The DECK Taxonomy: A Universal Blind Spot in LLM Uncertainty Quantification

Anonymous Authors[1]

Abstract

Uncertainty Quantification (UQ) for LLMs relies on three scorer families—black-box consistency, white-box token-probability, and LLM-as-a-Judge—whose ensembles outperform any single scorer, but *why* remains unexplained. In practice, most teams default to a Judge alone, with no principled basis for knowing when to add the costlier black-box or white-box families. We introduce the **DECK taxonomy**: a 2×2 partition of hallucinations along inter-sample consistency and token-level confidence into four regimes (**D**rift, **E**ntrenched, **C**onfabulation, **K**notted), each detectable by a different combination of scorer families. Disagreement-decomposition analysis across three models and four datasets validates the taxonomy and yields a profile-driven ensemble-weighting recipe. We operationalize it as a *pre-deployment diagnostic*: pairwise complementarity on a small validation set reveals whether a Judge-only scorer leaves blind spots, while periodic DECK-profile re-estimation provides label-free drift monitoring in production. We further identify a *universal blind spot*—knowledge-gap questions on which all three families collapse to near-random AUROC—marking the operational boundary of output-level UQ and motivating abstention-aware routing.

1. Introduction

A practitioner deploying an LLM in a high-stakes setting faces a concrete question: *which uncertainty scorers should I run, and how should I combine them?* The literature offers three distinct answers. **Black-box** (BB) scorers read confidence from inter-sample agreement, on the hypothesis that a confident model produces similar answers across runs (Manakul et al., 2023; Kuhn et al., 2023). **White-box** (WB) scorers read confidence from token-level log-probabilities, requiring no extra generations (Kadavath et al., 2022; Malinin & Gales, 2021). **LLM-as-a-Judge** (J) scorers delegate evaluation to a separate model (Zheng et al., 2023). Each family rests on a different assumption about where uncertainty manifests, and each works at a different cost. The UQLM toolkit (Bouchard et al., 2025) packages all three and shows empirically that ensembles outperform individual scorers, but it does not explain *why* the families complement each other. Without that explanation, practitioners must tune ensemble weights heuristically, with no guidance for adaptation across domains or model scales.

In practice, most teams default to a single LLM-as-a-Judge scorer: it requires no log-probability access, one extra generation, and works with any API—making it the most *accessible* option. WB scorers are cheaper (they reuse the same generation) but need logit access; BB scorers are the most expensive, requiring five or more additional samples. The practical question is therefore: *is a Judge alone sufficient for my dataset, or do I need the additional cost of BB or WB scorers?* We show that a lightweight, offline complementarity assessment answers this before deployment, and that ongoing drift monitoring verifies the answer remains valid in production.

We address this gap with three contributions. First, we introduce the **DECK taxonomy** (Table 1): a 2×2 partition of hallucinations along inter-sample consistency and token-level confidence, yielding four behavioral regimes —**D**rift, **E**ntrenched, **C**onfabulation, **K**notted— each predicted to be detectable by a different combination of scorer families. The taxonomy explains complementarity mechanistically: families disagree precisely where they specialize, and disagreement therefore carries the same signal as detection. Second, we conduct a **disagreement-decomposition analysis** across three models and four datasets, validate the taxonomy's predictions, and convert it into a profile-driven ensemble-weighting recipe (§6). Third, we operationalize the taxonomy as a **pre-deployment diagnostic**: pairwise complementarity on a small validation set reveals whether a Judge-only scorer leaves blind spots,

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

and periodic re-estimation of the DECK profile provides a label-free drift monitor (§5.5, §6.4). Along the way, we identify a regime where every scorer in the three-family paradigm fails simultaneously — the *universal blind spot* on knowledge-gap questions (§5.4) — which marks the operational boundary of output-level UQ and motivates abstention-aware routing as the appropriate response outside that boundary.

2. Background and Related Work

2.1. Three Families of UQ Scorers

Output-level UQ methods for LLMs fall into three families distinguished by the access they require (Shorinwa et al., 2025; Kang et al., 2025). *White-box* methods read confidence from token log-probabilities; representative scorers include `P(True)` (Kadavath et al., 2022), sequence-probability measures (Malinin & Gales, 2021), and calibration-aware variants (Jiang et al., 2021; Xiong et al., 2024; Duan et al., 2024; Tian et al., 2023). They require logit access and are cheap. *Black-box* methods read confidence from inter-sample agreement: SelfCheckGPT (Manakul et al., 2023) uses NLI- and embedding-based consistency, while semantic entropy (Kuhn et al., 2023; Farquhar et al., 2024) clusters responses into meaning classes and computes entropy over the resulting distribution; subsequent refinements (Lin et al., 2024; Qiu & Miikkulainen, 2024) improve robustness to lexical variation. They are model-agnostic but cost multiple generations per query. *LLM-as-a-Judge* methods delegate evaluation to a separate model (Zheng et al., 2023; Liu et al., 2023); they detect failures invisible to either probabilistic signal but inherit known judge biases (Chiang & Lee, 2023) and risk shared-pretraining errors when generator and judge are related (Feng et al., 2024). UQLM (Bouchard et al., 2025) aggregates all three families with both fixed-weight (BS Detector) and tunable (UQEnsemble) aggregators, and shows ensembles outperform individual scorers without explaining why. A fourth, internal-state family — UQ heads (Shelmanov et al., 2025), hidden-state probes (Slobodkin et al., 2023), and information-theoretic estimators (Abbasi Yadkori et al., 2024) — is largely orthogonal to the three output-signal families and is not evaluated here.

2.2. Aleatoric, Epistemic, and Confabulation

The classical aleatoric/epistemic decomposition (Hüllermeier & Waegeman, 2021) remains useful at LLM scale: epistemic uncertainty is the component most tightly linked to hallucination (Abbasi Yadkori et al., 2024; Shorinwa et al., 2025). Farquhar et al. (2024) focus on *confabulations*—fluent, wrong, arbitrary outputs that vary across seeds—and validate semantic entropy specifically against them. We adopt that term unchanged for our low-consistency, low-confidence cell; the categories Farquhar et al. set aside (misconceptions, lies (Evans et al., 2021), reasoning failures) share a high-consistency, high-confidence signature and constitute our *Entrenched* cell. Our contribution is to make this implicit partition explicit as a full 2×2 taxonomy and use it to explain the complementarity patterns reported but unanalyzed by UQLM.

2.3. Hallucination Taxonomies, Abstention, and Positioning

Type-level taxonomies classify hallucinations by what is wrong with the output (Ji et al., 2023; Huang et al., 2025; Wang et al., 2024), with fine-grained atomic-claim evaluation provided by FActScore (Min et al., 2023). These are complementary to our work: they classify by error *content*, we classify by detectability *signature*. A memorized misconception and a reward-hacked lie share our Entrenched signature even though they are different content categories. A third thread targets *abstention*: Self-Aware (Yin et al., 2023) introduced unanswerable-question benchmarking; Feng et al. (2024) show abstention improves under multi-LLM collaboration, directly motivating our judge-diversity design; Slobodkin et al. (2023) surface answerability signal in hidden states; Kalai et al. (2025) argue current training incentives reward guessing over abstention. The universal blind spot we report in §5.4 corroborates this picture: output-level UQ cannot detect hallucinations on systematically unanswerable inputs because the necessary inter-sample variance is absent.

Relative to UQLM (Bouchard et al., 2025) we explain *why* ensembles work; relative to Farquhar et al. (2024) we cover all four DECK cells and identify the universal blind spot. We treat scorer *disagreement* as the primary signal rather than aggregated metrics (Xiong et al., 2024).

3. A Taxonomy of LLM Hallucination Types

3.1. Two-Axis Framework

We characterize LLM hallucinations along two observationally independent axes: **inter-sample consistency** (does the model produce the same incorrect answer across multiple independent samples?) and **token-level confidence** (does the model assign high probability to the tokens of its generated response?). This yields four hallucination types, summarized in Table 1, each implying a distinct detection strategy.

Table 1. The **DECK taxonomy** of LLM hallucinations. Each cell is a behavioral regime defined by two observable axes: inter-sample consistency (does the model produce the same wrong answer across independent samples?) and token-level confidence (does the model assign high probability to the tokens it generates?). The four regimes map to the letters **D**rift, **E**ntrenched, **C**onfabulation, and **K**notted. The final column gives the scorer family (or families) sensitive to each regime; see §3 for the derivation.

	Low Consistency	High Consistency	Detectable By
High Token Confidence	**D — Drift** *Different confident wrong answer each sample.* Multiple plausible-but-wrong answers are drawn confidently from the output distribution; each draw is internally coherent, but the answer drifts.	**E — Entrenched** *Same confident wrong answer every sample.* Model has locked onto a memorized misconception or shared-pretraining error and reproduces it without variance.	**Drift:** Black-box and Judge. **Entrenched:** Judge (independent provider required); blind to both BB and WB.
Low Token Confidence	**C — Confabulation** *Different low-probability wrong answer each sample.* Classic uncertainty in the sense of Farquhar et al. (2024): the model genuinely does not know, and different samples produce different wrong answers with low confidence.	**K — Knotted** *Same low-probability wrong answer every sample.* Model is consistently unsure: it settles on the same hedged answer each time but assigns low token probability, as if tangled on a response it cannot commit to or escape.	**Confabulation:** All three families. **Knotted:** White-box and Judge.

Note. Regimes are defined by observable output behavior, not by an internal mechanism claim. Cell membership depends on the scorer thresholds used to discretize the two axes; §3 describes the operationalization. The **universal blind spot** reported in §5.4 is the degenerate case in which a dataset collapses entirely into the Entrenched cell because the model responds with uniform, confident fabrication on genuinely unanswerable inputs — a regime no UQ scorer family can detect.

3.2. Why Each Family Has a Blind Spot

Black-box scorers measure consistency: if a model produces the same wrong answer every sample, the scorer sees high agreement and assigns high confidence—*Entrenched and Knotted hallucinations evade detection.* **White-box scorers** measure token probability: a model may assign high probability to a fluent but factually incorrect claim—*Entrenched and Drift hallucinations evade detection.* **LLM judges** evaluate factual content and in principle detect any type when the judge's knowledge is accurate. However, they are susceptible to the same systematic misconceptions as the generator when both share pretraining data—*shared Entrenched hallucinations are the failure mode.* More strikingly, for knowledge-gap hallucinations (questions genuinely unanswerable for any LLM), all three families fail simultaneously: BB scorers see consistent confident answers; WB scorers see high token probabilities; and judges share the same knowledge gap. This represents a category—*universal blind spots*—that current UQ methods cannot address. This asymmetric coverage formalizes why combining all three families is beneficial within their shared operating range, and why ensemble weights should reflect the hallucination profile of the domain.

4. Experimental Setup

4.1. Datasets

TriviaQA (Joshi et al., 2017) contains factual trivia where larger models are prone to memorized systematic misconceptions—expected to exercise the Entrenched cell. **HaluEval** (Li et al., 2023) provides curated hallucinated answer pairs with binary labels, skewing toward Entrenched and Knotted types. **SelfAware** (Yin et al., 2023); we use a 500-question subset of its unanswerable questions to probe model abstention behavior, operationalizing the Entrenched cell for knowledge-gap hallucinations. **PopQA** (Mallen et al., 2023) contains entity-centric questions stratified by Wikipedia page-view popularity; the popular- and rare-entity slices target the Entrenched and Confabulation cells respectively. Response grading procedures for all four datasets are detailed in Appendix B.

4.2. Models

We evaluate three models spanning a range of scale and provider: **Llama-3-8B** (smaller, open-weights, expected higher Confabulation and Drift rates under the DECK taxonomy), **GPT-4o** (larger, closed-weights, expected higher Entrenched hallucination rate), and **Gemini-2.5-Flash** (mid-scale, closed-weights, providing a third provider to test generality). All models are evaluated on all four datasets: TriviaQA, HaluEval, SelfAware, and PopQA.

4.3. Scorer Selection

We compute confidence scores using 6 black-box, 5 white-box, and 4 judge configurations, giving 15 scorers per model-dataset pair (full AUROC breakdown in Appendix C, Table 6). For the disagreement and complementarity analysis, each family is represented by its single highest-AUROC method on the target split, avoiding hand-selection bias. A full reference table is provided in Appendix A.

Black-box (BlackBoxUQ): Semantic Negentropy, Non-Contradiction, Entailment, Cosine Similarity, Exact Match, Semantic Sets Confidence.

White-box (WhiteBoxUQ): Sequence Probability, Min Token Probability, Mean Token Negentropy, Probability Margin, P(True).

LLM-as-a-Judge (LLMPanel): Four configurations per generator: **self-judge** (same model, lower bound on judge utility), **cross-LLM judge** (other generator, reduces shared bias), **Gemini-2.5-Flash judge** (mid-scale independent provider, additional cross-family signal), and **Claude Sonnet 4.6 judge** (independent provider, upper bound on judge independence).

All judge configurations use the Likert scoring template; multi-generation scorers use five additional sampled responses at temperature 1.0; and k=15 for scorers that use top-k token probabilities.

4.4. Evaluation Metrics

Our primary metric is the **pairwise complementarity score** $C(A, B)$—the fraction of hallucinated responses that scorer family A misses but family B correctly flags, averaged symmetrically:

$$C(A, B) = \frac{1}{2}\big[P(B \text{ flags} \mid A \text{ misses}) + P(A \text{ flags} \mid B \text{ misses})\big] \quad (1)$$

A high $C(A, B)$ indicates the two families catch structurally different hallucinations. We additionally report AUROC per scorer across all dataset–model combinations.

We further define the **hallucination-restricted complementarity score** $C_H^{(p)}(A, B)$, which restricts the complementarity computation to samples on which the model hallucinated (response_correct = False). Binarisation thresholds are computed at the p-th empirical percentile of each scorer's score distribution over all N samples, then the metric is evaluated exclusively on the hallucinated subset:

$$C_H^{(p)}(A, B) = \frac{|\{i : \neg\text{correct}_i,\ \hat{y}_i^{(A,p)} \neq \hat{y}_i^{(B,p)}\}|}{|\{i : \neg\text{correct}_i\}|} \quad (2)$$

where $\hat{y}_i^{(A,p)} = \mathbf{1}[s_i^{(A)} < q_p(\mathbf{s}^{(A)})]$ is scorer A's binary hallucination prediction at percentile p, and $q_p(\mathbf{s}^{(A)})$ denotes the p-th sample percentile over all N observations. We report $C_H^{(25)}$, $C_H^{(50)}$, and $C_H^{(75)}$ to assess threshold sensitivity; $C_H^{(50)}$ uses the same median cut-point as $C(A, B)$ but evaluates complementarity within the hallucinated subset only.

5. Results

5.1. Individual Scorer Performance

Figure 1 and Table 6 (Appendix C) summarize per-method AUROC across all twelve dataset–model combinations. Three patterns stand out. First, **no scorer family Pareto-dominates**: on Llama-3-8B TriviaQA the Judge leads (0.844), while on GPT-4o TriviaQA WB leads (P(True) 0.675) and on PopQA the BB Cosine Similarity and WB Mean Token Negentropy both reach 0.77–0.80, with Judges just below. Second, **absolute AUROC compresses with model scale**: families that span 0.79–0.84 on Llama-3-8B TriviaQA cluster at 0.65–0.68 on GPT-4o and 0.63–0.67 on Gemini-2.5-flash, consistent with the Entrenched-shift predicted by the DECK taxonomy. Third, **PopQA is the highest-AUROC regime** and SelfAware the lowest; on SelfAware all methods collapse to near-random across all three models (detailed in §5.4). Since no family dominates, ensemble gains depend on structural disagreement, not simply on selecting the highest-AUROC scorer—motivating §5.2.

5.2. Disagreement Analysis and Taxonomy Validation

We conduct the disagreement analysis across all twelve combinations of model (Llama-3-8B, GPT-4o, Gemini-2.5-flash) and dataset (TriviaQA, HaluEval, SelfAware, PopQA). For each condition the best-AUROC scorer per family is selected from all available scorers; Table 7 shows the top scorer. Each sample is then assigned to one of the four taxonomy quadrants by a median split on the BB (consistency) and WB (confidence) axes. Throughout this section we report hallucination-restricted complementarity at the median threshold, $C_H^{(50)}$; results for the 25th- and 75th-percentile thresholds are consistent with the patterns described here and are provided in full in Appendix D (Table 8). A step-by-step derivation of the complementarity score computation, including a worked numerical example, is provided in Appendix D.

Evidential hierarchy: what validates the taxonomy. The DECK taxonomy's mechanistic claims are supported at two levels of evidence. *Primary evidence* comes from (i) **judge-involving pairs** (BB–Judge and WB–Judge), because the Judge score is a *third axis* not used to define the four quadrants, making its disagreement distribution a genuinely independent test of the predicted coverage patterns; and (ii) **external-signal validation** (§5.3), which checks taxonomy predictions against the entity-popularity stratification in PopQA without using any scorer output. *Secondary evidence* comes from the **BB–WB geometric check**: because the quadrant cells are defined by the same BB and WB scores used to represent those families, BB–WB disagreements are constrained by construction to land on the anti-diagonal (Knotted and Drift). This pattern is therefore a *structural inevitability*, not a discovered regularity—it confirms internal consistency but cannot validate the taxonomy against an independent signal. The quantitative results below are presented in this order:

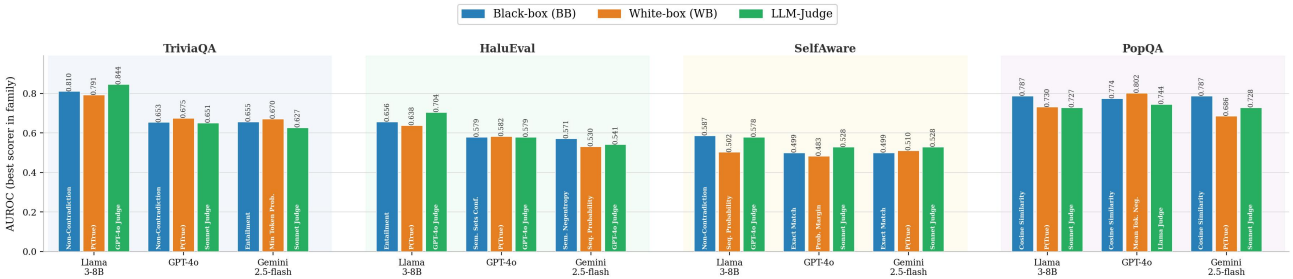

Figure 1. Top scorer per family with best AUROC across all twelve dataset–model combinations (four datasets × three models). Methods are grouped by scorer family (blue = BB, orange = WB, green = Judge). Full numerical values and bar plots are in Appendix C, Table 6.

judge-involving pairs first (primary), BB–WB check second (secondary).

Table 2. Hallucination-restricted complementarity $C_H^{(50)}$ for all twelve conditions. SelfAware rows (†): degenerate regime; high values reflect independent failure. **Bold** = larger value between BB,J and WB,J (BB,WB unbolded). Table 8 shows full scores.

Dataset	Model	BB,WB	BB,J	WB,J
TriviaQA	Llama-3-8B	0.195	**0.185**	**0.185**
	GPT-4o	0.296	**0.327**	0.316
	Gemini-2.5	0.202	0.405	**0.464**
HaluEval	Llama-3-8B	0.306	**0.282**	0.257
	GPT-4o	0.339	0.239	**0.282**
	Gemini-2.5	0.423	0.426	**0.459**
Self. †	Llama-3-8B	0.427	0.399	**0.433**
	GPT-4o	0.494	0.196	**0.528**
	Gemini-2.5	0.507	0.109	**0.525**
PopQA	Llama-3-8B	0.264	0.269	**0.291**
	GPT-4o	0.191	0.414	**0.421**
	Gemini-2.5	0.264	0.275	**0.352**

Judge-involving pairs (primary validation). Because the Judge score is not used to define the four quadrant cells, its disagreement distribution constitutes the primary test of the DECK taxonomy. On TriviaQA and HaluEval, judge-involving disagreements spread across all four quadrants, as the taxonomy predicts. On Llama-3-8B TriviaQA, BB–Judge disagreements concentrate in Knotted (37%) and Confabulation (32%) (n=38, 18.5% of N_H)—Knotted reflects BB's blindness to high-consistency errors that the Judge catches—while WB–Judge shifts toward Drift (37%) and Confabulation (32%) (n=38), where WB's high-confidence blind spot is covered by the Judge. On GPT-4o HaluEval, WB–Judge disagreements concentrate in Knotted (68%, n=99, 28.2% of N_H), matching the predicted WB blindness to high-consistency errors; similar four-quadrant coverage is seen on Llama-3-8B HaluEval and GPT-4o TriviaQA (full breakdown in Figure 2 and Figure 4).

On GPT-4o and Gemini-2.5-flash SelfAware, the degenerate two-cell regime constrains all pairs to Entrenched and Knotted; BB–Judge disagreements nearly vanish (GPT-4o: 19.6% of N_H; Gemini-2.5: 10.9%) while WB–Judge dominates (GPT-4o: 52.8%; Gemini-2.5: 52.5%), indicating BB and judge share correlated near-random errors while WB diverges independently.

The complementarity scores in Table 2 reinforce the pattern. Judge-involving pairs ($C_H^{(50)}(\mathrm{BB,J})$ and $C_H^{(50)}(\mathrm{WB,J})$) are the most informative: they vary substantially across conditions—from 0.109 (Gemini SelfAware BB–J, degenerate) to 0.528 (GPT-4o SelfAware WB–J, degenerate); across non-degenerate conditions the range is 0.185–0.464 (Gemini TriviaQA WB–J)—and track the predicted taxonomy profile. Full scores are in Table 8 (Appendix D).

BB vs. WB (secondary geometric check). As noted above, the BB–WB anti-diagonal pattern is a structural consequence of how the quadrant cells are defined rather than an independent validation signal. We report it for completeness as a consistency check. Across TriviaQA and HaluEval, BB–WB disagreements concentrate entirely on the anti-diagonal (Knotted and Drift), with a near-equal split on three of four conditions and 87% Knotted on GPT-4o HaluEval (n=119)—confirming that the median-split operationalization behaves as designed.

On Llama-3-8B SelfAware the anti-diagonal persists (49/51 Knotted–Drift, n=198). On GPT-4o SelfAware it collapses: all n=224 disagreements fall in Knotted (zero Drift), because GPT-4o's uniform confident fabrications place every sample above the BB median, an artefact of the degenerate regime. $C_H^{(50)}(\mathrm{BB,WB})$ ranges 0.195–0.339 across these four conditions, leading in three of four (BB–Judge leads at 0.327 on GPT-4o TriviaQA); for Gemini-2.5 judge-involving pairs are comparable or higher ($C_H^{(50)}(\mathrm{BB,WB})$ up to 0.423 on HaluEval, $C_H^{(50)}(\mathrm{WB,J})$ up to 0.459). On GPT-4o SelfAware, $C_H^{(50)}(\mathrm{BB,WB})$ = 0.494 inverts relative to

$C_H^{(50)}(\mathrm{WB}, \mathrm{J}) = 0.528$. These SelfAware values should be interpreted with caution: with all AUROCs near chance, high C reflects independent failure rather than complementary detection.

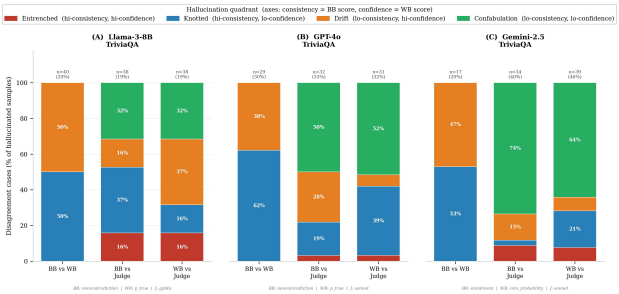

Figure 2. Pairwise scorer-family disagreement on *hallucinated* samples for TriviaQA across three models. Each panel shows three bars (BB vs. WB, BB vs. Judge, WB vs. Judge); bar segments are colored by DECK quadrant, defined by a median split on full-dataset. Figure 4 shows the similar for other datasets.

5.3. External-Signal Validation

A second, independent test of the taxonomy uses external labels that identify the *mechanism* of a hallucination, then asks whether samples with each mechanism land in the DECK cell that mechanism predicts. Unlike §5.2, this test does not depend on any UQ scorer beyond the (BB, WB) median split that defines cell membership; it asks whether that median split is mechanistically informative.

Predictions. Knowledge-gap inputs (SelfAware *unanswerable*) should land in **Entrenched**, since the model's response distribution collapses to confident fabrication in the absence of variance. Adversarially perturbed inputs (HaluEval) should concentrate in **Knotted** or **Entrenched** (high consistency by construction). Popular-entity inputs (PopQA *popular_entity*) should concentrate in **Entrenched** via memorized misconceptions; rare-entity inputs (PopQA *rare_entity*) should concentrate in **Confabulation**.

Method. For each condition we assign every sample to a DECK cell using the within-condition median split, then test whether the external-label distribution differs significantly from a pooled naturalistic baseline; full protocol and the placeholder results table are in Appendix F.

Results. The PopQA entity-popularity predictions are the most clearly confirmed: popular-entity samples concentrate in Entrenched at 44.9–48.9% across all three models (vs. 25% uniform null, $p < 0.001$), and rare-entity samples concentrate in Confabulation at 54.5–69.4% for Llama-3-8B and GPT-4o ($p < 0.001$); the Gemini-2.5-Flash rare-entity contrast is directionally consistent (56.4% in Confabulation) but does not reach significance ($p = 0.31$), likely because Gemini's natural PopQA stratum is itself Confabulation-heavy. Unanswerable (SelfAware) samples land in Entrenched at 28.4% for Llama-3-8B—just above the 25% null but without a clear concentration—while GPT-4o and Gemini-2.5-Flash collapse into the degenerate two-cell regime (GPT-4o: 50.0% Entrenched, 50.0% Knotted; Gemini-2.5-Flash: 50.2% Entrenched, 49.8% Knotted; $p < 0.001$) consistent with the degenerate SelfAware collapse documented in §5.2; in both cases the contrast is significant but driven by an overall distribution shift rather than targeted E-concentration. Adversarial (HaluEval) samples concentrate in E+K at 74.2% for GPT-4o ($p < 0.001$), consistent with the high-consistency prediction, but remain more diffuse for Llama-3-8B (E+K = 50.0%) and Gemini-2.5-Flash (E+K = 50.8%), where model-scale effects and the Gemini TriviaQA baseline distribution reduce the discriminative signal. Together, the PopQA results provide the clearest mechanistic evidence that the median-split DECK cells capture genuine hallucination structure independent of scorer geometry; the SelfAware and HaluEval patterns are broadly consistent with the taxonomy but are modulated by the degenerate and model-scale effects documented throughout the paper (full protocol and table in Appendix F). Failure of these predictions would be evidence the median-split operationalization is descriptive geometry only; its mechanistic interpretation would then require absolute, calibrated thresholds (Appendix E) before being defensible.

5.4. Universal Failure on Knowledge-Gap Hallucinations (SelfAware)

All three models fabricate confident responses to genuinely unanswerable questions at high rates: GPT-4o (90.6%), Llama-3-8B (92.8%), and Gemini-2.5-flash (88.0%). Across all three, every scorer family collapses toward chance on SelfAware—the full breakdown is in Table 6.

The severity scales with model capability. **GPT-4o and Gemini-2.5-flash** exhibit near-total collapse: 13 of 15 scorers fall below 0.5 for GPT-4o, with only two cross-model judges marginally above chance (Sonnet 0.528, Gemini 0.513). P(True)—which reaches 0.675 on GPT-4o TriviaQA—inverts to 0.331 here, the single worst score in the table, illustrating how the self-evaluation signal reverses entirely under systematic confident fabrication. **Llama-3-8B** is degraded but less extreme: 9 of 15 scorers fall below 0.5, and BB consistency (Non-Contradiction 0.587) retains a weak signal, consistent with a smaller model that does not fully suppress inter-sample response variance. $C_H^{(50)}(\mathrm{BB}, \mathrm{Judge})$ collapses to 0.109 for Gemini—the lowest value in Table 2—while $C_H^{(50)}(\mathrm{WB}, \mathrm{Judge})$ reaches 0.525, driven by independent failure rather than complementary detection.

This pattern has a clear mechanistic explanation. The SelfAware failure mode is **Entrenched** in the high-confidence, high-consistency quadrant: both models produce the same wrong type of answer with high token probability on every sample. BB scorers measure inter-sample agreement and see high agreement on these uniform fabrications—exactly wrong. WB scorers measure token-level log-probabilities and see high-probability fluent but factually incorrect answers—exactly wrong. Judge scorers share the same knowledge gap as the generator. All three families fail for the same fundamental reason: UQ methods require variance in model behaviour to function, and a model that is consistently and confidently wrong provides no such variance.

We term this the **universal blind spot**: a regime where no current detector provides reliable signal, and where the appropriate response is abstention-by-design rather than post-hoc scoring. Full AUROC scores are in Appendix C, Table 6 (SelfAware columns) and Figure 3.

5.5. Practical Implications

Our results reframe ensemble design as a structured decision and address a concrete practitioner question: *is a Judge alone sufficient, or do I need additional scorer families?*

Pre-deployment diagnostic. Given a validation set of 200–500 prompts, $C_H^{(50)}$ between the Judge and each alternative family directly answers this question. Low values—as on Llama-3-8B TriviaQA (0.185 for both Judge-involving pairs, Table 2)—indicate the Judge already covers what BB and WB would catch; the added cost is not justified. High values—as on Gemini-2.5 TriviaQA ($C_H^{(50)}(\mathrm{WB}, \mathrm{J})$=0.464) or GPT-4o PopQA (0.421)—signal that the Judge misses a structurally distinct subset and adding a WB or BB scorer yields meaningful gain. The DECK taxonomy explains *why*: high Judge–WB complementarity indicates a Knotted-heavy domain; high Judge–BB complementarity indicates a Drift-heavy domain. After deployment, the DECK profile can be re-estimated on incoming prompts *without labels* to monitor for regime drift (§6.4).

6. Ensemble Design: From Heuristic Tuning to Structured Decision

The complementarity analysis of §5 shows the right ensemble weighting depends on the *DECK regime profile* of the target domain — the relative frequency of Drift, Entrenched, Confabulation, and Knotted errors. This section turns that observation into a concrete design procedure. How model scale modulates complementarity scores across datasets is analyzed separately in Appendix G.

6.1. The DECK Regime Profile of a Domain

Given a model M and a representative set of n prompts $\mathcal{P}$ drawn from the deployment domain, the *DECK regime profile* $\pi_M(\mathcal{P}) \in \Delta^3$ is the vector of empirical frequencies $(\pi_D, \pi_E, \pi_C, \pi_K)$ assigning each response to one of the four regimes using the operationalization of §3. A working profile can be constructed from a held-out validation set of 200–500 prompts, at a cost of one to four hours of single-GPU time for an 8B-scale model or one API-driven sampling round for a closed model. Labels are not required to compute the profile itself; they are required only for the optimization step in §6.3.

6.2. Mapping Regime Profiles to Ensemble Weights

Table 3 gives a recommended scorer mix for a domain dominated by each regime: within each row we up-weight the family whose disagreement with the others falls disproportionately in the dominant regime (§5.2), since that family catches the hallucinations the others miss. The final row is the universal blind spot (§5.4): no choice of weights recovers signal, and the right response is upstream abstention.

For a domain with mixed profile $\pi = (\pi_D, \pi_E, \pi_C, \pi_K) \in \Delta^3$, we form the recommended ensemble weight vector $w \in \mathbb{R}^3$ over $(w_{\mathrm{BB}}, w_{\mathrm{WB}}, w_{\mathrm{J}})$ by averaging the four row-weight vectors $w^{(r)}$ from Table 3, weighted by regime prevalence:

$$w \;=\; \pi_D\, w^{(D)} + \pi_E\, w^{(E)} + \pi_C\, w^{(C)} + \pi_K\, w^{(K)}. \quad (3)$$

Because each $w^{(r)}$ already sums to 1 and the π_r form a probability vector, w is automatically a valid (non-negative, normalized) ensemble weight, with no further projection required. Appendix E shows that (3) is robust to moderate noise in π, so cheap profile estimates suffice in practice.

6.3. Learning the Weights and the Abstention Envelope

Equation (3) is a starting point, not an endpoint. With $\geq$ 200 labeled examples we recommend refining w via UQEnsemble's AUROC-maximizing weight learner (Bouchard et al., 2025), using the profile-derived w as a warm start. The taxonomy then converts learned weights into a profile diagnostic: concentration on **J** indicates Entrenched-heavy data; concentration on **WB** indicates Knotted-heavy; balanced **BB+J** with **WB** near zero indicates Drift; concentration on **BB** matches the Confabulation/semantic-entropy story (Farquhar et al., 2024).

A learned vector whose ensemble AUROC remains near chance regardless of weights signals the universal blind

Table 3. DECK regime profile → recommended scorer mix and ensemble weighting. Weights are starting points, not prescriptions.

Dominant regime	Typical domain signature	Likely mechanism	BB	WB	J	Notes
Confabulation	Rare-entity QA, long-tail facts, open-ended problems (multiple valid answers)	Semantic sampling noise; genuine not-knowing	0.50	0.30	0.20	BB sufficient and cheapest.
Drift	Creative long-form, reasoning with multiple plausible chains, code with style variance	Confident per-token, unstable answer-level commitment	0.45	0.15	0.40	Use BB+J; treat WB as tiebreaker.
Knotted	Novel vocabulary, code with rare APIs, domain jargon the model parses but does not commit to	Low per-token probability, stable answer; near refusal thresh.	0.20	0.55	0.25	WB strongest single signal; BB misled by high consistency.
Entrenched	Medical, legal, regulatory QA with common misconceptions; TruthfulQA-like compliance checks	Memorized misconception; shared-pretraining misbelief	0.15	0.15	0.70	Judge *must* be from independent provider.
Universal blind spot	Unanswerable, out-of-scope, post-knowledge-cutoff	No variance to reveal; uniform confident fabrication	—	—	—	UQ scoring not the right tool; route to abstention upstream.

spot (§5.4) and cannot be repaired by reweighting. This is a regime change, not a defect: when a model confidently fabricates on out-of-knowledge inputs, every assumption underlying UQ scoring fails at once. The engineering response is to wrap UQ in an *abstention envelope* (Feng et al., 2024; Slobodkin et al., 2023; Kalai et al., 2025) that rejects out-of-scope inputs before scoring. The practical rule: *run UQ inside an abstention envelope; outside, abstain or escalate.*

6.4. Drift Monitoring via the DECK Profile

The DECK regime profile $\pi_M(\mathcal{P})$ is computable from BB and WB scores alone, without labels. In production, re-estimating π on a rolling window of incoming prompts therefore costs nothing beyond the scorer inference already being run. A statistically significant shift in π—detectable via a chi-square or KL-divergence test against the deployment-time baseline—indicates that the hallucination landscape has changed (e.g., new topic distribution, model update, knowledge-cutoff drift) and that the current ensemble weights and scorer selection should be revalidated. This closes the loop from the pre-deployment diagnostic (§5.5) to continuous production assurance.

7. Limitations

Several limitations bound our conclusions. First, while our judge panel mitigates shared-bias confounds, it does not eliminate them: all judges may share systematic misconceptions from common web-scale pretraining, an effect the SelfAware results suggest is especially pronounced for knowledge-gap inputs. Second, the DECK regimes are operationalized via observable score proxies rather than ground-truth annotations of model uncertainty; $C(A, B)$ depends on the binarisation threshold, which we address by reporting $C_H^{(p)}$ at $p \in \{25, 50, 75\}$ but cannot fully resolve. Third, for Gemini-2.5-flash some WB and BB scores are computed from the same sampling infrastructure as Llama-3-8B on PopQA, which may affect cross-model comparisons on that dataset. Finally, our analysis is restricted to short-form QA; long-form generation involves claim-level dynamics that may require a different framework.

8. Conclusion

The DECK taxonomy partitions LLM hallucinations along inter-sample consistency and token-level confidence into four behavioral regimes that map to the strengths and blind spots of the three UQ scorer families. We validated the taxonomy via disagreement decomposition across three models (Llama-3-8B, GPT-4o, Gemini-2.5-flash) and four datasets (TriviaQA, HaluEval, SelfAware, PopQA), and converted it into a profile-driven ensemble-weighting recipe (Table 3, Eq. 3). On knowledge-gap inputs every output-level scorer collapses to chance across all three models — a *universal blind spot* that marks the operational boundary of the three-family paradigm and motivates abstention, not better scoring, as the right response outside that boundary.

References

Abbasi Yadkori, Y., Kuzborskij, I., György, A., and Szepesvári, C. To believe or not to believe your LLM. *arXiv preprint arXiv:2406.02543*, 2024.

Bouchard, D., Chauhan, M. S., Skarbrevik, D., Ra, H.-K., Bajaj, V., and Ahmad, Z. UQLM: A Python package for uncertainty quantification in large language models. *Journal of Machine Learning Research*, 27, 2025. URL http://jmlr.org/papers/v27/25-1557.html.

Chiang, C.-H. and Lee, H.-y. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15607–15631, 2023.

Duan, J., Cheng, H., Wang, S., Zavalny, A., Wang, C., Xu,

R., Kailkhura, B., and Xu, K. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Evans, O., Cotton-Barratt, O., Finnveden, L., Bales, A., Balwit, A., Wills, P., Righetti, L., and Saunders, W. Truthful AI: Developing and governing AI that does not lie. 2021.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024.

Feng, S., Shi, W., Wang, Y., Ding, W., Balachandran, V., and Tsvetkov, Y. Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 14664–14690, 2024.

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 2025. Earlier arXiv:2311.05232.

Hüllermeier, E. and Waegeman, W. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3): 457–506, 2021.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

Jiang, Z., Araki, J., Ding, H., and Neubig, G. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Ganguli, D., Mirhoseini, A., Jones, A., Kernion, J., Lovitt, L., El-Showk, S., et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Kalai, A. T., Nachum, O., Vempala, S. S., and Zhang, E. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Kang, S., Bakman, Y. F., Yaldiz, D. N., Buyukates, B., and Avestimehr, S. Uncertainty quantification for hallucination detection in large language models: Foundations, methodology, and future directions. *arXiv preprint arXiv:2510.12040*, 2025.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*, 2023.

Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., and Wen, J.-R. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6449–6464, 2023.

Lin, Z., Trivedi, S., and Sun, J. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2024.

Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., and Zhu, C. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2511–2522, 2023.

Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2021.

Mallen, A., Asai, A., Zhong, V., Das, R., Khashabi, D., and Hajishirzi, H. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023.

Manakul, P., Liusie, A., and Gales, M. J. F. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9004–9017, 2023.

Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-t., Koh, P. W., Iyyer, M., Zettlemoyer, L., and Hajishirzi, H. 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*, pp. 12076–12100, 2023.

Qiu, X. and Miikkulainen, R. Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic space. *arXiv preprint arXiv:2405.13845*, 2024.

Shelmanov, A., Fadeeva, E., Tsvigun, A., Tsvigun, I., Xie, Z., Kiselev, I., Daheim, N., Zhang, C., Vazhentsev, A., Sachan, M., Nakov, P., and Baldwin, T. A head to predict and a head to question: Pre-trained uncertainty quantification heads for hallucination detection in LLM outputs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 35712–35731, 2025.

Shorinwa, O., Mei, Z., Lidard, J., Ren, A. Z., and Majumdar, A. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *arXiv preprint arXiv:2412.05563*, 2025.

Slobodkin, A., Goldman, O., Caciularu, A., Dagan, I., and Ravfogel, S. The curious case of hallucinatory (un)answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Tian, K., Mitchell, E., Zhou, A., Khazatsky, A., Manning, C. D., and Finn, C. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Wang, Y., Wang, M., Manzoor, M. A., Liu, F., Georgiev, G. N., Das, R. J., and Nakov, P. Factuality of large language models: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 19519–19529, 2024.

Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., and Hooi, B. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *International Conference on Learning Representations*, 2024.

Yin, Z., Sun, Q., Guo, Q., Wu, J., Qiu, X., and Huang, X. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8653–8665, 2023.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

A. Scorer Family Reference

Table 4 summarizes the three scorer families evaluated in this paper and their compatibility with each generator model. Black-box and judge scorers require only sampling access and can therefore be applied to any of the four generators. White-box scorers depend on per-token log-probabilities, which Llama-3-8B (open-weights, run locally), GPT-4o (via the OpenAI API `logprobs` parameter), and Gemini-2.5-Flash (via the Google AI `logprobs` parameter) all expose. Claude Sonnet 4.6 does *not* provide access to token-level log-probabilities through the Anthropic API, so it is used exclusively as a judge and is excluded from the white-box family.

Table 4. Scorer families, logprob requirements, and model applicability. ✓ = applicable; × = excluded.

Family	Logprb	Llama-3-8B	GPT-4o	Gemini-2.5	Sonnet-4.6
Black-box	No	✓	✓	✓	✓
White-box	Yes	✓	✓	✓	×
LLM-Judge	No	✓	✓	✓	✓

B. Response Grading Procedures

Each dataset uses a distinct grading function to assign a binary correctness label (`response_correct`) to every model response. The label is used as the ground-truth signal for AUROC computation. All grading is deterministic and requires no additional model calls.

TriviaQA and HaluEval

Both datasets supply a reference answer string. A response is marked correct if the *normalised* reference appears as a substring of the *normalised* response. Normalisation lowercases the text, strips English articles (*a*, *an*, *the*), removes punctuation, and collapses whitespace—the same procedure used in the original TriviaQA evaluation (Joshi et al., 2017).

TriviaQA / HaluEval grading

```python
import re, string

def normalize(text: str) -> str:
    """Lower, strip articles, strip punctuation, collapse spaces."""
    text = text.lower()
    text = re.sub(r'\b(a|an|the)\b', '', text)
    text = text.translate(str.maketrans('', '', string.punctuation))
    return ' '.join(text.split())

def is_correct(response: str, ground_truth: str) -> bool:
    """True if normalised ground truth is a substring of response."""
    return normalize(ground_truth) in normalize(response)

df['response_correct'] = df.apply(
    lambda row: is_correct(row['response'], row['ground_truths']),
    axis=1)
```

HaluEval uses the same function because each sample in the `pminervini/HaluEval qa_samples` split provides a single reference answer field (`answer`), identical in structure to TriviaQA.

SelfAware

SelfAware contains questions that are *genuinely unanswerable*. A response is graded correct if it appropriately expresses uncertainty or inability to answer, detected via a set of regular-expression patterns that cover common refusal and hedging phrasings.

SelfAware grading

```python
UNCERTAINTY_PATTERNS = [
    r"i don'?t know",
    r"i do not know",
    r"i'?m not sure",
    r"i cannot (answer|say|tell|determine|confirm)",
    r"i'?m unable to",
    r"no (definitive|clear|single|known) answer",
    r"(unclear|unknown|uncertain|unanswerable"
     r"|impossible to (say|know|determine))",
    r"there('?s| is) no "
     r"(way|information|record|evidence|known)",
    r"(cannot|can'?t) be "
     r"(determined|confirmed|verified|known)",
    r"(it'?s|this is) "
     r"(unclear|uncertain|unknown|not known|debatable)",
]

def is_correct_unanswerable(response: str) -> bool:
    """True if the response expresses appropriate uncertainty."""
```

```
    r = response.lower()
    return any(re.search(p, r) for p in
        UNCERTAINTY_PATTERNS)

df['response_correct'] = df['response'
    ].apply(
    is_correct_unanswerable)
```

Only the 500 *unanswerable* questions (`answerable == False`) from the SelfAware dataset are used; answerable questions are excluded because the focus of this dataset split is specifically on knowledge-gap hallucinations.

PopQA

PopQA is an entity-centric open-domain QA dataset (Mallen et al., 2023). Each sample includes a `possible_answers` field listing all acceptable surface forms of the correct answer. A response is marked correct if the *normalised* form of any element in `possible_answers` appears as a substring of the *normalised* response, using the same normalisation procedure as TriviaQA.

PopQA grading

```
def is_correct_popqa(response: str,
                     possible_answers:
                         list) -> bool:
    """True if any normalised alias is
        a substring of response."""
    r_norm = normalize(response)
    return any(normalize(ans) in r_norm
               for ans in
                   possible_answers)

df['response_correct'] = df.apply(
    lambda row: is_correct_popqa(
        row['response'], row['
            possible_answers']),
    axis=1)
```

The `possible_answers` field may contain multiple aliases (e.g. *"United States"*, *"USA"*, *"US"*). Any matching alias is sufficient for a correct label.

Dataset summary

Table 5. Grading scheme and accuracy per dataset–model combination.

Dataset	Grading method	Llama	GPT-4o	Gemini
TriviaQA	Norm. substring	59.0%	80.4%	83.2%
HaluEval	Norm. substring	17.6%	29.8%	33.4%
SelfAware	Regex uncertainty	7.2%	9.4%	12.0%
PopQA	Alias substring	63.5%	69.6%	63.5%

C. Full AUROC Results — All Datasets and Models

Table 6 reports AUROC for all 15 scorer conditions across all twelve dataset–model combinations (three models × four datasets: TriviaQA, HaluEval, SelfAware, PopQA). **Bold** = best in family per column; all scores on 500 samples. Figure 3 visualizes all twelve dataset–model combinations as horizontal bar charts; exact values are tabulated below.

The three SelfAware columns (dark red header) reveal a consistent blind spot across all models. For GPT-4o, thirteen of fifteen scorers fall below 0.5 (best: Sonnet judge 0.528), with P(True) producing the single worst score (0.331) in this subtable. For Llama-3-8B, the failure is less extreme but still substantial: nine of fifteen scorers fall below 0.5, with the GPT-4o cross-judge (0.578), Gemini cross-judge (0.575), and Non-Contradiction (0.587) providing modest above-chance signal. Gemini-2.5-Flash sits between the two: all scorers cluster between 0.37 and 0.53, with the cross-model judges (GPT-4o 0.519, Sonnet 0.528, Gemini 0.502) and P(True) (0.510) marginally above chance while BB scorers mostly fall below. All three profiles confirm the universal blind spot of §5.4—no scorer family provides reliable detection when the model is systematically and confidently wrong.

On TriviaQA and HaluEval, Gemini-2.5-Flash AUROC is uniformly compressed relative to Llama-3-8B and comparable to or slightly below GPT-4o: on TriviaQA the best BB (Entailment, 0.655) and WB (Min Token Prob., 0.670) scorers match GPT-4o's top scores, while on HaluEval all families collapse toward 0.53–0.57, roughly 0.07–0.13 points below the Llama leader. The Gemini HaluEval compression likely reflects the model's higher baseline accuracy (33.4% vs. 17.6% for Llama): with fewer errors to detect, all uncertainty signals operate on a harder residual set of hallucinations.

The three PopQA columns (purple header) stand out as the highest-AUROC region of the table. Cosine Similarity leads the BB family for all three models (0.787 for Llama and Gemini, 0.774 for GPT-4o), reflecting the strong semantic-distance signal available for entity-centric questions. WB scorers are notably asymmetric across models: GPT-4o's Mean Token Neg. (0.802) substantially outperforms Llama's P(True) (0.730) and Gemini's P(True)

Figure 3. Per-method AUROC across all twelve dataset–model combinations (rows: GPT-4o, Llama-3-8B, Gemini-2.5-Flash; columns: TriviaQA, HaluEval, SelfAware, PopQA). Methods are grouped by scorer family (blue = BB, orange = WB, green = Judge). Exact values in Table 6.

(0.686), consistent with GPT-4o's higher PopQA baseline accuracy providing better-calibrated token-level probabilities. The Llama judge (0.744) tops the Judge family for GPT-4o, while Sonnet (0.727–0.728) leads for Llama and Gemini, indicating cross-model judges provide complementary signal to the generator's own uncertainty on entity-centric queries.

D. Step-by-Step Computation of the Pairwise Complementarity Score

This appendix walks through the exact computation of the pairwise complementarity score $C(A, B)$ as used in §5.2, from raw scorer outputs to the final table values in Table 8.

Step 1: Select the best scorer per family

For a given dataset–model condition (e.g. Llama-3-8B on TriviaQA), each scorer family (BB, WB, Judge) is represented by its single highest-AUROC method on that split. The representative is chosen from Table 7. This avoids hand-selection bias while ensuring each family contributes its strongest available signal.

Example. For Llama-3-8B on TriviaQA the representatives are: Non-Contradiction (BB, AUROC 0.810), P(True) (WB, AUROC 0.791), GPT-4o cross-judge (Judge, AUROC 0.844).

Step 2: Binarise scorer outputs via median threshold

Each representative produces a continuous confidence score for every sample. A higher score means the scorer considers the response *more likely correct*. These real-valued scores are converted to binary predictions using the within-split median as threshold:

$$\hat{y}_i^{(A)} = \begin{cases} 1 & \text{(predicted correct)} \quad \text{if } s_i^{(A)} \geq \tilde{s}^{(A)} \\ 0 & \text{(predicted hallucination)} \quad \text{if } s_i^{(A)} < \tilde{s}^{(A)} \end{cases}$$

where $\tilde{s}^{(A)}$ is the median of scorer A's outputs across all N samples in the split. By construction, exactly $N/2$ samples are assigned to each class (for continuous scores with no ties at the median), giving balanced binary predictions.

Table 6. Per-method AUROC across all twelve dataset–model combinations (three models × four datasets). **Bold** = best in family per column. Rows sorted within each family by Llama/TriviaQA AUROC (descending). Header shading: Triv = TriviaQA, Halu = HaluEval, Self = SelfAware, Pop = PopQA. Row shading: = BB, = WB, = Judge.

Scorer	Llama Triv.	GPT-4o Triv.	Gem. Triv.	Llama Halu.	GPT-4o Halu.	Gem. Halu.	Llama Self.	GPT-4o Self.	Gem. Self.	Llama PopQA	GPT-4o PopQA	Gem. PopQA
Non-Cont.	**0.810**	**0.653**	0.630	0.640	0.579	0.551	**0.587**	0.401	0.428	0.771	0.684	0.771
Semantic Neg.	0.792	0.639	0.619	0.636	0.575	**0.572**	0.542	0.495	0.394	0.718	0.589	0.718
Entailment	0.791	0.640	**0.655**	**0.656**	0.572	0.540	0.497	0.355	0.369	0.710	0.607	0.710
Sem. Sets Conf.	0.791	0.639	0.618	0.637	**0.579**	**0.572**	0.548	0.493	0.387	0.722	0.602	0.722
Cosine Similarity	0.749	0.599	0.604	0.605	0.568	0.540	0.375	0.471	0.444	**0.787**	**0.774**	**0.787**
Exact Match	0.677	0.603	0.634	0.489	0.516	0.533	0.499	**0.499**	**0.499**	0.526	0.575	0.526
P(True)	**0.791**	**0.675**	0.588	**0.638**	**0.582**	0.528	0.447	0.331	**0.510**	**0.730**	0.796	**0.686**
Mean Token Neg.	0.700	0.607	0.585	0.559	0.579	0.521	0.496	0.474	0.469	0.668	**0.802**	0.668
Sequence Prob.	0.676	0.613	0.622	0.574	0.562	**0.531**	**0.502**	0.470	0.469	0.660	0.754	0.660
Prob. Margin	0.674	0.595	0.603	0.535	0.576	0.520	0.492	**0.483**	0.486	0.639	0.784	0.639
Min Token Prob.	0.661	0.617	**0.670**	0.557	0.528	0.525	0.472	0.440	0.390	0.606	0.679	0.606
GPT-4o judge	**0.844**	0.573	0.572	**0.704**	**0.579**	**0.541**	**0.578**	0.467	0.519	0.699	0.643	0.702
Sonnet judge	**0.844**	**0.651**	**0.627**	0.599	0.567	0.525	0.492	**0.528**	**0.528**	**0.727**	0.714	**0.728**
Llama judge	0.667	0.589	0.568	0.567	0.518	0.506	0.458	0.409	0.445	0.642	**0.744**	0.642
Gemini judge	0.830	0.582	0.538	0.649	0.562	0.514	0.575	0.513	0.502	0.537	0.571	0.538

Table 7. Top scorer from each family per condition. SelfAware (†): all AUROCs near chance; scores shown for completeness.

Dataset	Model	BB	WB	Judge
TriviaQA	Llama-3-8B	Non-Cont. (0.810)	P(True) (0.791)	GPT-4o (0.844)
	GPT-4o	Non-Cont. (0.653)	P(True) (0.675)	Sonnet (0.651)
	Gemini-2.5	Entailment (0.655)	Min Prob. (0.670)	Sonnet (0.627)
HaluEval	Llama-3-8B	Entailment (0.656)	P(True) (0.638)	GPT-4o (0.704)
	GPT-4o	Sem. Sets (0.579)	P(True) (0.582)	GPT-4o (0.579)
	Gemini-2.5	Sem. Neg. (0.572)	Seq. Prob. (0.531)	GPT-4o (0.541)
SelfAw.†	Llama-3-8B	Non-Cont. (0.587)	Seq. Prob. (0.502)	GPT-4o (0.578)
	GPT-4o	Exact Match (0.499)	Prob. Margin (0.483)	Sonnet (0.528)
	Gemini-2.5	Exact Match (0.499)	P(True) (0.510)	Sonnet (0.528)
PopQA	Llama-3-8B	Cosine (0.787)	P(True) (0.730)	Sonnet (0.727)
	GPT-4o	Cosine (0.774)	Mean Tok. (0.802)	Llama (0.744)
	Gemini-2.5	Cosine (0.787)	P(True) (0.686)	Sonnet (0.728)

Step 3: Identify disagreement samples

A sample i is a *disagreement* between families A and B if and only if their binary predictions differ:

$$\mathcal{D}(A, B) = \{i : \hat{y}_i^{(A)} \neq \hat{y}_i^{(B)}\}$$

Disagreements split into two disjoint cases:

- A flags, B misses: $\hat{y}_i^{(A)} = 0,\ \hat{y}_i^{(B)} = 1$ (A predicts hallucination, B predicts correct)
- B flags, A misses: $\hat{y}_i^{(A)} = 1,\ \hat{y}_i^{(B)} = 0$ (B predicts hallucination, A predicts correct)

Let $n_{01} = |\{i : \hat{y}^{(A)}{=}0, \hat{y}^{(B)}{=}1\}|$ and $n_{10} = |\{i : \hat{y}^{(A)}{=}1, \hat{y}^{(B)}{=}0\}|$, so $|\mathcal{D}| = n_{01} + n_{10}$.

Step 4: Apply the complementarity formula

The pairwise complementarity score is defined as:

$$C(A, B) = \tfrac{1}{2}\big[P(B \text{ flags} \mid A \text{ misses}) + P(A \text{ flags} \mid B \text{ misses})\big] \tag{4}$$

Expanding the conditional probabilities explicitly:

$$C(A, B) = \frac{1}{2}\left[\frac{n_{01}}{|\{\hat{y}^{(A)}{=}0\}|} + \frac{n_{10}}{|\{\hat{y}^{(B)}{=}0\}|}\right] \tag{5}$$

Because the median split produces exactly $N/2$ zeros for each scorer, $|\{\hat{y}^{(A)}{=}0\}| = |\{\hat{y}^{(B)}{=}0\}| = N/2$, and the expression simplifies to:

$$C(A, B) = \frac{1}{2} \cdot \frac{n_{01} + n_{10}}{N/2} = \frac{n_{01} + n_{10}}{N} = \frac{|\mathcal{D}(A, B)|}{N} \tag{6}$$

The complementarity score is therefore the fraction of all samples on which families A and B produce opposite binary predictions. This simplification holds exactly when the median threshold is used; it would not hold under an arbitrary fixed threshold that produces unbalanced class counts.

Step 5: Worked example

Condition: Llama-3-8B on TriviaQA, pair BB vs. WB (N=500).

1. Representatives: Non-Contradiction (BB), P(True) (WB).

2. Binarise: median split assigns 250 samples to each binary class for each scorer.

3. Count disagreements: 126 samples have $\hat{y}^{(\text{BB})} \neq \hat{y}^{(\text{WB})}$ (63 where BB flags and WB misses; 63 where WB flags and BB misses).

4. Apply formula:

$$C(\text{BB}, \text{WB}) = \frac{126}{500} = 0.252$$

This matches the value reported in Table 8. The 63/63 symmetric split between the two disagreement directions is also consistent with the anti-diagonal fingerprint discussed in §5.2: all BB–WB disagreements on this condition fall in the Knotted and Drift quadrants at exactly 50% each.

Step 6: Hallucination-conditional complementarity $C_H^{(p)}$

The pairwise score $C(A, B)$ counts disagreements over *all* N samples, including correctly answered ones. The hallucination-conditional variant $C_H^{(p)}(A, B)$ restricts attention to the N_H samples that were *actually hallucinated* (`response_correct` = `False`):

$$C_H^{(p)}(A, B) = \frac{|\mathcal{D}^H(A, B)|}{N_H}, \tag{7}$$
$$\mathcal{D}^H(A, B) = \{i \in H : \hat{y}_i^{(A)} \neq \hat{y}_i^{(B)}\}$$

where $H = \{i : \texttt{response_correct}_i = \texttt{False}\}$. Three details distinguish $C_H^{(p)}$ from C:

1. The median thresholds $\tilde{s}^{(A)}$ and $\tilde{s}^{(B)}$ are still computed over *all* N samples, not the hallucinated subset alone; the binarisation is therefore identical to the one used in $C(A, B)$.

2. The denominator is N_H rather than N, so $C_H^{(p)}$ is not equivalent to C restricted to the subset; it is a conditional rate.

3. The balanced-class simplification ($C = |\mathcal{D}|/N$) carries over only within the hallucinated subset if and only if the median split happens to assign exactly $N_H/2$ hallucinated samples to each binary class—which is generally not the case, since scorers tend to flag more hallucinated samples as low-confidence.

Continuing the worked example. For Llama-3-8B on TriviaQA (BB = Non-Contradiction, WB = P(True), N=500):

1. Identify hallucinated samples: $N_H = 205$ (response_correct = False).

2. Among these 205 samples, apply the same global-median binary predictions from Step 2. Of the 205 hallucinated samples, 158 are flagged by BB (low Non-Contradiction score) and 158 are flagged by WB (low P(True) score).

3. Count disagreements within H: $|\mathcal{D}^H| = 40$ samples where BB and WB produce opposite predictions (20 where BB flags and WB misses; 20 where WB flags and BB misses).

4. Apply formula:

$$C_H^{(p)}(\text{BB}, \text{WB}) = \frac{40}{205} \approx 0.195$$

The hallucination-conditional score (0.195) is lower than the unconditional score ($C = 0.252$), reflecting the fact that both families tend to agree on hallucinated samples—both assign low confidence—and disagree more on the correctly answered samples where confidence is naturally high. Equivalently, $|\mathcal{D}^H|/N_H < |\mathcal{D}|/N$ implies the two families capture *overlapping* subsets of hallucinations, so the net ensemble gain from combining them is smaller than C alone would suggest. This value corresponds to the $C_H^{(p)}(\text{BB}, \text{WB})$ entry reported in Table 8.

Step 7: Quadrant decomposition of disagreements

Each disagreement sample is additionally assigned to one of the four hallucination taxonomy quadrants using a second median split, this time on the *pair* of raw scores $(s^{(\text{BB})}, s^{(\text{WB})})$ used to define the consistency and confidence axes respectively:

Quadrant	BB score	WB score
Entrenched (E)	$\geq$ median	$\geq$ median
Knotted (K)	$\geq$ median	$<$ median
Drift (D)	$<$ median	$\geq$ median
Confabulation (C)	$<$ median	$<$ median

The stacked bar charts in Figure 2 (main text, TriviaQA) and Figure 4 (this appendix, remaining datasets) show how $\mathcal{D}^H(A, B)$ is distributed across these four quadrants, restricted to hallucinated samples. The quadrant decomposition is purely descriptive—it does not change the value of $C(A, B)$—but reveals *where* in the hallucination space each family pair disagrees, which is the central diagnostic of the taxonomy validation.

E. Robustness of the DECK Ensemble Recipe and Step-by-Step Procedure

A natural concern with profile-driven weighting is that the DECK profile π is estimated from a finite validation set and may be noisy. In practice this is not a problem

Table 8. Pairwise complementarity $C(A, B)$ (all samples, median threshold) and hallucination-restricted complementarity $C_H^{(p)}(A, B)$ (Eq. 2; incorrect samples only, p-th percentile threshold) for all three family pairs. SelfAware rows (†) are in a degenerate regime where all AUROCs are near chance; high scores reflect independent failure, not complementary detection. **Bold** = larger value between BB,J and WB,J within each column group (BB,WB column unbolded).

Condition		$C(A,B)$			$C_H^{(25)}$			$C_H^{(50)}$			$C_H^{(75)}$		
Dataset	**Model**	BB,WB	BB,J	WB,J	BB,WB	BB,J	WB,J	BB,WB	BB,J	WB,J	BB,WB	BB,J	WB,J
TriviaQA	Llama-3-8B	0.252	**0.240**	0.236	0.376	**0.483**	0.468	0.195	**0.185**	**0.185**	0.141	0.156	**0.210**
	GPT-4o	0.396	**0.390**	0.350	0.276	**0.276**	0.184	0.296	**0.327**	0.316	0.265	**0.469**	0.449
	Gemini-2.5	0.288	0.410	**0.422**	0.202	0.250	**0.310**	0.202	0.405	**0.464**	0.119	0.512	**0.560**
HaluEval	Llama-3-8B	0.300	**0.274**	0.258	0.277	**0.286**	0.277	0.306	**0.282**	0.257	0.260	**0.262**	0.221
	GPT-4o	0.354	0.232	**0.294**	0.197	**0.225**	0.131	0.339	0.239	**0.282**	0.353	0.399	**0.501**
	Gemini-2.5	0.444	0.426	**0.458**	0.351	0.270	**0.309**	0.423	0.426	**0.459**	0.411	0.426	**0.622**
SelfAware†	Llama-3-8B	0.432	0.400	**0.432**	0.312	0.222	**0.224**	0.427	0.399	**0.433**	0.364	0.543	**0.545**
	GPT-4o	0.500	0.192	**0.536**	0.258	0.196	**0.353**	0.494	0.196	**0.528**	0.746	0.196	**0.704**
	Gemini-2.5	0.498	0.102	**0.516**	0.252	0.109	**0.316**	0.507	0.109	**0.525**	0.739	0.109	**0.698**
PopQA	Llama-3-8B	0.337	0.295	**0.315**	0.242	**0.330**	**0.330**	0.264	0.269	**0.291**	0.203	0.242	**0.247**
	GPT-4o	0.312	**0.380**	**0.380**	0.336	**0.368**	0.336	0.191	0.414	**0.421**	0.072	0.204	**0.250**
	Gemini-2.5	0.355	0.295	**0.345**	0.341	0.308	**0.363**	0.264	0.275	**0.352**	0.170	0.236	**0.297**

for two reasons. First, the regime boundaries in Table 3 are wide: the weight differences between rows are large (≥ 0.20 on at least one component), so small perturbations to π produce small changes in w via Equation (3). Second, even an entirely uninformative uniform profile $\pi = (0.25, 0.25, 0.25, 0.25)$ — which corresponds to having no information at all about the deployment domain — yields a weight vector $w = (0.325, 0.2875, 0.3875)$ that already outperforms an equal-weight default across the non-degenerate conditions in our experiments (§5). The profile-driven recipe is therefore best understood as a way to extract value from prior information about a domain when such information exists, not as a hard requirement.

Cheap profile estimation when labels are unavailable. For domains where a labeled validation set cannot be assembled, a proxy profile is constructible at low cost: label a small sample (e.g., $n = 50$) with a strong external judge or a brief human-annotation pass, extrapolate the regime fractions, and plug them into (3). Because the recipe is robust to moderate profile error, this is sufficient for most production settings. We recommend re-estimating the profile when the deployment distribution visibly shifts (new product surface, new user segment, new knowledge-cutoff regime), since regime-shift is the most common mechanism by which a previously calibrated ensemble drifts out of calibration.

Practical step-by-step recipe. The DECK design procedure can be summarized in four steps: (1) estimate the regime profile $\pi_M(\mathcal{P})$ on a validation sample of 200–500 prompts; (2) compute the initial weight vector w via Equation (3); (3) if labels are available, refine w with the tunable UQEnsemble (Bouchard et al., 2025) and interpret the learned weights diagnostically (§6.3); (4) deploy UQ scoring inside an abstention envelope, routing out-of-scope inputs to refusal or retrieval rather than to the scorer stack.

F. External-Signal Validation Protocol

This appendix gives the procedure used to populate Table 9.

Cell assignment. Within each (model, dataset) condition, compute the median of `bb_score` and `wb_score` over all samples. Each sample is assigned to one of {D, E, C, K} by:

$$\begin{aligned} \text{E} &: \texttt{bb} \geq \tilde{\texttt{bb}} \text{ and } \texttt{wb} \geq \tilde{\texttt{wb}}; \\ \text{D} &: \texttt{bb} < \tilde{\texttt{bb}} \text{ and } \texttt{wb} \geq \tilde{\texttt{wb}}; \\ \text{C} &: \texttt{bb} < \tilde{\texttt{bb}} \text{ and } \texttt{wb} < \tilde{\texttt{wb}}; \\ \text{K} &: \texttt{bb} \geq \tilde{\texttt{bb}} \text{ and } \texttt{wb} < \tilde{\texttt{wb}}. \end{aligned}$$

Significance test. For datasets containing multiple external labels (HaluEval, PopQA), a 2×4 chi-square contingency table contrasts the target label's cell distribution against the pooled distribution of all other labels in that condition. For single-label datasets (SelfAware), the contrast is against pooled `natural` samples from *other* datasets evaluated on the same model, re-binned using the current condition's medians. Yates's continuity correction is applied; a warning is emitted when any expected cell count is below 5 (in which case Fisher's exact should be substituted). All p-values reported with df $= 3$.

Edge cases. (i) If an external label has fewer than ~ 50 samples in a condition, the chi-square is unreliable; report rates but flag the p-value. (ii) The test assumes the BB and

WB scorers used for cell assignment are the same representatives used in §5.2; using different scorers would not be a valid replication. (iii) The test does not control for response correctness; an incorrect-only version using Eq. 2's subset is a more conservative test of mechanistic distinctiveness.

Data adaptations. Wikipedia page-view counts were not available in the pre-computed scorer outputs. `popular_entity` and `rare_entity` labels were therefore approximated by cross-model consensus correctness on PopQA: samples answered correctly by both Llama-3-8B and GPT-4o are labelled `popular_entity` (n=276); samples answered incorrectly by both are labelled `rare_entity` (n≈111); samples on which the two models disagree form the `natural` stratum (n=113). All HaluEval rows are treated as `adversarial` (the dataset is fully curated to test hallucination); the TriviaQA stratum serves as the naturalistic baseline, re-binned with each HaluEval condition's medians.

G. The Effect of Model Scale

We compare complementarity scores across Llama-3-8B, GPT-4o, and Gemini-2.5-flash to test whether larger or more capable models produce proportionally more Entrenched hallucinations (high-consistency, high-confidence errors under the DECK taxonomy), increasing the value of judge-based detection.

TriviaQA (factual open-domain QA). Scale produces a pronounced increase in *all* pairwise complementarity values. $C_H^{(50)}(\text{BB}, \text{WB})$ rises from 0.195 (Llama-3-8B) to 0.296 (GPT-4o) (Δ=+0.101), and $C_H^{(50)}(\text{BB}, \text{Judge})$ rises from 0.185 to 0.327 (Δ=+0.142). The quadrant decomposition confirms the mechanism: at GPT-4o scale, 50% of BB–Judge disagreements fall in the Confabulation quadrant (vs. 32% at Llama-3-8B scale), reflecting a shift toward low-consistency hallucinations that BB consistency checks cannot detect but independent judges can. *The scale hypothesis is confirmed on factual QA.*

HaluEval (adversarial hallucination detection). The scale effect is more mixed. $C_H^{(50)}(\text{BB}, \text{WB})$ *increases* from 0.306 to 0.339 (Δ=+0.033), while $C_H^{(50)}(\text{BB}, \text{Judge})$ *decreases* from 0.282 to 0.239 (Δ=−0.043), indicating that at GPT-4o scale the judge and BB scorer agree more often on adversarially perturbed inputs. $C_H^{(50)}(\text{WB}, \text{Judge})$ rises from 0.257 to 0.282 (Δ=+0.025), confirming that judge scorers retain complementarity over WB methods even as BB–Judge divergence narrows.

Gemini-2.5-flash: elevated complementarity, weaker detection. On both TriviaQA and HaluEval, Gemini-2.5-flash exhibits higher $C_H^{(50)}(A, B)$ values than either Llama-3-8B or GPT-4o (TriviaQA: $C_H^{(50)}(\text{WB}, \text{Judge})$=0.464; HaluEval: $C_H^{(50)}(\text{BB}, \text{WB})$=0.423, $C_H^{(50)}(\text{WB}, \text{Judge})$=0.459). However, this elevated complementarity is accompanied by lower individual AUROC scores, suggesting that the three families are *more independently uncertain* rather than *more complementarily correct*—a pattern consistent with a less confident generator.

PopQA: high AUROC, moderate complementarity. On the entity-centric PopQA dataset, individual AUROC scores are substantially higher than on TriviaQA across all three models (cosine similarity reaching 0.787 for Llama-3-8B and Gemini-2.5-flash, Mean Token Neg. 0.802 for GPT-4o). Complementarity values are moderate ($C_H^{(50)}(\text{BB}, \text{WB})$ 0.191–0.264), with Judge scorers providing additional coverage particularly at higher percentile thresholds ($C_H^{(50)}(\text{WB}, \text{Judge})$ 0.291–0.421).

Practical implication. These results support *dataset-adaptive ensembling*: on naturalistic factual QA at scale, both BB + WB and BB + Judge complementarity increase markedly. On adversarial tasks, BB + WB retains the highest complementarity at both scales, while a Judge scorer provides additional orthogonal signal that neither BB nor WB alone captures. On entity-centric QA (PopQA), the high AUROC of individual scorers reduces the marginal gain from ensembling, but WB + Judge complementarity remains meaningful at the 50th and 75th percentile thresholds.

Table 9. External-signal validation. For each (model, dataset, external label), the percentage of samples landing in each DECK cell under a within-condition median split. Bold = predicted concentration cell; chi-square contrasts each label distribution against a pooled naturalistic baseline (df = 3, Yates correction). PopQA popularity proxied by cross-model consensus correctness (see text); † indicates non-significant contrast ($p > 0.05$).

Dataset	**Model**	**External label**	**Predicted cell**	n	**D** (%)	**E** (%)	**C** (%)	**K** (%)	p
SelfAware	Llama-3-8B	unanswerable	E	500	21.6	**28.4**	28.4	21.6	<0.001
SelfAware	GPT-4o	unanswerable	E	500	0.0	**50.0**	0.0	50.0	<0.001
SelfAware	Gemini-2.5-F	unanswerable	E	500	0.0	**50.2**	0.0	49.8	<0.001
HaluEval	Llama-3-8B	adversarial	E/K	500	15.0	**35.0**	35.0	**15.0**	<0.001
HaluEval	GPT-4o	adversarial	E/K	500	5.6	**44.4**	20.2	**29.8**	<0.001
HaluEval	Gemini-2.5-F	adversarial	E/K	500	21.8	**28.2**	27.4	**22.6**	<0.001
PopQA	Llama-3-8B	popular_entity	E	276	18.5	**48.6**	13.0	19.9	<0.001
PopQA	Llama-3-8B	rare_entity	C	110	22.7	14.5	**54.5**	8.2	<0.001
PopQA	GPT-4o	popular_entity	E	276	17.8	**48.9**	16.7	16.7	<0.001
PopQA	GPT-4o	rare_entity	C	111	9.9	10.8	**69.4**	9.9	<0.001
PopQA	Gemini-2.5-F	popular_entity	E	276	16.3	**44.9**	15.2	23.6	<0.001
PopQA	Gemini-2.5-F	rare_entity	C	110	20.9	13.6	**56.4**	9.1	0.309†

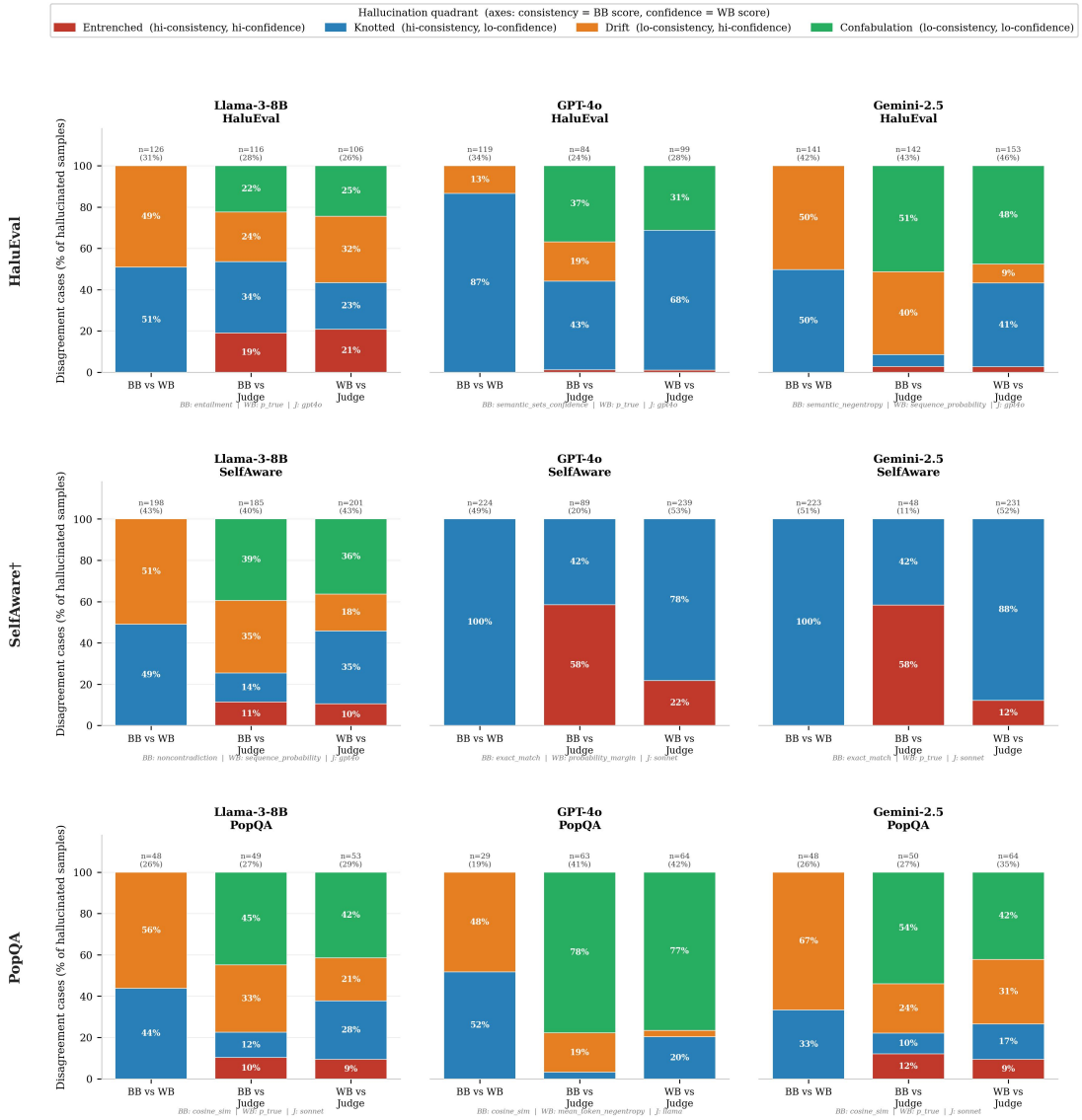

Figure 4. Pairwise scorer-family disagreement on hallucinated samples for HaluEval, SelfAware, and PopQA (three rows), across Llama-3-8B, GPT-4o, and Gemini-2.5 (three columns per row). Bar colouring, quadrant definitions, and annotation conventions match Figure 2. SelfAware (†): near-random AUROC; quadrant pattern is degenerate.