymbol{z}) = & \left[P(y_s \mid \boldsymbol{z}) - P_\star(y_s \mid \boldsymbol{z}) \right] \\ & + \left[P_\star(y^* \mid \boldsymbol{z}) - P(y^* \mid \boldsymbol{z}) \right]. \end{aligned}$$

Therefore, the two-coordinate contribution to the total variation distance is at least

$$\frac{\gamma(\boldsymbol{z}) + \gamma_\star(\boldsymbol{z})}{2}.$$

Hence,

$$\ell(\boldsymbol{z}) \geq \frac{\gamma(\boldsymbol{z}) + \gamma_\star(\boldsymbol{z})}{2}.$$

This completes the proof.